

E.Sh. Nazirova, Sh.B. Abidova, M.M. Mahmudova

MULTIMEDIALI MA'LUMOTLAR BAZASI



O'ZBEKISTON RESPUBLIKASI
RAQAMLI TEXNOLOGIYALAR VAZIRLIGI

MUHAMMAD AL-XORAZMIY NOMIDAGI TOSHKENT
AXBOROT TEXNOLOGIYALARI UNIVERSITETI

E.Sh. Nazirova, Sh.B. Abidova, M.M. Mahmudova

MULTIMEDIALI MA'LUMOTLAR BAZASI

Muhammad Al-Xorazmiy nomidagi Toshkent axborot
texnologiyalari universiteti tomonidan o'quv qo'llanma sifatida
tavsiya etilgan

“Multimedial ma'lumotlar bazasi” fanidan “5330500” –
Kompyuter injiniringi (“Multimedia texnologiyalari”) mutaxassisligi
bakalavriat talabalari uchun

TOSHKENT
“METODIST NASHRIYOTI”
2024

UDK: 004.6(075.8)
KBK: 32.973a7
N 18

E.SH. Nazirova, Sh.B. Abidova, M.M. Mahmudova
Multimediali ma'lumotlar bazasi. / O'quv qo'llanma– Toshkent:
"METODIST NASHRIYOTI", 2024. – 160 b.

Ushbu o'quv qo'llanmada ma'lumotlar bazasi asoslari, multimediali ma'lumotlar turlari, multimediali ma'lumotlar bazasi boshqaruvini tashkil qilish, multimediali ma'lumotlar bazasini ishlab chiqishda relyatsion yondashuvdan foydalanish, tasvirli, video va audio ma'lumotlarni saqlash, multimediali ma'lumotlarni izlash, ma'lumotlarni izlashda doimiy ifodalardan foydalanish, shuningdek, multimedia ma'lumotlarni saqlashda zamonaviy ma'lumotlar bazasini boshqarish tizimlaridan foydalanishning usullari bayon etilgan.

O'quv qo'llanma oliy ta'lim muassasalari bakalauriat talabalari, oliy o'quv yurtlari magistrantlari, ilmiy tadqiqotchilar hamda multimedia va axborot texnologiyalari sohasiga qiziquvchi kitobxonlar ommasiga mo'ljallangan.

Taqrizchilar:

B.Yu. Palvanov Muhammad al-Xorazmiy nomidagi TATU Urganch filiali "Axborot xavfsizligi va ta'lim texnologiyalari" kafedrası mudiri, (PhD) dotsent

Sh.A. Anarova Muhammad al-Xorazmiy nomidagi TATU, "Axborot texnologiyalari" kafedrası mudiri, t.f.d., dotsent

Muhammad Al-Xorazmiy nomidagi Toshkent axborot texnologiyalari universiteti kengashining 2022-yil 22-dekabrda 5(727)-sonli qaroriga asosan nashr etishga ruxsat berilgan.

ISBN 978-9910-03-191-5

© E.SH. Nazirova va boshq., 2024.
© "METODIST NASHRIYOTI", 2024.

KIRISH

"Multimediali ma'lumotlar bazasi" fani talabalarning zamonaviy ma'lumotlar bazalari va ma'lumotlar bazalarini boshqarish tizimlari, ma'lumotlar bazasining hayotiy tsikllari, dizayn texnologiyalari va relyatsion ma'lumotlar bazalarini tahlil qilish sohasidagi kasbiy bilim va ko'nikmalarini shakllantiradigan asosiy fanlardan biridir.

"Multimediali ma'lumotlar bazasi" fanining maqsadi - talabalarni multimedia ma'lumotlar bazalarini tashkil etishning asosiy printsiplari bilan tanishtirish; multimediali ma'lumotlar bazalarini loyihalash va rivojlantirish bo'yicha nazariy bilim va amaliy ko'nikmalarni olish; multimediali ma'lumotlar bazasini loyihalashning asosiy bosqichlari, ma'lumotlar modellari (ierarxik, tarmoq va relyatsion), munosabatlarni normallashtirish tamoyillari to'g'risida bilimlarga ega bo'lish; nisbiy ma'lumotlar bazasini ichki tashkil qilish to'g'risida bilimlarni olish; mijoz-server texnologiyalari, zamonaviy sanoat ma'lumotlar bazalari va ularning rivojlanish istiqbollari bilan tanishish. Multimedia ma'lumotlarini saqlashning umumiy tushunchasi bilan talabalarni tanishtirish. Katta va ko'p o'lchovli multimediyali ma'lumotlar bazasini loyihalash va boshqarishning nazariy, tashkiliy va uslubiy masalalari bayonidan iborat.

"Multimediali ma'lumotlar bazasi" fanining vazifalari:

- ma'lumotlar bazasini boshqarish tizimlaridan foydalanish qobiliyati;
- video, audio, grafika, gipermatnli ma'lumotlarni saqlash bo'yicha umumiy ko'nikmalar;
- multimedia texnologiyalaridan foydalangan holda multimedia mahsulotlarini ma'lumotlar bazasida saqlash imkoniyati.

Fanni o'rganish jarayonida olingan bilimlar kelajakda talabalarga axborot madaniyatining umumiy darajasini oshirish muammolarini mustaqil hal qilishga imkon yaratishi kerak.

1-BOB.
MULTIMEDIA MA'LUMOTLAR BAZASI FANIGA KIRISH

1.1. Ma'lumotlar bazasi asoslari. Multimedia MBBT

Keng ma'noda **Ma'lumotlar bazasi (MB)** deganda real dunyoning aniq obyektlari haqidagi ma'lumotlar to'plamini tushirish mumkin. Lekin ma'lumotlar hajmi oshib borishi bilan bu masalalarni hal etish murakkablashadi. Yuzaga kelgan muammo obyekt va ma'lumotlarni strukturalash, ya'ni tizimga solish yo'li bilan hal qilinadi. *Obyekt*-bu mavjud va farqlanishi mumkin bo'lgan narsadir. Obyektlarga tegishli bir qator ma'lumotlar borki, ularning to'plami MB bo'la oladi. Masalan, har bir akademik-litsey yoki kasb-hunar kolleji-bu obyektlar bo'lsa, ulardagi o'quvchilar haqidagi ma'lumotlar to'plami MBga misol bo'la oladi.

Har qanday jiddiy MBning yaratilishi uning loyihasi tuzishdan boshlanadi. MB loyihalovchisining asosiy vazifasi obyektlar va ularni tavsiflovchi parametrlarni tanlash, ma'lumotlar orasidagi ma'lumotlarni o'rnatishdan iborat.

MBni yaratish jarayonida, foydalanuvchi ma'lumotlarni turli belgilar bo'yicha tartiblashga va belgilarning turli birikmalari bo'yicha zarur ma'lumotlarni tez topish uchun imkoniyatlar yaratilishiga harakat qiladi. Bu ishlarni ma'lumotlar strukturalangan (tuzilmalangan) bo'lgandagina bajarish mumkin.

Strukturalash-bu obyektlar va ma'lumotlarning o'zaro bog'lanishi tasvirlash usullari haqidagi kelishuvni kiritishdir.

1-misol: strukturalanmagan ma'lumotlar

Shaxsiy ish №16493; Aliyev Karim Ergashevich; tug'ilgan sana 1 yanvar 1979 yil; Shaxsiy ish №16498; Boqiyev Dilmurod Raxmatullayevich; tug'ilgan sana 5 dekabr 1985 yil; Shaxsiy ish №16595; Zokirov Anvar Rashidovich; tug'ilgan sana 15 may 1984 yil.

2-misol: strukturalangan ma'lumotlar

Shaxsiy ishi №	Familiyasi	Ismi	Otasining ismi	Tug'ilgan sana
16493	Aliyev	Karim	Ergashevich	01.01.79
16498	Boqiyev	Dilmurod	Raxmatullayevich	05.12.85
16595	Zokirov	Anvar	Rashidovich	15.05.84

Zamonaviy MB texnologiyasida MBni yaratish, unga xizmat ko'rsatish va foydalanuvchilarni MB bilan ishlashga imkon yaratish maxsus dasturiy uskunalar yordamida amalga oshiriladi. Bunday dasturiy uskunalar yordamida amalga oshiriladi. Bunday dasturiy uskunalar majmuasi *Ma'lumotlar bazasini boshqarish tizimlari (MBBT)* deb ataladi.

MBBT-MBni yaratish, uni dolzarb holatda ushlab turish, kerakli axborotni topishni tashkil etish va boshqa xizmat ko'rsatish uchun zarur bo'ladigan dasturiy va til vositalari majmuasidir.

Ko'pchilik MBlar jadval tuzilmasiga ega. Unda ma'lumotlar adresi satr va ustunlar kesishmasi bilan aniqlanadi. MBda ustunlar-*maydonlar*, satrlar esa *yozuvlar* deb ataladi. Maydonlar MBning tuzilmasini, yozuvlar esa, unda joylashgan ma'lumotlarni tashkil etadi.

Maydonlar-MB tuzilmasining asosiy elementlaridir. Ular ma'lum xususiyatlarga ega bo'ladilar. Har qanday maydonning asosiy xususiyati uning *uzunligidir*. Maydon uzunligi undagi belgilar soni bilan ifodalanadi.

Maydonning yana bir xususiyati, uning *nomidir*. Maydonda uning nomidan tashqari yana *imzo* xususiyati ham mavjud. Imzo-ustunning sarlavhasida aks ettiriladigan axborotdir. Uni maydon nomi bilan aralastirib yubormaslik lozim. Agar imzo berilmagan bo'lsa sarlavhada maydon nomi yozib qo'yiladi. Turli tipdagi maydonlar turli maqsadlarda ishlatiladi va turli xossalarga ega bo'ladi.

Maydonlarning xususiyati bilan tanishib chiqamiz:

1. **Oddiy matn maydoni**. Belgilar soni 255 dan oshmasligi kerak.

2. **MEMB-katta o'lchamli matn maydoni**. Belgilar soni 65535dan oshmasligi shart. Oddiy matn va MEMB maydonida hisob ishlarini bajarib bo'lmaydi.

3. **Sonli maydon**. Sonli ma'lumotlarni kiritishga xizmat qiladi va hisob ishlarini bajarishda foydalaniladi. Bu maydon 1,2,4,8 va 16 baytli bo'lishi mumkin.

4. **Sana va vaqt maydoni.** Bu maydon sana va vaqtni formatlangan holda saqlab qo'yish imkonini beradi (01.06.01 20:29:59). 8 bayt o'lchamga ega.

5. **«Pul birligi» nomi bilan ataluvchi maydon.** Bu maydondan hisob kitob ishlarini yuritishda foydalaniladi.

6. **Hisoblagich maydoni.** Bu maydon 4 bayt uzunlikka va avtomatik ravishda ma'lum songa oshib borish xususiyatiga ega. Ushbu maydondan yozuvlarni nomerlashda foydalanish qulaydir.

7. **Mantiqiy amal natijasini saqlovchi maydon.** Bu maydon «rost» (true) yoki «yolg'on» (false) qiymatni saqlaydi. Maydon o'lchami 1bayt.

8. **OLE-nomi bilan yurituvchi maydon.** Bu maydon Excel jadvalini, Word hujjatini, rasm, ovoz va boshqa shu kabi ma'lumotlarni ikkilik sanoq sistemasida saqlaydi. Maydon o'lchami 1Gbaytgacha.

9. **Gipersilka maydoni.** Bu maydon belgi va sonlardan iborat bo'lib, biror fayl yoki saytga yo'l ko'rsatadi.

10. **Qiymatlar ro'yxatidan iborat bo'lgan maydon.** Bu maydon bir qancha qiymatlardan iborat bo'lgan ro'yxatdan tanlangan aniq bir qiymatni saqlaydi.

Jadvallar orasidagi munosabatlar ishonchli ishlashi va bir jadvaldagi yozuv orqali ikkinchi jadvaldagi yozuvni topish uchun jadvalda alohida maydon – unikal maydon bo'lishini ta'minlash kerak.

Unikal maydon – bu qiymatlari takrorlanmaydigan maydondir.

Misol sifatida talabalar haqidagi ma'lumotlarni saqlovchi ma'lumotlar bazasining bir qismini keltiramiz.

Maydon nomi	Maydon xususiyati	Maydon hajmi
Talabaning ID	Hisoblagich maydoni	4 bayt
Talaba F.I.Sh.	Oddiy matnli maydon	255 belgi
Talabaning tug'ilgan joyi haqida	Oddiy matnli maydon	255 belgi
Talabaning tug'ilgan kuni	Sana va vaqt maydoni	8 bayt
Talabaning kursi	Qiymatlar ro'yxatidan iborat bo'lgan maydon	
Talabaning rasmi	OLE-nomi bilan yuritiluvchi maydon.	1G bayt
Talaba haqida qo'shimcha ma'lumotlar	MEMB-katta o'lchamli matn maydoni	65535 belgi

Multimediali ma'lumotlar

Multimediali taqdimot – bugungi kunda axborot taqdim etishning yagona va eng zamonaviy shakli hisoblanadi. Bu matnli ma'lumotlar, rasmlar, slayd-shou, diktir jo'rligidagi ovoz bilan boyitilgan, videoparcha va animatsiya, uch o'lchamli grafika tarzidagi dasturiy ta'minot bo'lishi mumkin. Taqdimotning ma'lumot taqdim etishning boshqa shakllaridan asosiy farqi ularning mazmunan boyitilganligi va interfaolligidir, ya'ni belgilangan shaklda o'zgarishga moyilligi va foydalanuvchi faoliyatiga munosabatini bildirishidir. Bundan tashqari, taqdimot sizning saytingiz kaliti ham bo'lishi mumkin. Ya'ni internetga chiqish imkoniyati mavjud bo'lgan paytda sichqonchani bir martagina bosish orqali taqdimotni ko'rib, kompaniya saytidan eng yangi ma'lumotni olish mumkin.

Multimediali texnologiya (multi – ko'p, media – muhit) bir vaqtning o'zida ma'lumot taqdim etishning bir necha usullaridan foydalanishga imkon beradi: matn, grafika, animatsiya, videotasvir va ovoz.

Multimediali texnologiyaning eng muhim xususiyati interfaolik – axborot muhiti ishlashida foydalanuvchiga ta'sir o'tkaza olishga qodirligi hisoblanadi.

So'nggi yillar davomida ko'plab multimediali dasturiy mahsulotlar yaratildi va yaratilmoqda: ensiklopediyalar, o'rgatuvchi dasturlar, kompyuter taqdimotlari va boshqalar.

Kompyuter taqdimotlari (Kompyuter vositasida tayyorlangan taqdimotlar) ma'ruza, doklad yoki boshqa chiqishlarda odatda ko'rgazmali namoyish etish vositasi sifatida plakatlar, qo'llanma, laboratoriya tajribalaridan foydalaniladi. Bu maqsadda diaproektorlar, kodoskoplar, grafik tasvirlarni ekranda namoyish etuvchi slaydlardan foydalaniladi. Kompyuter va multimediali proektorning paydo bo'lishi ma'ruzachi nutqini ovoz, video va animatsiya jo'rligida sifatli tashkil etishning barcha zaruriy jihatlarni o'zida mujassam qilgan ko'rgazmali materiallarni taqdimot sifatida tayyorlash va namoyish etishga imkon berdi.

Ma'lumki, inson ma'lumotning ko'p qismini ko'rish (~80%) va eshitish (~15%) organlari orqali qabul qiladi (bu avvaldan aniqlangan va kino hamda televideniya undan samarali foydalaniladi). Multimediali texnologiyalar ushbu muhim sezgi organlarining bir

vaqtda ishlashiga yordam beradi. Dinamik vizual ketma-ketlik (slyd-shou, animatsiya, video)ni ovozli tarzda namoyish etish orqali insonlarning e'tiborini ko'proq jalb qilamiz. Shundan kelib chiqib, multimediali texnologiyalar axborotni maksimal samarali tarzda taqdim etishga imkon beradi.

Videodan farqli ravishda multimediali texnologiyalar axborotlarni boshqarishga imkon beradi, ya'ni interfaol bo'lishi mumkin. Multimediali taqdimot ma'lumotni to'g'ridan to'g'ri qabul qilishni ta'minlaydi. Foydalanuvchi taqdim etilayotgan barcha ma'lumotlarni ko'radi va o'zini qiziqtirgan qismlaridan foydalana oladi. Ma'lumotni qabul qilish katta mehnat va vaqt talab qilmaydi.

Ma'lumot taqdim etishning boshqa shakllaridan farqli ravishda multimediali taqdimot bir necha o'n minglab sahifa matn, minglab rasm va tasvirlar, bir necha soatga cho'ziladigan audio va video yozuvlar, animatsiya va uch o'lchamli grafikalarini o'z ichiga olgan bo'lishiga qaramay, ko'paytirish xarajatlarining kamligini va saqlash muddatining uzoqligini ta'minlaydi.

Multimediali ma'lumotlarni yaratish texnologiyasi

Hozirgi vaqtda ko'plab kompaniya va firmalarda seminarlar, uchrashuvlar, treninglar va boshqa tadbirlarni o'tkazish uchun turli kompyuter texnologiyalaridan foydalanilmoqda. Ma'lumot mazmunga boy, esda qoladigan va ko'rgazmali bo'lishi uchun ko'proq multimedia texnologiyalari ishlatiladi. Bular matn, grafika va ovoz kabi ma'lumotning turli shakllarini qayta ishlashga imkon beruvchi multimedia apparat vositalari bo'lish bilan birga amaliy dasturlar paketlari ham.

Multimedia bir necha ta'rifga ega:

- *Multimedia* – turli ma'lumotlarni ishlab chiqish, ishga tushirish, qayta ishlash vositalarini qo'llash tartiblarini ta'riflovchi texnologiya;

- *Multimedia* – kompyuter apparat ta'minoti (kompyuterda kompakt-disklar o'qish qurilmasi – CD-Rom Drive, uning yordamida ovozli va videoma'lumotni eshittirishga yordam beradigan ovoz va videoplata, joystik va boshqa maxsus qurilmalarning mavjudligi);

- *Multimedia* – bu bir necha ma'lumot taqdim etish vositalarining bir tizimga birlashishi. Odatda multimedia deganda matn, ovoz, grafika, multiplikatsiya, videotasvir va fazoviy modellashtirish kabi ma'lumot

taqdim etish vositalarining kompyuter tizimidagi birlashishi tushuniladi. Bunday vositalarning birlashishi ma'lumot qabul qilishning yangi sifatlari darajasini ta'minlaydi: inson passiv ravishda mahliyo bo'libgina o'tirmasdan, balki faol ishtirok etadi ham. Multimedia vositalari bilan ishlovchi dasturlar ko'p modallidir, ya'ni ular bir necha sezgi organlariga bir vaqtda ta'sir qilgani uchun auditoriyaning qiziqishi va e'tiborini tortadi.

Multimedia ilova mazmuni muallif tomonidan ssenariy tayyorlash paytida atroflicha o'ylanadi va texnologik ssenariyni ishlab chiqishda aniqlashtiriladi. Ta'limiy ma'lumot taqdim etishning an'anaviy shakli – matn va statik grafika ko'p asrlik tarixga ega bo'lsa, multimedidan foydalanish tajribasi yillar bilan o'lchanadi.

Illustratsiya, jadval va chizmalar mavjud bo'lgan animatsiya elementlari va ovoz jo'rligida taqdim etilayotgan chiroyli bezalغان multimediali ilova o'rganilayotgan materialni qabul qilishni osonlashtiradi, tushunish va eslab qolishga yordam beradi, ta'lim oluvchining bilim olishga bo'lgan faolligini oshirib, predmetlar haqida aniqroq va to'laroq tushunchaga ega bo'lishini ta'minlaydi.

Sifatli multimedia ilovalari ishlab chiqishga yo'naltirilgan turli-tuman, bir-biridan farqli texnologik usullar mavjud. Ushbu ilovalarni yaratish va ulardan foydalanishda bir qator asosiy texnologik tavsiyalarga amal qilish kerak.

Multimedia ilovalarini yaratish uchun asos sifatida materialni elementlarga bo'lish va ierarxiya tarzida ko'rgazmali ravishda taqdim etishga asoslangan holda tizimlashtirish usulini o'zida mujassam etgan materialni qamrab olish modeli olish mumkin.

Multimedia ilovasini loyihalashning boshlang'ich bosqichida materialni qamrab olish modeli quyidagilarga imkon beradi:

- material mazmunini aniq belgilash;

- mazmunni ko'rgazmali, aniq va ochiq tarzda taqdim etish;

- multimedia ilova komponentlari tarkibini aniqlash.

Kompyuter ekranidagi ma'lumotni vizuallashtirish usullarini ishlab chiqishda psixologiya yutuqlarini hisobga olish kerakligi bir qator umumiy tavsiyalarni shakllantirishga yordam beradi:

- ekrandagi ma'lumot tizimga keltirilgan bo'lishi kerak;

- vizual ma'lumot davriy ravishda tovushli ma'lumotga almashib turishi kerak;

- rang yorqinligi yoki tovush balandligi davriy ravishda o'zgarib

turishi kerak;

- vizuallashtirilayotgan material mazmuni juda ham sodda yoki juda ham murakkab bo'lmashligi kerak.

Ekrandagi kadr formatini ishlab chiqishda va yaratishda ko'rish maydonini tashkil etishni belgilovchi obyektlar orasida maqsad va munosabat borligini hisobga olish tavsiya etiladi. Obyektlarni quyidagicha joylashtirish tavsiya qilinadi:

- bir-biriga yaqin, ya'ni ko'rish maydonida obyektlar bir-biriga qanchalik yaqin bo'lsa, ular shunchalik katta ehtimol bilan yagona, yaxlit obrazni tashkil qiladi;

- jarayonlar o'xshashligi, ya'ni obrazlarning o'xshashligi va yaxlitligi qanchalik ko'p bo'lsa, ular shunchalik katta ehtimol bilan uyg'unlashadi;

- davom etish xususiyatlarini hisobga olgan holda, ya'ni ko'rish maydonida tartibli ketma-ketlikning davom etishiga mos keladigan joylarda joylashgan elementlar qanchalik ko'p bo'lsa, ular shunchalik katta ehtimol bilan yaxlit yagona obrazga birlashadi;

- obyektlar shakli, harf va raqamlar o'lchami, ranglarning to'qligi, matnning joylashishi va boshqalarni tanlashda predmet va fonni belgilashning o'ziga xosligini hisobga olgan holda;

- vizual axborotlarni ortiqcha detallar, yorqin va kontrast ranglar bilan bezamagan holda;

- xotirada saqlash uchun mo'ljallangan materialning rangini o'zgartirish, tagiga chizish, shrift o'lchamini va stilini o'zgartirish orqali ajratish.

1.2. Multimediali ma'lumotlar turlari

Quyida multimedia ilovalari haqida asosiy tushunchalar va ularning ko'rinishlari to'g'risida ma'lumot berilgan.

Taqdimot/prezentatsiyalar

Taqdimot (ing. *presentation*) – audiovizual vositalardan foydalanib ko'rgazmali shaklda ma'lumot taqdim etish shakli. Taqdimot yagona manbaga umumlashgan kompyuter animatsiyasi, rafika, video, musiqa va ovozni o'zida mujassam etadi. Odatda taqdimot ma'lumotni qulay qabul qilish uchun syujet, ssenariy va ukturaga ega bo'ladi.

Animatsion rolklar

Animatsiya – multimediali texnologiya, tasvirning harakatlanayotganligini ifodalash uchun tasvirlarning ketma-ket namoyishi. Tasvir harakatini tasvirlash effekti sekundiga 16 ta kadrdan ortiq videokadrlarning almashinishida hosil bo'ladi.

O'yinlar

O'yin dam olish, ko'ngil ochish ehtiyojlarini qondirish, organizmdagi zo'riqishni yo'qotish hamda ma'lum malaka va ko'nikmalarni rivojlantirishga yo'naltirilgan multimedia ilovalaridandir.

Videoilovalar

Videoilovalar – harakatlanuvchi tasvirlar ishlab chiqish texnologiyasi va namoyishi. Video tasvirlarni o'qish qurilmalari – videofilmlarni boshqaruvchi dasturlar.

Multimedia-galereyalar

Galereyalar – ovoz jo'rligidagi harakatlanuvchi suratlar to'plami.

Audioilovalar

Ovozli fayllarni o'quvchi qurilmalar – raqamli tovushlar bilan ishlovchi dasturlar. Raqamli tovush – bu elektrik signal amplitudasining diskret sonlar bilan ifodalanishi.

Web uchun ilovalar

Web uchun ilovalar – bu alohida veb-sahifalar, uning tarkibiy qismlari (menyu, navigatsiya v.b.), ma'lumot uzatish uchun ilovalar, ko'p kanalli ilovalar, chatlar va boshqalar.

Multimedia ilovalari yaratish texnologiyasini o'rganishda ularning qanday yaratilishini ifodalovchi ssenariy ishlab chiqiladi. Bundan kelib chiqib, har bir multimedia ilovasi turli tarkibiy qismlar (turli mavzular)dan tashkil topadi, degan mantiqiy xulosaga kelishimiz mumkin. Multimedia ilovalari tarkibini quyidagi qismlarga bo'lish mumkin: yaratilayotgan multimedia ilovasi uchun mavzu tanlash, ish maydonini belgilash (masshtab va fon), kadrlar, qatlamlardan foydalanish, turli shakllar simvollarini yaratish, dasturlash tilida o'zgaruvchilar kiritish va skriptlar yozish, tovushli fayllar bilan ishlash, matn qo'shish, effektlar yaratish, rasmlardan foydalanish va import qilish, kutubxonadagi tayyor komponentlardan foydalanish, navigatsiyani yaratish, matn razmetkasi tillari va skriptlash tillaridan foydalanish.

O'z navbatida multimedia ilovalarini quyidagi turosti turlarga bo'lish mumkin.

Taqdimot:

- Chiziqli taqdimot – murakkab grafika, videoqo'yilma, ovoz jo'rligidagi va navigatsiya tizimiga ega bo'lmagan dinamik rolik.

- Interfaol taqdimot – ierarxik tamoyillar bo'yicha tuzilmaga keltirilgan va maxsus foydalanuvchi interfeysi orqali boshqariladigan multimediali komponentlar to'plami.

Animatsiya:

- Kadrlar animatsiyasi – tasvirlar harakati taassurotini beradigan suratlarining kadrlar bo'yicha almashishi. Dasturiy animatsiya – dasturlangan amallar ketma-ketligi yordamida tasvirlar almashadigan animatsiya (ya'ni algoritim va o'zgaruvchilar yordamida). Asosiy obyektlarni chizish qo'lda amalga oshiriladi yoki biror bir kolleksiya yoxud galereyadan import qilib olinadi, shundan keyingina unda qaysidir dasturlash tili imkoniyatlari ishga solinadi.

O'yinlar:

Ko'ngil ochuvchi o'yinlar – foydalanuvchiga bo'sh vaqtini o'tkazishga imkon beradigan dasturlar. O'rgatuvchi o'yinlar – taqdim etilgan yengil o'yin shaklida foydalanuvchiga u yoki bu soha bo'yicha o'z malaka va ko'nikmalarini oshirishga yordam beradigan dasturlar.

Video o'qish qurilmalari:

Film kadrlari harakatini shakllantirish – harakat taassurotini uyg'otuvchi rasmlar, kadrlar ketma-ketligini, tasvirlarni tayyorlash va joylashtirish. Videotasvirlar oqimini o'qish qurilmasi – videooqim formatlari avi, mpeg v.b.ni o'z ichiga olgan o'qish qo'rilmasini shakllantirish, shundan keyingina ushbu oqimni boshqarish imkoniyati tug'iladi (masalan, videofaylni ishga tushirish, pauza, video boshiga tez o'tish kabi buyruqlarni ishlatish).

Multimedia-galereyalar:

Tasvirlarning kadrlarda almashishi – tasvirlarning belgilangan vaqt oralig'ida almashish tartibi. Panorama – katta ochiq maydonni erkin tomosha qilishga imkon beruvchi keng va katta plandagi manzara. Interfaol galereya – foydalanuvchiga boshqarish imkonini beradigan galereya (tasvirlar bo'yicha harakat).

Tovush o'qish qurilmalari:

Bitta ovozli fayl o'qish qurilmasi – wav, mp3 va boshqa formatdagi ovozli faylni multimedia ilovaga qo'shish va uni qaytadan

eshitish.

Turli ovozli fayllarni o'qish qurilmasi – bitta ovozli fayl o'qish qurilmasiga o'xshash, lekin bajarish ketma ketligini o'zgartirish imkoniyati qo'shilgan holda. Virtual musiqa instrumentlari – haqiqiy musiqa instrumentlari imitatsiyasi.

web uchun ilovalar:

Bannerlar – internetdagi reklama xarakteridagi tasvir yoki matn bloki. U reklama beruvchining veb-saytiga yoki mahsulot yoxud xizmat turi atroflicha bayon qilingan sahifalarga giper murojaatdan iborat.

Bannerlar tashrifchilarni jalb etish uchun, imidjni shakllantirish yoki shu resursni siljitish uchun turli Internet resurslarda joylashtiriladi. Ma'lumot uzatish ilovalari (masalan, mehmonlar kitobi).

1.3. Multimedia ma'lumotlarni tayyorlash va saqlash

Yaratuvchi gipermatn sahifalarini yaratishda ishlatiladigan dasturni tanlab olishi kerak. To'liq funksional multimedia dasturlarini yaratishga imkon beruvchi bir qator kuchli multimedia yaratish vositalari mavjud. Macromedia Director, Macromedia Flash yoki Authoware Professional kabi paketlar yuqori darajadagi professional va qimmat vositalar hisoblanadi, shu bilan birga FrontPage, mPower 4.0, HyperStudio 4.0 va Web Workshop Pro ularning oddiyroq va arzonroq analoglari hisoblanadi. PowerPoint va matn muharrirlari (masalan, Word) kabilardan ham chiziqli va chiziqsiz multimediali resurslar yaratishda foydalanish mumkin. Borland Delphi ham multimediali ilovalar ishlab chiqish vositasi hisoblanadi. Sanab o'tilgan vositalar oson o'qish va tushunish mumkin bo'lgan to'liq hujjatlar bilan ta'minlangan. Albatta, yana ko'plab boshqa ishlab chiqish vositalari borki, sanalganlar o'miga ulardan foydalanib ham bir xil natijaga erishish mumkin. Hozirgi kunda multimedia ilovalari yaratish texnologiyalarini o'rgatuvchi avtomatlashtirilgan tizimlar juda kam, ularni topishning iloji ham yo'q. Ushbu mavzuda darslar, kitoblar va maqolalar to'plamiga ega bo'lgan Internet tarmog'i sahifalari ham shunday tizimlarga o'xshaydi. Bunday saytlarning asosiy qismi "Multimedia elementlari yaratish uchun flash darslari" yoki Macromedia Directorda multimedia yaratish" mavzulariga yo'naltirilgan.

Maktabda foydalanilayotgan multimedia vositalari ta'rifiga rasmiy yondashuv shundan darak beradiki, har xil turdagi axborotlarni ta'limiy faoliyatga olib kirishi mumkin bo'lgan har qanday vosita multimedia vositasi bo'lishi mumkin. Biroq ko'p hollarda multimedida vositalari to'g'risida so'z borganda kompyuter va uning atrofidagi jihozlar tushuniladi. Shuningdek, maktabda o'qituvchilar va o'quvchilar tomonidan nafaqat matnlar yoki tasvirlar uchun qo'llaniladigan, balki audio- yoki to'g'ridan-to'g'ri boshqa axborotlar bilan ishlash imkonini beradigan multimedia vositalarini sanab o'tish joizdir. Turli yillarda maktab ta'limida mutaxassislarni samarali tayyorlash maqsadini ko'zlovchi, umumiy o'rta ta'lim tizimida sifatli axborot ta'minotiga erishishga yo'naltirilgan har xil vositalar kirib keldi. Bugungi kunda maktablarda quyidagilarni uchratish mumkin:

- Ovoz yozib olish va ularni tinglash uchun vositalar (elektrofonlar, magnitofonlar, CDdan o'quvchilar);
- Telefon, telegraf va radio aloqa vositalari va tizimlari (telefon apparatlari, faksimil apparatlari, teletayplar, telefon stansiyalari, radioaloqa tizimlari);

Televidenie, radioeshitirish (tele- va radiopriyomniklar, o'quv televidenie va radio, DVD) tizimi va vositalari, optik va proeksion kino va fotoapparaturalar (fotoapparatlarni, kinokameralar, diaproektorlar, kinoproektorlar, epidiaskoplar), axborotlarni va hujjatlarni ko'paytirish va saqlash uchun mo'ljallangan poligrafiya, nusxa olish, ko'paytirish va boshqa texnikalar (rotaprintlar, kserokslar, rizograflar, mikrofilmlar olish tizimi), axborotlarni qayta ishlash va saqlash, elektron ko'rinishini taqdim etishga mo'ljallangan kompyuter vositalari (kompyuterlar, printerlar, skanerlar, grafiklar hosil qiluvchi), aloqa kanallari orqali axborotlarni uzatishni ta'minlovchi telekommunikatsion tizimlar (modemlar, o'tkazish tarmoqlari, sputnik, optik to'lqinlar, radioreleylar va axborotlarni uzatishga mo'ljallangan boshqa turdagi aloqa kanallari). Texnik vositalarning ta'lim tizimiga kirib kelishi ta'limiy faoliyatda axborotlarni ovozli, matnli, foto va video tasvirlar tarzida taqdim etish imkonini yaratdi. Bunday vositalarga ko'p hollarda murakkab texnik va texnologik jihatlari tufayli multimedia vositalari sifatida qaraladi.

Kompyuterning ta'lim sohasiga kirib kelishi axborotlarni qayta ishlashning universal vositasi sanaladi. Uning universalligi bir tomondan har xil tipdagi axborotlarni qayta ishlash imkoniga egaligi

bilan belgilansa (multimedia axborotlarini), boshqa tomondan bir xil tipdagi axborotlar bilan bir qator operatsiyalarni bajaradi. Shu tufayli kompyuter o'zining atrofidagi qator vositalar bilan ta'limdagi multimedia-vositalarining barcha funksiyalarini ta'minlash imkoniyatiga ega. Maktab ta'limida qo'llanilayotgan kompyuterlar markasi, modeli, yaratilgan vaqti va qo'llanilishi sohasidan qat'i nazar quyidagilardan iborat umumiy xususiyatga ega:

- Yagona foydalanuvchi kompyuter vositasidan ish jarayonida faqat bir kishi foydalanadi; shunday bo'lsada, axborotlarni qayta ishlashda bir necha operatsiyalarni bir vaqtning o'zida qo'llanish imkoniyati cheklanmaydi;
- Matn, raqamli ma'lumotlar, grafik tasvirlar, ovoz va boshqalar (multimedia axborotlar)dan iborat axborotlarni qayta ishlash, saqlash, taqdim etish va uzatish imkonini;
- Foydalanuvchi bilan yagona muloqot tilining mavjudligi;

Multimedia qurilmalarining har xil turdagi apparatlar bilan birgalikda qo'llanilishi shaxsiy kompyuterlarning har xil tipdagi axborotlarni qayta ishlash, saqlash, taqdim etish va uzatish imkoniyatlarini oshiradi;

Axborotlarni qayta ishlashni maxsus kompyuter dasturlari orqali amalga oshirish kompyuterni har xil tizimli vazifalar, amaliy topshiriqlarni hal etish, inson faoliyatini axborotlashtirish uchun yo'naltirishga mo'ljallangan. Multimedia texnologiyasi har xil turdagi axborotlarning mazmun va uyg'unligi ta'minlangan holda integratsiyalashuvini ta'minlaydi. Bu kompyuter yordamida har xil shakldagi axborotlarni taqdim etish imkonini beradi:

- rasmlar, chizmalar, kartalar va slaydlardan nusxa olish orqali hosil qilingan tasvirlar;
- ovoz yozish, ovoz effektlari va musiqalar;
- video, murakkab videoeffektlar;
- animatsiyalar va animatsiyali imitatsiyalar.

O'z navbatida zamonaviy kompyuter multimedia vositalari jadal rivojlanayotgan kompyuter telekommunikatsiyalari bilan bog'liq. Barcha kompyuter tarmoqlarida e'lon qilingan axborot resurslari amaliy jihatdan multimedia resurslari sanaladi. Aksariyat multimedia resurslari va texnologiyalar telekommunikatsion tartibda ishlashga mo'ljallanmoqda. Maktabda multimedia resurslari va

texnologiyalaridan foydalanishda telekommunikatsiya tarmoqlarini qo'llash bir qator imkoniyatlarni yuzaga chiqarishga olib kelmoqda:

- o'quv-uslubiy multimedia axborotlariga kirish imkonini kengaytiradi;
- o'quvchilarda kommunikativ malakalarni, muomala madaniyatini, multimedia axborotlarini izlash uquvini shakllantiradi;
- tezkor maslahat yordamini tashkil etadi;
- mustaqil ta'lim olish uchun individual ta'lim bazasini rivojlantiradi;
- aniq vaqt birligida virtual o'quv mashg'ulotlari (seminarlar, ma'ruzalar) o'tkazilishini ta'minlaydi;
- masofaviy ta'limni tashkillashtiradi;
- hamkorlikdagi tadqiqot loyihalarini tashkil etishni uyushtirish;
- ilmiy tadqiqot faoliyatini modellashtirish;
- o'qituvchilarning tarmoqdagi o'zaro hamjamiyatini shakllantirish;
- o'quvchilarning tarmoqdagi uyushmasini shakllantirish.

Nazorat savollari

1. Chiziqli taqdimot deganda nimani tushunasiz?
2. Kadrlar animatsiyasi nima?
3. Multimedia ma'lumotlar bazasi qanday xarakteristikalariga ega bo'lishi kerak.
4. Multimedia-galereyalar larning afzalliklari va kamchiliklari.
5. Obyektga yo'naltirilgan MBBT ning asosiy komponentlari

2-BOB.

MULTIMEDIA MA'LUMOTLAR MODEL TUSHUNCHASI

2.1. Ierarxik, taqsimlangan va relyatsion ma'lumotlar modeli

MB bitta yoki bir necha modellarga asoslangan bulishi mumkin. Xar kaday modelga uzining xossalari (parametrlari) bilan biror amal (ish) bajarsa bo'ladi. MB modellarning quyidagi asosiy turlari mavjud:

Relyatsion (lotin tilidagi relatio-munosabat so'zidan olingan) modelda ma'lumotlarni saqlash uni tashkil etuvchi qismlari orasidagi munosabatlarga asoslangan. Eng sodda holda u ikki o'lchovli massiv yoki jadvaldan iborat bo'ladi. Murakkab axborot modellari ana shunday jadvallarning o'zaro bog'langan to'plamidan iborat.

MBning **ierarxik** modeli pastki pog'onadagi yuqori pog'onadagiga bo'ysinish tartibida joylashgan elementlar to'plamidan iborat bo'ladi va ag'darilgan daraxt(graf)ni tashkil etadi. Ushbu model **sath, tugun, bog'lanish** kabi parametrlar bilan tavsiflanadi. Uning ishlash tamoyili shundayki, quyi sathdagi bir necha tugunlar bog'lanish yordamida yuqoriroq sathdagi bitta tugun bilan bog'langan bo'ladi. Tugun – bu ierarxiyaning berilgan sathida joylashgan elementning axborot modelidir.

MBning semantik tarmoq modeli ierarxik modelga o'xshashdir. U ham tugun, sath, bog'lanish kabi asosiy parametrlarga ega. Lekin semantik tarmoq modelida turli sathdagi elementlar orqali «erkin», ya'ni «har biri hamma bilan» ma'noli bog'lanish qabul qilingan.

Taqsimlangan ma'lumotlar bazasi va uning arxitekturasi.

Ma'lumotlar bazasini hisoblash tizimlarida qo'llanilishi ma'lumotlarni qayta ishlashning eski usullarini almashishiga sabab bo'ldi. Yangi usulda har bir ilova uchun alohida ma'lumotlar to'plami aniqlangan va qo'llab – quvvatlangan. Eski usulda barcha ma'lumotlar markazlashgan holda aniqlangan va qo'llab quvvatlangan. So'nggi paytlarda tarmoq va ma'lumotlar almashishining texnologiyalari shiddat bilan rivojlanmoqda. Bu internet tarmog'i, mobil va simsiz

hisoblash vositalari, hamda " intellektual " qurilmalarni paydo bo'lishi bilan bog'liq. Taqsimlangan ma'lumotlar bazasining texnologiyasi ma'lumotlarni qayta ishlashning markazlashgan turidan teskarisi bo'lgan markazlashmagan turiga o'tishga zamin yaratmoqda. Taqsimlangan ma'lumotlar bazasini boshqarish tizimlari texnologiyalari ma'lumotlar bazasi sohasidagi eng katta yutuqlardan biri hisoblanadi.

Taqsimlangan ma'lumotlar bazasi bilan bog'liq muammolarni tahlil qilishdan oldin taqsimlangan ma'lumotlar bazasi nima ekanligini aniqlab olish zarur.

Taqsimlangan ma'lumotlar bazasi - bu o'zaro matiqan bog'langan va kompyuter tarmog'ida fizik jihatdan tarqoq joylashgan taqsimlangan ma'lumotlar to'plamidir.

Bu ta'rifdan taqsimlangan MBBT ni ta'ri fi kelib chiqadi. **Taqsimlangan MBBT.** Taqsimlangan ma'lumotlar bazasini boshqarish uchun mo'ljallangan va foydalanuvchilarga taqsimlangan ma'lumotlarga oson murojaat usulini ta'minlab beruvchi dasturiy kompleks.

2.2. Taqsimlangan ma'lumotlar bazasi.

Ma'lumotlarni qayta ishlashning bu kesishuvchi va xatto takrorlanuvchi ma'lumotlar saqlanadigan bir nechta serverlardan foydalanishni talab etadi. Bunday ma'lumotlar bazasi bilan ishlash uchun taqsimlangan ma'lumotlar bazasini boshqarish tizimlari (TMBBT) ishlatiladi.

Taqsimlangan ma'lumotlar bazasini boshqarish tizimlari (Taqsimlangan MBBT) bir qancha fragmentlarga bo'lingan va mantiqiy jihatdan yagona bo'lgan ma'lumotlar bazasidan iborat. Ma'lumotlar bazasining har bir fragmenti alohida MBBT boshqaruvchi ostida va o'zaro aloqa tarmog'i orqali bog'langan bitta yoki bir nechta kompyuterda saqlanadi. Tarmoqdagi har bir tugun foydalanuvchilarning lokal saqlanayotgan ma'lumotlarga bo'lgan so'rovlarini o'zi mustaqil qayta ishlash (ya'ni, har bir uzal ma'lum darajadagi avtonomlikka ega), hamda tarmoqning boshqa kompyuteridagi ma'lumotlarni qayta ishlash imkoniga ega.

Foydalanuvchilar taqsimlangan ma'lumotlar bazasiga dastur yordamida murojaat etadi. Dasturlar boshqa tugundagi ma'lumotlarga murojaatni talab etmaydigan (lokal dasturlar) va talab etadigan (global dasturlar) dasturlarga bo'linadi. Taqsimlangan MBBT da kamida bitta

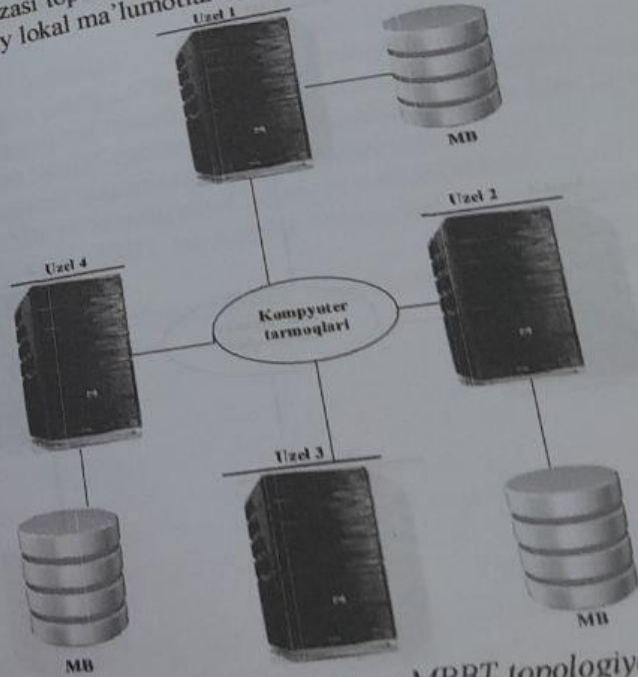
global dastur bo'lishi kerak, shuning uchun ixtiyoriy TMBBT quyidagi xarakteristikalariga ega bo'lishi kerak.

- mantiqan bo'lingan ma'lumotlar to'plami mavjud bo'lishi kerak.
- saqlanadigan ma'lumotlar bir qancha fragmentlarga bo'lingan.
- ma'lumotlar fragmentining replikatsiyasi mavjud bo'lishi mumkin.

Fragmentlar va ularning nusxalari turli tugunlarga taqsimlanadi.

- tugunlar bir - biri bilan tarmoq orqali bog'lanadi.
- har bir tugundagi ma'lumotlar bir qancha fragmentlarga taqsimlanadi.
- MBBT har bir tugundagi lokal dasturlarni avtonom ishlashini ostida amalga oshadi.
- MBBT har bir tugundagi lokal dasturlarni avtonom ishlashini qo'llab - quvvatlash imkoniga ega.
- har bir tugundagi MBBT kamida bitta global dasturni qo'llab - quvvatlaydi.

Misol tariqasida 1-rasmdagi ko'rsatilgan taqsimlangan ma'lumotlar bazasi topologiyasidagi kabi tizimning har bir tugunida o'zining xususiy lokal ma'lumotlar bazasi mavjud bo'lishi shart emas.

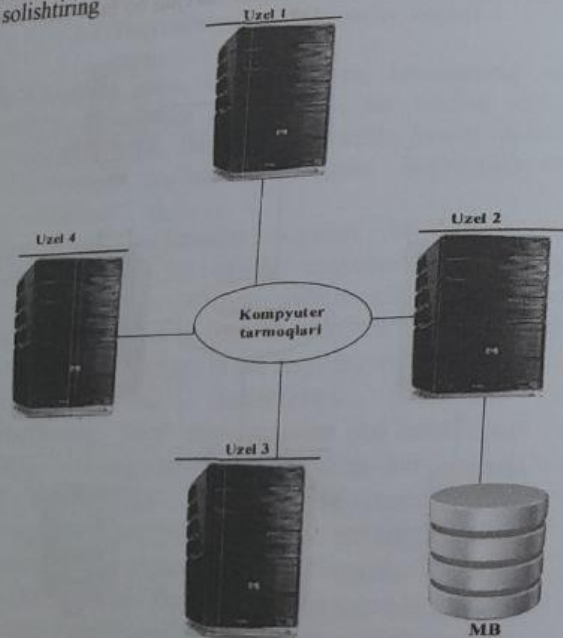


2.1-rasm. Taqsimlangan MBBT topologiyasi

Taqsimlangan ma'lumotlarni qayta ishlash

Taqsimlangan MBBT bilan taqsimlangan qayta ishlash vositalari o'rtasidagi farqni anglash muxim hisoblanadi.
Taqsimlangan qayta ishlash. Markazlashgan ma'lumotlar bazasini qayta ishlashni turli kompyuterlardan turib amalga oshirilishidir.

Taqsimlangan MBBT ta'riflashda bu tizim fizik jixatdar taroqda taqsimlangan ma'lumotlar bilan ishlashi asosiy hisoblanadi. Agar ma'lumotlar markazlashgan holda saqlanayotgan bo'lsa, u holda xatto ixtiyoriy foydalanuvchi bu ma'lumotlarga tarmoqdagi ixtiyoriy kompyuterdan murojaat etayotgan bo'lsada bu tizim taqsimlangan qayta ishlashni qo'llab-quvvatlovchi hisoblanadi va taqsimlangan MBBT sifatida qaralmaydi. Taqsimlangan qayta ishlash topologiyasi sxemasi 2-rasmda tasvirlangan. Bu sxemada tasvirlangan 2-tugundagi markazlashgan ma'lumotlar bazasini 1-rasmdagi ma'lumotlar bazasi bilan solishtiring



2.2-rasm. Taqsimlangan qayta ishlash topologiyasi.

2.1-jadval. Taqsimlangan MBBT avfzalliklari va kamchiliklari

Avfzalligi	Kamchiliklari
Strukturani akslantirish	Murakkablikning yuqoriligi
Bo'linuvchan va mahalliy avtonomlik	Qiymatning ortishi
Ma'lumotlarga kirishning yuqoriligi	Ximoyada muammolarning mavjudligi
Ishonchlilikning yuqoriligi	Ma'lumotlar butunligini nazorat qilishda qiyinchilik mavjudligi
Ishlab chiqarishning yuqoriligi	Standartlarning mavjud emasligi
Iqtisodiy foydalar	Mahorat etmasligi
Tizimning modulliligi	Ma'lumotlar bazasi ishlab chiqish jarayonlarini qiyinlashtirish

Nazorat savollari

1. Teskari ravishda ma'lumotlarni qayta ishlashning markazlashmagan usuliga o'tishga nima zamin yaratdi?
2. Taqsimlangan ma'lumotlar bazasi va taqsimlangan MBBT nima?.
3. Relyatsion MBBT qanday xarakteristikalariga ega bo'lishi kerak?
4. Ma'lumotlarni taqsimlangan qayta ishlash nima.
5. Ieraxik MBBT larning afzalliklari va kamchiliklari.
6. Taqsimlangan MBBT lar qanday funksional imkoniyatlarga ega bo'lishi kerak?
7. Obyektga yo'naltirilgan MBBT ning asosiy komponentlari

3-BOB.
MULTIMEDIA MB LOYIHALASHDA RELYATSION
YONDOSHUV

3.1. MMB konseptual va mantiqiy loyahasiga relyatsion yondoshuv

Relyatsion ma'lumotlar bazasi nazariyasi 70-yillar boshida munosabatlarning matematik nazariyasi asosida Kodd tomonidan ishlab chiqilgan. Relyatsion ma'lumotlar bazasida barcha ma'lumotlar jadval ko'rinishida saqlanadi, bunda barcha amallar jadvallar bilan ishlashga keltiriladi. Bu nazariyada asosiy tushunchalar jadval, munosabat, misra, ustun, kalitlardan iborat. Xar qanday ma'lumotlar bazasining asosi jadvallardir. Jadval misralar va ustunlardan tashkil topadi va ma'lumotlar bazasida noyob nomga ega. Ma'lumotlar bazasi, ular oralaridagi bog'lanish mos keluvchi ustma-ust tushuvchi) maydonlar yordamida o'rnatiladigan, sanoqsiz jadvallardan tuzilgan. Jadvallarning xar birida qandaydir biror bir tur (guruh) obyektlar to'g'risida ma'lumot bor.

Relyatsion model

Ma'lumotlarning relyatsion modeli 1970 yil IBM firmasining xodimi Edgar Kodd tomonidan taklif etilgan. Relyatsion ma'lumotlar bazasining asosiy g'oyasi ma'lumotlar bazasi strukturasi soddalashtirishga qaratilgan. Unda ierarxik modeldagi kabi avlod va ajdodlarga yaqqol ko'rsatgichlar yo'q, barcha ma'lumotlar star va ustunlarga bo'lingan oddiy jadvallarda tasvirlanadi. Relyatsion modeldagi asosiy tushuncha "munosabat" (relation) hisoblanadi..

Relyatsion modelda o'ziga xos atamalar ishlatiladi, biroq bu model mohiyatini o'zgartirmaydi. Masalan, mantiqiy darajada element atribut deb ataladi. Bundan tashqari u uchun "kolonka", "ustun" va "maydon" atamalari ham ishlatiladi. Atributlar to'plami kortejni (qator, yozuv, satr) hosil qiladi. Kortejlar to'plami munosabatni (MB jadvali yoki faylini) hosil qiladi.

Relyatsion modelda fayllar o'rtasidagi bog'lanishlar yaqqol tarzda tavsiflanmasligi ham mumkin. Bu bog'lanishlar ma'lumotlarni qayta ishlash vaqtida dinamik ravishda mos maydonlarning qiymatlari

bo'yicha o'rnatiladi. Relyatsion MB da yozuvlar chiziqli strukturaga ega. Har bir munosabat kalitga, ya'ni kortejni bir qiymati identifikatsiyalovchi atributga (oddiy kalit) yoki atributlar to'plamiga (tarkibiy kalit) ega.

Qaralayotgan munosabatda kalit bo'lmagan atribut yoki atributlar guruhi boshqa munosabatda kalit bo'lsa, bu atribut yoki atributlar guruhi tashqi kalit deb ataladi.

Agar biror jadval tashqi kalitga ega bo'lsa u holda u: a) mos birlamchi kalitga ega bo'lgan jadval bilan mantiqan bog'langan; b) bu bog'lanish birga ko'p xarakterga bo'ladi.

Shunday qilib relyatsion model 1969 -1970 yillarda Kodd tomonidan munosabatlarning matematik nazariyasi asosida yaratilgan bo'lib, quyidagi asosiy tushunchalarga asoslanadi jadval, munosabat, satr, ustun, birlamchi kalit, tashqi kalit.

Ma'lumotlarning shunday modeliga relyatsion model deyiladi, unda barcha ma'lumotlar foydalanuvchiga jadval shaklida xavola etiladi va ma'lumotlar bazasi ustidagi barcha amallar jadvallar ustiga amallarga olib kelinadi. Har bir jadval ma'lumotlar bazasi tarkibida o'z nomiga ega bo'ladi, hamda satr va ustunlardan iborat bo'ladi. Har bir jadval real dunyodagi obyektlar (mohiyat) tipini aks ettiradi, har bir satri esa obyektning konkret nusxasini bildiradi. Relyatsion modelning har bir tushunchasini "Xodim" - mohiyati (obyekti) misolida ko'rib chiqamiz:

Tabel nomer	Familiya	Atribut		munosabat sxemasi	
		Ism	Otasining ismi	Bo'lim	Lavozimi
100	Abdullaev	Abdulla	Abdullaevich	002	Boshliq
101	Ismoilov	G'ani	Komilovich	001	Injener
102	Karimov	Boltaboy	To'raevich	002	injener

Atribut, maydon qiymati — Kortej, yozuv, satr domen, ustun

3.1- rasm. "Xodimlar" obyekti haqidagi jadval.
Har bir ustun o'z nomiga ega bo'ladi va ustun nomi odatda

jadvalning ustki qismida ko'rsatiladi. Uning nomi joriy jadvalda boshqa ustun uchun takrorlanmasligi kerak, biroq bu nom boshqa jadvalning ustun nomi sifatida ishlatilishi mumkin. Ya'ni turli jadvallar bir xil nomli ustunga ega bo'lishi mumkin.

Ixtiyoriy jadval kamida bitta ustunga ega bo'ladi; ustunlar jadvalda ularni yaratish tartibida joylashadi. Satrlar ustunlardan (atributlar) farqli ravishda nomga va joylashish tartibiga ega emas. Ularning miqdori chegaralanmagan.

Ixtiyoriy jadval shunday ustunga yoki ustunlarga (kalitga) ega bo'ladi, undagi qiymatlar har bir satrni takrorlanmas qiymat bilan identifikatsiyalaydi. Rasmda keltirilgan misolda kalit bu "tabel nomeri" ustunidir.

Atributlar qiymati eng kichik axborot birligi – domen yordamida ko'rsatiladi. Boshqacha aytganda domen – bu obyekt atributining mumkin bo'lgan barcha qiymatlar to'plamidir. Yana ikkita tushunchani ko'rib o'tamiz. Bu "daraja" va "Kardinal son" tushunchalari. Munosabatlarning kardinal soni deganda kortejlar soni tushuniladi, munosabat darajasi – bu joriy munosabatning atributlar sonidir.

Jadvallar o'rtasidagi aloqa relyatsion ma'lumotlar modelining asosiy elementi hisoblanadi. Bu aloqa tashqi kalitlar yordamida qo'llab – quvvatlanadi. Misol sifatida biror korxonaning MB sida saqlanayotgan xodimlar ("xodimlar" jadvali) va boshliqlar ("boshliqlar" jadvali) haqidagi ma'lumotlarni ko'rib chiqamiz.

"Boshliqlar" jadvalida birlamchi kalit "nomer" ustini. "Familiya" ustunining qiymatlari takrorlanuvchi bo'lganligi uchun birlamchi kalit sifatida qo'llash mumkin emas. "Xodimlar" jadvalida "Boshliq nomeri" ustuni tashqi kalit hisoblanadi.

MB da ma'lumotlar bilan birga ma'lumotlar lug'ati va boshqa obyektlar, masalan, ekran formalari, hisobotlar, tasvirlovchi formalar (views) va amaliy dasturlar joylashadi.

Relyatsion MB ni butunlik cheklanishi ba'zi talablarni, masalan, atribut qiymati faqat mos domen ichidan olinishini, yoki tashqi kalit jadvaldagi mavjud bo'lmagan satrni ko'rsatmasligi kerak (ko'rsatgich bo'yicha yaxlitlik).

"Munosabat" tushunchasini batafsil ko'rib chiqamiz.

Relyatsion modellarda o'zgaruvchi munosabatlar va munosabat qiymatini farqlash kerak. O'zgaruvchi munosabat – bu xuddi dasturlash tilidagi kabi oddiy o'zgaruvchilar, ya'ni qiymati vaqt o'tishi bilan

o'zgaruvchi nomlangan obyekt. Ushbu o'zgaruvchining ixtiyoriy vaqt momentidagi qiymati munosabat qiymati hisoblanadi.
Birlamchi kalit

№	Xodimlar		Lavozim
	Familiya	boshliq	
4781	Ivanov	5742	M.n.s
5325	Petrov	6931	S.p.s
3120	Sidorov	5742	P.s
1230	Yashin	2345	Inj
2138	Yudin	6931	Bosh inj

№	Familiya	Boshliqlar	
		Bo'lim	Staj
5742	Ivanov	SAPR	12
6931	Petrov	№5	25
2345	Sidorov	laboratoriya	21

3.2. Relyatsion algebra

Relyatsion algebra Kodd tomonidan aniqlangan ikkita guruhga bo'lingan 8 ta operatoridan iborat.

Birinci guruhga to'plar ustida bajariluvchi an'anaviy amallar kiradi: birlashtirish (\cup), kesishma (\cap), ayirish ($-$) va dekart ko'paytirish (*). Bunda barcha amallarda operandlar ixtiyoriy to'plam deb emas, balki munosabatlar deb qaraladi.

Ikkinchi guruh maxsus relyatsion amallarni tashkil etadi: tanlash, proeksiya, birlashtirish va bo'lish.

Ushbu amallarning munosabalarida qo'llagandagi natijalarni batafsil ko'rib chiqamiz.

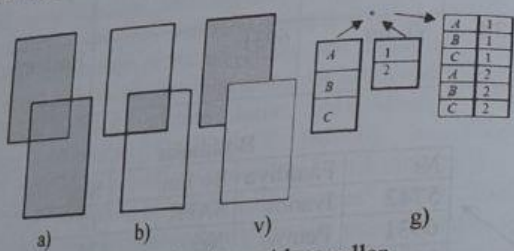
Birlashtirish (U). Berilgan ikkita munosabatdan biriga va ikkalasiga ham tegishli bo'lgan kortejlardan iborat yangi munosabatni qaytaradi. (3.2-rasm, a).

Kesishma (∩). Bir vaqtning o'zida ikkala munosabatga ham tegishli bo'lgan barcha kortejlardan iborat yangi munosabatni qaytaradi. (3.2-rasm, b).

Ayirish (−). Berilgan ikkala munosabatdan faqat birinчисiga tegishli bo'lgan kortejlardan iborat yangi munosabatni qaytaradi. (3.2-rasm, v).

Dekart ko'paytirish (*). berilgan ikkala jadvaldagi kortejlarning kombinatsiyalaridan iborat yangi jadval hosil qiladi (3.2-rasm, g).

Tanlash – berilgan jadvaldagi ma'lum shartni qanoatlantiruvchi barcha kortejlardan iborat yangi jadval hosil qiladi. Bu algebraik cheklanish hisoblanadi. (3.2-rasm, a).

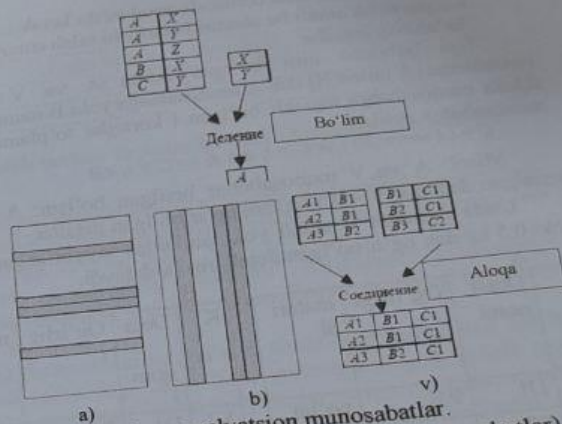


3.2-rasm. To'plamlar ustida amallar.

a-birlashtirish, b – kesishma, v- ayirish, g – dekart ko'paytirish
 Proeksiya – berilgan jadvaldagi ba'zi kortejlarni istisno (chiqarib tashlash) qilib qolgan kortejlardan (podkortej) yangi munosabat hosil qiladi (3.3-rasm, b).

Ulash – berilgan ikkiala jadvalda umumiy qiymatga ega bo'lgan kortejlarning ulanishidan iborat yangi jadval hosil qiladi. Natijaviy jadvalda umumiy qiymat faqat bir marta qatnashadi. Bunday ulash tabiiy ulash deb ataladi (3.3-rasm, v).

Bo'lish – berilgan binar va unar ikkita jadval uchun unar jadvalning barcha qiymatlari bilan moslashgan binar jadvaldagi bitta atributning qiymatlaridan iborat jadval (3.3-rasm).



3.3-rasm. Maxsus relyatsion munosabatlar.
 a- namuna olish, b – proeksiya, v- aloqa (o'zaro munosabatlar)

Jadval ustidagi har qanday amal natijasi jadvaldan iborat bo'ladi. Ushbu relyatsion xususiyat yopiqlik xossasi deb ataladi.

Biror amal natijasi boshqa amal uchun boshlang'ich ma'lumot sifatida qo'llanilishi mumkin. Shuning uchun, masalan, birlashtirish proeksiyasini olish, yoki ikkita tanlanma ulanmasini olish mumkin. Bunday ifodalar murakkab hisoblanadi.

Har bir munosabat sarlavhaga, tanasiga, kalitlar potensialiga (zahirasiga) ega bo'ladi. Relyatsion amallarni bajarishda atribut nomlarini yaratilishi va potensial kalitlar o'zgarishiga e'tibor qilish kerak.

Standart relyatsion amallar

Jadvallar ustida amallar bajarilishini batafsil ko'rib o'tamiz.
 Birlashtirish (union), kesishma (intersect) va ayirish (minus) amallari uchun quyidagi 2 ta xossa qanoatlantirilishi kerak:

– operandlar bir xil darajaga ega bo'lishi kerak;

- mos atributlar bitta domenda aniqlanishi kerak.
Ko'paytirish amali bu shartni bajarilishi talab etmaydi.

An'anaviy amallar
Tipi bo'yicha mos bo'lgan ikkita A va V munosabatni birlashmasi (A union B) deb A munosabatga yoki B munosabatga yoki ikkala munosabatga tegishli bo'lgan t kortejlar to'plamidan iborat S munosabatga aytiladi.

$$C = (A \cup B) | t_i \in C \forall t_j \in A \& t_i \in C \forall t_j \in B.$$

Misol: A va V munosabatlar berilgan bo'lsin: A - po'latdan yasalgan detallar, V - 0.5 kg dan og'ir bo'lgan detallar.

Unda A union B amali yoki po'latdan tayyorlangan detallarni, yoki 0.5 kg dan og'ir bo'lgan detallarni ifodalaydi.

A				B			
K	Detal nomi	Og'irligi	material	K	Detal nomi	Og'irligi	material
1	D1	0.8	po'lat	K1	D1	0.8	po'lat
2	D2	1.0	po'lat	K2	D2	1.0	po'lat
K3	D3	0.5	po'lat	K4	D4	0.7	alyuminiy

Natijada 6 ta emas, 4 ta kortej olinadi va takrorlanuvchi kortejlar o'chiriladi.

K	Detal nomi	Og'irligi	Material
K1	D1	0.8	po'lat
K2	D2	1.0	po'lat
K3	D3	0.5	po'lat
K4	D4	0.7	alyuminiy

Tipi bo'yicha mos bo'lgan ikkita A va V munosabatni kesishmasi (A intersect B) deb bir vaqtning o'zida ikkala A va B munosabatga tegishli bo'lgan t kortejlar to'plamidan iborat S munosabatga aytiladi.

$$C = (A \cap B) | \forall t_i \in C | t_i \in A \& t_i \in B.$$

Misol: yuqoridagi keltirilgan munosabatlar uchun A intersect B amali natijasi po'latdan tayyorlangan va og'irligi 0.5 kg dan og'ir

bo'lgan detallarni tasvirlaydi.

C	Detal nomi	Og'irligi	Material
K1	D1	0.8	po'lat
K2	D2	1.0	po'lat

3.3. Bog'lanish turlari va xususiyatlari: Atributlar va kalitlar

Ma'lumotlar bazasida jadvallar orasida munosabatlar o'rnatiladi. Jadvallar orasidagi munosabatlarning to'rtta turi mavjud: bir-birga, bir-ko'pga, ko'p-birga, ko'p-ko'pga.

Relyatsion ma'lumotlar bazasini loyihalash masalalariga o'tishdan avval mumkin bo'lgan munosabatlar turining har biriga batafsilroq to'xtaymiz.

Bir-birga munosabatning ma'nosi shuki bitta jadvalning xar bir yozuvi boshqa jadvaldagi faqat bitta yozuviga to'g'ri keladi (mos keladi). Misol uchun, agar ikkita jadvalni ko'radigan bo'lsak, ulardan birida korxonada ishchilari to'g'risida ma'lumotlar, ikkinchisida esa professional ma'lumotlar bo'lsa, u holda bu jadvallar orasida bir-birga munosabat mavjud, chunki bitta jadvalda ma'lumoti bo'lgan odam uchun professional ma'lumotlari bo'lgan ikkinchi jadvalda faqat bitta yozuv bo'lishi mumkin.

Hammadan ko'p uchraydigan ma'lumotlar bazasidagi munosabatlar turi bu bir-ko'pga munosabat. Berilgan munosabatlar turini ko'rsatish uchun, korxonada mijozlari va ular bergan buyurtmalar to'g'risida ma'lumotlar bo'lgan jadvallarga murojaat qilishimiz mumkin. Boshqa misollar sifatida korxonada va unda ishlaydigan mexnatkashlar orasidagi munosabatlar ko'rilishi mumkin. Huddi shunday munosabatlar kompyuter va unga kiruvchi komponentlar orasida mavjud va x.k.

Ko'p-birga munosabat avval ko'rilgan tur bir-ko'pga munosabat bilan o'xshash. Obyektlar orasidagi munosabatlar turi sizning nuqtai nazaringizga bog'liq. Misol, agar siz qilingan buyurtmalar va mijozlar orasidagi munosabatlarni ko'rsangiz, u holda ko'p-birga munosabat olasiz.

Ikkita jadvallar orasida ko'p-ko'pga munosabat hosil bo'ladi, qachonki:

Birinchi jadvaldagi bitta yozuv ikkinchi jadvaldagi bittadan ortiqroq yozuv bilan bog'lanishi mumkin bo'lsa;
Ikkinchi jadvaldagi bitta yozuv birinchi jadvaldagi bittadan ortiqroq yozuv bilan bog'lanishi mumkin bo'lsa.

Misol uchun ulgurji savdo magaziniga murojaat qilamiz. Ikki guruh obyektlarni ko'rib chiqamiz: yetkazib beruvchi (ta'minlovchi) korxonalar ishlab chiqaradigan tovarlar ro'yxati va iste'molchilar buyurtma qilgan tovarlar ro'yxati. Bu ma'lumotlarni o'z ichiga olgan jadvallar orasida ko'p-ko'pga munosabat mavjud, chunki har qaysi yetkazib berilayotgan maxsulotga bittadan ortiq buyurtma bo'lishi mumkin. Huddi shunday, xar qaysi buyurtma qilingan maxsulot bittadan ortiq korxonalar tomonidan ishlab chiqarilishi mumkin.

Berilgan ma'lumotlarni normallashtirish

Normallashtirish – bu ma'lumotlar bazasidagi ma'lumotlarni ortiqchaligini kamaytirishdir.

Normallashtirish – bu jadvalni ma'lumotlarni qo'shish, o'zgartirish va o'chirish uchun qulay bo'lgan ikkita yoki undan ko'p jadvallarga bo'lishdir.

Normallashtirishni asosiy maqsadi – shunday ma'lumotlar bazasini yaratishga qaratilgan, unda faktlar bir marta qayt qilinadi, ya'ni ma'lumotlar ortiqcha yoki takrorlanishi mumkin emas.

Bu amal saqlanadigan ma'lumotlarni qarama – qarshiligini bartaraf etish uchun zarur.

Jadvallarni normallashtirish mantiqiy loyihalashning eng formal (tartibga solingan) metodi bo'lib, uni qo'llanganda ER-diagrammani qurish talab etilmaydi.

Normallashtirish jarayoni bir normal formadan boshqasiga o'tish amallaridan iborat bo'lib, unda har bir navbatdagi jadval oldingisiga nisbatan yaxshi xossalarga ega bo'ladi.

Har bir normal forma uchun bir qator talablar qo'yilgan va jadval normallashtirilgan deyiladi agar u qo'yilgan talablarga javob bersa.

Relyatsion ma'lumotlar bazasi nazariyasida quyidagi normal formalar ketma – ketligi ajratib ko'rsatiladi:

- birinchi normal forma (1NF);
- ikkinchi normal forma (2NF);
- uchinchi normal forma (3NF);

- Boys-Kodd normal formasi (BCNF);
- to'rtinchi normal forma (4NF);
- beshinchi normal forma yoki proeksiya – ulash normal formasi (5NF) yoki PJ/NF).

Normal formalarining asosiy xossalari:

- har bir navbatdagi normal forma qaysidir ma'noda oldingisiga qaraganda yaxshi;
- navbatdagi normal formaga o'tganda oldingi normal forma xossalari saqlab qolinadi.

Berilgan ma'lumotlar bazasini loyihalashda siz eng qulay berilgan ma'lumotlar bazasi strukturasi to'g'risidagi masalani hal qilishingiz zarur. Bunda izlanayotgan asosiy maqsadlar:

- Jadvaldagi ma'lumotlarga tez yetishishni ta'minlash;
- Kiritishdagi xatolar sababi va kompyuter diski maydonidan noratsional foydalanish sababi bo'lishi mumkin bo'lgan, keraksiz berilgan ma'lumotlarni qaytarilishiga yo'l qo'ymaslik;
- Berilgan ma'lumotlar butligini shunday ta'minlash kerakki, bitta obyekt o'zgartirilganda avtomatik ravishda ular bilan bog'liq bo'lgan obyektlarni o'zgarishi ro'y bersin.

Berilgan ma'lumotlar bazasidagi ma'lumotlarning ortiqchaligini kamaytirish jarayoni normallashtirish deb ataladi. Berilgan ma'lumotlar bazasi nazariyasida murakkab tuzilishga ega bo'lgan ma'lumotlarni bir nechta jadvallarga bo'lishni yetarlicha rasmiylashtirish yondoshishlar ishlab chiqilgan. Bu savollar batafsil maxsus adabiyotlarda yoritilgan.

Biz jadvallarni normallashtirishni, ularni nazariy asoslanishiga qarshi ba'zi bir amaliy tomonlariga to'xtaymiz.

Normallashtirish nazariyasi jadvallarning 5ta normal shakli bilan ishni tutatadi. Bu shakllar, birinчисidan beshinchi normal shakligacha ichiga olib, ortiqcha ma'lumotlarni kamaytirish uchun atalgan. Shuning uchun xar qaysi keyingi normal shakl avvalgi shaklning talablarini va ba'zi bir qo'shimcha shartlarni qoniqtirishi kerak.

Samrali strukturali ko'p jadvalli bazani loyihalashda nazariya natijalaridan foydalanamiz. Misol sifatida xaridor va qilingan buyurtmalar haqida quyidagi ma'lumotni o'z ichiga olgan jadvalni ko'rib chiqaylik:

Xaridorlar to'g'risida ma'lumot;
Buyurtma sanasi va buyurtma qilingan tovar miqdori;

Buyurtma bajarilgan sana va sotilgan tovar miqdori;
Sotilgan tovarning sifati (nomi, qiymati).

Bu jadvalni bir jadvalli ma'lumotlar bazasi sifatida ko'rish mumkin. Asosiy muammo shundaki, unda ko'p miqdorda qaytarilayotgan ma'lumot u qilgan har qaysi buyurtma uchun qaytariladi.

Bunaday tuzilish ma'lumotlar bazasi bilan ishlaganda hosil bo'ladigan quyidagi muammolarga sabab bo'ladi:

Qaytarilayotgan ma'lumotlarni kiritish uchun ko'p vaqt sarflashga to'g'ri keladi. Misol, xaridorlardan bittasi qilgan barcha buyurtmalar uchun xaridor to'g'risidagi bir xil ma'lumotlarni har safar kiritishingizga to'g'ri keladi.

Adres yoki telefon o'zgaranda xaridor to'g'risidagi ma'lumotlarni o'z ichiga olgan barcha yozuvlarni to'g'rilash kerak.

Qaytarilayotgan ma'lumotlar bazasining borligi uning o'lchamlarini asossiz kattalashishiga olib keladi. Natijada so'rovlarning bajarilish tezligi pasayadi. Bundan tashqari, qaytariladigan ma'lumotlar kompyuter diski maydonidan noratsional foydalanishiga olib keladi.

Har qanday odatdan tashqari holat talab qilingan ma'lumotni olish uchun talay vaqt talab qiladi. Misol, qaytarilayotgan ma'lumotlarni ko'p marotaba kiritishda xato ehtimolligi ortib boradi. Jadvallarni o'lchamlari katta bo'lganda xatolarni izlash ko'p vaqtni oladi.

Normallashtirish nazariyasining amaliy maslaxatlaridan qilingan buyurtmalar haqidagi jadval asosida camarali tuzilishga ega bo'lgan ko'p jadvalli ma'lumotlar bazasini ishlab chiqish uchun foydalanamiz.

Qilingan buyurtmalarni o'z ichiga olgan jadval normalashmagan bo'ladi. Ma'lumotlar bazasining jadvali birinchi normal shaklda, misol uchun, quyidagi talablarga javob berishi kerak:

Jadvaldavalda qaytarilgan maydonlar guruhi bo'lmasligi lozim;

Satrlar tartibga solinmagan bulishi kerak;

Ustunlar tartibga solinmagan bo'lishi kerak.

Birinchi shart bajarilishi uchun har bir jadval indeksga ega, noyob ma'lumotlarni o'z ichiga olgan maydon qo'shish kerak. Misol, xaridorlar to'g'risidagi ma'lumotni o'z ichiga oluvchi jadvalga xaridor kodi tushirilgan maydon qo'shish kerak.

Ikkinchi talab qaytariladigan guruhlar yo'q qilishni asoslaydi.

Xar qaysi xaridor bir nechta telefon nomeriga ega bo'lishi va qaysisida o'z navbatida bir nechta tovarlar buyurilgani bir nechta buyurtma qilishi mumkin bo'lganligi uchun bizga to'rtta jadval zarur. Bu jadvallarning har bir yozuvi quyidagi ma'lumotlarni o'z ichiga oladi.

Jadval	Ma'lumotlar
Birinchi	Xaridorlar to'g'risida ma'lumotlar
Ikkinchi	Xaridor telefonlarining ro'yxati
Uchinchi	Xaridor buyurtmasining nomlari va sanasi, menedjer to'g'risidagi ma'lumot
To'rtinchi	Buyurtma qilingan va sotilgan tovarning kodi, nomeri, miqdori

Bog'lanilayotgan jadvallar orasidagi munosabat birga-ko'p ekanligini takidlaymiz.

Endi birinchi va ikkinchi jadval normalashgan. Uchinchi jadvalda buyurtmani rasmiylashtirgan menedjer to'g'risida qaytarilayotgan ma'lumot joylashgan. Demak, bu jadvalni ikkiga ajratishimiz mumkin. Ularning biriga firma menedjerlari to'g'risidagi ma'lumotni, ikkinchisiga esa xaridor buyurtmasining nomeri va sanasini, hamda buyurtmani rasmiylashtirgan menedjer kodini kiritamiz. Buyurtma qilingan va sotilgan mahsulot kodini nomlanishini, miqdorini o'z ichiga olgan to'rtinchi jadvalni ko'rib chiqamiz. U yana sotilgan tovar to'g'risidagi qaytarilayotgan ma'lumotni o'z ichiga oladi. Ma'lumotlar bazasining bu ma'lumotini firma sotayotgan tovarlar ro'yhatining alohida jadvalida joylashtirish mumkin. O'tkazilgan normalashtirish natijasida bitta boshlang'ich jadval o'miga biz qaytarilmayotgan ma'lumotni o'z ichiga olgan oltita oddiy jadval hosil qildik.

Birlamchi va tashqi kalitlar haqida tushuncha

Kalit - bu talab etilayotgan mohiyat nusxasini topishda yordam beradigan atributlarning minimal to'plami. Minimal deyilishining sababi shundaki, agar to'plamdagi istalgan biror-bir atribut yo'qotilsa, qolgan atributlar yordamida mohiyatni identifikatsiyalash mumkin

emas, o'ar mohiyat hech bo'lmaganda bitta kalitni tasvirlaydi. Ulardan biri birinchi kalitni qabul qiladi. Birinchi kalitni tanlash faqat minimal sonli atributlardan tashkil topgan maydon nazarda tutuladi. Bundan tashqari kalitni ishlatishda uzun matoli va qiymatli maydonni olish tavsiya etilmaydi (butun sonli atributlarni ishlatish foydaliroqdir). Talabalar identifikatsiyasi uchun unikal (qaytarilmaydigan yagona ma'nosida) nomer bo'lgan imtihon daftarchasi nomerini ishlatish mumkin, yoki familiyasini, guruh nomerini va boshqa qo'shimcha atributlarni. Lekin bunday hollarda ikkita bitta familiyali talaba chiqib qolishi ham mumkin.

Birinchi kalit sterjnl mohiyat bolishligi ruxsat etilmaydi (birinchi kalitda qatnashuvchi har qanday atribut). Aks holda qarama-qarshi holat yuz beradi. Su sabab unikal birinchi kalitni tashkil qilish va ta'minlash kerak.

Agar C mohiyati ikkita A va B mohiyatni bog'lasa, u holda A va B mohiyatlar birinchi kalitga mos tashqi kalitni tashkil etish kerak.

Agar B mohiyati ikkinchi A mohiyatni belgilasa, u holda A mohiyat birinchi kalitga mos tashqi kalitni tashkil etish kerak. har qanday o'zaro bog'lanishlarda bo'lgan mohiyatlarni belgilash uchun (sterjnl, xarakteristik, belgilash assotsiatsiya kabi) yangi birlashtirilgan termin «Maqsad» yoki «Maqsadli mohiyat» ishlatiladi.

Assotsiyalarni ko'rsatish usulini tanlash muammosini qarashda va ma'lumotlar bazasida "qaysi kalit tashqi kalit?" degan savolga javob olish kerak bo'ladi. Keyinchalik, har bir tashqi kalit uchun uchta savolni echish kerak bo'ladi:

Ishlatilayotgan tashqi kalit aniqlanmagan qiymat qabul qilishi mumkinmi (NULL-qiymat)?

Tashqi kalitga murojaat qilayotgan maqsadli mohiyatni o'chirishga harakat bo'lganda nima yuz berishi mumkin?

Tashqi kalitga murojaat qilayotgan maqsadli mohiyat birinchi kalitni yangilashga harakat qilinganda nima yuz berishi mumkin?

Shunday qilib, har bir tashqi kalit uchun ma'lumotlar bazasini loyihalovchi loyihada tashqi kalitni tashkil qiluvchi maydon yoki maydonlar kombinatsiyasini tashkil etish kerak.

Nazorat savollari

1. Relyatsion model tarixi. Relyatsion modelning asosiy mohiyati.
2. Relyatsion modelning asosiy mohiyati. Relyatsion model terminologiyasi.
3. Relyatsionnoy model kalitlari. Oddiy va tarkibiy kalitlar
4. Ma'lumotlarning qanday modeli relyatsion deb ataladi.
5. relyatsion jadvalning asosiy elementlarini misol bilan tushuntiring.
6. Relyatsion ma'lumotlar modelida jadvallar o'rtasidagi bog'lanish qanday o'rnatiladi? Misol keltiring.
7. Relyatsion modelning kortej, domen, atribut atamalariga ta'rif bering.
8. birlamchi va tashqi kalit nima?
9. Relyatsion algebraning asosiy operatorlari?
10. Ma'lumotlarni normalashtirish deganda nimani tushunasiz?
11. MB ni mantaqiy loyihalashning maqsadi.
12. MB ni noreallashtirish nima?
13. MB normalashtirishdan asosiy maqsad nima?
14. Qanday jadval normalashtirilgan jadval deyiladi?
15. Relyatsion jadvalning maydonlari o'rtasidagi bog'lanishlarning asosiy turlari.

4-BOB. MULTIMEDIA MB SO'ROVLARI

4.1. SQL tili haqida tushuncha

SQL (Structured Query Language) *strukturalashgan so'rov tili* ma'nosini bildirib, u relyatsion ma'lumotlar bazasi bilan ishlash imkonini yaratib beradigan tildir.

Ma'lumki, relyatsion modelning tarixi (va bilvosita SQL tarixi ham) 1970 yil Y.F.Koddni (bu paytda u IBM korporatsiyasining San Xosedagi tadqiqot markazida ishlagan) maqolasi chiqqan davrdan boshlanadi. 1974 yil shu laboratoriyada ishlovchi D. Chamberlen "Structured English Query Language" yoki SEQUEL deb nomlangan tilni e'lon qiladi. 1976 yil bu tilning qayta ishlangan SEQUEL/2 versiyasi yaratildi va u rasmiy ravishda SQL deb atalgan. Xozirgi kunda SQL qisqartmasini ba'zilar "sikvel" deb talffuz etadi. Biroq rasmiy ravishda u "es-kyu-el" deb o'qilishi kerak.

SQL tili relyatsion algebra paydo bo'lgandan keyin paydo bo'ldi va uning birinchi prototipi IBM Research kompaniyasi tomonidan 70 yillar oxirida yaratilgan. Bu til birinchi IBM System R nomli MBBT tarkibiga kiritilgan. Keyinchalik bu til ko'pgina tijorat MBBT tarkibida qo'llanilgan va keng tarqalganligi sababli vaqt o'tishi bilan relyatsion MBBT larda ma'lumotlar ustida amallar bajaruvchi tillarning norasmiy standarti bo'lib qoldi. SQL tilining birinchi ramiy standarti 1989 yil qabul qilingan. Ko'pgina MBBT lar ushbu standartni qo'llab – quvvatlaydi. Biroq ma'lumotlar bazasi bilan bog'liq axborot texnologiyalarining rivojlanishi va ba'zi talablarning paydo bo'lishi birinchi SQL standartini qayta ishlash va kengaytirishni taqozga etdi.

1992 yil oxirida SQL tilining yangi xalqaro standarti (SQL/92 yoki SQL2) qabul qilindi unda ham ba'zi kamchiliklar aniqlangan, biroq shunga qaramasdan SQL/89 ga nisbatan aniq va to'liqroq xisoblanadi. Xozirgi paytda ko'pgina MBBT ishlab chiqaruvchilar o'z maxsulotlarini SQL2 standartini qanoatlantiradigan qilib o'zgartirdilar.

1999 yil SQL3 deb atalgan yangi standart paydo bo'ldi. Agar SQL1 va SQL2 standartlari biri –biridan miqdor jixati bilan farq qilgan bo'lsa, SQL3 standarti sifat jixatlari bilan farqlanadi. SQL3 ga murakkab strukturaga ega ma'lumotlar tipini ishlatish imkonini beradigan yangi ma'lumotlar tipi kiritilgan. Bu tipni obyektga

mo'ljallanganlik darajasi yuqori xisoblanadi. SQL tilini tula qonli an'anaviy dasturlash tillari tarkibiga kiritib bo'lmaydi. Chunki unda dastur bajarilishini boshqaruvchi va boshqa ko'pgina an'anaviy operatorlar yo'q. Unda faqat ma'lumotlar bazasida saqlanayotgan ma'lumotlarga murojaat qiluvchi operatorlar mavjud.

SQL tili foydalanuvchi relyatsion ma'lumotlar bazasi bilan muloqat qilishi uchun mo'ljallangan bo'lib, quyidagi 3 ta qismdan iborat:

- DDL (Data Definition Language) – ma'lumotlarni aniqlash tili. Ma'lumotlar bazasini (jadvallarini, indekslarini va x.k.) yaratish va uning sxemasini taxrirlash uchun mo'ljallangan..
- DCL (Data Control Language) – ma'lumotlarni boshqarish tili. Foydalanuvchilarning ma'lumotlar bazasi obyektlariga murojatini chegaralash operatorlaridan iborat.
- DML (Data Manipulation Language) – ma'lumotlarni qayta ishlash tili. Ma'lumotlar bazasi jadvallariga o'zgartirishlar kiritish uchun mo'ljallangan.

Ma'lumotlar bazasi bilan ishlovchi ixtiyoriy til foydalanuvchiga quyidagi imkoniyatlarni yaratishi lozim:

- strukturasi to'la tavsiflagan holda ma'lumotlar bazasini va jadvallarini yaratish;
- ma'lumotlar ustida manipulyatsiya amallarini bajarish, masalan, jadvallardan ma'lumotlarni kiritish, taxrirlash, va o'chirish;
- oddiy va murakkab so'rovlarni bajarish.

Bundan tashqari, ma'lumotlar bazasi bilan ishlovchi til yuqoridagi amallarni bajarish uchun foydalanuvchilardan kam urinishlarini talab qilishi, hamda komandalarining sintaksisi va tuzilishi o'zganish uchun ososn va tushunarli bo'lishi kerak. Nixoyat bu til universal bo'lishi kerak. Bu bir MBBT dan boshqasiga o'tganda komandalarni bir xil strukturasi va sintaksisidan foydalanishni ta'xminlaydi. SQL tili bu talablarni barchasini qanoatlantiradi.

4.2. SQL tilida ma'lumotlar turlari

SQL tilida ma'lumotlarning quyidagi asosiy turlari ishlatilib, ularning formatlari har xil MBBT lar uchun farq qilishi mumkin:

INTEGER	Butun son (odatda 10 tagacha qiymatli raqam va ishora).
SMALLINT	Qisqa butun (odatda 5 tagacha qiymatli raqam va ishora).
DECIMAL(p,q)	o'qli son, p raqam va ishoradan iborat (0 < p < 16).
FLOAT	haqiqiy son 15 ta qiymatli raqam va butun darajadan iborat.
CHAR(n)	uzunligi o'zgarmas, n ga teng bo'lgan simvolli qator (0 < n < 256).
VARCHAR(n)	uzunligi o'zgaruvchi, n simvoldan oshmagan simvolli qator.
DATE	sana (yy/mm/dd).
TIME	vaqt (hh mm ss).
DATETIME	sana va vaqt kombinatsiyasi.
MONEY	pul birligi simvoli (\$, rub, ...)

4.3. Jadvallar yaratish va ularga cheklovlar kiritish

Jadvallarni yaratish. Jadvallar CREATE TABLE buyrug'i bilan yaratiladi. Bu buyruq qatorlarsiz bo'sh jadval yaratadi. U jadval nomini, ma'lum tartibda ko'rsatilgan ustunlar nomlari ketma - ketligi, ma'lumotlar turlari va ustunlar o'lchovini aniqlaydi.

CREATE TABLE buyrug'ining umumiy yozilishi:

```
CREATE TABLE <jadval nomi>
(<ustun nomi> <ma'lumot turi>[(<ustun o'lchovi>)],
<ustun nomi> <ma'lumot turi>[(<ustun o'lchovi, ...>);
```

Jadval yaratishda va ular ustida ish yuritishda quyidagi 2 ta jadvaldan iborat ma'lumotlar bazasini misol sifatida qaraymiz. Sotuvchilar (Salepeople):

Snum	Sname	City	Comm
11	Peel	London	0.12
12	Serres	San Jose	0.13
14	Motika	London	0.11

SNum - xar bir sotuvchi unikal nomeri,
 SName - sotuvchi nomi,
 City - sotuvchi adresi (shaxri),
 Comm - sotuvchilarning o'qli shakldagi komission foydasi.
 Buyurtmachilar (Customers):

Cnum	Cname	City	Rating	SNum
21	Hoffman	London	100	11
22	Giovanni	Rome	200	13
23	Liu	SanJose	200	12

CNum - xar bir buyurtmachi unikal nomeri;
 CName - buyurtmachi nomi;
 City - buyurtmachi adresi (shaxri);
 Rating - buyurtmachining boshqalardan ustunlik darajasini ko'rsatuvchi kod;
 SNum - shu buyurtmachiga tayinlangan sotuvchi nomeri.
 Misol uchun sotuvchilar jadvalini yaratish:

```
CREATE TABLE Salepeople
(SNum integer, SName char(10), City char(10), Comm
```

decimal);
Cheklovlar kiritish. Jadvalni yaratayotganda (yoki uni o'zgartirayotganda), maydonlarga kiritilayotgan qiymatlarga cheklovlar o'rnatish mumkin. Bu holda SQL cheklovlarga to'g'ri kelmaydigan hamma qiymatlarni rad etadi.

Maydonga bo'sh (NULL) qiymatlar kiritilishi oldini olish uchun CREATE TABLE buyrug'ida NOT NULL cheklovi ishlatiladi. Masalan, birlamchi kalitlar xech qachon bo'sh bo'lmasliklari kerak, shuning uchun Salepeople jadvalini quyidagicha yaratish mumkin:

```
CREATE TABLE Salepeople
(Snum integer NOT NULL,
Sname char(10), city char(10), comm decimal);
```

Ko'p hollarda ustunga kiritilgan qiymatlar bir biridan farq qilishi kerak bo'ladi. Agar ustunga UNIQUE cheklovi o'rnatilsa, unda ustunga qiymat kiritishga urinish rad etiladi. Bu cheklov bo'sh bo'lmaydigan (NOT NULL) deb e'lon qilingan maydonlarga qo'llaniladi. Masalan:

```
CREATE TABLE Salepeople
(SNum integer NOT NULL UNIQUE,
Sname char(10), city char(10), comm decimal);
```

Jadval cheklovi UNIQUE maydonlar guruhiga ham o'rnatilishi mumkin. Bu bir necha maydonlar qiymatlari kombinatsiyasi unikalligini ta'minlaydi.

4.4. Maydon qiymatlarini tekshirish (CHECK cheklovi)

CHECK cheklovi jadvalga kiritilayotgan ma'lumot qabul qilinishidan oldin mos kelishi lozim bo'lgan shart kiritishga imkon beradi. CHECK cheklovi CHECK kalit so'zi ko'rsatilgan maydondan foydalanuvchi shartli ifodadan iboratdir. Misol uchun Salepeople jadvali Comm ustuniga kiritilayotgan qiymat 1 dan kichik bo'lsin.

```
CREATE TABLE Salepeople
```

```
(SNum integer NOT NULL PRIMARY KEY,  
SName char(10) NOT NULL UNIQUE,  
City char(10),  
Comm decimal CHECK (Comm < 1));
```

CHECK cheklovidan maydonga ma'lum qiymatlarini kiritishdan himoya qilib, xatolar oldini olish uchun foydalanish mumkin. Masalan, mahsulotni sotish shaxobchalariga ega bo'lgan shaharlar faqat London, Barselona, San Xose va Nyu York bo'lsin.

```
CREATE TABLE Salepeople
```

```
(SNum integer NOT NULL PRIMARY KEY,  
SName char(10) NOT NULL UNIQUE,  
City char(10) CHECK (City IN ('London','New York','San  
Jose','Barselona')), Comm decimal CHECK (Comm < 1));
```

CHECK jadval cheklovi sifatida kelishi mumkin. Bu shartga bir necha maydon kiritishga imkon beradi. Masalan:

```
CREATE TABLE Salepeople
```

```
(SNum integer NOT NULL PRIMARY KEY,  
SName char(10) NOT NULL UNIQUE,  
City char(10),  
Comm decimal,  
CHECK (Comm < .15 OR City = 'Barcelona'));
```

Ko'zda tutilgan qiymatlarni o'rnatish

Biror bir maydon uchun qiymat ko'rsatmagan holda jadvalga satr qo'shilish kerak bo'lsa, SQL bunday maydonga kiritish uchun ko'zda tutilgan qiymatga ega bo'lishi kerak, aks holda buyruq rad etiladi. Eng umumiy ko'zda tutilgan qiymat NULL qiymatdir. CREATE TABLE buyrug'ida ko'zda tutilgan qiymat DEFAULT operatori orqali, ustun cheklovi sifatida ko'rsatiladi. Masalan:

```
CREATE TABLE Salepeople
```

```
(SNum integer NOT NULL PRIMARY KEY,  
SName char(10) NOT NULL UNIQUE,  
City char(10) DEFAULT 'New York',  
Comm decimal CHECK (Comm < 1));
```

Maydonlar qiymatlarini kiritish, o'chirish va o'zgartirish
Jadvallarni o'chirish. Faqat bo'sh jadvalni o'chirish mumkin.
Jadvalni o'chirish buyrug'i quyidagi ko'rinishga ega:

```
DROP TABLE <jadval nomi>; Masalan: DROP TABLE  
Salepeople;
```

Jadvalni yaratilgandan so'ng o'zgartirish. Jadvalni o'zgartirish uchun ALTER TABLE buyrug'idan foydalaniladi. Bu buyruqda jadvalga yangi ustunlar qo'shish, ustunlarni o'chirish, ustunlar kattaligini o'zgartirish, hamda cheklovlarini qo'shish va olib tashlash imkoniyatlariga ega.

Jadvalga ustun qo'shish buyrug'i:

```
ALTER TABLE <jadval nomi> ADD <ustun nomi>  
<ma'lumot turi> <o'lchami>;
```

Masalan:

```
ALTER TABLE Salepeople ADD Phone CHAR(7);
```

Qiymatlarni kiritish. Hamma satrlar SQLda INSERT buyrug'i yordamida kiritiladi. INSERT quyidagi formatga ega:

```
INSERT INTO <table name | view name> [(column [,column]  
...)]
```

```
VALUES (<value> [,<value>] ... );
```

Masalan, sotuvchilar jadvaliga yangi satr kiritish uchun quyidagi buyruqdan foydalanish mumkin:

```
INSERT INTO Salepeople VALUES (11, 'Peel', 'London',
```

```
.12);
```

Ustun nomlarini ko'rsatish ham mumkin, masalan:

```
INSERT INTO Salepeople (Sname, Comm, SNum)
```

```
VALUES ('Peel', .12, 11);
```

Bu yerda e'tibor berilsa City ustuni tashlab yuborilgan, chunki unga ko'zda tutilgan qiymat kiritiladi.

Satrlarni o'chirish. Satrlarni jadvaldan DELETE buyrug'i bilan o'chirish mumkin. U alohida qiymatlarni emas faqat satrlarni o'chiradi. DELETE quyidagi formatga ega:

```
DELETE FROM <table name | view name> [WHERE
```

search-condition];

Masalan, sotuvchilar jadvalidagi hamma satrlarni o'chirish uchun, quyidagi shartni kiritish mumkin: **DELETE FROM Salepeople;**

Ma'lum satrlarni o'chirish uchun shartlardan foydalaniladi. Masalan, jadvaldan Axelrod sotuvchini o'chirish uchun uning nomerini shartda berish kerak:

```
DELETE FROM Salepeople WHERE SNum = 13;
```

Maydon qiymatlarini o'zgartirish. O'zgartirish UPDATE buyrug'i yordamida bajariladi. Bu buyruqda UPDATE ifodasidan so'ng jadval nomi va SET ifodasidan so'ng ma'lum ustun uchun o'zgartirish ko'rsatiladi. UPDATE ikki formatga ega. Ulardan birinchisi:

```
UPDATE <table name | view name>
```

```
SET column = expression [, column = expression]
```

```
[WHERE search-condition]
```

bu yerda expression - bu ustun | ifoda | konstanta | o'zgaruvchi.
Ikkinchi variant:

```
UPDATE <table name>
```

```
SET column = expression, ...
```

```
[ FROM table-list ]
```

```
[ WHERE search-condition ]
```

Masalan, hamma buyurtmachilar bahosini 200 ga o'zgartirish mumkin:

```
UPDATE Customers SET Rating = 200;
```

Ma'lum satrlarni o'zgartirish uchun DELETE dagi kabi shartlardan foydalanish kerak. Masalan, Peel (SNum=11) sotuvchining hamma buyurtmachilari uchun bir xil o'zgartirish quyidagicha kiritiladi:

```
UPDATE Customers SET Rating = 200 WHERE SNum = 11;
```

SET vergul bilan ajratilgan ixtiyoriy sondagi ustunlarga qiymat tayinlashi mumkin. Masalan:

```
UPDATE Salepeople SET SName = 'Gibson', City = 'Boston', Comm = .10
```

```
WHERE SNum = 14;
```

UPDATE buyrug'ining SET jumlasida ifodalarni ham ishlatish mumkin. Masalan: **UPDATE Salepeople SET Comm = Comm * 2;**

4.5. Oddiy SELECT so'rovlar va mantiqiy operatorlar

SELECT operatori MB jadvallaridan natijaviy to'plam olish uchun mo'ljallangan. SELECT operatori yordamida MBga so'rov beriladi va u foydalanuvchiga ma'lumotlarning natijaviy to'plamini qaytaradi. Bu ma'lumotlar jadval shaklida qaytariladi. Bu jadval keyingi SELECT operatori tomonidan yana qayta ishlanishi ham mumkin.

SELECT operatori SQL quyidagi ko'rinishga ega:

```
SELECT [ALL] <uctunlar>
```

```
FROM jadval
```

```
WHERE izlash sharti
```

```
GROUP BY ustunlar
```

```
HAVING izlash sharti
```

```
ORDER BY tartiblash spesifikatori
```

Masalan, OFFICES jadvalidagi hamma yozuvlarni qaytaruvchi sodda so'rov quyidagicha yoziladi. **SELECT * FROM OFFICES**

Misol: Hamma xizmatchilarning nomlari, ofislari va ishga olish sanalari ro'yxatini hosil qilish.

```
SELECT NAME, REP_OFFICE, HIRE_DATE FROM SALESREPS
```

SELECT operatori WHERE sharti berilgan shart asosida kerakli ma'lumotlarni qaytarish uchun xizmat qiladi. Masalan, sotuvlarda haqiqiy hajmi rejadan oshgan ofislarni ko'rsatish kerak.

```
SELECT CITY, SALES, TARGET FROM OFFICES WHERE SALES > TARGET
```

Nomeri 105 ga teng bo'lgan xizmatchi nomi, haqiqiy va rejadagi sotuvlar hajmini ko'rsatish:

```
SELECT SALES, NAME, QUOTA FROM SALESREPS WHERE EMPL_NUM = 105
```

Agar izlash sharti TRUE bo'lsa qator natijaviy to'plamga qo'shiladi, agar izlash sharti FALSE bo'lsa, qator natijaviy to'plamga qo'shilmaydi, agar NULL bo'lsa ham natijaviy to'plamdan chiqariladi. O'z ma'nosiga ko'ra WHERE, kerakli yozuvlarni qoldiruvchi filtr sifatida ishlatiladi.

Mantiqiy operatorlar

BETWEEN va **IN** operatorlari. **BETWEEN** operatori - bu qiymatlar diapazoniga tegishlilikni tekshirishdir. Misol: Narxi har xil diapazonga mos keluvchi buyurtmalarni topish.

```
SELECT ORDER_NUM, AMOUNT FROM ORDERS  
WHERE AMOUNT BETWEEN 20.000 AND 29.999
```

NOT ifodasi shartni teskarisiga o'giradi, yani tegishli emas ma'nosini bildiradi. **NOT** ifodasi yordamida berilgan diapazonga tegishlilikni tekshirish mumkin, masalan: sotuvlar haqiqiy hajmlari rejaning 80 dan 120 protsentgacha bo'lgan oraliqqa tushmaydigan xizmatchilar ro'yxatini chiqarish.

```
SELECT NAME, SALES, QUOTA FROM SALESREPS  
WHERE SALES NOT BETWEEN (0.8 * QUOTA) AND (1.2
```

* QUOTA)
IN operatori to'plamga tegishlilikni tekshiradi. Masalan, to'rtta aniq xizmatchilar tomonidan olingan hamma buyurtmalarni aniqlash.

```
SELECT ORDER_NUM, REP, AMOUNT FROM ORDERS  
WHERE REP IN (107, 109, 101, 103)
```

NOT IN yordamida diapazonga "tegishli emaslikni" tekshirish mumkin.

LIKE operatori. Quyidagicha '%' shablonli **LIKE** operatorini qaraymiz:

```
SELECT COMPANY, CREDIT_LIMIT FROM  
CUSTOMERS
```

```
WHERE COMPANY LIKE '%n'
```

Bu holda **LIKE** '%n' operatori 'n' harfiga tugaydigan hamma yozuvlarni ko'rsatadi, agar % shablone birinchi kelsa:

```
SELECT COMPANY, CREDIT_LIMIT FROM  
CUSTOMERS
```

```
WHERE COMPANY LIKE '%gan'
```

Agar faqat bitta simvol ixtiyoriy bo'lsa '_' shablone qo'llaniladi.

Masalan:

```
SELECT COMPANY, CREDIT_LIMIT FROM  
CUSTOMERS
```

```
WHERE COMPANY LIKE 'Ap_lsin'
```

Yozuvlarni tartiblash, **ORDER BY** ifodasi. **SELECT** operatori tarkibida natijaviy yozuvlarni tartiblangan holda taqdim etish uchun

ORDER BY ifodasi ko'zda tutilgan. Masalan, agar o'quvchilar ro'yxatini alfavit tartibida yoki tovarlar narxini kamayish tartibida chiqarish zarur bo'lsa, u holda bu ifodadan foydalanish kerak bo'ladi.

Quyidagi misolni ko'ramiz: Har bir ofis uchun sotuvlar haqiqiy hajmlarini regionlar nomlari, har bir regionda esa shaharlar nomlari bo'yicha alfavit tartibida chiqarish.

```
SELECT CITY, REGION, SALES FROM OFFICES  
ORDER BY REGION, CITY
```

Masalan: Sotuvlari haqiqiy hajmlari kamayish tartibida bo'lgan ofislar ro'yxatini chiqarish.

```
SELECT CITY, REGION, SALES FROM OFFICES  
ORDER BY SALES DESC
```

Sotuvlar hajmlarini **DESC** predikatini qo'llab kamayish tartibida chiqaramiz. O'sish tartibida chiqarish uchun **ASC** predikati qo'llanadi. Bu predikat ko'zda tutilgan bo'lib, uni ko'rsatish shart emas.

Bir necha jadvallar bilan ishlash

Jadvallarni jamlashtirish. Jamlashtirish relyatsion ma'lumotlar bazasi operatsiyalaridan biri bo'lib, jadvallar orasidagi aloqani belgilaydi va ulardan ma'lumotni bitta buyruq yordamida ajratishga imkon beradi. Jamlashda jadvallar **FROM** buyrug'idan so'ng ro'yhat sifatida tasvirlanadi. So'rov predikati ixtiyoriy jadval ixtiyoriy ustuniga tegishli bo'lishi mumkin. Jamlashning eng soddasi bu dekart ko'paytmasidir, uni quyidagicha bajarish mumkin:

```
SELECT Customers.*, Salepeople.* FROM Salepeople,
```

Customers;

Lekin bu yerda hosil bo'lgan jadval keraksiz ma'lumotlarga ega. Keraksiz satrlarni olib tashlash uchun **WHERE** jumlasidan foydalaniladi.

Masalan: berilgan shahardagi sotuvchilar va buyurtmachilar ixtiyoriy kombinatsiyasini ko'rish uchun quyidagini kiritish lozim:

```
SELECT Customers.CName, Salepeople.SName,  
Salepeople.City
```

```
FROM Salepeople, Customers
```

```
WHERE Salepeople.City = Customers.City;
```

Jamlashda **SQL** bir necha jadval satrlari kombinatsiyasini predikatlar bo'yicha solishtirishdir. Misol: har bir sotuvchiga mos

keluvchi buyurtmachilar ro'yxatini chiqarish:

```
SELECT Customers.CName, Salepeople.SName  
FROM Customers, Salepeople  
WHERE Salepeople.SNum = Customers.SNum;  
Sodda joylashtirilgan ostki so'rovlar.
```

SQL yordamida so'rovlarni bir birining ichiga joylashtirish ham mumkin. Odatda ichki so'rov qiymat hosil qiladi va bu qiymat tashqi predikat tomonidan tekshirilib, to'g'ri yoki noto'g'riligi tekshiriladi.

Misol: bizga sotuvchi nomi ma'lum: Motika, lekin biz SNUM maydoni qiymatini bilmaymiz va buyurtmachilar jadvalidan hamma buyurtmalarni ajratib olmoqchimiz. Buni quyidagicha amalga oshirish mumkin:

```
SELECT * FROM Orders WHERE SNum =  
( SELECT SNum FROM Salepeople  
WHERE SName = 'Motika' );
```

Agar ostki so'rovda IN operatoridan foydalanilsa, ixtiyoriy sondagi satrlar hosil qilish mumkin. Misol: Londondagi sotuvchilar uchun hamma buyurtmalarni ko'rsatish.

```
SELECT * FROM Orders WHERE SNum IN  
( SELECT SNum FROM Salepeople WHERE City =  
'London' );
```

Bu natijani jamlanma orqali ham hosil qilish mumkin. Lekin odatda ostki so'rovli so'rovlar tezroq bajariladi. Ostki so'rovlarni HAVING izlash sharti ichida ishlatish ham mumkin. Bu ostki so'rovlar agar ko'p qiymatlar qaytarmasa agregat funksiyalaridan yoki GROUP BY yoki HAVING operatorlaridan foydalanishi mumkin. Misol:

```
SELECT Rating, COUNT (DISTINCT CNum) FROM  
Customers
```

GROUP BY Rating

```
HAVING Rating >( SELECT AVG (Rating) FROM
```

Customers

```
WHERE City = 'San Jose' );
```

Bu buyruq San Jose dagi baholari o'rtachadan yuqori bo'lgan buyurtmachilarni aniqlaydi.

UNION ifodasidan foydalanish. UNION ifodasi bir yoki bir necha so'rovlar natijasini birlashtirishga imkon beradi.

Misol: Londonda joylashgan hamma sotuvchilar va buyurtmachilarni bitta jadvalda chiqarish.

```
SELECT SNum, SName FROM Salepeople WHERE City =  
'London'  
UNION  
SELECT CNum, CName FROM Customers WHERE City =  
'London';
```

So'rovlarda funksiyalar

Agregat funksiyalar qo'llanishi.

Agregat (yoki STATIK) funksiyalar sonli yoki hisoblanuvchi ustunlar bilan ishlaydi. Agregat funksiya argumenti butun ustun bo'lib, bitta qiymat qaytaradi. Bu funksiyalarga quyidagilar kiradi:

- SUM() – ustundagi hamma qiymatlar summasini hisoblash.
- AVG() – ustundagi hamma qiymatlar o'rtachasi qiymatini

hisoblash.

- MIN() – ustundagi hamma qiymatlar eng kichigini aniqlash.
- MAX() – ustundagi hamma qiymatlar eng kattasini aniqlash.
- COUNT() – ustundagi qiymati sonini aniqlash.
- COUNT(*) – so'rov natijasi jadvalidagi satrlar sonini

aniqlash.

Agregatlash argumenti bo'lib ustun nomidan tashqari ixtiyoriy matematik ifoda xizmat qilishi ham mumkin. Misol: Sotuv kompaniyada reja bajarilishining o'rtacha protsentini aniqlash.

```
SELECT AVG(100 * (SALES/QUOTA)) FROM  
SALESREPS
```

Masalan, sotuv kompaniyasida sotuvlar xajmini chiqarish.

```
SELECT SUM(QUOTA), SUM(SALES) FROM  
SALESREPS
```

AVG() agregatlash funksiyasiga yana bir sodda misolni ko'ramiz.

Masalan: "ACI" ishlab chiqaruvchi mollari o'rtacha narxini hisoblash.

```
SELECT AVG(PRICE) FROM PRODUCTS  
WHERE MFR_ID = 'ACI'
```

Ekstremumlarni topishda MIN() va MAX() funksiyalari sonli ustunlar, sanalar va satrli o'zgaruvchilar bilan ishlaydi. Eng sodda qo'llanish sonlar bilan ishlash. Masalan, eng ko'p va kam sotuvlar rejadagi hajmini chiqarish.

```
SELECT MIN (QUOTA), MAX (QUOTA) FROM  
SALESREPS
```


Masalan, buyurtmalardan eng oldin berilgan so'rov sanasini topish.

```
SELECT MIN(ORDER_DATE) FROM ORDERS
MBdagi yozuvlar sonini sanash uchun COUNT() qo'llaniladi. Bu funksiya son qiymat qaytaradi. Masalan: kompaniya mijozlari sonini chiqarish.
```

```
SELECT COUNT(CUST_NUM) FROM CUSTOMERS
COUNT(*) funksiyasi satrlar sonini hisoblaydi. Misol
SELECT COUNT(*) FROM ORDERS
```

Agregat funksiyalar jadval uchun natijaviy satr hosil ham qiladi. Masalan: Buyurtma o'rtacha narxini chiqarish.

```
SELECT AVG(AMOUNT) FROM ORDERS
```

4.6. SQL turlari va strukturasi

Yuqorida ta'kidlib o'tilganidek SQL tili tarkibida xisoblash jarayonini boshqarish imkonini beruvchi IF ... THEN ...ELSE, GO TO, DO ... WHILE kabi buyruqlar mavjud emas. Bunday masalalar dasturiy yo'l bilan (dasturlash tili yoki masalalarni boshqarish tili) yoki interaktiv holda (foydalanuvchining so'rovlari asosida) amalga oshiriladi. Imkoniyati cheklanganligi sababli (xisoblash jarayonini boshqarish imkoniyati) SQL tili 2 ta usulda qo'llanilish mumkin. Birinchi usulda *interaktiv* ishlash nazarda tutiladi. Bunda foydalanuvchi SQL operatorlarini terminaldan beradi. Ikkinchi usulda protsedurali tildagi dasturga SQL tili operatorlari kiritiladi.

Interaktiv rejimda ma'lumotlar bazasi bilan ishlashda foydalanuvchi muloqat rejimida ishlaydi, ya'ni SQL tilidagi so'rovni kiritadi va natijani oladi, yani so'rovni kiritadi va natijaga ega bo'ladi va x.k.

Kiritilgan SQL rejimida boshqa dasturlash tillarida yaratigan dastur tarkibiga kiritiladi. Bu ma'lumotlar bazasi bilan boshqa algoritmik tillarda yaratilgan amaliy dasturlar orqali ishlashni ta'minlaydi. Biroq bu yerda qo'shimcha dasturiy vosita kerak bo'ladi. U dasturlash tili bilan SQL operatorlari o'rtasidagi interfeysni ta'minlab beradi.

Atamalar

SQL til yordamida ma'lumotlar bazasiga beriladigan so'rov deganda joriy buyruq tavsiflagan va ma'lumotlar bazasini boshqarish tizimi tomonidan bajarish uchun mo'ljallangan ma'lumotlar bazasi ustida bajariladigan buyruq tushuniladi.

So'rov SQL tili operatorlari yordamida yaratiladi. Operatorlar so'zlar deb ataluvchi aloxida ma'noli qismlardan tashkil topadi. Operatorlar sintaksisi SQL tilining standartida belgilab berilgan.

SQL tili relyatsion ma'lumotlar bazasi bilan ishlashiga qaramasdan "munosabat" atamasi o'miga "jadval" atamasi, "kortej" va "atribut" atamalari o'miga "satr" va "ustun" atamalari ishlatiladi.

SQL strukturasi va operatorlari

SQL tili 1 jadvalda tasvirlangan bo'limlardan iborat.

4.1-jadval. Ma'lumotlarni aniqlash operatorlari DDL (Data Definition Language — ma'lumotlarni aniqlash tili)

Operator	Ma'nosi	Amal
CREATE TABLE	Jadval yaratish	MB da yangi jadval yaratish
DROP TABLE	Jadvalni o'chirish	MB dan jadvalni o'chirish
ALTER TABLE	Jadvalni o'zgartirish	Mavjud jadval strukturasi o'zgartirish yoki joriy jadval uchun o'rnatilgan butunlik cheklanishlarini o'zgartirish
CREATE VIEW	Tasvir yaratish	Biror SQL -so'rovga mos virtual jadvalni yaratish

4.2-jadval. Ma'lumotlarni manipulyatsiyalash operatorlari Data Manipulation Language (DMP)

Operator	Ma'nosi	Amal
DELETE	Satrni o'chirish	Filtr mezonlariga mos keladigan bir yoki bir nechta qatorlarni asosiy jadvaldan olib

		tashlaydi. Operatoridan foydalanish samoviylikni saqlash tamoyillariga mos keladi, shuning uchun bu operator sintaktik jihatdan to'g'ri yozilgan bo'lsa ham, har doim ham to'g'ri bajarilmasligi mumkin.
INSERT	Satmi qo'yish	Jadvalga bitta satrni qo'yish. Bu operatorning boshqa modifikatsiyalarida biror jadvalning bir necha stari yoki so'rov natijasi jadvalga qo'yilishi mumkin.
UPDATE	Satmi yangilash	Bitta yoki filtratsiya shartini qanoatlantiruvchi bir necha ustunlardagi bitta yoki ko'proq ustunlardagi qiymatlarni almashtirish

4.3-jadval. Data Query Language (DQL) so'rov tili

Operator	Ma'nosi	Amal
SELECT	Satmi tanlash	Relyatsion algebraning barcha amallarini bajaruvchi operator bo'lib so'rovga mos natijaviy jadvalni xosil qiladi.

4.4-jadval. Tranzaksiyalarni boshqarish operatorlari

Operator	Ma'nosi	Amal
COMMIT	Tranzaksiyani tugatish	Tranzaksiyani tashkil qiluvchi ma'lumotlarni qayta ishlovchi murakkab va o'zaro bog'langan amallarni tugatish
ROLLBACK	Tranzaksiyani bekor qilish	Tranzaksiya bajarilishi natijasida yuz bergan o'zgarishlarni bekor qilish
SAVEPOINT	Tranzaksiya bajarilishida oraliq nuqta saqlash	MB ni oraliq xolatini saqlash. Bu keyinchalik shu xolatga qaytish uchun zarur bo'ladi.

4.5-jadval. Ma'lumotlarni boshqarish operatorlari DCL (Data Control Language) – ma'lumotlarni boshqarish

Operator	Ma'nosi	Amal
ALTER DATABASE	MB o'zgartirish	Butun ma'lumotlar bazasiga tegishli bo'lgan obyektlar va cheklanishlar to'plamini o'zgartirish
ALTER DBAREA	MB saqlash soxasini o'zgartirish	Avval yaratilgan saqlash soxasini o'zgartirish
ALTER PASSWORD	Parolni o'zgartirish	Butun ma'lumotlar bazasi parolini o'zgartirish
CREATE DATABASE	MB yaratish	Barcha parametrlarini ko'rsatgan holda yangi ma'lumotlar bazasini yaratish
CREATE DBAREA	Saqdash soxasini yaratish	Yangi saqlash soxasini yaratish va unda ma'lumotlar joylashtirishga ruxsat etishi
DROP DATABASE	MB ni o'chirish	Mavjud ma'lumotlar bazasini o'chirish (bunday amal bajarish vakolatiga ega bo'lgan foydalanuvchilar uchun)
DROP DBAREA	MB saqlash soxasini o'chirish	Mavjud saqlash soxasini o'chirish (agar unda joriy vaqtda faol ma'lumotlar joylashmagan bo'lsa)
GRANT	Vakolat berish	MB ba'zi obyektlari ustida bir qator amallarni bajarish xuquqini berish
REVOKE	Xuquqdan maxrum qilish	Biror obyektga yoki obyekt ustida bajariladigan ba'zi amallarni bajarish xuquqidan maxrum qilish

Ko'pincha ustunlardagi qiymatlarning maksimal, minimal va o'rtacha qiymatlarini hisoblashga to'g'ri keladi. Masalan, o'rtacha ballni hisoblash zarurati paydo bo'ladi. Bunday hisoblashlarni bajarish uchun SQL tilida maxsus agregat funksiyalari mavjud:

MIN – ustundagi minimal qiymat;

MAX – ustundagi maksimal qiymat;

SUM – ustundagi qiymatlar yig'indisi;
 AVG – ustundagi qiymatlarning o'rtachasi;
 COUNT – ustundagi NULL dan farqli bo'lgan qiymatlar miqdori.
 Quyidagi so'rov talabalar imtixonlarda olgan ballarning o'rtachasini aniqlaydi.
 SELECT AVG(mark) FROM mark_st

SELECT ma'lumotlarni tanlash operatori

Ma'lumotlarni tanlash SELECT operatori yordamida bajariladi. Bu SQL tilining eng ko'p qo'llaniladigan operatori xisoblanadi. SELECT operatorini sintaksisi quyidagicha:

```
SELECT [ALL/DISTINCT] <atributlar ro'yxati> *
FROM <jadvallar ro'yxati >
[WHERE <tanlash sharti >]
[ORDER BY < atributlar ro'yxati >]
[GROUP BY < atributlar ro'yxati >]
[HAVING <shart>]
[UNION< SELECT operatorli ifoda>]
```

Kvadrat qavslrda operatori yozishda qatnashishi shart bo'lmagan elementlar ko'rsatilgan. ALL kalit so'zi natijaga shartni qanoatlantiruvchi barcha satrlar, shuningdek takrorlanuvchi satrlar ham kirishini bildiradi. DISTINCT kalit so'zi natijaga takrorlanuvchi satrlar kiritilmasligini bildiradi. Keyin bolang'ich jadvaldagi atributlar ro'yxati ko'rsatiladi. Bu atributlar natijaviy jadvalga kiritiladi. * simvoli natijaviy jadvalga boshlang'ich jadvalning barcha atributlari kiritilishini bildiradi.

Operatorda qatnashishi shart bo'lgan so'zlardan FROM so'zi xisoblanadi. Bu so'zdan keyin tanlov bajariladigan jadvallar nomi ko'rsatiladi.

Tanlash ifodasida WHERE kalit so'zidan keyin jadval satrlarini tanlab olish sharti ko'rsatiladi. Bunda natijaviy jadvalga WHERE ifodasidagi shart rost qiymat qabul qiladigan satrlar kiritiladi.

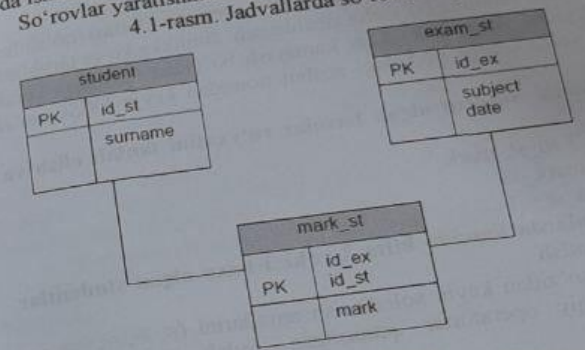
ORDER BY kalit so'zi natijaviy jadval satrlarini ko'rsatilgan ustunlar ro'yxati bo'yicha tartiblash amalini bildiradi.

GROUP BY kalit so'zidan keyin gruppalanadigan atributlar ro'yxati ko'rsatiladi.

HAVING ifodasida har bir gruppaga qo'yiladigan shartlar

ko'rsatiladi. (GROUP BY va HAVING kalit so'zlari keyinroq tushuntiriladi)
 FROM, WHERE va ORDER BY kalit so'zlari SQL tilining qolgan ma'lumotlarni manipulyatsiyalash operatorlarida ham shu tarzda ishlatiladi.
 So'rovlar yaratishni aniq misol uchun ko'rib o'tamiz (4.1-rasm)

4.1-rasm. Jadvallarda so'rovlarni yaratish.



Student		exam_st		
id_st	Surname	id_ex	Subject	Date
1	Xasanov	1	Matem	10.05.2009
2	Karimov	2	Fizika	15.05.2009
3	Jabborov	3	Informatika	25.05.2009

mark_st		
id_ex	id_st	mark
2	2	3
3	1	2
1	1	4
3	2	5
2	1	3
1	2	2
2	3	4
1	3	5
3	3	3

Barcha studentlar ro'yxatini tanlash va tasvirlash.

```
SELECT *  
FROM student
```

yoki
SELECT id_st, surname
FROM student

Agar ushbu so'rovga ORDER BY surname ifodasi qo'shilsa, uxolda ro'yhat familiya bo'yicha tartiblanadi. Jimlikka ko'ra tartiblash o'sish bo'yicha bajariladi. Agar kamayish bo'yicha tartiblash kerak bo'lsa, u xolda oxirgi ifodadagi atribut nomidan keyin DESC so'zi qo'shiladi.

«1» kodli student olgan baxolar ro'yxatini tanlab olish va tasvirlash

```
SELECT id_st, mark  
FROM mark_st  
Where id_st = 1
```

Ekzamenlarda kamida bitta 2 yoki 3 baxo olgan studentlar kodini tanlab olish

WHERE so'zidan keyin solishtirish amallarini (<, >, =, <> va x.k.) va mantiqiy operatorlar qatnashgan ifodalarni joylashtirish mumkin.

```
SELECT id_st, mark  
FROM mark_st  
WHERE ( MARK >= 2 ) AND ( MARK <= 3 )
```

SQL tilida shart ifodalarini tuzish uchun solishtirish va mantiqiy operatorlardan tashqari yana bir qator maxsus operatorlar qo'llaniladi. Bu operatorlar dasturlash tillarida mavjud emas. Bu operatorlar:

IN – biror qiymatlar to'plamiga tegishlilikini tekshirish;
BETWEEN – biror qiymatlar diapozoniga tegishlilikini tekshirish;

LIKE – namuna bilan mosligini tekshirish;

IS NULL – qiymat mavjudmasligini tekshirish.

IN operatori biror qiymatlar to'plamiga tegishlilikni tekshirish uchun ishlatiladi.

Quyidagi so'rov oxirgi keltirilgan misoldagi so'rov natijalarini beradi (ekzamenlarda kamida bitta 2 yoki 3 baho olgan studentlarni identifikatorini tanlab oladi)

```
SELECT id_st, mark
```

```
FROM mark_st  
WHERE mark IN (2,3)  
Xuddi natijani BETWEEN operatoridan foydalanib olish  
mumkin:  
SELECT id_st, mark  
FROM mark_st  
WHERE mark BETWEEN 2 AND 3
```

Familiyalari A xarfi bilan boshlanuvchi studentlar ro'yxatini tanlab olish

Bunday xolatda LIKE operatoridan foydalanish qulay xisoblanadi. LIKE operatori faqat simvolli maydonlar uchun qo'llaniladi va maydon qiymati operatorida ko'rsatilgan namunaga mosligini tekshirish imkonini yaratadi. Namuna quyidagi maxsus simvollaridan tashkil topadi:

_ (tagiga chizish belgisi) – bitta ixtiyoriy simvolni bildiradi;
% (foiz belgisi) – ixtiyoriy miqdordagi simvollar ketma – ketligini bildiradi.

```
SELECT id_st, surname  
FROM student
```

```
WHERE surname LIKE 'A%'
```

Ko'pincha ustunlardagi minimal, maksimal yoki o'rtacha qiymatlarni xisoblashga to'g'ri keladi. Masalan, keltirilgan jadvallarda o'rtacha baxoni xisoblash mumkin. Bunday xisoblashni bajarish uchun SQL tilida maxsus agregat funksiyalari mavjud:

MIN() – ustundagi minimal qiymat;

MAX() – ustundagi maksimal qiymat;

SUM() – ustundagi qiymatlar summasi;

AVG() – ustundagi qiymatlarni o'rtachasi;

COUNT() – ustundagi NULL dan farqli qiymatlar miqdori.

Quyidagi so'rov studentlar olgan baxolarning o'rtachasini xisoblaydi.

```
SELECT AVG(mark)  
FROM mark_st
```

Albatta, agregat funksiyalarini WHERE so'zi bilan birgalikda qo'llash mumkin. Quyidagi so'rov 100 kodli studentning egzamenlarda olgan baxolarining o'rtachasini xisoblaydi:

```
SELECT AVG(mark)
```

```
FROM mark_st  
WHERE id_st = 100  
Navbatdagi so'rov 10 kodli studentning o'rtacha baxosini
```

```
xisoblaydi:  
SELECT AVG(mark)  
FROM mark_st  
WHERE id_ex = 10  
SQL tili ko'rib o'tilgan mexanizmiga qo'shimcha ravishda
```

agregat funksiyalarni to'la jadval uchun emas, balki gruppallangan qiymatlar uchun qo'llash imkonini beruvchi vosita mavjud. Buning uchun SQL da maxsus GROUP BY konstruksiyasi mavjud bo'lib unda ko'rsatilgan ustun qiymatlari bo'yicha gruppalash amalga oshiriladi. Masalan, har bir studentni ekzamenlarda olgan baxolarining o'rtachasini aniqlash mumkin. Buning uchun quyidagi so'rov beriladi:

```
SELECT id_st, AVG(mark)  
FROM mark_st  
GROUP BY id_st
```

Bu imkoniyat ham odatdagidek WHERE so'zi bilan birgalikda qo'llanilishi mumkin. Bu so'rovni bajarishda MBBT avval jadvaldan WHERE ifodasidagi shartni qanoatlantiruvchi satrlarni tanlab oladi, keyin tanlangan satrlarni gruppalashva agregatlash amalini bajaradi.

Quyidagi so'rov har bir studentning 100 kodli ekzamendan olgan baxolarining o'rtachasini aniqlaydi.

```
SELECT id_st, AVG(mark)  
FROM mark_st  
WHERE id_ex = 100  
GROUP BY id_st
```

Ko'rib turganimizdek gruppalash bittadan ko'p maydon bo'yicha bajarilishi mumkin. GROUP BY seksiyasiga ega bo'lgan so'rovlar uchun quyidagi muxim cheklanish mavjud: bunday so'rovlar natijasi tarkibiga gruppalash bajarilgan ustunlar va agregatlash natijasiga ega bo'lgan ustunlar kiradi.

Biror xulosaga kelishdan oldin SQL tilining barcha imkoniyatlarini ko'rib chiqish kerak. Masalan, ixtiyoriy tekstni so'rov tarkibiga kiritish mumkin. Bunga misol keltiramiz:

```
SELECT 'O'rtacha ball =', AVG(mark)  
FROM mark_st  
WHERE id_ex = 10
```

Ushbu so'rov natijasida foydalanuvchi faqat oddiy sonlarni emas, balki tekstni ham ko'radi.

Bir nechta jadvallardan ma'lumotlarni tanlash uchun SQL ni qo'llash

Sh paytgacha faqat bitta jadvaldan ma'lumotlarni tanlash miollari keltirildi. Relyatsion amallarga mos bo'lgan bir nechta jadvallardan ma'lumotlar tanlab olishni ham bajarish mumkin. Bir nechta jadvallardan ma'lumotlarni tanlab olishga to'liq misollar keltirish imkoni yo'q. Bunga doir ba'zi misollarni ko'rib o'tamiz.

Qoidaga ko'ra, ma'lumotlar tanlab olinadigan jadvallar u yoki bu shaklda bir biri bilan bog'langan. Masalan, birga ko'p va x.k.

1 rasmdagi ER-diagrammaga qarang. Bu miolda bog'langan jadvallar mavjud. student, mark_st va exam_st jadvallari ko'rib chiqamiz.

mark_st jadvali id_ex maydoni bo'yicha exam_st jadvali bilan bog'langan.
mark_st jadvali id_st maydoni bo'yicha student jadvali bilan bog'langan.

Masalan, studentlarni ro'yxatini ular ekzamenlardan olgan baxolari bilan birgalikda tanlash zarur bo'lsin. Buning uchun quyidagi so'rov beriladi:

```
SELECT student.surname, mark_st.id_ex, mark_st.mark  
FROM student, mark_st  
WHERE student.id_st = mark_st.id_st
```

Keltirilgan ko'p jadvali so'rov bir jadvali so'rovdan quyidagilar bilan farq qiladi.

1. FROM seksiyasida ikkita jadval ko'rsatilgan.
2. jadvalr soni bitta ko'p, shuning uchun ko'rsatilgan maydonlar nomining bir qiymatligi yo'qoladi. Masalan, ko'p xollarda maydonni FROM da ko'rsatilgan jadvallar ro'yxatidagi qaysi jadvaldan olish noma'lum bo'lib qoladi. Maydon nomlarining ko'p qiymatligini bataraf etish uchun maydon nomida perefiks - jadval nomi qo'shimcha qilinadi. Jadval nomi maydon nomidan nuqta bilan ajratiladi.

3. WHERE ifodasida jadvallarni birlashtirish sharti ko'rsatiladi. Ko'rinib turganidek jadval nomidan iborat prefiksdan foydalanish so'rovni murakkablashtiradi. Bunday murakkablikni bartaraf etish

uchun psevdonim ishlatiladi. Yuqoridagi so'rovni quyidagicha yozish mumkin:

```
SELECT E.surname, M.id_ex, M.mark  
FROM student E, mark_st M  
WHERE E.id_st = M.id_st
```

INSERT ma'lumotlarni kiritish operatori:

```
INSERT INTO jadval_nomi [(<ustunlar ro'yxati >)] VALUES  
(<qiymatlar ro'yxati >)
```

Bunday sintaksis jadvalga faqat bitta star kiritish imkonini beradi. Agar satrdagi barcha ustunlarga qiymat kiritilayotgan bo'lsa, so'rovda barcha ustunlar nomini ko'rsatish zarur emas.

Masalan, BOOKS jadvaliga yangi kitob ma'lumotlari kiritiladi

```
INSERT INTO BOOKS ( ISBN, TITL, AUTOR, COAUTOR,  
YEARIZD, PAGES)  
VALUES ("5-88782-290-2", "Apparatsnye sredstva IBM  
PC. Ensiklopediya",
```

"Guk M.", "", 2000, 816)

Bu kitob avtori faqat bitta va soavtor (hammuallif) mavjud emas, biroq ustunlar ro'yxatida COAUTOR ustuni ham ko'rsatilgan. Shuning uchun VALUES bo'limida bu ustunga mos qiymatni ko'rsatish zarur. Misolda bu maydon uchun bo'sh satr ("") ko'rsatilgan. Bu soavtor yo'qligini bildiradi. Shuningdek bu yerda aniqlangmagan NULL qiymatini ko'rsatish ham mukin edi.

Satrdagi barcha ustunlarga qiymat kiritishda ustunlar ro'yxatini ko'rsatish zarur emas. Bunda faqat qiymatlar ro'yxatini ko'rsatish yetarli bo'ladi. Bunday xolda operator ko'rinish quyidagicha shaklda bo'ladi:

```
INSERT INTO BOOKS VALUES ("5-88782-290-2",  
"Apparatsnye sredstva IBM PC. Ensiklopediya". "Guk  
M.", "", 2000, 816)
```

Misolda keltirilgan ikkiala operator ham bir xil amalni bajaradi.

Shuningdek to'liq miqdorda bo'lmagan qiymatlarni ko'rsatish mumkin. Ya'ni iymatlar qatorida soavtorni ko'rsatmaslik mumkin, chunki jeoriy kitobda soavtor yo'q. Biroq bunda qiymat kiritiladigan ustun nomlarini quyidagicha shaklda ko'rsatish kerak bo'ladi:

```
INSERT INTO BOOKS ( ISBN, TITL, AUTOR,
```

```
YEARIZD, PAGES)
```

```
VALUES ("5-88782-290-2", "Apparatsnye sredstva IBM PC.  
Ensiklopediya". Guk M.", 2000, 816)
```

Bu xolda COAUTOR ustuniga NULL qiymati yoziladi.

Agar jadvalni yaratishda ustun yoki atributga majburiy qiymat (NOT NULL) belgisi qo'yilgan bo'lsa, u xolda INSERT operatorida joriy ustunning har bir satriga kiritiladigan qiymatg ko'rsatilishi kerak. Shuning uchun, agar jadvalning hamma ustuni majburiy qiymat li bo'lsa, u xolda har bir yangi kiitladigan satrda barcha ustun uchun qiymat mavjud bo'lishi kerak va bunda ustunlar ro'yxatini ko'rsatish shar emas. Aks xolda jadvalda kamida bitta maburiy qiymatli bo'lmagan ustun bo'lsa, u xolda albatta ustunlar ro'yxatini ko'rsatish shart bo'ladi.

Qiymatlar ro'yxatida maxsus funksiyalar va ifodalar ko'rsatish ham mumkin. Bunda ushbu funksiyalarning qiymatlari ma'lumotlarni kiritish momentida xisoblangan bo'lishi zarur.

Ma'lumotlarni kiritish operatori birdaniga bir necha satrlarni kiritish imkoniga ham ega. Bunda qiymatlar satri boshqa bir jadvaldan tanlab olinadi. Masalan studentlar xaqidagi jadval mavjud bo'lsin. Unda studentlarning familiyasi, adresi, uy telefoni va tug'ilgan sanasi ko'rsatilgan bo'lsin. U xolda bitta operator yordamida ularni bibliotekaning kitobxonlariga aylantirish mumkin:

```
INSERT INTO READER (FIO_students, Adres, Telefon,  
Data_roid)  
SELECT (FIO_students, Adres, Telefon, Den_roid)  
FROM STUDENT
```

DELETE o'chirish operatori:

Ma'lumotlarni o'chirish operatori jadvaldan shartni qanoatlantiruvchi bir yoki bir neta satrlarni o'chirishi mumkin.

```
DELETE FROM jadval_nomi [WHERE tanlash_sharti]
```

Agar satrlarni tanlash sharti ko'rsatilmasa, u xolda jadvaldagi barcha satrlar o'chiriladi. Natijada ma'lumotlarga ega bo'lmagan bo'sh bo'lgan jadval xosil bo'ladi.

Agar jadvaldan oldingi sessiya natijalarini o'chirish kerak bo'lsa, u xolda R1 jadvalidagi barcha satrlar o'chiriladi:

```
DELETE FROM R1
```

WHERE qismidagi shart ifodasi xuddi SELECT operatoridagi filtrlash shartiga o'xshash bo'ladi. Bu shart jadvaldan qaysi satrlar o'chirilishi kerakligini aniqlaydi.

Masalan, student Mironova A.V. o'chirilmaligi kerak bo'lsa, quyidagi so'rov beriladi:

```
DELETE FROM R2 WHERE FIO = "Mironov A.V."
```

WHERE qismida biror so'rov ko'rsatilishi mumkin. Masalan, agar jadvaldan o'zlashtirmagan studentlarni o'chirish kerak bo'lsin. Oliy ta'lim qonuniga ko'ra oxirgi sessiyada ikkita va undan ortiq fandan ikki baxo olgan student o'zlashtirmagan xisoblanadi. U xolda tanlab olish sharti ikkita va undan ko'p ikki baxo olgan studentlarni va ikkita undan ko'p ekzamenlarni topshirmagan studentlarni aniqlashi kerak. Bunday studentlarni aniqlash uchun R1 jadvalidan 2 baxoli va baxo ko'rsatilmagan satrlar tanlab olinishi, keyin olingan natija FIO ustuni ko'rsatilmagan satrlar tanlab olinishi, keyin har bir gruppada satrlar soni aniqlanadi (bu har bir studentning olgan ikki baxolari bilan topshirmagan ekzamenlar sonini bildiradi) va ikkitadan ko'p satrga ega ustunlar tanlab olinadi. Endi ushbu murakkab bo'lgan konstruksiyani SQL tilida yozamiz va sodda ko'rinishga ega bo'lishini ko'ramiz.

```
DELETE FROM R2 WHERE R2.FIO IN (SELECT R1.FIO
FROM R1
```

```
WHERE Otsenka = 2 OR Otsenka IS NULL GROUP BY R1.FIO
HAVING COUNT(*) >= 2
```

DELETE operatsiyasini bajarishda unda qatnashgan qism so'rovda satrlar o'chiriladigan jadval ko'rsatilmaligi kerak.

Ma'lumotlarni manipulyatsiyalash operatsiyalarining barchasi ma'lumotlar bazasining butunligi tushunchasi bilan bog'langan. Manipulyatsiyalash amallari sintaktik jixatdan to'g'ri bo'lsada butunlik talablari tufayli bajarilmaligi mumkin.

UPDATE ma'lumotlarni yangilash operatsiyasi

o'zgarish yuz berganda va mos xolda bu o'zgarishni ma'lumotlar bazasida akslantirish uchun ishlatiladi.

```
UPDATE jadval_nomi SET ustun_nomi = yangi_qiymat
[WHERE tanlash_sharti]
```

Bu yerda ham WHERE qismi DELETE operatoridagi kabi ko'rsatilishi shart emas. U DELETE operatoridagi kabi bir xil vazifani

bajaradi va o'zgartirish amali bajariladigan satrlarni tanlash imkonini beradi. Agar tanlash sharti (WHERE qismi) ko'rsatilmagan bo'lsa, u xolda o'zgartirish amali jadvalning barcha satrlari uchun bajariladi.

Masalan, student Stepanova K. Ye. ma'lumotlar bazasi fandan "2" baho oldi, keyin uni "3" bahoga qayta topshirgan bo'lsin. Bu xolatga mos xolda R1 jadvalini o'zgartirish amali quyidagi operator bilan amalga oshiriladi:

```
UPDATE R1
SET R1.Otsenka = 3
WHERE R1.FIO = "Stepanova K.Ye." AND R1.Dissiplina =
```

"Bazы dannыx"

Qanday xolatlarida bir nechta satrlarni o'zgartirish zarurati paydo bo'ladi? Bu kam uchaydigan masala emas. Masalan, agar guruxlar jadvalidagi guruxlarni kursini bittaga oshirish zarr bo'lsa quyidagi o'zgartirish amali bajarishi mumkin. Guruxlar jadvali quyidagicha sxemaga ega bo'lsin:

```
R4 = < Gurux, Kurs>
UPDATE R4
SET R4.Kypc = R4.Kypc + 1
```

Ma'lumotlarni turi. Ma'lumotlar bazasini yaratish

MYSQL tilida jadvaldagi maydonlarni quyidagi tiplari mavjud

[2]: 4.6 - jadval. Butun sonli ma'lumotlar tiplari.

Tip	Ta'rifi
TINYINT	-128 dan +127 gacha sonlarni saqlashi mumkin
SMALLINT	-32 768 dan 32 767 gacha
MEDIUMINT	-8 388 608 ... 8 388 607
INT	Diapazoni -2 147 483 648 ... 2 147 483 647
BIGINT	Diapazoni -9 223 372 036 854 775 808 ... 9 223 372 036 854 775 807

4.7 - jadval. Ratsional sonlar tiplari

Tip	Ta'rif
FLOAT	kichik aniqlikdagi o'zgaruvchan vergulli son
DOUBLE	lkkilangan aniqlikdagi o'zgaruvchan vergulli son
REAL	DOUBLE sinonimi
DECIMAL	satr shaklida saqlanuvchi kasr son
NUMERIC	DECIMAL sinonimi

4.8 - jadval.. Satrlar tiplari

Tip	Ta'rif
TINYTEXT	255 ta simvolgacha bo'lgan satr
TEXT	65535 simvolli satr
MEDIUMTEXT	16 777 215 gacha bo'lgan simvol saqlashi
mumkin	
LONGTEXT	4 294 967 295 SIMVOL

4.9 - jadval. Binar ma'lumotlar

Tip	Ta'rif
TINYBLOB	255 ta simvol saqlash mumkin
BLOB	65535 simvolli satr
MEDIUMBLOB	16 777 215 gacha bo'lgan simvol saqlashi
mumkin	
LOB	4 294 967 295 SIMVOL

4.10 - jadval. Sana va vaqt

Tip	Ta'rif
DATE	yyyy-oo-ss formatdagi sana
TIME	ss:mm:ss formatdagi vaqt
DATETIME	yyyy-oo-ss ss:mm:ss formatdagi sana va vaqt
TIMESTAMP	timestamp formatdagi sana va vaqt

Ma'lumotlarni aniqlash

SQL ning DDL (Data Definition Language) ma'lumotlarni aniqlash tili ma'lumotlar bazasining sxema, domenlar, jadvallar, tasvirlar va indekslar kabi obyektlarini yaratish va o'chirish uchun ishlatiladi.

Quyida SQL da ma'lumotlarni aniqlash tilining asosiy operatorlari nomi keltirilgan.

CREATE SCHEMA
 DROP SCHEMA
 CREATE DOMAIN
 DROP DOMAIN
 CREATE TABLE
 DROP TABLE
 CREATE VIEW
 DROP VIEW

Bu operatorlar konseptual sxemaning tarkibiga kiruvchi strukturalarni yaratish, taxrirlash va o'chirish uchun ishlatiladi. Ba'zi MBBT larda quyidagi 2 ta operator xam mavjud:

ALTER DOMAIN
 ALTER TABLE
 DROP INDEX

Jadvallarni yaratish

Jadvallar CREATE TABLE komandasi bilan yaratiladi. Bu komanda qatorlarsiz bo'sh jadval yaratadi. CREATE TABLE komandasi jadval nomini va jadvalning o'zini ko'rsatilgan tartibdagi ustunlar ketma - ketligi ko'rinishida aniqlaydi. Unda har bir ustundagi ma'lumotlarning tiplari va ustunlar o'lchovi ko'rsatiladi. Har bir jadval juda bo'lmaganda bitta ustunga ega bo'lishi kerak.

CREATE TABLE komandasi sintaksisi:

```
CREATE TABLE <table-name >
( <column name> <data type>[( <size>)],
  <column name> <data type>[( <size>)], ... );
```

Argument qiymati kattaligi ma'lumot turiga bog'liqdir. Agar siz maxsus ko'rsatmasangiz, tizim avtomatik qiymatni o'rnatadi.

Misol uchun sotuvchilar jadvalini yaratishni ko'rib chiqamiz:
 CREATE TABLE Salepeople
 (SNum integer,


```
SName char (10),
City char (10),
Comm decimal);
```

Jadvallarni o'chirish

Jadvalni o'chirish imkoniga ega bo'lish uchun, jadval egasi (Ya'ni yaratuvchisi) bo'lishingiz kerak. Faqat bo'sh jadvalni o'chirish mumkin. Qatorlarga ega bo'lgan, to'ldirilgan jadvalni o'chirish mumkin emas, Ya'ni jadval o'chirishdan oldin tozalangan bo'lishi kerak. Jadvalni o'chirish komandasi quyidagi ko'rinishga ega:

```
DROP TABLE <table name >;
```

Masalan: **DROP TABLE Salepeople;**

Jadvalni o'zgartirish

Jadvalni o'zgartirish uchun ALTER TABLE komandasidan foydalaniladi. Bu komanda jadvalga Yangi ustunlar qo'shish, ustunlarni o'chirish, ustunlar kattaligini o'zgartirish, hamda cheklanishlarni qo'shish va olib tashlash imkoniyatlariga ega. Bu komanda ANSI standarti qismi emas, shuning uchun har xil tizimlarda har xil imkoniyatlarga ega.

Jadvalga ustun qo'shish uchun komandaning tipik sintaksisi:

```
ALTER TABLE <table name> ADD <column name>
<data type> <size>;
```

Masalan:

```
ALTER TABLE Salepeople ADD Phone CHAR(7);
```

Jadvallar uchun cheklanishlar

Cheklanishlarni aniqlash. Ko'p xollarda ustunga kiritilgan qiymatlar bir biridan farq qilishi kerak. Agar ustun uchun UNIQUE cheklanishi o'tatilsa, bu ustungsha mavjud qiymatni kiritishga urinish rad etiladi. Bu cheklanish bo'sh bo'lmaydigan (NOT NULL) debe'lon qilingan maydonlarga qo'llanishi mumkin.

Masalan:

```
CREATE TABLE Salepeople
```

```
(SNum integer NOT NULL UNIQUE,
```

```
SName char (10),
City char (10),
Comm decimal);
```

Unikalligi talab qilinadigan maydonlar (birlamchi kalitlardan tashqari) kandidat kalitlar yoki unikal kalitlar deyiladi.

Jadval cheklanishi UNIQUE maydonlar guruxiga o'tmatilishi mumkin. Bu bir necha maydonlar qiymatlari kombinatsiyasi unikalligini ta'minlaydi. Bizning ma'lumotlar bazamizda har bir buyurtmachi bitta sotuvchiga birlashtirilgan. Ya'ni Buyurtmachilar jadvalida buyurtmachi nomeri (cnum) va sotuvchi nomeri (snum) kombinatsiyasi unikal bo'lishi kerak. Bu cheklanishni UNIQUE (cnum, snum) yordamida, Customers jadvalini yaratishda kiritish mumkin. Bu ustunlar uchun NOT NULL cheklanishini kiritish zarurdir.

Birlamchi kalitlar cheklanishlari. SQL birlamchi kalitlarni to'g'ridan to'g'ri birlamchi kalit (PRIMARY KEY) cheklanishi orqali ta'riflaydi. PRIMARY KEY jadvalni yoki ustunlarni cheklashi mumkin. Bu cheklanish UNIQUE cheklanishi kabi ishlaydi, jadval uchun faqat bitta birlamchi kalit (ixtiyoriy sondagi ustunlar uchun) aniqlanishi mumkin bo'lgan xoldan tashqari. Birlamchi kalitlar NULL qiymatga ega bo'lishi mumkin emas.

Misol:

```
CREATE TABLE Salepeople
(SNum integer NOT NULL PRIMARY KEY,
SName char (10),
City char (10),
Comm decimal);
```

Maydon qiymatlarini tekshirish (CHECK cheklanishi). CHECK cheklanishi jadvalga kiritilayotgan ma'lumot qabul qilinishidan oldin mos kelishi lozim bo'lgan shart kiritishga imkon beradi. CHECK cheklanishi CHECK kalit so'zi ko'rsatilgan maydondan foydalanuvchi predikat ifodalaridan iboratdir.

Misol: Salepeople jadvali Comm ustuniga kiritilayotgan qiymat 1 dan kichik bo'lish sharti.

```
CREATE TABLE Salepeople
```

```
(SNum integer NOT NULL PRIMARY KEY,
SName char(10) NOT NULL UNIQUE,
City char(10),
```

Comm decimal CHECK (Comm < 1));

Ko'zda tutilgan qiymatlarni (poumolchaniyu) o'rnatish. Biror bir maydon uchun qiymat ko'rsatmagan xolda jadvalga satr qo'shsangiz, SQL bunday maydonga kiritish uchun ko'zda tutilgan qiymatga ega bo'lishi kerak, aks xolda komanda rad etiladi. Eng umumiy ko'zda tutilgan qiymat NULL qiymatdir. CREATE TABLE komandasida ko'zda tutilgan qiymat DEFAULT operatori orqali, ustun cheklanishi sifatida ko'rsatiladi. Masalan:

```
CREATE TABLE Salepeople  
(SNum integer NOT NULL PRIMARY KEY,  
SName char(10) NOT NULL UNIQUE,  
City char(10) DEFAULT 'New York',  
Comm decimal CHECK ( Comm < 1 ));
```

Ma'lumotlar yaxlitligini ta'minlash

Jadval bir maydonidagi hamma qiymatlar boshqa jadval maydonida aks etsa, birinchi maydon ikkinchisiga ilova qiladi deyiladi. Bu ikki maydon orasidagi bog'liqlikni ko'rsatadi. Masalan, buyurtmachilar jadvalida har bir buyurtmachi, sotuvchilar jadvalida o'ziga birlashtirilgan sotuvchiga ilova qiluvchi SNum maydoniga ega. Bir maydon ikkinchisiga ilova qilsa tashqi kalit, u ilova qilayotgan maydon ajdod kalit deyiladi. Buyurtmachilar jadvalidagi SNum maydoni tashqi kalit, sotuvchilar jadvalidagi SNum - ajdod kalitdir.

Tashqi kalit bitta maydondan iborat bo'lishi shart emas. Birlamchi kalit kabi, tashqi kalit bitta modul sifatida qayta ishlanuvchi bir necha maydonlarga ega bo'lishi mumkin. Maydon tashqi kalit bo'lsa ilova qitlayotgan jadval bilan ma'lum usulda bog'liqdir. Tashqi kalit har bir qiymati (satri), ajdod kalitning bitta va faqat bitta qiymatiga (satri) ilova qilishi kerak. Bu xolda tizim ilovali yaxlit xolatda deyiladi

Shu bilan birga ajdod kalit qiymati tashqi kalit bir necha qiymatlariga ilova qilishi mumkin.

Cheklanish FOREIGN KEY. SQL ilovali yaxlitlikni FOREIGN KEY yordamida ta'minlaydi. Tashqi kalit vazifasi ajdod kalitda ko'rsatilmagan qiymatlarni tashqi kalit maydonlariga kiritmaslikdir. FOREIGN KEY cheklanishi sintaksisi:

```
FOREIGN KEY <column list> REFERENCES  
<pktable> [<column list>]
```

Birinchi ro'yhat komanda tomonidan o'zgartiriluvchi ro'yxatdir. Pktable - bu ajdod kalitli jadval. Ikkinchi ustunlar ro'yxat bo'lgan ajdod kalitni tashkil qiluvchi ustunlardir. Misol uchun Sotuvchilar jadvaliga ilova qiluvchi tashqi kalit sifatida e'lon qilingan SNum maydoniga ega bo'lgan Buyurtmachilar jadvalini yaratamiz:

```
CREATE TABLE Customers  
(CNum integer NOT NULL PRIMARY KEY,  
CName char(10),  
City char(10),  
SNum integer,  
FOREIGN KEY (SNum) REFERENCES Salepeople  
(SNum));
```

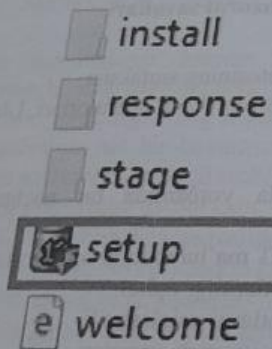
Nazorat savollari

1. SELECT tanlash operatorining sintaksisi
2. IN, BETWEEN, LIKE, IS NULL operatorlari. Ularni ta'riflang va misollar keltiring.
3. Agregat funksiyalari.
4. SELECT operatorida yordamida bir nechta jadvallardan ma'lumotlarni tanshlash
5. SQL tilining INSERT ma'lumotlarni operatorlari.
6. SQL tilida ma'lumotlarning tiplari.
7. SQL DDL ma'lumotlarni aniqlash tili
8. SQL tilida jadvallar yaratish operatori

5-BOB. MULTIMEDIALI MA'LUMOTLAR BAZASINI TASHKIL QILISH

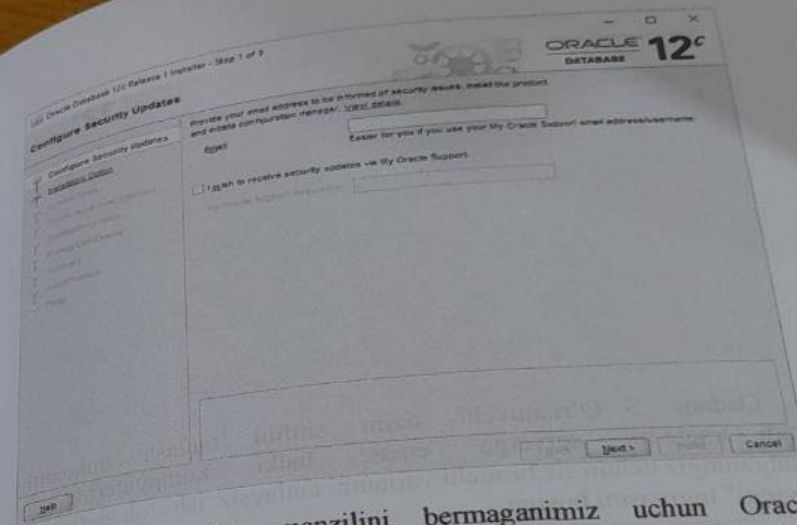
5.1. Oracle ma'lumotlar bazasini boshqarish tizimi

Oracle ma'lumotlar bazasini kompyuteringizga o'rnatish uchun siz Oracle veb-saytining yuklab olish sahifasidan o'rnatuvchini yuklab olishingiz kerak. ZIP formatidagi o'rnatish fayllarini olganingizdan so'ng, ularni kompyuteringizdagi ma'lum bir papkaga chiqarib olishingiz kerak. Quyidagi rasmda Oracle o'rnatish fayllari chiqarilgandan so'ng papkasining tuzilishi ko'rsatilgan.

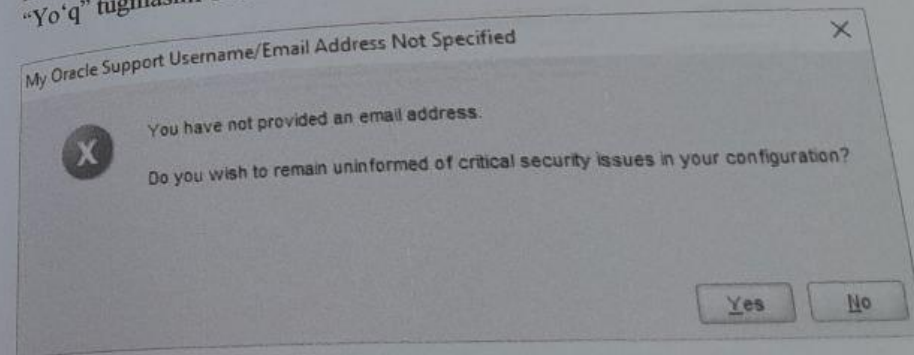


Endi o'rnatish jarayonini boshlash uchun setup.exe faylini ikki marta bosishingiz kerak. Ko'pincha avtomatik ravishda bajariladigan 9 ta qadam bo'ladi.

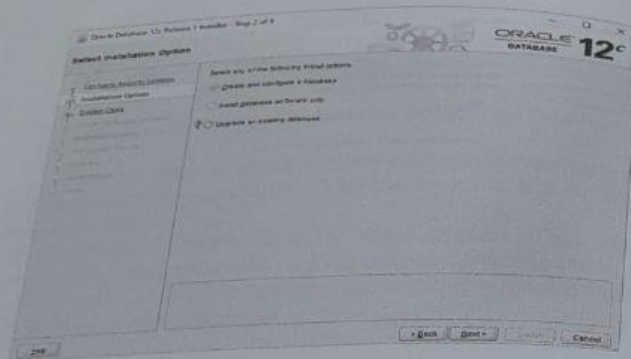
1-qadam. O'rnatuvchi so'nggi xavfsizlik muammolari va yangilanishlarini olish uchun elektron pochta manzilingizni ko'rsatishingizni so'raydi. Keyingi tugmani bosish orqali uni e'tiborsiz qoldirishingiz mumkin



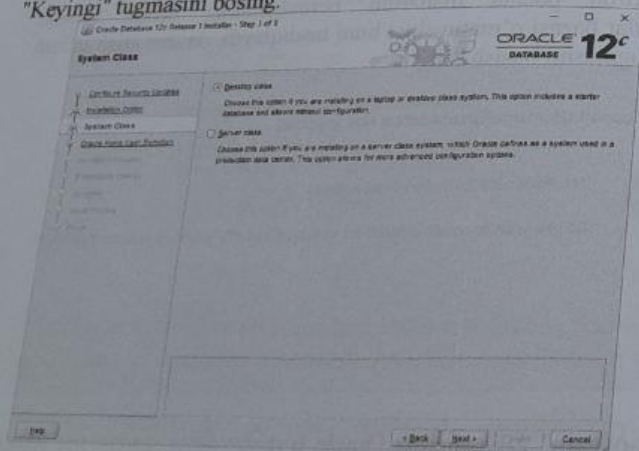
Elektron pochta manzilini bermaganimiz uchun Oracle ma'lumotlar bazasi o'rnatuvchisi buni tasdiqlaydi, davom etish uchun "Yo'q" tugmasini bosish kifoya.



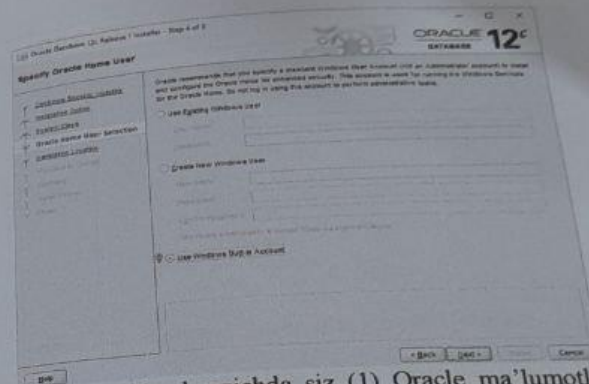
2-qadam. 2-bosqichda Oracle o'rnatuvchisi sizdan ma'lumotlar bazasini yaratish va sozlashni, faqat ma'lumotlar bazasi dasturini o'rnatishni yoki mavjud ma'lumotlar bazasini yangilashni xohlaysizmi, so'raydi. Oracle ma'lumotlar bazasini birinchi marta o'rnatganingiz uchun 1-variantni tanlang va "Keyingi" tugmasini bosish.



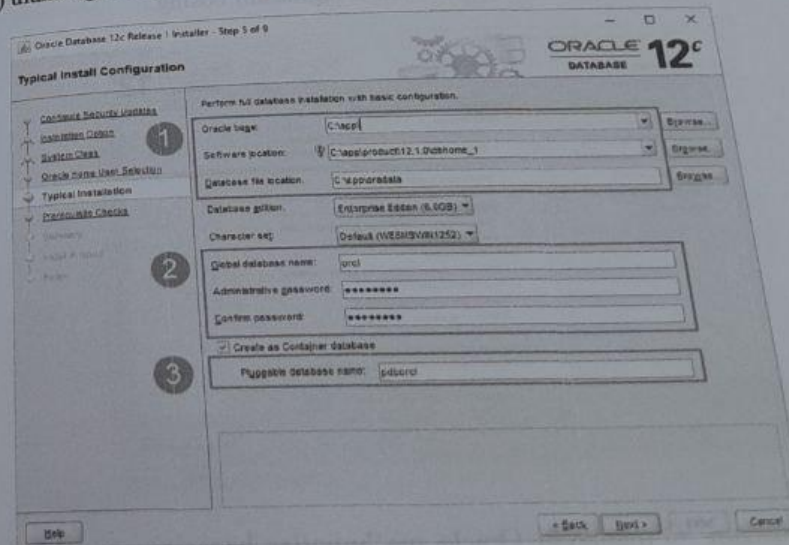
Qadam 3. O'matuvchi tizim sinfini tanlash imkonini beradi. Oracle-ni serverga emas, balki kompyuteringizga o'matganingiz uchun siz birinchi variantni tanlaysiz: ish stoli sinfini va "Keyingi" tugmasini bosing.



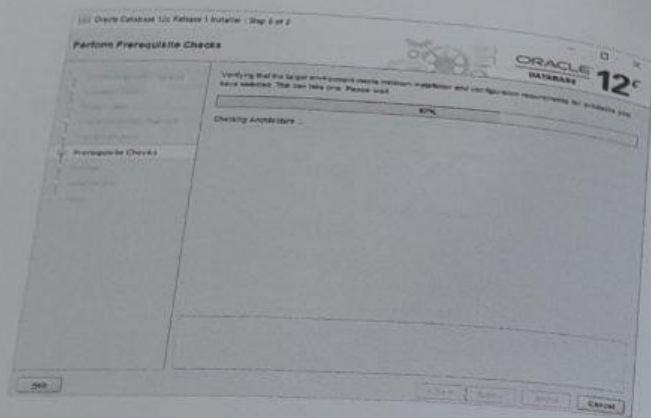
4-qadam. Bu qadam Oracle Home-ni kuchaytirilgan xavfsizlik uchun o'matish va sozlash uchun Windows foydalanuvchi hisobini belgilash imkonini beradi. Uchinchi variantni tanlang: "Windows o'matilgan hisob qaydnomasidan foydalaning".



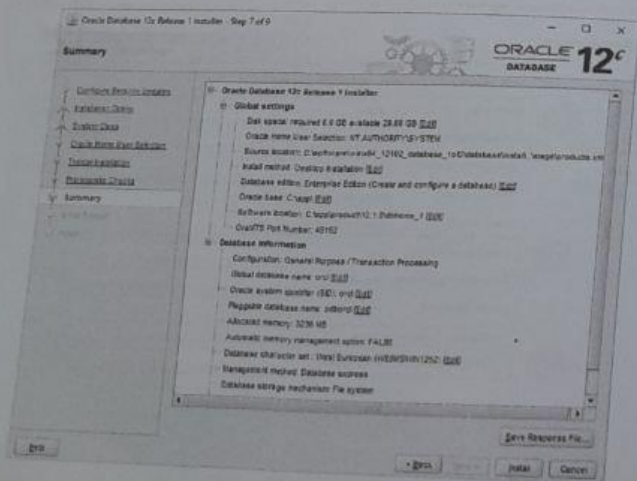
5-qadam. Ushbu bosqichda siz (1) Oracle ma'lumotlar bazasi o'matiladigan papkani, (2) global ma'lumotlar bazasi nomi va parolini, (3) ulanadigan ma'lumotlar bazasi nomini tanlashingiz mumkin.



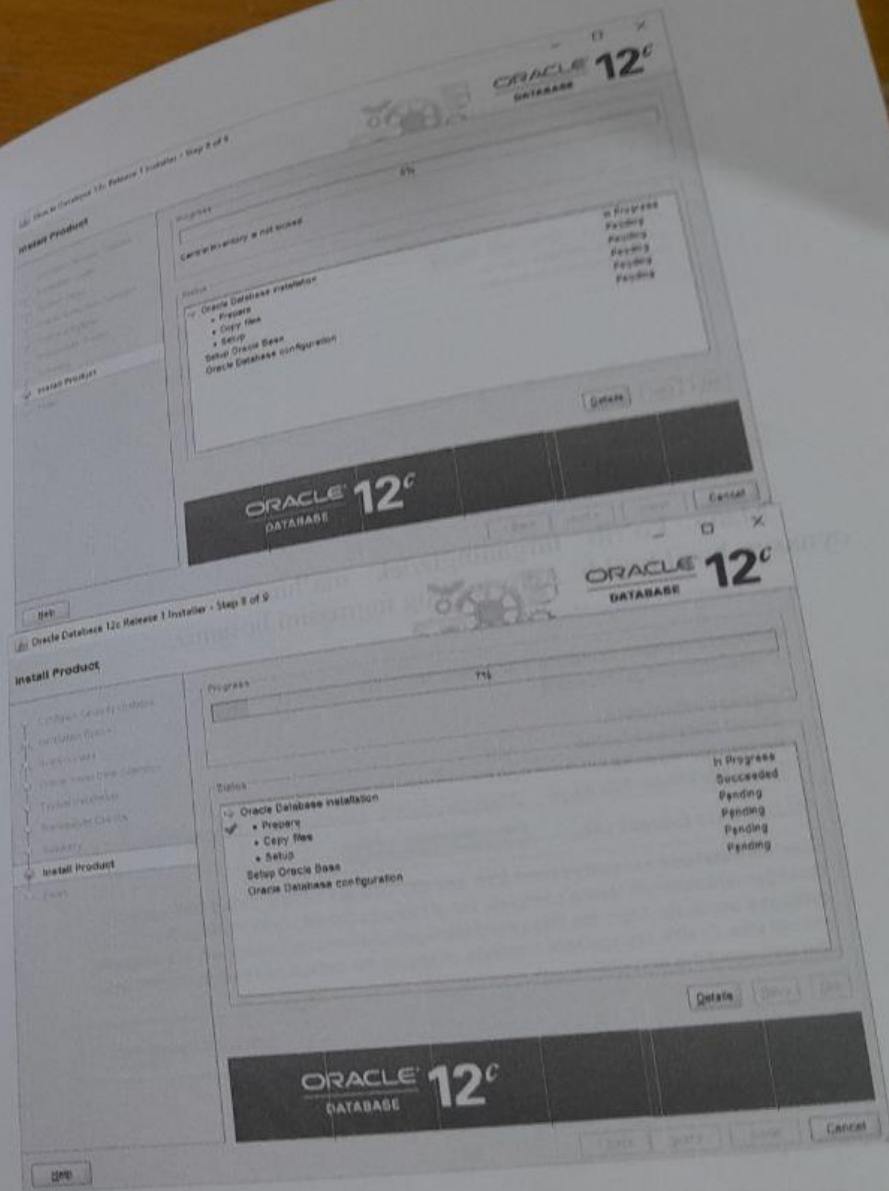
Qadam 6. O'matuvchi zaruriy tekshiruvni amalga oshiradi.

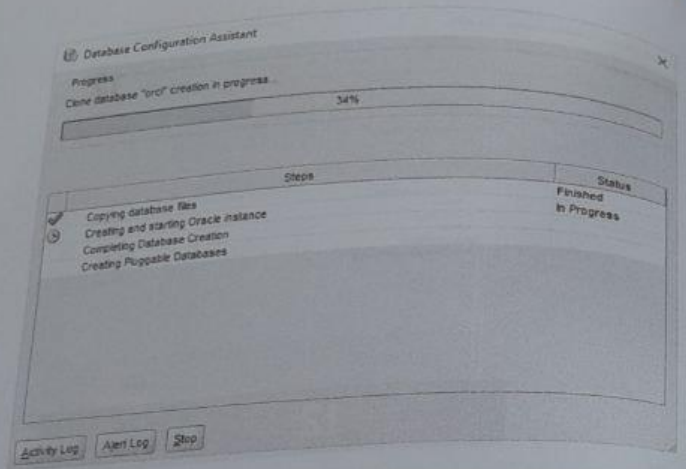


7-qadam. O'ratuvchi sizga global sozlamalar, ma'lumotlar bazasi ma'lumotlari va boshqalar kabi ma'lumotlarning qisqacha mazmunini ko'rsatadi. Ma'lumotni ko'rib chiqishingiz kerak va agar hamma narsa yaxshi bo'lsa, o'ratish tugmasini bosing.

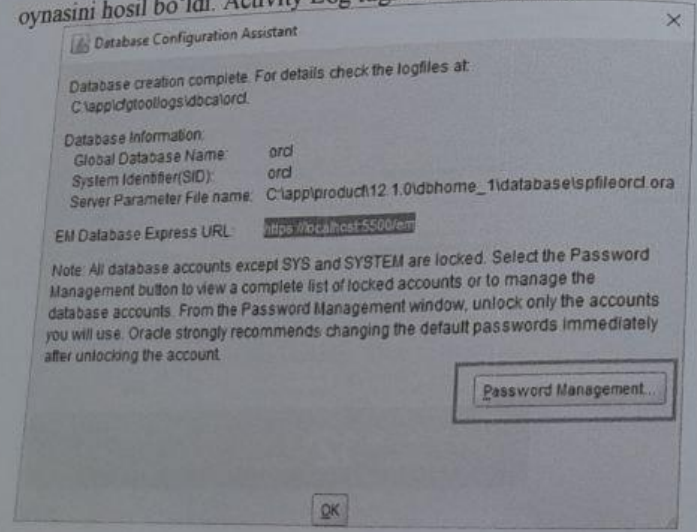


8-qadam. O'ratuvchi Oracle ma'lumotlar bazasini o'ratishni oshlaydi. Kompyuteringiz parametrlariga qarab, bir necha daqiqa avom etadi.

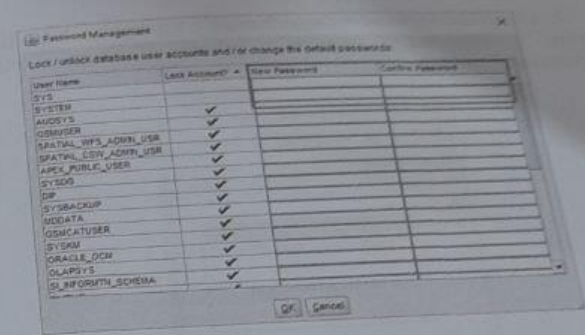




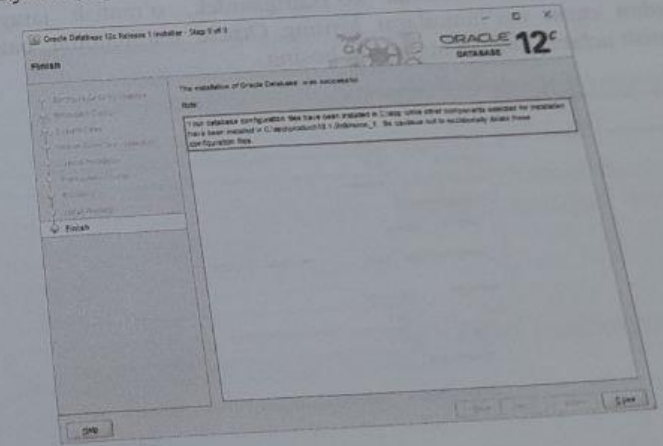
Mana ko'rib turganingizdek ma'lumotlar bazasini sozlash oynasini hosil bo'ldi. Activity Log tugmasini bosamiz.



SYS va SYSTEM bandlariga parolni kiriting, so'ngra OK tugmasini bosing.



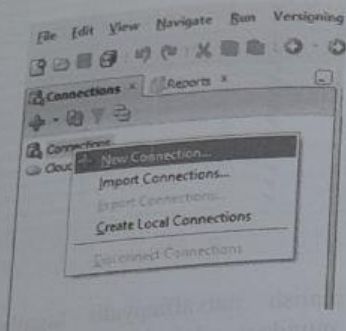
Qadam 9. O'matish muvaffaqiyatli tugallanganidan so'ng, o'matuvchi sizga quyidagi skrinshotda ko'rsatilganidek xabar beradi. Oynani yopish uchun Yopish tugmasini bosing.



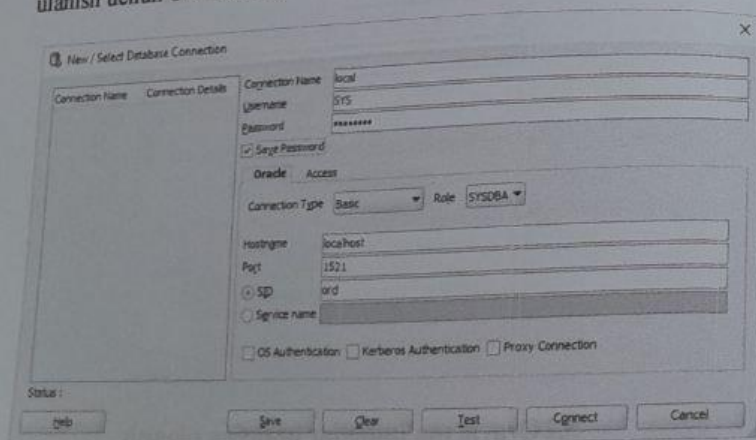
5.2 Oracle ma'lumotlar bazasiga ulanish

1. Oracle Database tomonidan taqdim etilgan SQL dasturchi ilovasini ishga tushiring.

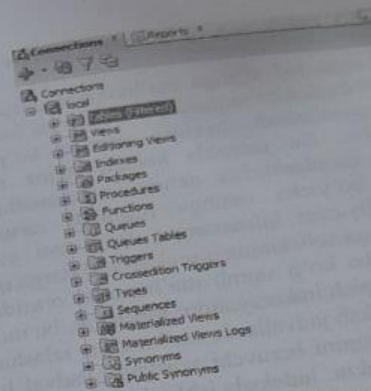
2. Ulanishlar tugunini sichqonchani o'ng tugmasi bilan bosib va yangi ulanishni yaratish uchun "Yangi..." menyusini tanlang.



3. Quyidagi skrinshotda ko'rsatilganidek, o'rnatish jarayonida taqdim etgan ma'lumotlarni kiriting. Oracle ma'lumotlar bazasiga ulanish uchun Connect tugmasini bosib.



SQL dasturchisi barcha obyektlarni quyida ko'rsatilganidek ko'rsatadi.



Oracle Database 12c ni muvaffaqiyatli o'rnatdingiz. Keling, Oracleni o'rganishni boshlaylik.

Tnsnames.ora fayliga yozuv qo'shilmoqda
tnsnames.ora Fayl odatda quyidagi katalogda joylashgan:
C:\app\Yuqoridagi o'rnatish bosqichlarini bajarsangiz, fayl quyidagi katalogda joylashgan:

C:\app\product\12.1.0\dbhome_1\network\admin\
Avval tnsnames.ora faylni Notepad yoki Notepad++ kabi istalgan matn muharriri yordamida ochasiz. Keyin faylning oxiriga quyidagi qatorlarni qo'shishingiz kerak:

```
PDBORCL =
(DESCRIPTION =
(AADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT =
1521))
(CONNECT_DATA =
(SERVER = DEDICATED)
(SERVICE_NAME = pdborcl)
) )
```

Shundan so'ng siz faylni saqlashingiz va uni yopishingiz mumkin. Oracle ma'lumotlar bazasi serveriga namunaviy ma'lumotlar bazasini yuklab olish va yuklash vaqti keldi.

5.3. Indekslash va ma'lumot turlari

Ma'lumotlar bazasi jadvalariga ma'lumotlar ixtiyoriy tartibda kiritiladi va shu tartibda diskda saqlanadi. Kerakli ma'lumotni topish bunday jadvaldan qiyinlashadi, ayniqsa agar u ko'p miqdorda ma'lumotni izlashni osonlashtirish uchun, ma'lumotlarni to'g'risidagi tartibda familiyalar bo'yicha tartibga tushirish zarur. Agarda, buyurtmachining familiyasini bilmasangiz, lekin uni yashayotgan tumanlar bo'yicha tartibga solishingiz mumkin. Bazalarga qo'yiluvchi asosiy talablardan biri, bu ko'p xajmli ma'lumotlar orasidan kerakli bo'lgan yozuvlarni tez topish imkoniyatidir. Indekslar, bu indeksni yo'q darajada tezlashtirish imkonini beruvchi vositadir. Jadval bir nechta indeksga ega bo'lishi mumkin. Indeksda ishlatiladigan maydonlarning miqdoriga, bog'liq ravishda oddiy (bittadan kalit) va tashkil etilgan indekslar (bir nechta maydonlar bo'yicha) ajratiladi.

Har qaysi indeksning qiymati uchun indeks faylidagi indeksga mos bo'lgan yozuv jadvalida joylashishini ko'rsatuvchi unikal dalil bor. Shuning uchun yozuvni axtarishda to'la jadvalni ketma-ket ko'rib chiqilmasdan, balki indekslarning tartibga tushirilgan qiymatlari asosida yozuvga to'g'ri kirib boriladi. Indeksning ahamiyatli xususiyati shuki, ularni birinchi kalitlar hosil qilish uchun ishlatish mumkin. Buni ma'nosi shuki, faqat bitta indeks maydoniga ega bo'lgan jadval uchun bu maydonning ahamiyati noyob bo'lishi kerak. Tarkibiy indekslar uchun indeks kattaligi indeks maydonlarining xar birida keltirilayotgan qiymatlarga ega bo'lishi mumkin. Lekin indeksli ifoda noyob bo'lishi kerak.

Indeksni noyob bo'lish talabi majburiy emas. Talab etilayotgan ma'lumotni izlashni (topishni) tezlashtirish uchun unikal (noyob) bo'lmagan indekslar ishlatilishi mumkin.

Oracle indeksi so'rovlar samaradorligini oshirish uchun samarali vositalardan biridir. Biroq, undan samarali foydalanish uchun uni to'g'ri tushunish kerak. Ushbu bo'lim so'rovlaringizni tezlashtirish uchun Oracle indekslarini tushunishga va ulardan foydalanishga yordam beradi.

Yangi indeks yaratish - CREATE INDEX jadvaldagi bir yoki bir nechta ustunlar uchun indeks yaratish uchun bayonotdan qanday foydalanishni

ko'rsatish - Indeksni olib tashlash - DROP INDEX bayonot yordamida mavjud indeksni qanday olib tashlashni o'rganing.
Noyob indeks(Unique index) - indeks ustuni yoki ustunlardagi qiymatlarning o'ziga xosligini ta'minlash uchun noyob indekslardan foydalaning.
Funksiyaga asoslangan indeks - funksiyalardan iborat ifodani o'z ichiga olgan so'rovlarni tezlashtiradi.
Bitmap indeksi - bir nechta aniq qiymatlarga ega bo'lgan ustun yoki ustunlarda bitmap indeksini yoki past kardinallikdan foydalaning.

Oracle ma'lumotlar turlari

Oracle ma'lumotlar turlari - sizga o'rnatilgan Oracle ma'lumotlar turlari haqida umumiy ma'lumot beradi.
NUMBER - sizni raqamli ma'lumotlar turi bilan tanishtiradi va jadvalda raqamli ustunlarni aniqlash uchun undan qanday foydalanishni ko'rsatadi.
FLOAT - amaliy misollar orqali Oracle da haqiqiy ma'lumotlar turini aniqlab berdi.
CHAR - belgilangan uzunlikdagi belgilar qatori haqida ma'lumot beradi.
NCHAR - Ruxsat etilgan uzunlikdagi Unicode belgilar ma'lumotlarini qanday saqlashni ko'rsatadi.
VARCHAR2 - sizni o'zgaruvchan uzunlikdagi belgilar bilan tanishtiradi va jadvaldagi o'zgaruvchan uzunlikdagi belgilar ustunlarini qanday aniqlashni ko'rsatadi.
NVARCHAR2 - ma'lumotlar bazasida o'zgaruvchan uzunlikdagi Unicode belgilarini o'zida saqlaydi.
DATE - sana va vaqt ma'lumotlarini turini ko'rsatadi.
TIMESTAMP - sana va vaqtni kasr soniya aniqligi bilan saqlaydi
INTERVAL - vaqt davrlarini saqlash uchun intervalli ma'lumotlarini ko'rsatadi.
TIMESTAMP WITH TIME ZONE - vaqt mintaqasi ma'lumotlarini sana va vaqtni o'zida saqlaydi.

Nazorat uchun savol va topshiriqlar.

1. Fayllarni tartiblash usullari.
2. Klasterlash va saxifalarga ajratish.
3. Indeksash nima?
4. Izlashning ikkilik daraxtlarini tushuntirib bering.
5. To'liq matnli izlash deganda nimani tushunasiz?

6-BOB.

MULTIMEDIALI MA'LUMOTLARNI IZLASH

6.1. Ma'lumotlarni izlash

Ushbu bo'lim Oracle ma'lumotlar bazasidan ma'lumotlarni izlashni o'rganishga yordam beradi. Biz bitta jadvaldan ma'lumotlarni olish imkonini beruvchi oddiy so'rovdan boshlaymiz. Oracle-da jadvallar ustunlar va qatorlardan iborat. Misol uchun, customers jadval namuna bazasi quyidagi ustunlar bor: customer_id, name, address, websiteva credit_limit. customersStol, shuningdek, bu ustunlar ma'lumotlarni ega.

CUSTOMERS
*CUSTOMER_ID
NAME
ADDRESS
WEBSITE
CREDIT_LIMIT

Jadvalning bir yoki bir nechta ustunlaridan ma'lumotlarni olish uchun SELECT buyrug'idan foydalanamiz :

```
SELECT  
column_1,  
column_2,
```

```
...
```

```
FROM  
table_name;
```

Ushbu SELECT qismida:

Birinchidan, ma'lumotlarni izlash uchun jadval nomini tanlaymiz. Ikkinchidan, ma'lumotlarni qaytarmoqchi bo'lgan ustunlarni ko'rsatamiz. Agar jadvaldagi bir nechta ustunlarni tanlamoqchi bo'lsak, ularning har birini vergul (,) bilan ajratishimiz kerak.

SELECT operatori juda murakkab bo'lib ORDER BY, GROUP

BY, HAVING, JOIN kabi ko'plab bo'laklardan iborat. Ushbu bo'linda biz faqat SELECT va FROM bandlari bilan ishlaymiz.

Oracle SELECT misollari:
 A) bitta ustundan ma'lumotlarni izlash
 Customers jadvalidan mijozlar nomlarini olish uchun biz quyidagi ketma-ketlikdan foydalanamiz:

```
SELECT
name
FROM
customers;
```

Quyidagi rasm natijani ko'rsatadi:

NAME
Kimberly-Clark
Hartford Financial Services Group
Kraft Heinz
Fluor
AECOM
Jabil Circuit
CenturyLink
General Mills
Southern
Thermo Fisher Scientific

B) Bir nechta ustunlardan ma'lumotlarni izlash
 Bir nechta ustunlardan ma'lumotlarni izlash uchun vergul bilan ajratilgan ustun nomlari ro'yxatini belgilaymiz.
 Quyidagi misolda ko'rsatilgandek, customer jadvalga customer_id, name hamda credit_limit ustunlarini kiritamiz.

```
SELECT
customer_id,
name,
credit_limit
FROM
customers;
```

Quyidagilar natijani ko'rsatadi:

CUSTOMER_ID	NAME	CREDIT_LIMIT
35	Kimberly-Clark	400
36	Hartford Financial Services Group	500
38	Kraft Heinz	500
40	Fluor	500
41	AECOM	500
44	Jabil Circuit	600
45	CenturyLink	600
47	General Mills	600
48	Southern	700
50	Thermo Fisher Scientific	700

C) Jadvalning barcha ustunlaridan ma'lumotlarni izlash.
 Quyidagi misol customers jadvalning barcha ustunlaridan barcha satrlarni oladi:

```
SELECT
customer_id,
name,
address,
website,
credit_limit
FROM
customers;
```

Natija quyidagicha:

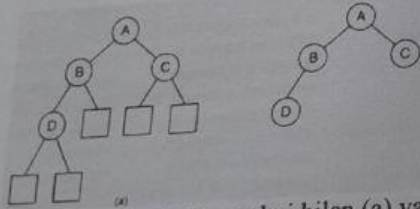
CUSTOMER_ID	NAME	ADDRESS	WEBSITE	CREDIT_LIMIT
1	Roytheon	514 W Superior St, Kokomo, IN	http://www.roytheon.com	100
2	Plains CP Holdings	2515 Bloyd Ave, Indianapolis, IN	http://www.plainsallamerican.com	100
3	US Foods Holding	8768 N State Rd 37, Bloomington, IN	http://www.usfoods.com	100
4	Above	6445 Bay Harbor Ln, Indianapolis, IN	http://www.above.com	100
5	Centene	4019 W 38th St, Bloomington, IN	http://www.centene.com	100
6	Community Health Systems	1608 Portage Ave, South Bend, IN	http://www.chs.net	100
7	Alcoa	23943 Us Highway 33, Elkhart, IN	http://www.alcoa.com	100
8	International Paper	136 E Market St # 800, Indianapolis, IN	http://www.internationalpaper.com	100
9	Emerson Electric	1905 College St, South Bend, IN	http://www.emerson.com	100
10	Union Pacific	3512 Rodville Rd # 137C, Indianapolis, IN	http://www.up.com	200

Buni qulay qilish uchun biz Oracle bazasiga jadvalning barcha ustunlaridagi ma'lumotlarni quyidagicha qaytarishni ko'rsatish uchun stenografiya yulduzcha (*) dan foydalanishimiz mumkin:
 SELECT * FROM customers;
 So'rovni ilovalarga joylashtirganimizda hech qachon yulduzcha (*) dan foydalanmasligimiz kerak. Jadvalning barcha ustunlaridan

ma'lumotlarni olishni istasangiz ham, ma'lumotlarni izlash uchun ustunlarni aniq belgilash yaxshi amaliyotdir. Faqat vaqtinchalik so'rovlar uchun yulduzcha (*) stenografiyasini qo'yishimiz kerak. Agar siz dastur kodida yulduzcha (*) dan foydalansangiz va jadvalda qat'iy ustunlar to'plami bor deb hisoblasangiz, ilova qo'shimcha ustunlarni qayta ishlamasligi yoki olib tashlangan ustunlarga kira olmasligi mumkin.

6.2. Izlashning ikkilik daraxtlari

Ikkilik daraxtlar ma'lumot izlashning samrali usuli hisoblanadi. Ikkilik daraxt strukturalangan tugunlar kolleksiyasidan iborat. Kolleksiya bo'sh bo'lishi mumkin va bu holda *bo'sh ikkilik daraxtga* ega bo'lamiz. Agar kolleksiya bo'sh bo'lmasa, u holda daraxt tugunlarning uchta ajratilgan oilasiga bo'linadi: asosiy ildiz tugun n (yoki ildiz), n ga nisbatan chap shohlarni tashkil qiluvchi ikkilik daraxt, n ga nisbatan o'ng shohlarni tashkil qiluvchi ikkilik daraxt. 1-rasmda A bilan asosiy ildiz tugun belgilangan. A tugunga nisbatan V tugun chap bolasi va chap shoxlarning ildizi hisoblanadi. A tugunga nisbatan S tugun o'ng bolasi va o'ng shoxlarning ildizi hisoblanadi.



6.1-rasm. Ikkilik daraxt tashqi tugunlari bilan (a) va ularsiz (b)

1a rasmdagi ikkilik daraxt to'rtta *ichki tugunlardan* (yumaloq qilib belgilangan) va beshta *tashqi* (so'nggi) tugunlardan (kvadrat qilib belgilangan) iborat. Ikkilik daraxtning o'lchami o'zidagi ichki tugunlarning soni bilan aniqlanadi. Tashqi tugunlar bo'sh ikkilik daraxtlarga to'g'ri keladi. Masalan, V tugunning chap bolasi bo'sh emas (D tugunga ega). V tugunning o'ng bolasi bo'sh daraxt. Sxemada

tasviri ko'rsatilmasligi ham mumkin. Ikkilik daraxtlar ichida tugunlar quyidagicha belgilanadi: r tugun n tugunning otasi hisoblanadi, agar n tugun r tugunning bolasi bo'lsa. Ikkita tugun aka-uka hisoblanadi, agar ularning tegishli bo'lsa.

Ikkita n_1 va n_2 tugunlar berilgan bo'lib, n_k tugun n_1 ildizli shohga deyiladi, n_1 tugun n_k tugunning *zurriyodi* (bolasi) mavjud. Bu yerda, n_1 va n_2, \dots, n_k tugunlar ketma-ketligida har bir n_i tugun n_{i-1} tugunning otasi hisoblanadi har bir $i = 1, 2, \dots, k-1$ uchun. *Yo'l uzunligi* deb, n_1 va n_2, \dots, n_k tugunlar orasidagi $(k-1)$ qovurg'alarining soniga aytiladi. Masalan, 1a rasmda A dan D gacha bo'lgan yagona yo'l ga teng. n tugunning ketma-ketligi orasidagi qovurg'alar soniga — ya'ni 2 aniqlanadi:

$$\text{chuqurlik}(n) = \begin{cases} 0 \\ \{ 1 + \text{chuqurlik}(\text{otasi}(n)) \} \end{cases}$$

Har-bir tugunning chuqurligi asosiy ildiz tugundan joriy tugungacha bo'lgan yagona yo'l uzunligiga teng. 1a rasmda A tugun chuqurligi 0 ga teng, D tugun chuqurligi 2 ga teng. n tugunning balandligi ham rekursiv yo'l bilan topiladi: agar n — tashqi tugun bo'lsa

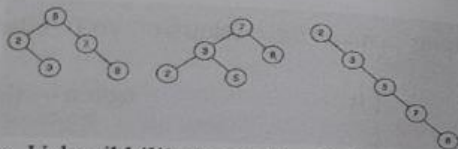
$$\text{balanlik}(n) = \begin{cases} 0 \\ \{ 1 + \max(\text{balanlik}(\text{chap}(n)), \text{balanlik}(\text{o'ng}(n))) \} \end{cases}$$

Bu yerda, $\text{chap}(n)$ va $\text{o'ng}(n)$ bilan mos ravishda n tugunning chap va o'ng zurriyodlari belgilangan. n tugunning balandligi deb, n tugundan boshlab pastga — tashqi tugungacha bo'lgan eng uzun yo'lga aytiladi. Ikkilik daraxt balandligi uning asosiy ildiz tuguni balandligi bilan aniqlanadi.

Masalan, 1a rasmda ikkilik darax balandligi 3 ga teng, D tugun balandligi 1 ga teng bo'radi.

Ikkilik daraxtlarning asosiy vazifasi ma'lumotlarni izlash samarasini oshirishdan iborat. Izlash jarayonida turli elementlar orasidan berilgan elementni topish, to'plam ichidan eng katta qiymatligini yoki eng kichik qiymatligini aniqlash, to'plam berilgan elementni o'z ichiga olishini aniqlash kabi operatsiyalar uchraydi.

Ikkilik darax ichidan samarali izlashni amalga oshirish uchun uning elementlari kerakli tashkil qilingan bo'lishi kerak. Masalan, ikkilik daraxt – *ikkilik izlash daraxti* deb atalishi uchun, uning elementlari daraxtda shunday joylashtirilgan bo'lishi kerakki, har-bir n element uchun uning chap shohlaridagi elementlar n dan kichik bo'lishi, uning o'ng shohlaridagi elementlari n dan katta bo'lishi kerak. 6.2-rasmda uchta shunaqa ikkilik darax ko'rsatilgan. Ularning uchchalasi ham bir xil butun sonli elementlarga ega.



6.2-rasm. Uchta ikkilik daraxt bir xil elementlari bilan.

Umuman olganda turli formadagi, ixtiyoriy berilgan elementlar to'plamini o'zida jamlagan juda ko'p ikkilik izlash daraxtlari mavjud.

Bu yerda elementlar *chiziqli tartibda* joylashgan va ixtiyoriy ikkita elementni o'zaro taqqoslash mumkin deb tahmin qilinadi. Chiziqli tartibga misol tariqasida o'sish bo'yicha tartiblangan butun yoki haqiqiy sonlar qatorini yoki alfavit tartibida joylashtirilgan simvollar satrini olish mumkin. Izlash ikkita ixtiyoriy elementni ularning chiziqli tartibiga nisbatan taqqoslovchi funksiyaga murojaat qilish yo'li bilan amalga oshiriladi.

To'liq matnli izlash

To'liq matnli izlash (*Full text searching*) – hujatlarni nomidan emas, to'liq matnidan yoki matnining bir qismidan avtomatik izlash.

To'liq matnli izlash dasturining dastlabki versiyalarida, biror bir

so'z yoki jumla qidirilganda, barcha hujatlarni matnini to'liq o'tib izlash nazarda tutilgan edi. Bunday texnologiya bo'yicha izlash, bazaning hajmiga qarab, juda ko'p vaqt talab qiladi. Internetda umuman amalga oshirib bo'lmadi. Zamonaviy algoritmlarda izlash uchun to'liq matnli indeks (lug'at) oldindan shakllantirib qo'yiladi. Unda barcha so'zlar qo'lda uchirishi qayd qilib qo'yilgan bo'radi. Bunday indeks bilan osongina kerakli so'z yoki jumlani topib olish mumkin bo'radi.

MySQL ma'lumotlar bazasida to'liq matnli indekslar «FULLTEXT» tipi bilan belgilanib, «VARCHAR» va «TEXT» tipidagi ustunlar uchun qo'llaniladi. Jadvalga matn kiritilayotganda «FULLTEXT» tipidagi indeks darhol xosil qilinadi. Bu ishni ancha sekinlatishi mumkin. Shuning uchun maydonlarga avval matnni kiritib keyin tahrirlash mumkin bo'radi. Izlashni MATCH() i AGAINST() funksiyalar yordamida amalga oshiriladi.

```
SELECT * FROM articles WHERE MATCH (title, body) AGAINST ('poisk');
```

Bu yerda izlanayotgan jumla aniq yozilishi kerak. Ya'ni yuqorida ko'rsatilgan misoldagi jumla bilan qidirilganda, «poiska», «poiskovik» so'zlar topilmaydi. Natijada, topilgan matn ichidagi mos kelgan jumalar ajratib ko'rsatib qo'yiladi.

Doimiy ifodalar (regular expressions)

Doimiy ifodalar (*regular expressions*) — matn ichidagi jumla satrlarni izlab topib, metasimvollardan foydalangan holda ular ustida muolaja o'tkazadigan formal tildir. Izlash uchun, simvollar va meta simvollardan tashkil topgan, izlash qoidasini ko'rsatuvchi shablon satrdan (Pattern) foydalaniladi. Matn ustida muolaja o'tkazish uchun, o'zida maxsus simvollar jamlagan qo'shimcha almashtirish satri beriladi.

Doimiy ifodalar ba'zi matn redaktorlari va utilitalar tomonidan matnni izlash va almashtirish uchun ishlatiladi. Masalan, doimiy ifodalar yordamida quyidagicha shablonlarni berish mumkin:

- Berilgan «ком» jumla yordamida, «ком», «комлета», «мерракомовый» kabi barcha simvollar ketma ketligini izlab topish;
- Matndan barcha "olma" so'zlarni izlab topib, uni "anor" so'ziga almashtirib chiqish;

- Matndan oldida “qandil” so’zi bo’lgan barcha “olma” so’zini izlab topish;
- Matndan “olma” yoki “anor” so’zlari qatnashgan barcha gaplarni olib tashlash;

Doimiy ifodalar bundan ham murakkabroq izlash va almashtirish shablonlarini berishi mumkin. Doimiy ifodalar bilan ishlash natijasida quyidagilarni amalga oshirish mumkin:

- Matndan izlanayotgan andozaga mos so’zlar borligini tekshirish;
- Andozaga mos jumlaning aniqlash;
- Andozaning ma’lum qismiga mos tushadigan simvollar guruhini aniqlash.

Agar doimiy ifodalar matni almashtirish uchun ishlatilayotgan bo’lsa, u holda natijada – ichidan topilgan jummalari o’chirilib, almashtiriladigan jumlar qo’yilgan yangi matn satri hosil bo’ladi. Hususan, agar almashtiriladigan jumla andozasi bo’sh bo’lsa, andoza bo’yicha topilgan jumlar o’chirib chiqiladi xolos.

Doimiy ifodalarda [] \ / ^ \$. | ? * + () { } maxsus simvollar ishlatiladi. Doimiy ifodalarning boshqa xilida bu maxsus simvollar boshqacha bo’lishi mumkin. Qolgan ishlatilgan simvollar o’zini anglatadi.

Andozada maxsus simvoldan birortasini ishlatish kerak bo’lib qolsa, oldidan “/” belgi qo’yiladi.

Andoza	Mos kelishi
a.?	a. yoki a
a\\b	a\b
a[F]	a[F]
\Q+*\AE	+*/

Ixtiyoriy simvol

Metasimvol “.” (nuqta) ixtiyoriy bitta simvolni anglatadi.

Simvolli sinflar (simvollar to’plami)

Kvadrat qavslarga [] olingan simvollar to’plami simvolli sinf deyiladi va doimiy ifodalar interpretatoriga satrning shu joyida qavs ichidagi simvoldan birortasi turishi mumkinligini anglatadi.

Xususan, [abv] andoza, matnda ushbu uchta simvoldan birortasi uchrashi mumkinligini anglatadi. [1234567890] andoza, matnda sonlardan birortasi uchrashi mumkinligini anglatadi. Simvollar diapazonini ko’rsatish imkoniyati ham bor: [A-Yaa-ya] alfavitning barcha harflariga mos keladi, o’qg’h dan tashqari, albatta.

Aksincha, matnda qatnashmaydigan simvollar ko’rsatish uchun, [^] ishlatiladi. [^0-9] andozasi sonlardan boshqa ixtiyoriy simvol bo’lishi mumkinligini anglatadi.

Andozada ba’zi simvolli sinflarni alohida metasimvollar bilan almashtirish mumkin:

Simvol	Ekvivalenti	Mos kelishi
\d	[0-9]	Sonlar
\D	[^0-9]	sonlardan tashqari simvollar
\s	[\f\n\r\t\v]	Probel simvol
\S	[^\f\n\r\t\v]	Probel simvoldan tashqari
\w	[[:word:]]	harf yoki sonli simvol yoki tag chiziq
\W	[^[:word:]]	harf yoki sonli simvol yoki tag chiziqdan tashqari ixtiyoriy simvol

Satr ichidagi pozitsiya

Quyidagi simvollar doimiy ifodani matn elementlariga nisbatan ko’rsatilgan pozitsiyaga joylashtirish imkoniyatini beradi: satr boshiga va oxiriga, so’z chetiga.

Ifoda	Pozitsiya	Misol uchun	Mos keladi
^	Matn boshi	^a	aaa aaa
\$	Matn oxiri	a\$	aaa aaa
\b	So’z cheti	a\b	aaa aaa
		\ba	aaa aaa
\B	So’z chetimas	\Ba\B	aaa aaa
\G	Avvalgi izlab topilgan	\Ga	aaa aaa (izlash 4- pozitsiyada to’xtagan —a topilmagan joyda)

Guruhni belgilash

Dumaloq qavslar harakat makonini va muolaja prioritetini aniqlash uchun ishlatiladi. Masalan, (tr[au]m-?)* jumla "tram-tram-trumtram-trum-tramtrum" ko'rinishdagi ketma-ketlikni izlab topadi.

Shart bo'yicha izlash

Doimiy ifodalarda izlash andozasining qacrida tekshirish qaysi yo'ldan borishini tanlash imkoniyati bor.

Doimiy ifoda	Izoh	misol	anglatadi
(?(?=yesli)to inache)	Agar tekshiruv muvoffaqiyatli tugasa, To qismi davom etadi, aks holda inache qismi davom etadi	(?(?<=a)m p)	mam,pap
(?(n)to inache)	Agar n-guruh javob qaytarsa, To qismi izlashda davom etadi, aks holda inache qismi izlashda davom etadi	(a)?(?(1)m p)	mam,pap

6.3. Oracle da ma'lumotlarni Order By orqali tartiblash

Oracle da jadval o'z satrlarini ma'lumotlar bazasiga qaysi qatorlar kiritilganidan qat'i nazar, aniqlanmagan tartibda saqlaydi. Ma'lumotlarni tartibga solish uchun Order by buyrug'idan foydalanamiz.

Misol uchun, barcha mijozlarni alifbo tartibida ismlari bo'yicha ro'yxatga olishni yoki barcha mijozlarni eng pastdan yuqoriga qadar kredit limitlari tartibida joylashtirishimiz mumkin.

Ma'lumotlarni saralash uchun SELECT so'roviga quyidagi ORDER BY bandni qo'shamiz :

```
SELECT  
column_1,  
column_2,  
column_3,
```

FROM

```
table_name  
ORDER BY  
column_1 [ASC | DESC] [NULLS FIRST | NULLS LAST],  
column_1 [ASC | DESC] [NULLS FIRST | NULLS LAST],  
...
```

Natijani ustun bo'yicha saralash uchun siz ushbu ustunni ORDER BY buyrug'idan keyin kiritamiz.

Ustun nomidan keyin saralash tartibi quyidagicha bo'lishi mumkin:
✓ ASC o'sib borish tartibida saralash uchun
✓ DESC kamayish tartibida saralash uchun

Odatda ORDER BY buyrug'i biz ko'rsatganimizdan qat'i nazar, tartibida saralashni istasak, DESC dan foydalanamiz.
NULLS FIRST NULL qiymatlarni NULL bo'lmagan qiymatlardan oldin va NULLS LAST NULL qiymatlarni NULL bo'lmagan qiymatlardan keyin qo'yadi.

Ushbu ORDER BY buyrug'i ma'lumotlarni bir nechta ustunlar bo'yicha saralash imkonini beradi, bunda har bir ustun turli xil tartiblarga ega bo'lishi mumkin.

E'tibor bering, ORDER BY buyrug'i har doim SELECT izlash buyrugining so'nggi bandida keladi.
Oracle ORDER BY buyrug'iga misollar:
Buning uchun biz Oracle ma'lumotlar bazasidagi customers jadvaldan foydalanamiz.

CUSTOMERS
* CUSTOMER_ID
NAME
ADDRESS
WEBSITE
CREDIT_LIMIT

Quyidagi so'rov mijoz nomi, manzili va kredit limitini customers jadvaldan oladi:

```
SELECT  
name,  
address,  
credit_limit
```

FROM
customers;

NAME	ADDRESS	CREDIT_LIMIT
Kimberly-Clark	1660 University Ter, Ann Arbor, MI	400
Hartford Financial Services Group	15713 N East St, Lansing, MI	400
Kraft Heinz	10315 Hickman Rd, Des Moines, IA	500
Fluor	1928 Sherwood Dr, Council Bluffs, IA	500
AECOM	2102 E Kimberly Rd, Davenport, IA	500
Jabil Circuit	221 3Rd Ave Se # 300, Cedar Rapids, IA	500
CenturyLink	2120 Heights Dr, Eau Claire, WI	500
General Mills	6555 W Good Hope Rd, Milwaukee, WI	600
Southern	1314 N Stoughton Rd, Madison, WI	600

Ko'rinib turibdiki qatorlar tartibi aniqlanmagan.

A) Ustunlar orqali qatorlarni saralash

Mijoz ma'lumotlarini nomlar bo'yicha alifbo tartibida o'sish tartibida saralash uchun quyidagi so'rov buyrug'idan foydalanamiz:

```
SELECT
name,
address,
credit_limit
```

```
FROM
customers
ORDER BY
name ASC;
```

NAME	ADDRESS	CREDIT_LIMIT
JM	Via Frenzy 6903, Roma,	1200
ADP	Langstr 14, Zuerich, ZH	700
AECOM	2102 E Kimberly Rd, Davenport, IA	500
AES	33 Fulton St, Poughkeepsie, NY	1200
AJG	12817 Coastal Hwy, Ocean City, MD	2400
AT&T	55 Church Hill Rd, Reading, PA	1200
AbbVie	6445 Bay Harbor Ln, Indianapolis, IN	100
Abbott Laboratories	3310 Dixie Ct, Saginaw, MI	200
Advance Auto Parts	2674 Collingwood St, Detroit, MI	3700
Aetna	200 E Fort Ave, Baltimore, MD	2400

Oracle bazasidagi jadval satrlarini tartiblashtirish uchun ASC

foydalanamiz. Agar uni ORDER BY buyrug'i orqali qo'llasak, qatorlarni ko'rsatilgan ustun bo'yicha o'sish tartibida tartiblaydi. Shunday qilib, quyidagi ifoda:

```
ORDER BY name ASC
```

quyidagilarga teng:

```
ORDER BY name
```

Mijozni nomi bo'yicha alifbo bo'yicha kamayish tartibida saralash uchun ORDER BY buyrug'iga DESC ni ustun nomidan keyin quyidagi tarzda qo'llaymiz:

```
SELECT
```

```
name,
address,
credit_limit
```

```
FROM
customers
```

```
ORDER BY
name DESC;
```

Quyidagi rasmda mijozlar ismlari alifbo tartibi bo'yicha kamayish tartibida tartiblangan natija ko'rsatilgan:

NAME	ADDRESS	CREDIT_LIMIT
eBay	Via Del Disegno 194, Milano,	1500
Yum Brands	Ruella Delle Spiriti, Roma,	500
Xerox	9936 Dexter Ave, Detroit, MI	400
Xcel Energy	1540 Stripes Crt, Baden-Daettwil, AG	400
World Fuel Services	Theresienstr 15, Munich,	2400
Whole Foods Market	4200 Yosemite Ave S, Minneapolis, MN	1200
Whirlpool	18305 Van Dyke St, Detroit, MI	200
Western Refining	5565 Baynton St, Philadelphia, PA	2400
Western Digital	33 Pine St, Lockport, NY	1200
WestRock	Chrottenweg, Bern, BE	5000

B) Bir nechta ustunlar bo'yicha qatorlarni saralash:

Bir nechta ustunlarni saralash uchun biz jadvaldagi har bir ustun nomini ORDER BY buyrugidan keyin vergul bilan yozib chiqamiz. Namuna ma'lumotlar bazasida quyidagi contacts jadvalga qarang.

CONTACTS	
*CONTACT_ID	
FIRST_NAME	
LAST_NAME	
EMAIL	
PHONE	
CUSTOMER_ID	

Masalan, kontaktlarni ismlari bo'yicha o'sish tartibida va familiyalarini kamayish tartibida saralash uchun biz quyidagi ifodadan foydalanamiz:

```
SELECT
  first_name,
  last_name
FROM
  contacts
ORDER BY
  first_name,
  last_name DESC;
```

Ushbu misolda Oracle dastlabki natijalar to'plamini yaratish uchun birinchi navbatda satrlarni nomlari bo'yicha o'sish tartibida tartiblaydi. Keyin Oracle familiya bo'yicha o'rnatilgan dastlabki natijani kamayish tartibida tartiblaydi.

Quyidagi natijaga qarang:

Corliss	Mcneil
Cristine	Bell
Daina	Combs
Daniel	Glass
Daniel	Costner
Darron	Robertson
Debra	Herring
Dell	Wilkinson
Delpha	Golden
Deneen	Hays
Denny	Daniel
Diane	Higgins
Dianne	Sen
Dianne	Derek
Dick	Lamb
Don	Hansen
Doretha	Tyler
Dorotha	Wong

Bu natijada:

- ✓ Birinchi ismlar o'sish tartibida tartiblangan.
 - ✓ Agar ikkita ism bir xil bo'lsa, familiyalar kamayish tartibida tartiblanadi, masalan, Daniel Glass va Daniel Costner, Dianne Sen va Dianne Derek, Doretha Tyler va Dorotha Wong.
- C) Ustun pozitsiyalari bo'yicha qatorlarni tartiblash.

Ma'lumotlarni saralash uchun ustun nomlarini ko'rsatish shart emas. Agar xohlasangiz, jadvaldagi ustun pozitsiyalaridan foydalanishingiz mumkin. Quyidagi misolga qarang:

```
SELECT
  name,
  credit_limit
FROM
  customers
ORDER BY
  2 DESC,
  1;
```

Ushbu misolda name ustunning o'rnini 1 va credit_limit ustun 2. Ushbu ORDER BY buyrug'ida biz Oracle bazasiga satrlarni saralashni ko'rsatish uchun ushbu ustun pozitsiyalaridan foydalanamiz.

D) NULL qiymatli qatorlarni saralashga doir misollar.

Namuna ma'lumotlar bazasida quyidagi locations jadvalga qarang:

LOCATIONS	
*LOCATION_ID	
ADDRESS	
POSTAL_CODE	
CITY	
STATE	
COUNTRY_ID	

Quyidagi misolda joy nomlarni tanlaydi, ularni shahar va shtat bo'yicha saralaydi:

```
SELECT
  country_id,
  city,
```



```

state
FROM
locations
ORDER BY
city,
state;

```

Natija quyidagicha:

COUNTRY_ID	CITY	STATE
IT	Roma	(null)
IT	Venice	(null)
JP	Hiroshima	(null)
SG	Singapore	(null)
CN	Beijing	(null)
UK	London	(null)
CH	Bern	BE
DE	Munich	Bavaria
US	South San Francisco	California
MX	Mexico City	Distrito Federal,
CH	Geneva	Geneve

NULL bo'lmagan qiymatlar bilan aralashtirilgan saralashda Oracle qaysi biri birinchi bo'lib ko'rinishini belgilash imkonini beradi. Masalan, quyidagi misolda joy nomlarni o'sish tartibida tartiblash uchun birinchi navbatda NULL qiymatlarni joylashtiradi.

```

SELECT
country_id,
city,
state
FROM
locations
ORDER BY
state ASC NULLS FIRST;

```

COUNTRY_ID	CITY	STATE
IT	Roma	(null)
IT	Venice	(null)
JP	Hiroshima	(null)
SG	Singapore	(null)
CN	Beijing	(null)
UK	London	(null)
CH	Bern	BE
DE	Munich	Bavaria
US	South San Francisco	California
MX	Mexico City	Distrito Federal,
CH	Geneva	Geneve
IN	Bombay	Maharashtra
UK	Stretford	Manchester
US	South Brunswick	New Jersey
AU	Sydney	New South Wales
CA	Toronto	Ontario
UK	Oxford	Oxford
BR	Sao Paulo	Sao Paulo
US	Southlake	Texas
JP	Tokyo	Tokyo Prefecture
NL	Utrecht	Utrecht
US	Seattle	Washington
CA	Whitehorse	Yukon

NULL bo'lmagan qiymatlardan keyin joy nomlarini qo'yish uchun NULLS LAST ga quyidagini kiritamiz:

```

SELECT
country_id,
city,
state
FROM
locations
ORDER BY
state
ASC NULLS LAST;
Natija:

```

COUNTRY_ID	CITY	STATE
CH	Bern	
DE	Munich	BE
US	South San Francisco	Bavaria
MX	Mexico City	California
CH	Geneva	Distrito Federal,
IN	Bombay	Geneve
UK	Stretford	Maharashtra
US	South Brunswick	Manchester
AU	Sydney	New Jersey
CA	Toronto	New South Wales
UK	Oxford	Ontario
BR	Sao Paulo	Oxford
US	Southlake	Sao Paulo
JP	Tokyo	Texas
NL	Utrecht	Tokyo Prefecture
US	Seattle	Utrecht
CA	Whitehorse	Washington
SG	Singapore	Yukon
		(null)

E) Funksiya yoki ifoda natijasi bo'yicha qatorlarni saralash
Ushbu ORDER BY buyrug'i orqali funksiya qo'llash imkonini mavjud, masalan, ustundagi satrli va matematik ma'lumotlarni funksiya natijasi bo'yicha tartiblaydi.

Masalan, quyidagi misolda mijoz nomlarini katta-kichik harflar bilan tartiblash uchun ORDER BY buyrug'iga UPPER() funksiyasini kiritamiz:

```
SELECT
  customer_id,
  name
FROM
  customers
ORDER BY
  UPPER( name );
```

Quyidagilar natijani ko'rsatadi:

CUSTOMER_ID	NAME
190 3M	
19	Abbott Laboratories
4	AbbVie
287	ADP
168	Advance Auto Parts
41	AECOM
80	AES
138	Aetna
16	Aflac
141	AIG

F) Sana bo'yicha saralash misoli
Namuna ma'lumotlar bazasidan quyidagi orders jadvalga qarang:

ORDERS
* ORDER_ID
CUSTOMER_ID
STATUS
SALESMAN_ID
ORDER_DATE

Ushbu misolda buyurtmalarni buyurtma sanasi bo'yicha saralash uchun ORDER BY buyrug'idan foydalanamiz:

```
SELECT
  order_id,
  customer_id,
  status,
  order_date
FROM
  orders
ORDER BY
  order_date DESC;
```

ORDER_ID	CUSTOMER_ID	STATUS	ORDER_DATE
88	6	Shipped	01-NOV-17
94	1	Shipped	27-OCT-17
1	4	Pending	15-OCT-17
14	40	Shipped	29-SEP-17
15	49	Shipped	27-SEP-17
17	17	Shipped	27-SEP-17
36	51	Shipped	05-SEP-17
57	60	Shipped	24-AUG-17
28	6	Canceled	15-AUG-17
29	44	Shipped	14-AUG-17
31	44	Canceled	12-AUG-17
30	45	Shipped	12-AUG-17
60	1	Shipped	30-JUN-17
21	21	Pending	27-MAY-17
20	20	Shipped	27-MAY-17
40	55	Shipped	11-MAY-17

6.4. Ma'lumotlarni filtrlash

DISTINCT - so'rov natijalaridan ikki nusxadagi qatorlarni qanday olib tashlaydi.

WHERE - so'rov orqali qaytarilgan natijalar to'plamidagi qatorlar uchun shartni qanday belgilashni ko'rsatadi.

AND - ikki yoki undan ortiq mantiqiy ifodalarni birlashtiradi va agar barcha ifodalar to'g'ri bo'lsa, true qiymatini qaytaradi.

OR - ikki yoki undan ortiq mantiqiy iboralarni birlashtirib, agar iboralardan biri rost bo'lsa, true qiymatini qaytaradi.

FETCH - qator cheklash bandidan foydalanib, so'rov orqali qaytarilgan qatorlarni qanday cheklashni ko'rsatadi.

IN - ro'yxatdagi qiymat yoki pastki so'rovdagi har qanday qiymatga mos kelishini aniqlaydi.

BETWEEN - ma'lumotlarni qiymatlar oralig'i asosida filtrlash.

LIKE - muayyan naqshlar asosida moslikni bajarish.

IS NULL va **IS NOT NULL** - ustundagi ifoda yoki qiymatlar **NULL** ga tengligi yoki yo'qligini tekshiradi.

Oracle da WHERE operatoriga kirish

Ushbu WHERE shart operatori SELECT buyrug'i tomonidan qaytarilgan qatorlar uchun qidiruv shartini

belgilaydi. Quyida WHERE shart operatorining sintaksisi tasvirlangan:

```
SELECT
  select_list
FROM
  table_name
WHERE
  search_condition
ORDER BY
  sort_expression;
```

FROM, WHERE shart operatoridan keyin ORDER BY buyrug'i keladi. WHERE shart kalit so'zdan keyin search_condition bajarilishi kerak bo'lgan shartni belgilaydi.

SELECT buyrug'idan tashqari, WHERE shart operatori uchun qatorlarni yangilash (UPDATE) yoki o'chirish (DELETE) buyrug'laridan foydalanishimiz mumkin. Oracle MB da WHERE shart operatoriga misol. Namuna ma'lumotlar bazasida quyidagi products jadvalidan foydalanamiz:

PRODUCTS
* PRODUCT_ID
PRODUCT_NAME
DESCRIPTION
STANDARD_COST
LIST_PRICE
CATEGORY_ID

A) Oddiy tenglik operatori yordamida qatorlarni tanlash. Quyidagi misol faqat 'Kingston' nomli mahsulotlarni qaytaradi:

```
SELECT
  product_name,
  description,
  list_price,
  category_id
FROM
  products
WHERE
  product_name = 'Kingston';
```

Quyidagi rasmda natijani ko'rishimiz mumkin:

PRODUCT_NAME	DESCRIPTION	LIST_PRICE	CATEGORY_ID
Kingston	Speed:DDR3-1333,Type:240-pin D3MM,CAS:9Module:4x16GBSize:64GB	671.38	3
Kingston	Speed:DDR3-1600,Type:240-pin D3MM,CAS:11Module:4x8GBSize:32GB	653.5	3
Kingston	Speed:DDR3-1600,Type:240-pin D3MM,CAS:11Module:4x16GBSize:64GB	644	3
Kingston	Speed:DDR4-2133,Type:289-pin D3MM,CAS:15Module:4x16GBSize:64GB	741.63	3

Ushbu misolda Oracle jadvallarni quyidagi tartibda ifodalaydi: FROM WHERE va SELECT

1. FROM operatorida ma'lumotlarni izlash uchun jadval ko'rsatilgan.

2. WHERE operatori esa shartga asoslangan qatorlarni filtrladi, masalan, product_name = 'Kingston').

3. SELECT operatori qaytarilishi kerak bo'lgan ustunlarni tanladi.

B) Taqqoslash operatori yordamida qatorlarni tanlash.

Tenglik operatoridan tashqari, Oracle bizga quyidagi jadvalda ko'rsatilgan boshqa ko'plab taqqoslash operatorlarini taqdim etadi:

Operator	Tavsif
=	Tenglik
!= <>	Tengsizlik
>	Katta
<	Kichik
>=	katta yoki teng
<=	kichik yoki teng
IN	Qiymatlar ro'yxatidagi har qanday qiymatga teng
ANY/SOME/ALL	Qiymatni ro'yxat yoki pastki so'rov bilan solishtirish Undan oldin =, >, < kabi boshqa operator bo'lishi kerak.
NOT IN	Qiymatlar ro'yxatidagi hech qanday qiymatga teng emas
[NOT] BETWEEN n and m	[Not] >= n va <= y ga teng.

Operator	Tavsif
[NOT] EXISTS	Agar quyi so'rov kamida bitta qatorni qaytarsa, true qiymatini qaytaradi.
IS [NOT] NULL	NULL testi

Misol uchun, ro'yxatdan narxlari 500 dan yuqori bo'lgan mahsulotlarni olish uchun quyidagi buyrug'lardan foydalanamiz:

```
SELECT
product_name,
list_price
FROM
products
WHERE
list_price > 500;
```

PRODUCT_NAME	LIST_PRICE
Gigabyte GA-Z270X-Gaming 9	503.98
Asus Rampage V Edition 10	519.99
Supermicro HSDG6-F	525.99
MSI X99A GODLIKE GAMING CARBON	549.59
Asus Z10FE-D8 WS	561.59
ASUS RAMPAGE V EXTREME	572.96
ASUS ROG MAXIMUS IX EXTREME	573.99
Asus X99-E-10G WS	649
Intel DP350PM	789.79

C) Ayrim shartlarga javob beradigan qatorlarni tanlash. Shartlarni birlashtirish uchun AND, OR va NOT mantiqiy operatorlardan foydalanishimiz mumkin. Masalan, 1-toifa identifikatoriga tegishli bo'lgan va ro'yxat narxi 500 dan yuqori bo'lgan barcha anakartlarni olish uchun quyidagi buyruqdan foydalanamiz:

```
SELECT
product_name,
list_price
FROM
products
WHERE
```

list_price > 500

AND category_id = 4;

Natijalar to'plami faqat ro'yxat narxi 500 dan yuqori bo'lgan anakartlarni o'z ichiga oladi.

PRODUCT_NAME	LIST_PRICE
Gigabyte GA-Z270X-Gaming 9	503.98
Asus Rampage V Edition 10	519.99
Supermicro H8DG6-F	525.99
MSI X99A GODLIKE GAMING CARBON	549.59
Intel Core i7-5930K	554.99
Asus Z10PE-D8 WS	561.59
Intel Xeon E5-1650 V3	564.89
Asus RAMPAGE V EXTREME	572.96
Asus ROG MAXIMUS IX EXTREME	573.99

D) Ikki qiymat orasidagi qiymatga ega bo'lgan qatorlarni tanlash

Ikki qiymat orasidagi qiymatga ega bo'lgan qatorlarni topish uchun

operatoridagi BETWEEN operatoridan foydalanamiz. Masalan, narxlar 650 dan 680 gacha bo'lgan mahsulotlarni olish uchun quyidagi buyrug'lar ketma-ketligidan foydalanamiz:

SELECT

product_name,
list_price

FROM

products

WHERE

list_price BETWEEN 650 AND 680

ORDER BY

list_price;

Quyidagi rasmda natijalar to'plami ko'rsatilgan:

PRODUCT_NAME	LIST_PRICE
Kingston	653.5
Corsair Dominator Platinum	659.99
Intel Core i7-3930K	660
Kingston	671.38
G.Skill Ripjaws V Series	677.99
Intel Core i7-7820X	678.75

Quyidagi iboralar ekvivalentdir:
list_price BETWEEN 650 AND 680
list_price >= 650 AND list_price <= 680

E) Qiymatlar ro'yxatida joylashgan qatorlarni tanlash. Qiymatlar ro'yxatidagi qatorlarni izlash uchun IN operatoridan quyidagi tarzda foydalanamiz:

SELECT

product_name,
category_id

FROM

products

WHERE

category_id IN(1, 4)

ORDER BY

product name;

Natija quyidagicha:

PRODUCT_NAME	CATEGORY_ID
AMD Opteron 6378	1
ASRock C2750D41	4
ASRock E3C224D4M-16RE	4
ASRock EP2C602-4L/D16	4
ASRock EP2C612 WS	4
ASRock Fatal1ty X299 Professional Gaming i9	4
ASRock X299 Taichi	4
ASRock X99 Extreme11	4
ASRock Z270 SuperCarrier	4
ASUS K6PFF-016	4

Ifodasi:

category_id IN (1, 4)

bir xil bo'ladi:

category_id = 1 OR category_id = 4

F) Qator qismi sifatida qiymatni o'z ichiga olgan qatorlarni tanlash. Quyidagi misolda Asus nomi bilan boshlangan mahsulotni tanlab oladi:

SELECT

product_name,

list_price

FROM

products

WHERE

```
product_name LIKE 'Asus%'
ORDER BY
list_price;
```

Ushbu misolda LIKE operatoridan foydalanib, qatorlarni moslashtirish ko'rib chiqilgan.

Oracle AND operatoriga kirish

AND operatori Boolean so'zlar va deklaratsiyalarini birlashtiruvchi mantiqiy operator hisoblanadi. Bu operator ikki ifoda ham rost bo'lganda rost, agar ifodalardan biri yolg'on bo'lsa, yolg'on qiymatni qaytaradi.

AND operatorning sintaksisi quyidagicha:
expression_1 AND expression_2

Quyidagi jadval AND operator yordamida rost va yolg'on qiymatlarini birlashtirgandagi natijani ko'rishimiz mumkin.

TRUE	TRUE	TRUE
TRUE	FALSE	FALSE
FALSE	TRUE	FALSE
FALSE	FALSE	FALSE

Agar MBda bir nechta mantiqiy operatorlardan foydalansangiz, Oracle har doim birinchi navbatda AND operatorini qo'llaydi. Biroq, baholash tartibini o'zgartirish uchun qavslardan foydalanishingiz mumkin.

Oracle AND operatoriga misollar

Namuna ma'lumotlar bazasida quyidagi orders jadvalga qarang:

ORDERS
* ORDER_ID
CUSTOMER_ID
STATUS
SALESMAN_ID
ORDER_DATE

A) Oracle MBda AND operatori orqali mantiqiy ifodalarni birlashtirish. Quyidagi misolda 2-mijozning kutilayotgan holatdagi buyurtmalari topiladi:

```
SELECT
order_id,
customer_id,
status,
order_date
FROM
orders
```

```
WHERE
status = 'Pending'
AND customer_id = 2
ORDER BY
order_date;
```

Ushbu misolda so'rov ikkala ifodani qoniqtiradigan barcha qiymatlarni qaytardi:
status = 'Pending'

va
customer_id = 2
Mana natija:

ORDER_ID	CUSTOMER_ID	STATUS	ORDER_DATE
78	2	Pending	14-DEC-15
44	2	Pending	20-FEB-17

B) AND operatoridan ikki va undan ortiq mantiqiy ifodalarni birlashtirish foydalanishimiz mumkin. Masalan, quyidagi misol barcha shartlarga javob beradigan buyurtmalarni o'z ichiga oladi:

```
SELECT
order_id,
customer_id,
status,
order_date
FROM
```

```

orders
WHERE
status = 'Shipped'
AND salesman_id = 60
AND EXTRACT(YEAR FROM order_date) = 2017
ORDER BY
order_date;

```

ORDER_ID	CUSTOMER_ID	STATUS	ORDER_DATE
77	1	Shipped	02-JAN-17
99	49	Shipped	07-JAN-17
104	18	Shipped	01-FEB-17

Ushbu misolda biz buyurtma sanasidan maydonni EXTRACT olish va uni 2017 yil bilan solishtirish uchun YEAR funksiyadan foydalandik.

C) Oracle da AND va OR operatorlarini birlashtirilgan holda misol ko'rib o'tamiz. Shart yaratish uchun AND operatoridan tashqari OR va NOT kabi boshqa mantiqiy operatorlardan ham foydalanishimiz mumkin.

Masalan, quyidagi so'rov mijoz identifikatori tomonidan berilgan buyurtmani topadi va uning holatini, bekor qilingan yoki kutilmoqda ekanligini aniqlab beradi.

```

SELECT
order_id,
customer_id,
status,
salesman_id,
order_date
FROM
orders
WHERE
(
status = 'Canceled'
OR status = 'Pending'
)
AND customer_id = 44
ORDER BY

```

ORDER_DATE	ORDER_ID	CUSTOMER_ID	STATUS	SALESMAN_ID	ORDER_DATE
	10		44 Pending	(null)	24-JAN-17
	69		44 Canceled		54 17-MAR-17

Oracle da BETWEEN operatoriga kirish

BETWEEN operatori sinov uchun diapazonni belgilash imkonini beradi. SELECT operatori orqali qaytariladigan qatorlar qidiruv shartini yaratish uchun BETWEEN operatori bilan qiyamatlari belgilangan diapazondagi qatorlar qaytariladi. Quyida BETWEEN operatorning sintaksisi tasvirlangan:

Ushbu sintaksisda:
A) quyi va yuqori
Quyi va yuqori qiymatlar tekshirish uchun diapazonning eng kichik va eng katta qiymatlarini belgilaydi. Quyi va yuqori qiymatlar harflar yoki ifodalar bo'lishi mumkin.

B) Ifoda
Quyi va yuqori diapazonni tekshirib ko'rish uchun berilgan ifoda hisoblanadi. Taqqoslash uchun quyi va yuqori ifodaning ma'lumotlar turlari bir xil bo'lishi kerak. C) AND operatori
AND operatori quyi va yuqori qiymatni ajratish uchun to'ldiruvchi vazifasini bajaradi. Ifodaning qiymati katta yoki teng (>=) dan quyigacha va yuqoridan kichik yoki tenggacha bo'lsa, BETWEEN operatori true qiymatini qaytaradi.

value >= min AND value <= max
NOT BETWEEN operatori BETWEEN operatorining natijasini inkor etadi.

BETWEEN operatori ko'pincha SELECT, DELETE va UPDATE operatorining WHERE qismida ishlatiladi. Endi, Oracle BETWEEN operatoridan foydalangan holda, ba'zi misollarini ko'rib chiqamiz.

A) Oracle MB da BETWEEN operatori orqali raqamli qiymatlarni filtrlash.
Namuna ma'lumotlar bazasida quyidagi products jadvalga qarang:

PRODUCTS	
PRODUCT_ID	
PRODUCT_NAME	
DESCRIPTION	
STANDARD_COST	
LIST_PRICE	
CATEGORY_ID	

Quyidagi misolda standart xarajatlari 500 dan 600 gacha bo'lgan mahsulotlarni qaytaradi:

```
SELECT
  product_name,
  standard_cost
FROM
  products
WHERE
  standard_cost BETWEEN 500 AND 600
ORDER BY
  standard_cost;
```

Ushbu misolda biz standart xarajat (standard_cost) ustunidagi qiymatlarni 500 dan 600 gacha bo'lgan oraliq bilan solishtirdik. So'rov faqat standart xarajatlari shu diapazon oraliq'ida bo'lgan mahsulotlarni qaytardi:

PRODUCT_NAME	STANDARD_COST
Asus Z10PE-D8 WS	504.14
G.Skill Trident X	507.32
Intel Core 17-3930K	509.32
G.Skill Ripjaws V Series	510.93
Intel Core 17-7820X	511.1
G.Skill Ripjaws V Series	517.78
EVGA 06G-P4-4998-NR	521.03
Intel Core 17-4830K	527.69
Intel Xeon E5-2630 V4	528.95
G.Skill Trident 2	532.27
G.Skill Trident 2	533.21
Zotac ZT-P10810C-10P	535.03
Intel Xeon E5-1650 V4	535.47
Corsair Dominator Platinum	537.63
Corsair Dominator Platinum	538.55

uchun Standart narxi 500 dan 600 gacha bo'lmagan mahsulotlarni izlash
 qo'shishimiz kerak: esa NOToperatorni yuqoridagi so'rovga quyidagicha

```
SELECT
  product_name,
  standard_cost
FROM
  products
WHERE
  standard_cost NOT BETWEEN 500 AND 600
ORDER BY
  product_name;
Natija quyidagicha:
```

PRODUCT_NAME	STANDARD_COST
ADATA ASU80055-128GT-C	37.78
ADATA ASU80055-512GT-C	113.29
AMD 100-5056062	1343.84
AMD 100-505989	2128.67
AMD 100-506061	706.99
AMD FirePro 57000	936.42
AMD FirePro W9100	2483.38
AMD Opteron 6378	651.92
ASRock C2750D41	339.55

B) Oracle da BETWEEN operatori orqali sanalari filtrlash.
 Namoyish uchun ma'lumotlar bazasidagi orders jadvaldan foydalanamiz:

ORDERS	
ORDER_ID	
CUSTOMER_ID	
STATUS	
SALESMAN_ID	
ORDER_DATE	

Quyidagi misolda mijozlar tomonidan December 1, 2016 va December 31, 2016 oraliq'ida berilgan buyurtmalarni qaytaradi:

```
SELECT
```



```

order_id,
customer_id,
status,
order_date
FROM
orders
WHERE
order_date BETWEEN DATE '2016-12-01' AND DATE '2016-12-
31'
ORDER BY
order_date;
Mana natija:

```

ORDER_ID	CUSTOMER_ID	STATUS	ORDER_DATE
87	7	Canceled	01-DEC-16
85	4	Pending	01-DEC-16
83	16	Shipped	02-DEC-16
82	44	Shipped	03-DEC-16
81	49	Shipped	13-DEC-16
80	3	Shipped	13-DEC-16
79	2	Shipped	14-DEC-16
102	45	Shipped	20-DEC-16

Ushbu bo'limda Oracle BETWEEN operatoridan ma'lum diapazondagi qatorlarni tanlash uchun qanday foydalanish haqida ma'lumot berilgan.

Nazorat uchun savol va topshiriqlar

1. MB qanday strukturalari mavjud?
2. Oracle MBBTda yozuvlar qanday indekslanadi?
3. Oracle MBBTda Order by qanday vazifani bajaradi?
4. Oracle MBBTda yozuvlar qanday o'chiriladi?
5. Oracle MBBTda jadvallar o'zaro qanday bog'lanadi?
6. Oracle MBBTda multimedia ma'lumotlari qanday saqlanadi?

7-BOB. SURATLARNI MA'LUMOTLAR BAZASIDA SAQLASH

7.1 Tasvirli ma'lumotlar shakllari

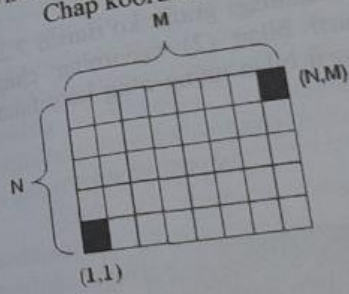
Raqamli tasvirlarni hosil qilish shakli va usullari Analog ikki o'lchovli signalni vaqt bo'yicha diskretlash va daraja bo'yicha kvantlash natijasida raqamli tasvir (RT) paydo bo'ladi. RTning eng kichik elementi piksel (pixel) deb ataladi. RT umumiy holda N ta qator va M ta ustundan iborat to'g'ri burchakli jadval ko'rinishida beriladi, bunda har bir element piksel bo'ladi. Bu jadvalni NxM elementlardan iborat matrisa ko'rinishida ham yozish mumkin.

RT pikselarini koordinatlarini grafik tasvirlash uchun turli usullardan foydalaniladi.

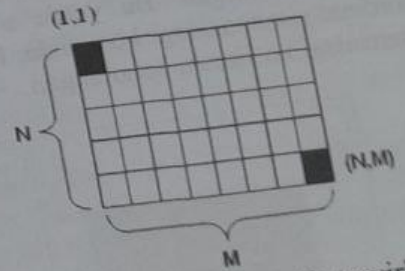
Tasvirlarni tanib olish masalalarida bitta RT turli usullarda keltirilishi mumkin, ya'ni dekart yoki qutbli koordinata sistemalarida.

7.1-rasmda RTni ikki xil usulda dekart koordinata sistemasida tasvirlash ko'rsatilgan.

Chap koordinat tizimi



O'ng koordinat tizimi



7.1-rasm. Dekart koordinat sistemasida RTni ikki xil usulda tasvirlash.

Chap koordinat sistema X o'qini chapdan o'ngga yo'nalishiga mos keladi. O'ng koordinat sistema Yo'qini pastdan yuqoriga yo'nalishiga mos keladi. Shu sababli RTni ifodalovchi matrisaning pastki chap tomonida (1,1) koordinatli piksel joylashadi, yuqori o'ng tomonda esa (N,M) koordinatli piksel joylashadi.

O'ng koordinat sistemada RT pikselarini tartibli hisobi unga matrisaning yuqori chap burchagidan boshlanib o'ng pastki burchak

koordinatlar. Koordinatning bunday ifodalaniشى umumiy qabul
 etilgan deb o'lganli uchun dekart sistemasiga mos kelmasda, a RT XU
 tekislikda esa ifodalashda ko'p qo'llaniladi. (x_1, u_1) va (x_2, u_2)
 koordinatlar deb piksel orasidagi d masofa quyidagicha aniqlanadi:

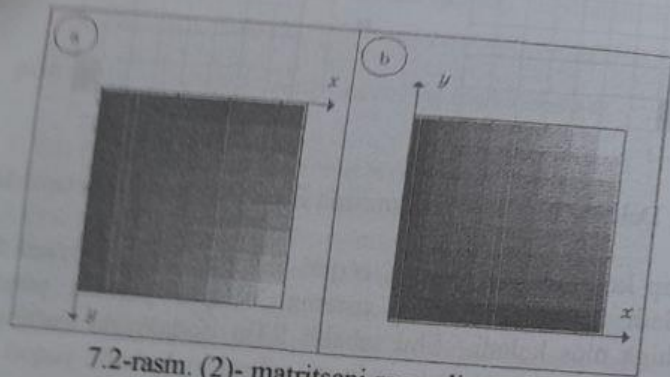
$$d_{12} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (1)$$

Bu piksel o'lchovli tasvirni aniqlovchi 8-tartibli matrisa
 berilgan.

1	2	3	4	5	6	7	8
2	4	6	8	10	12	14	16
3	6	9	12	15	18	21	24
4	8	12	16	20	24	28	32
5	10	15	20	25	30	35	40
6	12	18	24	30	36	42	48
7	14	21	28	35	42	49	56
8	15	24	32	40	48	56	64

(2)

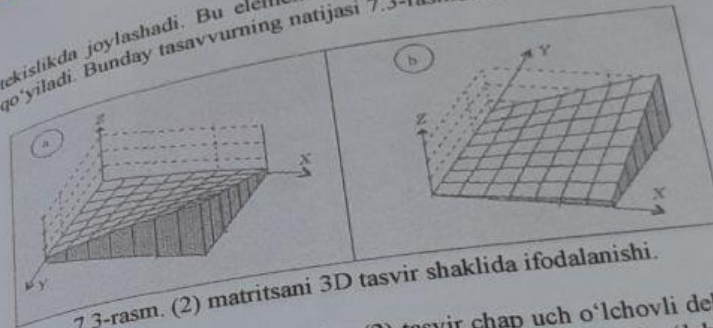
Bu tasvirni dekart koordinat sistemasidagi grafik ko'rinishi 7.2-
 rasmda ko'rsatilgan. Bu yerda a xarfi bilan (2) tasvirning chap
 koordinat sistemasidagi ko'rinishi, b xarfi bilan uning o'ng koordinat
 sistemasidagi ko'rinishi belgilangan.



7.2-rasm. (2)- matritsani raqamli tasviri.

(2) matrisa uch o'lchovli dekart koordinat sistemasida ham grafik
 ko'rinishda keltirilishi mumkin. Bu holda matrisaning elementlari XU

tekislikda joylashadi. Bu elementlarning qiymatlari Z o'qi bo'yicha
 qo'yiladi. Bunday tasavvurning natijasi 7.3-rasmda ko'rsatilgan.



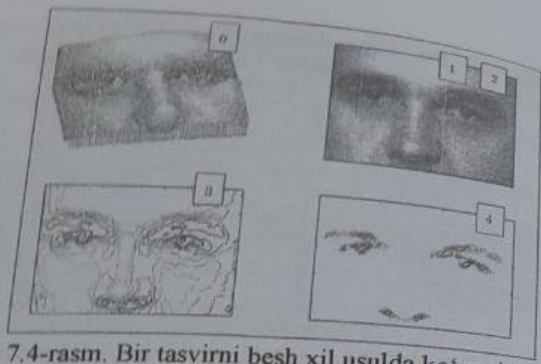
7.3-rasm. (2) matritsani 3D tasvir shaklida ifodalaniشى.

7.3-rasmda a xarf bilan (2) tasvir chap uch o'lchovli dekart
 dekart koordinat sistemasida belgilangan, b xarf bilan esa o'ng uch o'lchovli

Terminologiya bo'yicha 3D ko'rinishida keltirilgan tasvirlar "0"
 sinf raqamli tasvirlarga kiradi. «0 sinfini» tasvirlarni aniqlashda umum
 qabul qilingan sinf tushunchasi bilan adashtirmaslik uchun uni usul
 degan tushuncha bilan almashtiramiz. Raqamli tasvirlarni ta'riflash va
 ifodalash uchun beshta usul kiritilgan, ulardan 1-4 usullar tasvirlarni 2D
 shaklda ifodalashga mo'ljallangan. Oxirgi usul o'zining alohida
 nuqtalari yoki lokal sohalari bilan keltirilgan yarimtonli binar, konturli
 va tasvirlarga bo'lingan.

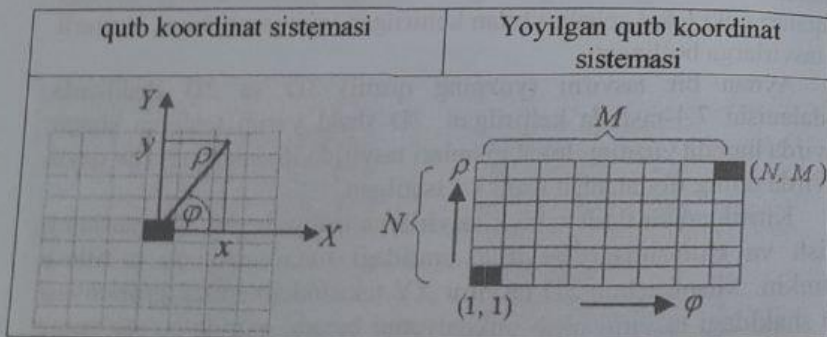
Aynan bir tasvirni (yuzning qismi) 3D va 2D shakllarda
 ifodalaniشى 7.4-rasmda keltirilgan. 2D shakl yarim tonli va kontur
 tasvirda hamda yuzning lokal qismlari tasvirida ifodalangan. Har qaysi
 tasvirda uning ifodalaniشى usuli ko'rsatilgan.

Kiritilgan sinflash tufayli tasvirlarga ishlov berish sistemasining
 kirish va chiqishdagi tasvirlar orasidagi munosabatlarni ta'riflash
 mumkin. Misol uchun 3D tasvirni XY tekislikka vertikal proeksiyasi
 2D shakldagi tasvirni olish imkoniyatini beradi, oxirida bo'sag'aning
 turli qiymatlarida kesiklari yarimtonli 2D tasvirni binar ko'rinishga
 o'tkazadi. O'z navbatida binar tasvir qandaydir konturlash prosedurasi
 yordamida osonlikcha konturga o'tkaziladi va x.k. RTning bunday
 shakl o'zgartirilishi timsollarni aniqlash masalalarida tasvirlardan
 belgilarni ajratib olishda tez-tez qo'llanib turiladi.



7.4-rasm. Bir tasvirni besh xil usulda ko'rsatish.

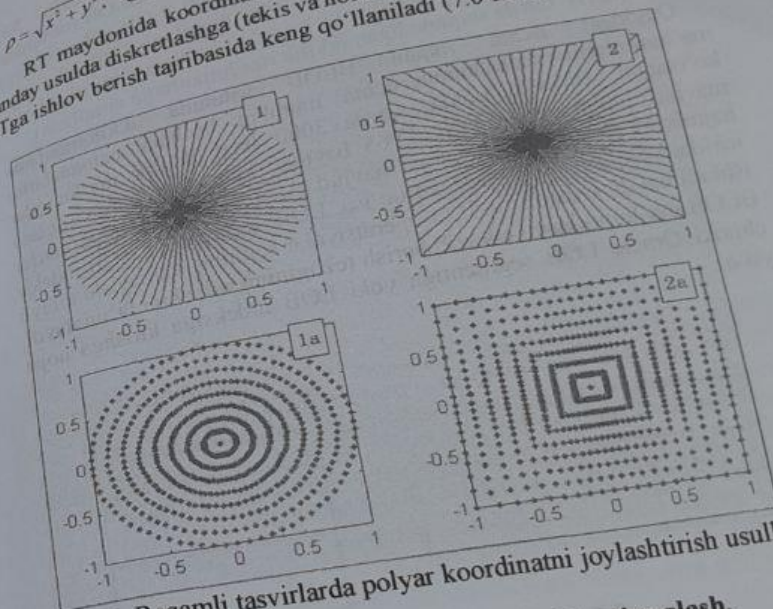
Raqamli tasvirlarni grafik ifodalash uchun boshqa koordinat sistemalarni (odekart) tashlash, yechilayotgan masalaning bevosita qo'llanish sohasi va undan kelib chiqadigan xususiyatlari asosida aniqlanadi. Masalan, timsollarni aniqlashda keng foydalaniladigan Fure-Mellin o'zgartirishida RT spektri dekart koordinat sistemasidan to'g'ri burchakli qutb koordinat sistemasiga o'tkaziladi. Bu oddiy qutb koordinat sistemasini yoyilmasi hisoblanadi. RTning bunday ifodalanishining xususiyatlari 7.5-rasmida ko'rsatilgan.



7.5-rasm. qutb koordinat sistemasining variantlari va piksellarning joylashuvi.

Bir sistemadan boshqa sistemaga o'tkazilayotganda piksellarning koordinatalari quyidagi munosabatlar bo'yicha hisoblanadi.

$\rho = \sqrt{x^2 + y^2}$; $tg(\varphi) = \frac{y}{x}$, $x \neq 0$; $x = \rho \cos(\varphi)$; $y = \rho \sin(\varphi)$. (3)
 RT maydonida koordinatalarni joylashtirish usuli qutb radiusini qanday usulda diskretlashga (tekis va notekis) bog'liq. Bu ikki usul ham RTga ishlov berish tajribasida keng qo'llaniladi (7.6-rasm).

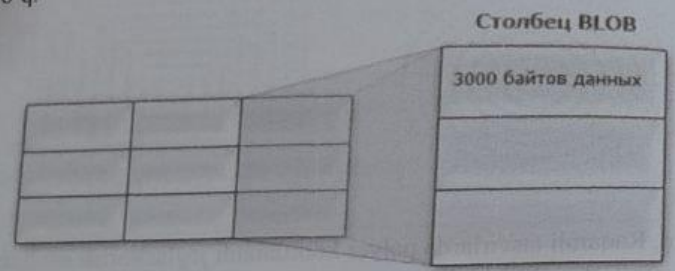


7.6-rasm. Raqamli tasvirlarda polyar koordinatni joylashtirish usullari.

7.2 Ma'lumotlar bazasida tasvirlarni saqlash.

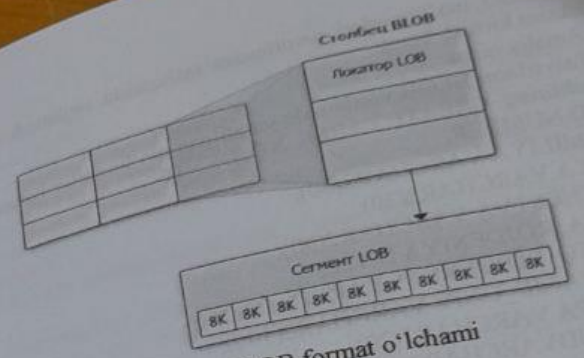
BLOB - bu katta ikkilikli obyektlarni belgilash uchun ma'lumotlar bazasida ishlatiladigan qisqartma. BLOB kolonkalarini bir necha yil oldin Oracle tomonidan ikkilik ma'lumotlarni saqlash uchun LONG RAW texnologiyasini almashtirish uchun joriy etilgan. BLOB ma'lumotlar turining arxitekturasi uchta asosiy qismga bo'linadi: BLOB ustuni, LOB segmenti va LOB indeksi. BLOB ustunida LOB qidiruvi (36 bayt) va ikkilik ma'lumotlari, agar satr uzunligi 3965 baytdan kam bo'lsa va ustun uchun satr saqlanmagan bo'lsa. Agar ikkilik ma'lumotlarning o'lchami 3964 baytdan oshsa, BLOB ustunidagi satrlarda bo'sh joy ajratilmaydi va LOB lokatori LOB

segmentida saqlanadigan ikkilik ma'lumotlarga tegishlidir. Shu munosabat bilan, har doim qatorlarda saqlanadigan BLOB ustunidagi ajratilgan joy) va 4000 baytgacha bo'lishi mumkin (LOB qidiruvchisiga ajratilgan bo'shliqning yig'indisi va mumkin bo'lgan (LOB lokatoriga joy miqdori). satrda saqlanadigan ikkilik ma'lumotlarga ajratilgan).
 Quyidagi uchta raqam BLOB ustunida ajratilgan).
 ma'lumotlarni saqlashning uchta mumkin bo'lgan namunalarni ko'rsatadi. Birinchi holda, satrda 3000 bayt ikkilik ma'lumotlar mavjud, chunki 3000 bayt 3,965 baytdan kam yoki satrda saqlash hajmining chegaraviy chegarasi mavjud. Agar BLOB ustunidagi saqlash ichidagi saqlash o'chirilmagan bo'lsa, LOB segmenti va LOB indeksi ishlatilmaydi. Odatda, bu ILO-operatsiyalar sonining kamayishi tufayli BLOB ma'lumotlarini etkazib berish tezligining oshishini ta'minlaydi, chunki Oracle LOB segmentiga yoki LOB indeksiga kirishga hojat yo'q.



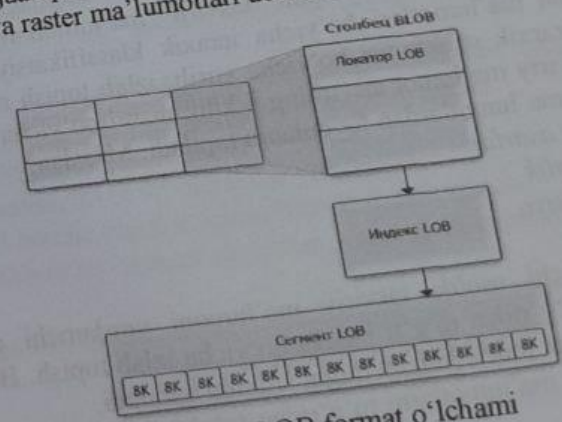
7.7-rasm.BLOB format o'lchami

BLOB hajmi 3965 baytdan kam bo'lgan ma'lumotlar satrlar ichida saqlanadi. 40-rasmda ikkilik ma'lumotlarning hajmi 3964 baytdan oshib ketadigan ikkinchi misol keltirilgan (bu holda ma'lumotlar hajmi 81920 bayt) va ma'lumotlar chiziqlar ichiga sig'maydi. Shu munosabat bilan, LOB lokatori LOB segmentida saqlanadigan ikkilik ma'lumotlarga ishora qiladi. Ikkilik ma'lumotlar LOB segmentida 12 dan ortiq bloklarni egallamaganligi sababli, LOB lokatori uning manzillarini saqlaydi. Bunday holda, LOB indeksi ishlatilmaydi.



7.8-rasm.BLOB format o'lchami

BLOB ma'lumotlari 3964 baytdan katta bo'lgan satrlar tashqarida saqlanadi. Segment jadvalidagi LOB lokatori ma'lumotlar saqlanadigan LOB segmentini bildiradi. 7.8-rasmda LOB indeksi ularni joylashirish uchun zarur bo'lgan shunday katta o'lchamdagi (106496 bayt) ikkilik ma'lumotni ko'rsatadi. Bunday holda, ikkilik ma'lumotlarning hajmi strings ichidagi saqlash uchun belgilangan chegaradan oshadi va ularni saqlash uchun LOB segmentida 12 ta blok kerak bo'ladi. Ushbu o'lchamdagi ma'lumotlar uchun LOB lokatori LOB segmentidagi bloklarning joylashishini aniqlash uchun LOB indeksiga murojaat qiladi. Ushbu holat vektor ma'lumotlari uchun juda kam uchraydi va raster ma'lumotlari uchun buni oldini olish mumkin.



7.9-rasm.BLOB format o'lchami

BLOB ma'lumotlari qatorlardan tashqarida saqlanadi va LOB indeksini kiritish kerak.

Amaliy qism

Tasvirli BLOB yordamida yaratilgan ma'lumotlar bazasi ustiga joylashtiring. CREATE TABLE STUDENT6

(ID NUMBER,
FAMILIYA VARCHAR2(30),
IMYA VARCHAR2(30),
OTCHESTVA VARCHAR2(30),
DATA_ROJDENIYA DATE,
MESTO_ROJDENIYA VARCHAR2(100),
NAZVANIE_UNIV VARCHAR2(100),
GRUPPA VARCHAR2(20),
STIPENDV VARCHAR2(50),
KARTINKA BLOB);

7.3 Suratlar MBda izlash

Tasvirli ma'lumotlarni saqlash, ularni kerak bo'lganda izlab topish vazifalari multimediali ma'lumotlar bazasining muhim vazifalaridan hisoblanadi. MB saqlanayotgan tasvirli ma'lumotlarni izlashning bir-necha xil usullari mavjud.

1. Tasvirli ma'lumot saqlanayotganda to'g'ridan-to'g'ri tasvirning ikkilik kodidan tashqari MB da tasvir haqidagi ma'lumotlar ham saqlanadi. Saqlanayotgan tasvirli ma'lumotlarni ana shu qo'shimcha ma'lumotlari bo'yicha ieraxik klassifikatsiyalab, izlash paytida ieraxik strukturasi bo'yicha yurib, izlab topish mumkin. Bu usulda tasviriy ma'lumot tasvirning ikkilik matni bo'yicha emas balki yordamchi ma'lumotlardan foydalanib topiladi. Masalan,

Badiiy asarlar

Rassomlik

O'rta asr

16 asr.

2. Ikkinchi usulda tasviriy ma'lumoni yordamchi atamalarini indekslanib, to'g'ridan to'g'ri indeks bo'yicha izlab topish. Bu usulda tasvirli ma'lumot indekslangan hujjat sifatida qaraladi.

3. Tasvirli ma'lumotlarni to'g'ridan-to'g'ri ikkilik kodi bo'yicha

ham izlash mumkin. Bunda ikkilik kodlar avvaldan olingan shablona solishtirib topiladi.

Tasvirli ma'lumotlarni ikkilik kodi bo'yicha izlanganda shablon va ikkilik matni yuz foiz mos kelishligi kutilmaydi.

- Tasvirli ma'lumotlarni MB izlashda so'rovlar turi:
 - Tasvirni ma'lum belgilariga ko'ra izlab topish (rang, tekstura, figura va h.k.)
 - Ma'lum tipdagi obyektlarni o'zida saqlagan tasvirlarni izlab topish.
- Tasvirlardagi obyektlarni o'zida saqlagan tasvirlarni izlab topish. Masalan, ma'lum figuralar (aylana, uchburchak va h.k.), o'lchovi, rangi bo'yicha va h.k.
- A tipidagi obyekt B tipidagi obyekt dan chapda joylashgan tasvirni izlab topish. (makondagi munosabatlar)
- O'xshashlik bo'yicha izlash: berilgan tasvirga (segmentga) o'xshash tasvirni (segmentlarni) izlab topish.

7.4 Suratlar uchun ma'lumotlar bazasi strukturasi

Tasvirli ma'lumotlarni quyidagicha strukturada saqlash mumkin:

- Piksellar matritsasini ketma-ket saqlash (siqilgan yo siqilmagan holda). Bunday struktura diskda katta joy talab qiladi.
- Har bir suratni alohida fayl sifatida saqlash.

Relyatsion strukturasi:

Suratni tasvirlash: surat identifikatori va uning asosiy xususiyatlari (atributlari)

- Obyektni tasvirlash: surat ichidagi obyektlar (segmentlar, to'rtburchaklar); qo'lda yoki avtomatik chiqarib olinadi.

Makonli strukturasi:

- Masalan, R- ili R*-daraxtlar yordamida
- MB barcha suratlar uchun bitta R-daraxt qurish
- Makonli bo'lmagan obyektlar uchun alohida indeks qurish

Tasvirlarga ishlov berish algoritmlarida foydalaniladigan belgilar sistemasi

Qandaydir RT piksellarning $N \times M$ o'lchovli matriwa ko'rinishida

yozilgan. U holda bu tasvirning har bir pikselini quyidagi ko'rinishda yozamiz

$$p(n, m), \quad (4)$$

Bu yerda $n=1, 2, \dots, N$ i $m=1, 2, \dots, M$.

Barcha tasvirlarni yoki ($N=M$ hol uchun) ko'rinishda yozish mumkin. Bunda matrisani belgilashdagi quyi indeks (ifoda) doim uning tartibini belgilaydi (yoki hol uchun o'lchov).

RT ifodalovchi har bir matrisaga transponirlash, aylantirish, kompleks qo'shish, darajaga ko'tarish va x.k. operatsiyalarni qo'llash mumkin. Ularni bu operatsiyalar uchun qabul qilingan belgilashlar ko'rinishida yozish mumkin. Masalan:

$$[P_N]^T, [P_N]^H, [P_N]^*, [P_N]^* \quad (5)$$

N tartibli nol va birlik matrisalarni belgilash uchun quyidagi belgilashlardan foydalaniladi:

$$[0]_N \text{ va } I_N, \text{ qachonki } [0] = 0 \text{ va } I_1 = 1. \quad (6)$$

quyida tahlil qilinadigan RT ga ishlov berish va aniqlash proseduralarida matrisalarni oddiy (dekart) ko'paytirishdan tashqari yana ikki tipdagi ko'paytirishdan foydalaniladi: to'g'ri va nuqtaviy.

va matritsalar uchun to'g'ri (kronekerov) ko'paytirish quyidagicha yoziladi:

$$A_N \otimes B_M = C_{(NM)} \quad (7)$$

bu yerda matrisa NM tartibga ega.

Matritsalarini to'g'ri ko'paytmasi o'ng va chap bo'lishi mumkin. Ikki matrisaning o'ng ko'paytmasida natija bloklar orqali shunday shakllanadiki, chap matrisaning har bir elementi o'miga shu elementni o'ng matrisaning barcha elementlariga ko'paytmasining natijasi yoziladi. - natijaviy matrisa quyidagi shaklga ega bo'ladi.

$$C_{(NM)} = \begin{bmatrix} a_{11}B_M & \dots & a_{1N}B_M \\ \dots & \dots & \dots \\ a_{N1}B_M & \dots & a_{NN}B_M \end{bmatrix} \quad (8)$$

A_N va B_M matritsalarining nuqtaviy ko'paytmasi quyidagicha yoziladi

$$A_N \odot B_M = C_N, \quad (9)$$

Bunda N tartibli C_N matritsa quyidagicha aniqlanadi:

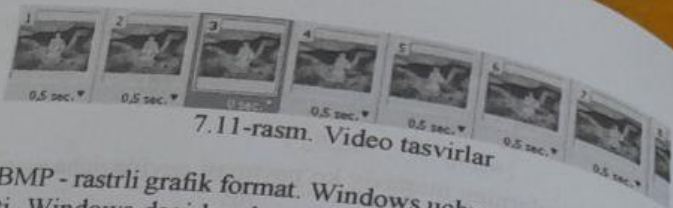
$$C_N = \begin{bmatrix} a_{11}b_{11} & \dots & a_{1N}b_{1N} \\ \dots & \dots & \dots \\ a_{N1}b_{N1} & \dots & a_{NN}b_{NN} \end{bmatrix} \quad (10)$$

Raqamli tasvirlar formatlari

Statik raqamli tasvir biror tasvirga oluvchi qurilma (fotokamera, skaner) yordamida olingan raqamli tasvir deyiladi (10-rasm). Ular *.bmp, *.jpg, *.pcx, *.tif kabi formatlarda saqlanadi. Bir necha statik tasvirlarni biror dastur orqali ketma-ket yoki takroriy tasvirlangan tasvirlar majmuasi dinamik tasvir deyiladi. Dinamik tasvirlar *.gif, *.avi, *.mov, *.mpg kabi formatlarda saqlanadi. Dinamik tasvirlarga animatsiyalar, video tasvirlarni misol qilib aytish mumkin (7.11-rasm). Statik tasvir video tasvir kadrigan ham olinishi mumkin.



7.10-rasm. Raqamli tasvirlar



7.11-rasm. Video tasvirlar

BMP - rastrli grafik format. Windows uchun standart grafik fayllar formati. Windows dagi barcha tasvirlarni uchun standart grafik fayllar fayllarni yarata va o'qiy oladi. Tasvirning tahrir qilish dasturlari BMP ranglarga belgilash mumkin. Bu formatda tasvir oq-qoradan to'liq Taxminan 16,7 million xil rang ishlatiladi.

GIF - Graphics Interchange Format (grafik ma'lumotlarni almashish formati). CompuServe firmasining standarti, u rastrli rangli tasvirlarni aniqlash uchun qo'llaniladi. Tarmoq orqali tarqatish uchun mo'ljallangan ushbu format fayllari NURISTON.TK juda kichkina bo'ladi. Format kulrangning 256 xil jilosi yordamida oq-qora tasvirni va 256 xil yoki undan kam rangni qo'llab, rangli tasvirni yozib oladi. Tasvirni animatsiyada ham ishlatish mumkin.

JPEG (JPG) - asosan rastrli tasvirlar (fotosuratlar, rasmlar va b.) uchun ishlatiladi. JPEG formatida yuqori darajada siqilganligi tufayli, tasvir fayli hajmi sezilarli darajada kamayadi. Lekin bunda tasvir sifati yo'qoladi. Bugungi kunda JPEG fotosuratlar va ko'psonli ranglar ishlatilgan tasvirlar uchun eng yaxshi format sanaladi. U Internetda ishlatish va elektron pochta orqali jo'natishga qulay. Taxminan 16,7 million xildagi rang qo'llaniladi.

PSD - Photo Shop Data. Fotosuratlarni qayta ishlovchi fotoshop dasturida yaratilgan fayllar formati. Taxminan 250 trillion xil rang qo'llaniladi.

TIFF (TIF) - Tagged Image File Format (tasvirli fayllarning birlashtirilgan formati), rastrli grafik format. Bu format tasvirlarni yuqori sifati ta'minlaydi va kompyuterlar o'rtasida ma'lumotlarni almashishdagi standart format sanaladi. TIFF formati tasvirni ma'lumotlarni yo'qotmay siqish imkonini beradi. Raqamli kameralar tasvirlarini saqlanuvchilari tomonidan keng qo'llaniladi. Taxminan 16,7 million xil ranglar mavjud.

Dinamik tasvirlarni qayta ishlashda barcha statik tasvirlarida bir xil xil rangni qayta ishlash usullari qo'llaniladi. Shuning uchun tasvirlar yaratish usullari va algoritmlarini statik raqamli tasvirlar uchun

qarash yetarli bo'ladi.

Tasvirlarni qayta ishlashda avvalo tasvirning rang xususiyatlari va unda ishlov usullarini o'rganib chiqish talab etiladi. Hozirgi zamonaviy kompyuterlarda grafik rejim ranglidir. Ya'ni bitta pikselda uchta rang (R-qizil, G-yashil, B-ko'k) aralashmasidagi rang qiymati bo'ladi. Unda mumkin bo'lgan ranglar soni $256^3=16777216$ taga yetadi. Bu rejim jonli tabiatdagi kuzatilgan ranglardan qolishmaydigan tasvirni saqlash, ishlov berish va uzatish imkonini beradi. Har qanday rangni quyidagi uchta asosiy bo'lgan - qizil, yashil va ko'k ranglarning aralashmasi yordamida tasvirlash mumkin. Agar biz 3 bayt yordamida nuqtaning rangini kodlashtirmoqchi bo'lsak, unda 1-bayt qizil, 2-bayt yashil, 3-bayt esa ko'k rangni ifodalaydi. Rangli to'planning bayt qiymati qanchalik katta bo'lsa, mazkur rang shunchalik aniq va ravshan bo'ladi. Agar nuqta oq rangdan iborat bo'lsa, demak unda ranglar mavjud bo'lib, u to'liq va ravshan bo'ladi. Shuning uchun ham oq rang uchta to'liq bayt 255,255,255 bilan kodlanadi. Qora rangda hamma mavjud ranglar (R-qizil, G-yashil, B-ko'k) bo'lmaydi, ya'ni jami ranglar to'plami nolga teng bo'ladi. o'ora rang 0,0,0 bilan kodlanadi. Kulrangda jami ranglarni tashkil etuvchi to'plam mavjud bo'lib, ular bir xil va bir-birini neytrallashtiradi. Masalan, kul rangni 80,80,80 yoki 120,120,120 bilan kodlashtirish mumkin. Ko'rinib turibdiki, ikkinchi holatdagi kodlashtirishda aniqlik va ravshanlik yuqori, ya'ni 80,80,80 bilan kodlashtirishga qaraganda 120,120,120 bilan kodlashtirish deyarli yorug'roqdir. o'ziz rangda esa qizil rangdan tashqari boshqa jami ranglarni tashkil etuvchilari nolga teng bo'ladi. Masalan, to'q qizil rang 125,0,0 yoki ochiq qizil rang 255,0,0 ko'rinishda kodlanadi. Dasturiy tizimda tasvirlarni piksellar bo'yicha aniqlanadi va qayta ishlanadi. Unda asosan BMP (Bitmap) kengaytmali grafik tasvirlar qayta ishlanadi. Tasvirdagi har bir piksel o'n olti yoki o'nli sanoq sistemasidagi sonlarni qabul qiladi. Nuqtadagi rang qiymatini qabul qilish uchun 000000(16) dan FFFFFFFF(16) gacha oraliqda bo'lgan o'n olti sonlar uchun oltita yacheyka (joy) ajratilgan. Bunda birinchi ikkita yacheyka ko'k rang uchun, keyingi ikkita yacheyka yashil rang uchun va nihoyat oxirgi ikkita yacheyka qizil rang qiymatlari uchun ajratilgan. Masalan, tasvirdagi ixtiyoriy (x,y) nuqtadagi rang qiymati $6BC8AD_{16}$ (7063725_{10}) ga teng bo'lsin. Bunda ko'k rang qiymati $6B_{16}$ (107_{10}) ga, yashil rang qiymati $C8_{16}$ (200_{10}) ga va qizil rang qiymati AD_{16} (173_{10}) ga teng. Shu tariqa biz yuqoridagi ma'lumotlar asosida grafik

tasvirlarga ishlov bera olamiz. Tasvirlarni qayta ishlash jarayonida turli usullar qo'llaniladi. Masalan, binar tasvirga o'tkazish, obekt chegaralarini aniqlash, sohalarni bo'laklash (segmentasiya), ingichkalashtirish, tasvir sifatini yaxshilash va x.k. o'uyida tasvirlarni qayta ishlash bilan bog'liq bo'lgan bir necha usullar keltirilgan.

Tasvir sifatini yaxshilash

Texnik qurilmalarning ma'lumotlarni qabul qilish xususiyati va suratga olish paytidagi yorug'lik darajalari kabi omillar tasvir sifatiga turlicha ta'sir qiladi. Agar tasvir sifati yomon bo'lsa, uni yaxshilash zarur. Chunki, sifatli bo'lmagan tasvirlarda yuz elementlarini topish qiyin kechadi. Tasvir sifatini yaxshilashning turli usullari mavjud [3-5]. Masalan, chegaralarni kuchaytirish, halaqitlarni yo'qotish, tiniqlikni oshirish va x.k.

Tasvir sifatini yaxshilashda mediana usulida filtrlash keng ko'llaniladi. Bu usulning mohiyati tasvir bo'ylab biror oyna bilan xarakterlanish va markaziy nuqta qiymati oynadagi qiymatlarni kattaligi bo'yicha tartiblanganda o'rta tushuvchi qiymat bilan almastiriladi. Misol uchun, 3×3 oyna markazida 5, ikki yonida 35, 40, yuqorisida 1, 41, 52 va pastida 23, 17, 89 qiymatlar joylashgan deb faraz qilaylik. Ularni tartiblaymiz: 1, 5, 17, 23, 35, 40, 41, 52, 89. Markazdagi qiymat (mediana) 35 ga teng. Demak, 5 o'miga 35 yoziladi: $g(m, n) = med(f(x, y))_{(x, y) \in W(m, n)}$.

$W(m, n)$ - markazi (m, n) dagi oyna, $f(x, y)$ - shu oynadagi nuqtalar qiymati. Natijada anchagina tekislangan tasvir xosil bo'ladi. Chiziqli tiniqlashtirish usuli ham tasvirlar sifatini oshirish masalalarida ko'p qo'llaniladi. Uning ko'rinishi quyidagicha:

$$G'(x, y) = 255 \cdot \frac{G(x, y) - G_{Min}}{G_{Max} - G_{Min}}$$

Bu yerda G_{Min} va G_{Max} - tasvirdagi eng kichik va eng katta rang qiymatlari, $G(x, y)$ - (x, y) koordinatadagi rang qiymati.

Shuningdek, tasvir gistogrammalari tahlili asosida ham tasvir tiniqligini oshirish mumkin. U quyidagi formula asosida amalga oshiriladi:

$$Q = \sum_{i=1}^255 (H_i)^2, \quad S_{0(n,n)} = \frac{255 \cdot Q(n,n)}{Q}$$

$$G'(x, y) = \begin{cases} 0, & S_{0(n,n)} \leq 0, \\ 255, & S_{0(n,n)} \leq 255 \end{cases}$$

(23)

bu yerda H_i - tasvirdagi i rangdagi piksellar soni ($0 \leq i \leq 255$), $G(x, y) - (x, y)$ koordinatada joylashgan pikselning rang qiymati, r - normallashtiruvchi parametr.

bo'lsa, u holda

$$I(x, y) = q * f(x, y) + C$$

yoki

$$I(x, y) = f(x, y) + C$$

almashtirishlar bajariladi, bu yerda q, S - konstantlar. Tasvirda halaqitlarni bartaraf etishda qo'llaniladigan filtrlar ikki sinfga ajratiladi: quyi chastotali va yuqori chastotali fazoviy filtrlar. Sochma dog'lar, g'alayonlar spektri uchun quyi chastotali filtrlar qo'llanilib, keyin obektدا chenarlarni belgilash yoki ajratib olishda yuqori chastotali filtrlar qo'llaniladi. Ularni ishlashi uchun tahlil etilayotgan

Tasvirda halaqitlarni yo'qotish usullari

Tasvirlarni qayta ishlash jarayonida turli usullar qo'llaniladi. Masalan, binar tasvirga o'tkazish, obekt chegaralarini aniqlash, sohalarni bo'laklash (segmentasiya), ingichkalashtirish, tasvir sifatini yaxshilash va x.k. o'uyida tasvirlarni qayta ishlash bilan bog'liq bo'lgan bir necha usullar keltirilgan.

Tasvir sezgichlar va tasvirni EHM xotirasiga uzatgichlar xatoligi odatda fazoviy korrelyatsiyaga ega bo'lmagan yakalangan nuqtalar tarzidagi halaqitdan iborat bo'ladi, ya'ni qiymati o'zgargan (buzilgan) nuqtalar atrofdagilaridan sezilarli farq qiladi. Bu xususiyat halaqitlarni yo'qotishga qaratilgan ko'pgina algoritmlarning asosi bo'lib xizmat qiladi. o'alaqlarni yo'qotishning keng tarqalgan usullaridan biri tekislashdir.

Umumiy usullar bilan tekislikdagi tasvirni, mahalliy usullarga ko'rib, umumiy usullar bilan tekislikdagi tasvirni qayta tekislikdagi tasvirga aylantirish mumkin. Bu usullar bilan tekislikdagi tasvirni qayta tekislikdagi tasvirga aylantirish mumkin. Bu usullar bilan tekislikdagi tasvirni qayta tekislikdagi tasvirga aylantirish mumkin.

Umumiy usullar bilan tekislikdagi tasvirni, mahalliy usullarga ko'rib, umumiy usullar bilan tekislikdagi tasvirni qayta tekislikdagi tasvirga aylantirish mumkin. Bu usullar bilan tekislikdagi tasvirni qayta tekislikdagi tasvirga aylantirish mumkin.

$$f(x, y) = \sum_{n, m} f(n, m)$$

bu yerda $f(x, y)$ va uning atrofini o'z ichiga oluvchi to'plan, n -to'plamdagi nuqtalar soni, $f(n, m)$ -ularning qiymatlari.

Tasvirlarni sonli qiymatga o'tkazish xarakteristiklarini yo'qotish uchun tasvirning sonli ko'rinishi bir necha marta hosil qilinadi.

Agar $f(x, y)$ halaqitlar ta'sir etmayan manba tasvir, $n(x, y)$ halaqitlar funksiyasi bo'lsa, buzilgan tasvir quyidagi ko'rinishga ega $(x, y) = f(x, y) + n(x, y)$, $f(x, y)$ ning bir necha sonli nusxasidan olingan o'rtacha $g(x, y)$ tasvir:

$$g(x, y) = 1/k \sum_{i=1}^k g(x, y)$$

k -nusxalar soni uchun $E\{g(x, y)\} = f(x, y)$ va $\sigma^2\{g(x, y)\} = \sigma^2\{n(x, y)\}/k$ bo'ladi, bu yerda $E\{g(x, y)\}$ g ning kutilgan qiymatlari, $\sigma^2\{g(x, y)\}$ va $\sigma^2\{n(x, y)\}$ - g va n funksiyalarining chetlanish darajalari. Umumiy ko'rinishda quyidagicha bo'ladi: $\sigma^2\{g(x, y)\} = \sigma^2\{n(x, y)\}$. Bu tenglamalardan ko'rinib turibdiki k oshgan sayin qiymatlarining chetlanishi kamayadi. [10]

Halaqit uchun fazoviy korrelyatsiya yo'qligi uning spektrida tasvir spektriga nisbatan yuqori chastotalar bo'lishiga olib keladi. Demak,

$A(m, n)$ etalon niqobli past chastotali filtrlash tasvirni tekislash uchun yaxshi samara beradi:

$$g(i, j) = \sum_{m=0}^M \sum_{n=0}^L f(i-n, j-m) A(m, n)$$

bu yerda (i, j) -natija, $f(i, j)$ - manba tasvirlar, niqob o'lchami $(2L+1) \times (2M+1)$.
 Tekislash uchun ishlatiladigan etalon niqobdagi qiymatlar musbat, markazida esa odatda atrofdagi qiymatlardan kichik bo'lmagan qiymar bo'ladi. 3×3 o'lchamli niqoblar keng tarqalgan, ularga misol qilib quyidagilarni ko'rsatish mumkin:

$$A_1(m, n) = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}; \quad A_2(m, n) = \frac{1}{10} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix}; \quad A_3(m, n) = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

bu niqoblarni qo'llaganda qiymatlar aniqlanish sohasidan chiqib ketmasligi uchun me'yorga soluvchi koeffitsientlar $(1/9, 1/10, 1/16)$ kiritilgan.

Ko'rib o'tilgan usullar tasvirdagi halaqitlarga qarshi anchagina samarali ishlashi mumkin, lekin ular tasvir chegaralari yoyilishiga (yuvilishiga), ya'ni tasvirning xiralashishiga (ayniqsa M, L oshgan sari) olib keladi.

Bulardan tashqari tasvirlarni tekislashning turli shart-sharoitlari va tasvir xususiyatlarini hisobga oluvchi bir qancha usullar mavjud. Chegaralarni kuchaytirish. O'tkazilgan tajribalar shuni ko'rsatdiki, chegarasi ajralib turgan fotografik va televizion tasvirlar odam tomonidan ranglar bir-biriga sezilarsiz o'tadigan tabiiy manzaradan ko'ra yaxshiroq qabul qilinarkan. Bu xususiyat va tasvirdagi chegaraning yoyilishi ko'rinishidagi xalaqitlarni yo'qotish muammolari tasvirlarga avtomatlashtirilgan ishlov berish oldiga chegarani kuchaytirish, ya'ni fon va obekt yorug'liklari farqini oshirish masalasini qo'ydi. Ushbu masalani yechish usullari tasvirlarga ishlov berishda keng qo'llaniladi.

Odatda chegara yuqori chastotali filtrlar yordamida ko'chaytiriladi:

$$A_1(m, n) = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}; \quad A_2(m, n) = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}; \quad A_3(m, n) =$$

$$\begin{pmatrix} 1 & -2 & 1 \\ 2 & 1 & -2 \\ 1 & -2 & 1 \end{pmatrix}$$

ko'rinib turibdiki bu filtrlarning ish niqoblari o'rtacha nol qiymatga ega bo'ladi, ya'ni niqobdagi manfiy va musbat qiymatlarni umumiy yig'indisi nolga teng (yoki yaqin). Buning sababi, niqob ko'llanganda birinchi maydon uchun nol natija, chegaraviy soha uchun esa noldan farqli natija olinishi kerak (2-darajali hosilaning xususiyati).
 Chegaraviy sohani ko'chaytirishning yana bir usuli bu statistik ayirmalashdir. Unda xar bir element qiymati o'rtacha kvadratik chetlashishning statistik bahosiga bo'linadi: $g_{ij} = f_{ij} / \sigma^2(i,j)$.

$$\sigma^2(i,j) = \frac{\sum_j \sum_l f_{ij} - \bar{f}_i \bar{f}_j}{f_{ij}}, i, j \in N(0, f)$$

Kadr o'lchamini kichiklashtirish

Rastrli ma'lumotga dastlabki ishlov berishning eng birinchi bosqichi odatda kadr o'lchamini kichraytirishdir. Boshlang'ich bosqichdanoq bunday operatsiyaning bajarilishi ikkita muxim afzallikni beradi:

- 1) Keyingi bosqichlarning xar birida qayta ishlash zarur bo'lgan axborot xajmi sezilarli qisqarishi;
- 2) Kiruvchi ma'lumotlarda additiv va impulsli xalaqitlarni susaytirilish.

Boshqacha qilib aytganda, bu bosqichni qiymatlari o'rtachalashlar piksellarga o'tish deb atash mumkin, ya'ni boshlang'ich pikselning rang komponentlari qiymati bu pikselning va qo'shni piksellarning rang komponentlarining o'rtacha olingan qiymatlari bilan almashtiriladi. [] da natijaviy pikselni boshlang'ich tasvirning minimal xududi deb atalgan, chunki keyingi barcha operatsiyalarda ishlatish uchun bunday almashtirishdan so'ng u tasvirning minimal birligi bo'lgan. Minimal xudud rangi L=16 va boshqa shunday piksellar ranglari qiymatining o'rtacha arifmetigi kabi hisoblanadi. Bunday almashtirish natijasida olinadigan rastr minimal xudud kartasi deb ataladi. Bitta minimal xududida L pikselni o'rtalashtirish xolati uchun almashtirish quyidagicha yoziladi:

$$R_{ij} = \frac{r_{2i-1,2j-1} + r_{2i-1,2j} + r_{2i,2j-1} + r_{2i,2j}}{4}$$

$$G_{ij} = \frac{g_{2i-1,2j-1} + g_{2i-1,2j} + g_{2i,2j-1} + g_{2i,2j}}{4}$$

$$B_{ij} = \frac{b_{2i-1,2j-1} + b_{2i-1,2j} + b_{2i,2j-1} + b_{2i,2j}}{4}$$

bu yerda R_{ij} , G_{ij} , B_{ij} (qizil) minimal xududning i-satri va j-ustunidagi piksel rangining qizil, yashil, ko'k komponentlari qiymati, r_{ij} , g_{ij} , b_{ij} - voshlang'ich rastrning i-satri va j-ustunidagi piksel rangining qizil, yashil, ko'k komponentlari qiymati, 4 piksellidan 1 kichik xududiga o'tish qayta ishlashda axborot xajmini 4 barobar qisqartiradi va 16 piksel xolati uchun esa 16 marta qisqartiradi. Bu esa keyinchalik qayta ishlash tezligini sezilarli darajada oshiradi va prosessor yuklamasini kamaytiradi. Bu almashtirish xalaqitlar darajasini sezilarli kamaytirishini ham sezish mumkin. Buda impulsiga bo'lgan ta'siri aditiv xalaqitga ham shunday ta'sir qiladi. Chunki, o'rtalashtirish alohida piksellarning rang qiymatining qo'shni rang qiymatlari bo'yicha ixtiyoriy oshib ketishini, bunday oshib ketishlarining tabiatiga bog'liq bo'lmagan holda 1 tekis silliqilaydi. Shuni yesda tutish kerakki, kichik hudud qartisiga o'tish boshlang'ich kadirida axborot qiymatini yuqotishga olib kelishi mumkin. Shuning uchun ortalashtirish piksellari soni 16dan oshganda tavsiya qilinmaydi. Bundan tashqari boshlang'ich tasvirning mayda detallari xam zarur bo'lganda bu almashtirish qo'llanilmaydi.

Nazorat uchun savol va topshiriqlar

1. Raqamli tasvir nima?
2. Raqamli tasvirlarning qanday formatlari bor?
3. Suratlarini qanday siqish usullarini bilasiz?
4. Suratlarini MBda izlash usullari.
5. Suratlarini saqlash uchun MB qanday strukturasi qo'llaniladi?.

8-BOB.
VIDEO VA AUDIO MA'LUMOTLARNI MA'LUMOTLAR
BAZASIDA SAQLASH

8.1. Video va audio ma'lumotlar shakllari

Video – vaqt bo'yicha tartiblangan o'zaro bog'liq tasvirlarning ketma-ketligi (freymlar yoki kadrlar deyiladi) video ketma-ketliklarning turli televizion standartlari mavjud:

- SECAM (Rossiya, Fransiya)
- PAL (bolshinstvo yevropeyskix stran, Kitay)
- NTSC (SShA, Yaponiya)

Standartlariga ko'ra tasvirlarning o'zgarish tezligi 25-30 freym/sek bo'lishi mumkin. Tasvir videoda lipillashini oldini olish uchun satrlar aro razvertka qilinadi (juft/toq qatorlar).

Video freymlar o'lchovi: 352 x 240, 768 x 576 (PAL), 720 x 576 (CCIR 601), 720 x 480 (NTSC), 1440 x 1152, 1920 x 1080 (HDTV) bo'lishi mumkin. Formati (eni bo'yiga nisbati): 4:3, 16:9 (keng ekranli). Tabiiy ovozlarni multimediyada WAVE (to'lqin) tarzida belgilanadi. Ovozlarni kompyuterda saqlash uchun asosan raqamli formatga o'giriladi.

Raqamli audioformat – raqamli ovoz yozish va uni saqlashda ovozli ma'lumotlarni taqdim etish formati.

Audiofayl – ovozning amplituda va chastotasi haqidagi ma'lumotlardan iborat kompyuter fayli.

Audio ma'lumotlarni MB saqlash, izlash va chiqarishda muhim masalalardan biri ma'lumot hajmi hisoblanadi. 11 kGs diskretlash chastotasida va amplitudaning sakkiz bosqichli yozuvida 1 minut ovoz 66 Kbayt xotira talab qiladi. 44 kGs va 16 bit stereo standartda 16 marta ko'proq xotira talab qiladi. Ya'ni 1 minut WAVE ovozni saqlash uchun 10 Mbayt joy talab qilinadi.

Hozirga kelib ovoz ma'lumotlarini siqadigan audiokodek usullar rivojlanib bormoqda. Dunyo bozorlarida kompressiya/dekompressiya apparat usullarini ishlatadigan ovoz kartalari ko'proq uchramoqda.

Mashhur audiokodeklar:

- MPEG-1 Layer III (MP3) — kompyuter texnikasida va raqamli ovoz eshittirish qurilmalarida foydalaniladigan audio yozuv kodeki.

• Advanced Audio Codec (AAC) — H.264 (AVC) videokodek bilan birga onlayn videoda qo'llaniladigan MP3 ga alternativ bo'lgan kodek.

• Ogg Vorbis (OGG) — kompyuter o'yinlarda va fayl almashish tarmoqlarida musiqani uzatish uchun keng ishlatiladigan erkin kodek

• Adaptive multi rate (AMR) — odam ovozinging mobil telefonlar va boshqa mobil qurilmalardagi yozuvi.

• G.723.1 — IP-telefoniya ilovalari uchun asosiy kodeklardan biri

• Internet Low Bitrate Codec (iLBC) — IP-telefoniya (xususan, Skype va Google Talk) uchun mashhur bo'lgan erkin kodek.

8.2. Ma'lumotlar bazasida video va audio ma'lumotlarni qidirish.

Katta obyektlar (LOB) - bu katta obyektlarni saqlash uchun ishlatiladigan ma'lumotlar turi - har xil matn formatlari, rasmlar, videolar, ovozi fayllar. Ma'lumotlarni saqlash uchun LOB-dan foydalanish dasturdagi ma'lumotlarni samarali boshqarish imkonini beradi.

LOB komponentlari. LOB lokator va qiymatdan iborat. Lokator - bu katta obyektning haqiqiy qiymatiga bo'lgan ichki ko'rsatkich. Qiymat - bu obyektning haqiqiy tarkibi.

LOB lokatori jadvalda va turli segmentlarda ma'lumotlarni saqlaydi, ma'lumotlar hajmi 4000 baytdan kam bo'lgan hollar bundan mustasno. Maksimal LOB hajmi (4 GB (4,294,967,295 bayt) - 1).

LOB turlari. Ham ichki, ham tashqi yirik obyektlardan jadval ustunlari, pl/sql da o'zgaruvchilar, obyekt atributlari sifatida foydalanish mumkin.

Internal LOB

Ichki yirik obyektlar - ma'lumotlar bazasi jadvallarida saqlanadi, quyidagi ma'lumotlar turlari qo'llab-quvvatlanadi: BLOB, CLOB va NLOB. Persistent and Temporary LOB

Ichki yirik obyektlar vaqtinchalik yoki doimiy bo'lishi mumkin. Ma'lumotlar bazasi jadvalining satrlarida doimiy LOBlar mavjud. Vaqtinchalik LOBlar faqat mahalliy dastur doirasida foydalanish uchun yaratilgan.

SQL Datatype	Ta'rif
BLOB	Ikki-kilik katta o'bekti(Binary Large Object) U ma'lumotlarni ikki-kilik formatda saqlaydi, rasmlar, audio va video, shuningdek kompilyatsiya qilingan dastur kodini saqlash uchun ishlatiladi
CLOB	Belgilar katta o'bekti(Character Large Object). Matn ma'lumotlarini DB formatida saqlaydi
NCLOB	Milliy belgilar to'plamida o'bekti (National Character Set Large Object). Matn ma'lumotlarini milliy kodlashda saqlaydi.
BFILE	Tashqi ikki-kilik fayl(External Binary File) Ma'lumotlar bazasidan tashqarida saqlanadigan fayl, operatsion tizim fayli kabi, ma'lumotlar bazasi uchun ruxsatga ega. BFILEs faqat o'qish uchun ruxsatga ega. LOB o'chirilganda, Oracle server o'zi faylni o'chirmaydi. BFILE faqat tasodifiy (ketma-ket bo'lmagan) o'qishni qo'llab-quvvatlaydi, bitimda qatnashmaydi.

8.1-jadval. LOB ma'lumotlar turlari

Vaqtinchalik misol, agar uni jadvallar qatoriga qo'shsangiz, doimiy bo'lib qoladi.

Temporary LOB xususiyatlari

- Ma'lumotlar jadvalda saqlanmaydi, lekin vaqtincha jadvallar makonida saqlanadi

- Vaqtinchalik LOB-lar doimiy LOB-larga qaraganda tezroq, chunki ularni qayta ishlab chiqarish va qaytarib berish ma'lumotlari yaratilmaydi

- Faqat seansda foydalaniladi. Mashg'ulot oxirida o'chiriladi

- Vaqtinchalik LOB yaratish uchun DBMS_LOB.CREATE_TEMPORARY dan foydalaning

External LOB.

Tashqi yirik obyektlar - ma'lumotlar bazasining jadval bo'shliqlaridan tashqarida, operatsion tizim fayllarida saqlanadigan ma'lumotlar turi. Amaldagi ma'lumotlar turi - BFILE. BFILE faqat tashqi bo'lishi mumkin. DBMS_LOB paketidan foydalanish

DBMS_LOB ichki va tashqi LOBlarni boshqarish usullarini taqdim etadi. DBMS_LOB usullarini ikki xil mutators va observers ga bo'lish mumkin:

- mutators - LOB ni o'zgartirishi mumkin: APPEND, COPY, ERASE, TRIM, WRITE, FILECLOSE, FILECLOSEALL, and FILEOPEN.
- observers - LOB o'qish mumkin: COMPARE, FILEGETNAME, INSTR, GETLENGTH, READ, SUBSTR, FILEEXISTS, and FILEISOPEN. LOB lokatorlarini ishga tushirish uchun quyidagi funktsiyalar qo'llaniladi:
 - EMPTY_CLOB() - funktsiya CLOB ustuniga LOB qidiruvchisini qaytaradi
 - EMPTY_BLOB() - funktsiya blob ustuniga LOB lokatorini qaytaradi

CLOB va BLOB ustunlari, agar uzunligi 4000 baytdan oshmasa, belgi yoki satr yordamida foydalanishni boshlash mumkin.

Amaliy qism

Ushbu bo'limda audio ko'rinishini obyekt ko'rinishi bilan qanday ishlatish kerakligi tasvirlangan. Ko'rinish virtual jadval kabi, obyekt ko'rinishi ham virtual obyektlar jadvali.

Oracle obyekt ko'rinishini asosiy relyatsion ko'rinish mexanizmining kengaytmasi sifatida ta'minlaydi. Obyekt ko'rinishini ishlatib, ma'lumotlar bazasida relyatsion yoki obyekt jadvallari ustunlarida saqlanadigan ma'lumotlardan virtual yoki ichki o'rnatilgan foydalanuvchi jadvallarini yaratishingiz mumkin.

Obyekt ko'rinishi ma'lumotlar bazasiga maxsus yoki cheklangan kirishni taklif qilishi mumkin. Masalan, maxfiy ma'lumotlar yoki o'chirish usulini o'z ichiga olgan atributlarga ega bo'lmagan xodim obyekt jadvalining versiyasini taqdim etish uchun siz obyekt ko'rinishini ishlatishingiz mumkin. Obyekt ko'rinishi shuningdek jadvallarni doimiy o'zgartirmasdan obyektga yo'naltirilgan dasturlashni sinab ko'rishga imkon beradi. Obyekt ko'rinishini ishlatib, ma'lumotlarni asta-sekin va shaffof ravishda relyatsion jadvallardan obyektga-alloqador jadvallarga aylantirishingiz mumkin.

Define a Relational Table Containing No ORDAudio Object

```
create table flat (
  id NUMBER,
  description VARCHAR2(4000),
  localData BLOB,
  srcType VARCHAR2(4000),
  srcLocation VARCHAR2(4000),
```

```

srcName VARCHAR2(4000),
upDateTime DATE,
local NUMBER,
format VARCHAR2(31),
mimeType VARCHAR2(4000),
comments CLOB,
encoding VARCHAR2(256),
numberOfChannels NUMBER,
samplingRate NUMBER,
sampleSize NUMBER,
compressionType VARCHAR2(4000),
audioDuration NUMBER,
)

```

-- Store audio data as SecureFiles LOBs.

-- LOB(localData) STORE AS SECUREFILE;

Define an Object View Containing an ORDAudio Object and Relational Columns

```

create or replace view object_audio_v as
select id,
ORDSYS.ORDAudio(T.description,
ORDSYS.ORDSource(
T.localData, T.srcType, T.srcLocation, T.srcName, T.updateTime,
T.local),
T.format,
T.mimeType,
T.comments,
T.encoding,
T.numberOfChannels,
T.samplingRate,
T.sampleSize,
T.compressionType,
T.audioDuration)
from flat T;

```

Obyekt ko'rishlari bir xil aloqador yoki obyekt ma'lumotlarini bir necha usulda qarashning moslashuvchanligini ta'minlaydi. Shuning uchun, ma'lumotlar bazasida ma'lumotlarni saqlash usulini o'zgartirmasdan, turli xil dasturlar uchun turli xil xotiradagi obyektli

tasvirlardan foydalanishingiz mumkin.
Tovushli jadvalni yaratish va to'ldirish uchun Create_soundtable.sql skript ovozli jadval yaratadi va ustuni bilan jadvalda BLOB ustuni mavjud; u jadvalni BLOB to'ldirishni namoyish qilish uchun yaratilgan.

Ushbu skript ovozli jadvalni yaratadi, bo'sh BLOB bilan qator ma'lumotlarining uzunligini tekshiradi. Siz create_soundtable.sql skriptidagi ma'lumotlar faylining nomi bilan almashtirishingiz mumkin. Ushbu skriptni ishga tushirishdan oldin CREATE TABLE imtiyoziga ega ekanligingizga ishonch hosil qiling.

Ovozli jadvalni yaratish uchun create_soundtable.sql skriptini ishlatiladi. Ushbu jadval FAQAT BLOB ustunidan ORDAudio ustuniga ma'lumotlarni qanday nusxalashni ko'rsatish uchun ishlatiladi.

Jadvalga bo'sh BLOB bilan qator qo'shing. BLOB ma'lumotlari bilan qatorni BFILE ma'lumotlar turidan foydalanib, ko'rsatilgan katalogdan yuklanadigan audio faylga ishora qilib yuklang.

Fayllarni yoping va bitimni bajaring. BLOB yuklangan uzunligini tekshiring.

```

SET SERVEROUTPUT ON;
CREATE TABLE soundtable ( id number,
sound BLOB default EMPTY_BLOB() )
Store audio data as SecureFiles LOBs.
LOB(sound) STORE AS SECUREFILE;
INSERT INTO soundtable(id, sound) VALUES (1, EMPTY_BLOB());
COMMIT;
DECLARE
f_lob BFILE := BFILENAME('MEDIA_DIR','aud1.wav');
b_lob BLOB;
length INTEGER;
BEGIN
SELECT sound INTO b_lob FROM soundtable WHERE id=1 FOR
UPDATE;
Open the LOBs.

```

```

dbms_lob.open(f_lob, dbms_lob.file_readonly);
dbms_lob.open(b_lob, dbms_lob.lob_readwrite);
Populate the BLOB from the 'aud1.wav' file in the BFILE.
dbms_lob.loadfromfile (b_lob, f_lob, dbms_lob.getlength(f_lob));
Close the LOBs.
dbms_lob.close(b_lob);
dbms_lob.close(f_lob);
COMMIT;
Check the length of the LOB.
SELECT dbms_lob.getlength(t.sound) INTO length FROM soundtable
t WHERE id = 1;
DBMS_OUTPUT.PUT_LINE('The length is '|| length);
END;

```

Create the audio_table Table
 Create_audtable.sql skripti ikkita ustun va id bilan audio_table jadvalini yaratadi. Ovoz ustuni ORDAudio turi sifatida aniqlangan.
 Ushbu skriptni ishga tushirishdan oldin CREATE TABLE imtiyoziga ega ekanligingizga ishonch hosil qiling.

```

create_audtable.sql
CREATE TABLE audio_table ( id NUMBER,
audio ORDAudio ) LOB(audio.source.localData) STORE AS
SECUREFILE;

```

Nazorat uchun savol va topshiriqlar

1. Raqamli video kodeklar turlari.
2. Blob va Lob operatorlari haqida ma'lumot bering?
3. Video va audio ma'lumotlarni MBda izlashni qanday tashkil qilinadi?

9-BOB. XML VA XML-MA'LUMOTLAR BAZASI

9.1. XML multimedia ma'lumotlar bazasi.

XMLning paydo bo'lgan yilini 1996 yil deb hisoblash ham bo'ladi. Hammasi 1986 yilda paydo bo'lgan SGML tilidan boshlanadi. SGML (Standard Generalized Markup Language — umumlashgan standart belgilash tili) belgilash tillarini xosil qiluvchi meta-til sifatida paydo bo'lgan.

SGML boshqa belgilash tillarini yaratishda keng qo'llaniladi. Aynan shu til yordamida gipertekst hujjatlarni belgilash tili — HTML ni yaratishda ishlatilgan.

XML (eXtensible Markup Language — kengaytirilgan belgilash tili) — Butun dunyo o'rgimchak to'ri Konsorsiumi tomonidan tavsiya etilgan, strukturalangan ma'lumotlarni saqlashda, matn hujjatlarni shakllantiruvchi til. Shuningdek dasturlar aro ma'lumot almashishda ham ishlatilishi mumkin.

XML yaratishdan maqsad strukturalangan ma'lumotlarni har-xil tizimlar orasida internet orqali o'zaro ma'lumot almashishda mutanosiblikni ta'minlash bo'lgan.

XML TYPE turi 9.0 versiyasida Oracle-da paydo bo'ldi. Bunga qadar hujjatlarni XML formatida saqlash uchun CLOB turi eng mos bo'lgan (va VARCHAR2 turi, eng ko'pi 4000 belgidan iborat edi, kamroq mos edi). O'z-o'zidan, yangi XMLTYPE obyekt CLOB yoki obyekt sifatida (9.2 versiyasidan boshlab) texnik jihatdan saqlanishi mumkin. Yana bir eslatma: XML va Java texnologiyalari «qo'lda» bo'lishiga qaramay, quyida keltirilgan misollar doirasi sizning ma'lumotlar bazangizdan Java-ning o'atilgan imkoniyatlarini talab qilmaydi.

Axborot olamida o'zaro XML ko'rinishidagi ma'lumotlar almashinishi soni ortib borishi bilan, saqlanadigan XML-hujjatlar hajmi ham tez sur'atlarda o'sib bormoqda.

Ko'pgina veb-saytlar ma'lumotlarni saqlanadigan joyda XML dan foydalanadilar. saqlanayotgan ma'lumot foydalanuvchilarga ko'rsatilishidan oldin HTML yoki XHTML o'zgartiriladi. Microsoft InfoPath va W3C XForms kabi dasturiy mahsulot va texnologiyalarning yangi avlodlari Microsoft Access yoki SQL Server

larni o'zining ma'lumotlarni saqlash joyi sifatida tanlab, XML ma'lumotlar bilan ishlashni boshlaydilar.

Shuningdek, ruxsatsiz foydalanishdan muxofazalangan konfidensial ma'lumotlarni saqlashda ko'pincha XML dan foydalaniladi.

Ma'lumotlarni XML ko'mishida saqlashning o'zi kifoya qilmaydi, shuningdek ma'lumotlar xavfsizligi va kengayishini ham tadbirini qilib borish lozim.

XML ma'lumotlarni saqlash joyi

XML ko'rinishidagi ma'lumotlarni saqlash haqida gap ketganda, ish samarasi muhim omillardan biri hisoblanadi.

Agar XML hujjat oddiy matnli hujjat sifatida saqlanadigan bo'lsa, qanday qilib unga samarali ishlov berishni ta'minlash mumkin?

Saqlanayotgan ma'lumotlar hajmi gigabaytlar bilan o'lchanganda, xotirada butunlay joylashadigan hujjatning obyekt modelini (DOM) hosil qilish samarasiz hisoblanib, o'zgacha yondashuvlarni izlash kerak bo'ladi.

Saqlanayotgan XML ma'lumotlar hajmini o'sib borishi bilan, katta hajli ma'lumotlar orasidan kerakligini izlab topish samarasini oshirish yanada muhimroq ahamiyat kasb etayapti.

XML ma'lumotlarni saqlashda qanday usul qo'llanishidan qat'iy nazar, ma'lumotni izlab topish tezligini oshirish uchun, ularni indekslash zarurligi yaqqol ma'lum.

Saqlanayotgan ma'lumotlar orasidan biror-bir to'plamni ajratib olish amaliyoti ko'p qo'llaniladi.

Internet tarmog'ida ishlayotganda, ma'lumot almashish tezligi muhim ahamiyatga ega. XML ko'rinishdagi ma'lumotlar bazasidan foydalanilganda, bunday ma'lumotlar bazasi kerakli tezlikni ta'minlab bera oladimi?

Bundan tashqari XML ma'lumotlar bazasi yetarlicha ishonchli ishlashi talab qilinadi.

XML va relyatsion ma'lumotlar

XML dan zamonaviy ma'lumotlar bazasida foydalanish masalasiga o'tishdan avval, relyatsion va XML ma'lumotlar rukturasini qisqacha solishtirib chiqamiz.

Relyatsion ma'lumotlar bazasida axborot ustun va satrlardan iborat jadvallarda saqlanadi. Ustunlarda jadvalning barcha yozuvlari bo'yicha bir xil ko'rinishdagi ma'lumotlar saqlanadi. Jadvalning har bir yozuvi satr ko'rinishdagi ma'lumotlar saqlanadi. Jadvalning har bir bog'liq emas. Aksincha, XML da hujjatning joriy ichki tartibi bilan bog'liq emas. Aksincha, XML da hujjatning joriy ichki tartibi XPath ga o'xshash funksiyalar qaytargan position() ma'lumotlarga o'z ta'sirini o'tkazadi.

Faqat eng sodda relyatsion ma'lumotlarga bitta jadvalda saqlanishi mumkin. Odatda relyatsion ma'lumotlar bazasi murakkab mantiqiy bog'lanishlar bilan o'zaro bog'langan jadvallardan iborat bo'ladi.

Har xil jadvallardagi ma'lumotlar o'zaro kalit maydonlar bilan bog'lanadi.

Odatda bunday kalit maydon sifatida yozuvning ID nomeri maydoni qatnashishi mumkin.

Ma'lumotlar orasidagi bog'lanish "birga-bir" (masalan, bitta farzandning bitta otasi bo'ladi), "birga-ko'p" (masalan, bitta otaning bir necha farzandi bo'lishi mumkin), "ko'pga-ko'p" (masalan, bitta tovar bir necha buyurtmalarda bo'lishi va bitta buyurtmada har xil tovar bo'lishi mumkin) tarzida bo'lishi mumkin.

Har qanaqa bunday bog'lanishlar ma'lumotni bir-nechta o'zaro bog'liq jadvallarda bo'lib saqlanishidan kelib chiqadi.

XPath, DOM yoki XML Infoset ma'lumotlar modelidan ko'rinib turganidak, ichki ierarxiyaga ega bo'lgan XML-hujjatlardan farqli o'laroq, relyatsion ma'lumotlar bazasi ierarxiyaga ega emas.

XML ma'lumotlar quyidagi misoldan ko'rinib turganidek ichki tartibga ega:

```
<Orders>
<Order Customer="Acme Industries" Date="2003-12-11"
Value="1234.56" Currency="US Dollars" />
<Order Customer="Fiction Fabricators" Date="2004-02-11"
Value="4300.12" Currency="US Dollars" />
<Order Customer="Aspiring Assemblers" Date="2005-07-11"
Value="10000.00" Currency="US Dollars" />
</Orders>
```

XMLning ichki ierarxiyasi XML-hujjatni to'g'ri tuzilganligini aniqlaydigan sharti hisoblanadi.

Shu kabi oddiy ma'lumotlarni relyatsion ma'lumotlar bazasida saqlash ularning tartibini yo'qolishiga olib keladi.

Indekslar

Hujjatlarni saqlaydigan joyi uchun o'rinli bo'lgan yana bir masala – bu so'rovlardir. Saqlanayotgan barcha ma'lumotlar ichidan faqat ma'lum bir qismini ajratib olish so'rovlar shaklida amalga oshiriladi. So'rovni amalga oshirish uchun, ma'lumot so'ralayotgan atributi bo'yicha indekslangan bo'lishi kerak.

Masalan, agar hujjatlarning katta kolleksiyasi ichidan, ma'lum avtor tomonidan yozilgan barcha hujjatlarni tezda ajratib olish zarur bo'lsa, buning uchun ma'lumotlarning avtor atributi bo'yicha indekslash kerak bo'ladi.

9.2. XML so'rovlari va qidiruvi

Relyatsion ma'lumotlar bazasi – ma'lumotlarni saqlashning mashxur usullaridan biri. Bunday usul uzoq vaqtdan beri strukturalangan ma'lumotlarni saqlashda izchil ishlatilib, bu vaqt mobaynida ancha mukammallashtirilgan. Ko'p sonli dasturchilar ulkan hajmdagi ma'lumotlarni saqlashda aynan shu ma'lumotlar bazasidan foydalanishi ham bejiz emas. Shu sabablarga ko'ra XML va relyatsion ma'lumotlar bazasini boshqarish tizimini birgalikda ishlatish mumkin bo'ladi.

Ushbu bo'limda Oracle-da XML-dan foydalanishning mantiqiy jihatlari ko'rib o'tamiz.

Oddiy misol

```
CREATE TABLE books
(id NUMBER PRIMARY KEY,
description XMLTYPE);
INSERT INTO books VALUES (100,
XMLTYPE('<cover><title>Oracle
SQL*Loader</title><author>Jonathan
Gennick</author><author>Sanjay
hra</author><pages>269</pages></cover>'));
SET long 1000 SELECT id,
```

```
description FROM books; SELECT id,
b.description XMLDATA FROM books b;
XMLDATA - XMLTYPE uchun maxsus yaratilgan "psevdo
ustun".
```

XMLTYPE - bu XML turi.XMLTYPE ma'lumotlar bazasiga kiritilgan matn shunchaki satr emas, balki XML hujjat satri ekanligini aytishga imkon beradi.

Birinchi holda, XML obyektini konstruktor yordamida yaratiladi, ikkinchisida NEW operatori ham ishlatiladi.Ikkinchisi Oracle-da obyekt bilan ishlashda ishlatiladi, ammo SQL-da mazmunli biron bir narsa bermagani uchun uni ishlatish faqat tavsifiy hisoblanadi.

Misol:

```
SELECT * FROM xbooks;
SELECT VALUE(x) FROM xbooks;
SELECT XMLDATA FROM xbooks;
```

Boshqa turdagi obyektlar jadvallarda bo'lgani kabi, XML obyektlari jadvalning elementlari havolalarga ega, ya'ni ular sizga boshqa turdagi va jadvallarda REF orqali murojaat qilishingizga imkon beradi:

```
SELECT REF(x) FROM xbooks x;
SELECT Deref(REF(x)) FROM xbooks x;
Ushbu turdagi xususiyatlar yo'q, ammo usullar mavjud. Siz buni SYS nomidan so'rov yuborib tasdiqlashingiz mumkin:
COLUMN text FORMAT A80
SELECT text
FROM user_source
WHERE name = 'XMLTYPE' AND type = 'TYPE'
ORDER BY line;
```

Shaharlar bilan jadval yaratish:

```
create table CITY (
id NUMBER not null, -- Shahar identifikatori
name NVARCHAR2(100) not null -- Наименованиегорода )
tablespace USERS
pctfree 10
intrans 1
maxtrans 255
```



```

storage
(
initial 64K
next 1M
minextents 1
maxextents unlimited );

```

Va ko'chalar bilan jadval yarating:
create table STREET

```

(
id NUMBER not null, -- Ko'chalar identifikatori
id_city NUMBER not null, -- Идентификатор города
name NVARCHAR2(100) not null -- Наименование улицы
)
tablespace USERS
pctfree 10
initrans 1
maxtrans 255
storage (
initial 64K
next 1M
minextents 1
maxextents unlimited );

```

Jadvallarimizni namunaviy ma'lumotlar bilan to'ldiring:

Shaharlarni qo'shing

```

insert into CITY (ID, NAME) values (1, 'Ташкент');
insert into CITY (ID, NAME) values (2, 'Бухара');
insert into CITY (ID, NAME) values (3, 'Самарканд');
commit;

```

Ko'chalarni qo'shing

```

insert into STREET (ID,
ID_CITY, NAME) values (1, 1, 'Мира');
insert into STREET (ID, ID_CITY, NAME) values (2, 1,
'Навои');

```

```

insert into STREET (ID, ID_CITY, NAME) values (3, 2,
'Амира Тимура');
insert into STREET (ID, ID_CITY, NAME) values (4, 2,
'Домоносова');
insert into STREET (ID, ID_CITY, NAME) values (5, 3,
'Чернышевского');
insert into STREET (ID, ID_CITY, NAME) values (6, 3,
'Карамзина');

```

Endi hamma narsa tayyor, davom eting.
1. XMLElement — XML tugunni tanlash
Oddiy shaklda, XML namunalari quyidagicha ko'rinadi:

```

select XMLElement("city", 'Ташкент') from dual;
Natija: <city>Moscow</city>

```

Endi biz CITY jadvalimiz bilan ham shunday qilamiz - barcha shaharlarni tanlang.

```

select XMLELEMENT("city",c.name) from city c;

```

Natija:
<city>Ташкент</city>
<city>Бухара</city>
<city>Самарканд</city>

Natijada biz bir qator elementlarga ega bo'ldik, endi biz ushbu elementlarning barchasini bitta ota tuguniga birlashtira olamiz.

2. XMLAGG — Ota tugunidagi elementlarning birlashishi (guruhlanishi).
Barcha shaharlarimizni avvalgi misoldan bitta ota-tugun country ga birlashtiring.

```

select
XMLELEMENT("country",XMLAGG(XMLELEMENT("city",c.name)
)) from city c;

```

Natija:
<country>
<city>Ташкент</city>
<city>Бухара</city>
<city>Самарканд</city>
</country>

Natijada biz shahar elementlarining ichki to'plamiga ega bo'lgan mamlakat elementiga ega bo'ldik.

Endi mamlakat nomini qo'shing va atributlarni qanday qo'shishni bilib oling.

3. XMLATTRIBUTES — Atributlarni XML elementiga qo'shish. Oddiydan boshlaylik, mamlakat nomini qo'shamiz. Buning uchun, tugun nomidan so'ng, qavs ichida ko'rsatilgan atribut parametrlari bilan boshqa XMLATTRIBUTES parametrini qo'shing.

```
select XMLELEMENT("country",xmlattributes('Ўзбекистан' as "name"), XMLAGG(XMLELEMENT("city",c.name))) from city c;
```

Natija:

```
<country name="Ўзбекистан">  
<city>Ташкент</city>  
<city>Бухара</city>  
<city>Самарканд</city>  
</country>
```

Nazorat uchun savol va topshiriqlar

1. XMLni ta'riflab bering.
2. Ma'lumotlarni saqlashda XMLning roli qanday?
3. XML hujjat ma'lumotlari qanday strukturada bo'ladi?
4. XML va relyatsion MB birgalikda qay tarzda ishlatiladi?

10-BOB. KATTA HAJMLI OBYEKT LARNI TASHKIL QILISH VA BOSHQARISH

Zamonaviy axborot tizimlarida foydalaniladigan axborot (ayniqsa video, audio multimediali ma'lumotlar) hajmini jadal o'sishi oldimizga ushbu axborotni ma'lumotlar bazasida saqlash va ularni boshqarishning yangi murakkab masalalarni qo'yadi. Quyida katta hajmdagi multimedia ma'lumotlarni tashkil qilishga, shuningdek ular bilan ishlash vositalari tahlil qilib chiqamiz.

10.1. Katta hajmli multimedia obyektlar ularni boshqarish usul va vositalari

Katta hajmli multimedia ma'lumotlar to'rtta asosiy xarakteristikaga ega bo'ladi: hajm, xilma-xillik, tezlik, narhi.

1. **Hajm.** Insonlar va mashinalar tomonidan xosil qilinayotgan ma'lumotlarning o'sib borayotgan soni, axborot texnologiyalari infrastrukturalariga ma'lumotlarni saqlash, ishlov berish va taqdim etishida yangi talablarni qo'yadi.

2. **Xilma-xillik.** Turli strukturalarda taqdim etilgan ma'lumotlar xilma-xil bo'ladi. Bular kredit kartalar bo'yicha operatsiyalar bo'ladi, ilmiy tadqiqot natijalari bo'ladi, fotografik suratlar bo'ladi, video va audio ma'lumotlar bo'ladi barchasi o'ziga xos ishlov va saqlash shart sharoitlarini talab etadi.

3. **Tezlik.** Tezlik deganda nafaqat ma'lumotlarning ma'lumotlar bazasiga kelib tushish tezligi, balki bu ma'lumotlar bazasidan olinadigan ma'lumotlarni chiqarib olish tezligi ham anglanadi.

4. **Narhi.** Katta hajmdagi ma'lumotlar — qimmat resurs hisoblanadi. Ma'lumot dolzarbligi, foydaliligi va ichidagi kontentiga qarab uning qanchalik qimmat baholigi aniqlanadi.

Katta hajmli ma'lumotlarni saqlash

Yangi axborot texnologiyalari, qurilmalari va kommunikatsiya vositalarining paydo bo'lishi bilan, insonlar tomonidan ishlab chiqarilayotgan ma'lumotlar soni ham geometrik progressiya bo'yicha

ortib bormoqda. Shu bilan birga ma'lumotlarning foydaliligi koeffitsienti juda past bo'lishi mumkin. Aksariyati befoyda, bekorchi ma'lumotlardan iborat. Shundan kelib chiqadiki, barcha ishlab chiqarilgan ma'lumotlardan oldin izchil ishlov berilgandan keyingina foydalanish mumkin bo'ladi.

Katta hajmdagi ma'lumotlar an'anaviy kompyuter tizimlari tomonidan amalga oshirib bo'lmaydigan ishlov berishni talab qiladi. Katta hajmdagi ma'lumotlar turli tizim va dasturlar tomonidan yaratilgan axborotni o'z ichiga oladi:

- **Qora quti:** vertolyot, samolyotlar, dengiz va kosmik kemalarning axborot ta'minoti qismiga kiradi. Uning vazifasiga boshqaruv obyekti xarakteristikalarini to'g'risidagi axborotni, ekipaj ovozi yozib borish kiradi.

- **Ijtimoiy multimedia ma'lumotlar:** ijtimoiy tarmoqlar orqali tarqatilgan axborot.

- **Fond birjalari:** kompaniyalar orasidagi oldi-sotdi muomalalar to'g'risidagi ma'lumotni saqlash.

- **Energotizimlar:** bunday ma'lumotlar energetik tarmoqning uzellari va kuchlanishlarini o'z ichiga oladi.

- **Transport tizimi:** modellar, xarakteristikalar, masofalar, GPS media ma'lumotlar – transport va yo'l tarmog'i haqidagi barcha ma'lumotlarni qamraydi.

- **Qidiruv tizimlari:** turli ma'lumotlar bazasidan izlash ishlari. Natijada, katta hajmdagi ma'lumotlar katta hajmga, katta tezlikka ega bo'lgan uch xil tipga ajratiladi:

- Strukturalangan ma'lumotlar – relyatsion MB
- Yarim strukturalangan ma'lumotlar - XML-fayllar
- Strukturalanmagan ma'lumotlar – tasvir, video, audio, Word, PDF, Text formatdagi fayllar.

10.2. Katta multimedia obyektlarini boshqarish usullari va vositalari.

BLOB, CLOB va NCLOB ma'lumotlar turlari ma'lumotlar bazasida saqlanadigan ichki LOBlarni, BFILE esa tashqi ko'rinishni ifodalaydi. BFILE obyektlari va ichki LOB o'rtasida uchta muhim farq bor:

- BFILE qiymati ma'lumotlar bazasida emas, balki operatsion tizim faylida saqlanadi.
- BFILE obyektlari tranzaksiyalarda qatnashmaydi (ya'ni ularga bekor qilish yoki saqlash mumkin emas).

Biroq, tranzaksiyalarda BFILE lokatoriga kiritilgan o'zgartirishlarni bekor qilish va saqlash mumkin.

- PL / SQL va Oracle ikkalasida ham BFILE obyektlarini o'qish mumkin.

BFILE turi orqali ma'lumotlarni yozish qo'llabquvvatlanmaydi. BFILE lokatorlari tomonidan ko'rsatilgan tashqi fayllar Oracle DBMS tashqarisida yaratilishi kerak.

PL / SQL-da BFILE kabi ma'lumotlar bilan ishlashda, ular oddiygina serverda saqlangan faylga ishora qiluvchi LOB lokatori bilan ishlaydi. Shuning uchun, BFILE turini o'z ichiga olgan ma'lumotlar bazasi jadvalining turli satrlarida bir xil faylga ishora qiluvchi qiymatlarni saqlash mumkin.

BFILE joylashtiruvchisi katalog taxalluslari va fayl nomlaridan iborat. Ushbu ikkita parametrdan foydalaniladi. Katalog olish uchun BFILENAME funksiyasidan foydalaniladi. Katalog taxallusi - bu Oracle-da aniqlangan operatsion tizim katalogiga mos keladigan nom. Taxalluslar PL / SQL dasturlariga operatsion tizimdan qat'iy nazar kataloglar bilan ishlashga imkon beradi. CREATE ANY DIRECTORY imtiyozlari bilan siz katalog taxallusini yaratishingiz mumkin (katalog o'zi fayl tizimida allaqachon mavjud bo'lishi kerak) va unga quyidagicha kirish huquqini berishingiz mumkin:

```
CREATE DIRECTORY bfile_data AS 'c:\PLSQL
Book\Ch13_Misc_Datatypes\
GRANT READ ON DIRECTORY bfile_data TO gennick;
```

Biroq, katalog taxalluslarini yaratish va ularga kirishni boshqarish bu PL / SQL bilan bevosita bog'liq bo'lmagan ma'lumotlar bazasi ma'muri funksiyalari, shuning uchun men ularni batafsil ko'rib chiqmayman. Agar sizga qo'shimcha ma'lumot kerak bo'lsa, ma'lumotlar bazasi ma'muri bilan gaplashing yoki CREATE DIRECTORY buyrug'idagi Oracle SQL Reference Manual qismini ko'ring. Kirish huquqiga ega bo'lgan kataloglar ro'yxati uchun ALL_DIRECTORIES ko'rinishini so'rang.

BFILE lokatorini yaratish.

BFILE lokatorini yaratish uchun siz BFILENAME funksiyasini chaqirib, unga katalog taxalluslari va fayl nomini berishingiz kerak. Quyidagi misol HTML-fayl uchun BFILE lokatorini yaratadi, u keyinchalik sharshalar jadvalida saqlanadi.

```
DECLARE  
web_page BFILE;  
BEGIN
```

Bu misol bir necha bor ishlay olishi uchun Tannery Falls liniyasini o'chirish.

```
DELETE FROM waterfalls WHERE falls_name='Tannery Falls';
```

Joylashtirishni yaratish uchun BFILENAME-ga qo'ng'iroq qiling. Lokatori waterfalls jadvalida saqlash.

```
INSERT INTO waterfalls (falls_name, falls_web_page)  
VALUES ('Tannery Falls', web_page);  
END;
```

BFILE joylashtiruvchisi - bu katalog taxalluslari va fayl nomlarining kombinatsiyasi. Haqiqiy fayl va katalog mavjud bo'lishi shart emas. Boshqacha qilib aytganda, Oracle mavjud bo'lmagan kataloglar uchun taxalluslar yaratishga imkon beradi, BFILENAME esa mavjud bo'lmagan fayllar uchun lokomotorlarga ruxsat beradi. Ba'zan bu qulay.

Siz BFILENAME qo'ng'iroqlaridagi katalog nomi ALL_DIRECTORIES ma'lumotlar lug'ati ko'rinishida ko'rsatiladigan holatga mos bo'lishi kerak. Dastlab men bfile_data yozuvini o'z misolimda kichik harfda ishlatganman - va tashqi BFILE ma'lumotlar fayliga kirishda ko'p xatolarga duch kelganman. Ko'p hollarda, BFILENAME-ni chaqirganda katalog nomi katta harf bilan yozilishi kerak.

BFILE ma'lumotlariga kirish.

BFILE lokatorini olganingizdan so'ng, siz tashqi fayllardan ma'lumotlarni BLOB obyektidagi kabi o'qishingiz mumkin. Quyidagi ol veb-sahifadan HTML-ning birinchi 60 baytini oladi. RAW tagi olingan ma'lumotlar UTL_RAW.CAST_TO_VARCHAR2 tsiyasi orqali belgilar qatoriga o'zgartiriladi.

```
DECLARE  
web_page BFILE;  
html RAW(60);  
amount BINARY_INTEGER := 60;  
offset INTEGER := 1;  
BEGIN
```

```
LOB web_sahifasi qidiruvchisini olish  
SELECT falls_web_page  
INTO web_page  
FROM waterfalls  
WHERE falls_name='Tannery Falls';  
Lokatori oching, 60 baytni o'qing va joylashuvni yoping  
DBMS_LOB.OPEN(web_page);  
DBMS_LOB.READ(web_page, amount, offset, html);  
DBMS_LOB.CLOSE(web_page);  
Natija o'n oltilikda ko'rsatilishi uchun keyingi qatordan sharh  
belgilarini olib tashlang  
DBMS_OUTPUT.PUT_LINE(RAWTOHEX(html));  
RAW ma'lumotlarini o'qiladigan belgilar qatoriga  
o'zgartiring DBMS_OUTPUT.PUT_LINE(UTL_RAW.CAST_TO_V  
ARCHAR2(html));  
END;
```

Sessiyada bir vaqtning o'zida ochiladigan BFILE obyektlarining maksimal soni SESSION_MAX_OPEN_FILES ma'lumotlar bazasi parametri bilan o'rtatiladi. Ushbu parametr bir vaqtning o'zida bir seansa ochilishi mumkin bo'lgan fayllar sonining yuqori chegarasini belgilaydi (nafaqat BFILE, balki UTL_FILE to'plamidagi dasturlardan foydalanib ochilgan fayllar ham).

Eslatib o'tamiz, Oracle faqat BFILE ma'lumotlarini o'qishni qo'llab-quvvatlaydi. BFILE turi DBMS (grafika va boshqalar) dan tashqarida yaratilgan ikkilik ma'lumotlar bilan ishlash uchun juda mos keladi. Masalan, raqamli kamera tomonidan olingan rasmlarni serverga yuklashingiz, ushbu fayllar uchun BFILE lokatorlarini yaratishingiz va ular bilan PL / SQL dasturlarida ishlashingiz mumkin.

LOB ustunlarini yuklash uchun BFILE-dan foydalanish.

BFILE turi nafaqat Oracle ma'lumotlar bazasidan tashqarida yaratilgan ikkilik fayllarga kirishni ta'minlaydi, balki tashqi ma'lumotlarni LOB ustunlariga yuklash uchun ham qulay vositadir. Oracle9i Release 1-da, DBMS_LOB.LOADFROMFILE funksiyasi BFILE obyektidan ikkilik ma'lumotni o'qish va uni BLOB ustuniga yozish uchun ishlatilgan, Oracle9i Release 2 ancha qulay vositalarni taqdim etgan:

- DBMS_LOB.LOADCLOBFROMFILE funksiyasi ma'lumotlarni BFILE obyektidan CLOB ustuniga kerakli belgi o'zgartirish bilan yuklaydi.
- DBMS_LOB.LOADBLOBFROMFILE funksiyasi BFILE obyektidan ma'lumotlarni BLOB ustuniga yuklaydi. U DBMS_LOB.LOADFROMFILE bilan bir xil, ammo uning interfeysi DBMS_LOB.LOADCLOBFROMFILE funksiyasining interfeysiga yaqin.

Tannery Falls sharsharasi BFILE DATA taxalluslari havolasida joylashgan katalogdagi TanneryFalls.directions nomli tashqi matn faylida joylashganligini tasavvur qiling. Quyidagi misol sharsharalar jadvalining CLOB ustuniga ma'lumotni yuklash uchun DBMS_LOB.LOADCLOBFROMFILE dan foydalanishni namoyish etadi:

```
DECLARE
  Tannery_Falls_Directions BFILE
:=BFILENAME('BFILE_DATA', 'TanneryFalls.directions');
directions CLOB;
destination_offset INTEGER := 1;
source_offset INTEGER := 1;
language_context INTEGER := DBMS_LOB.default_lang_ctx;
warning_message INTEGER;
BEGIN
  Ushbu misol bir necha bor ishlashi uchun 'Tannery Falls' satrini
olib tashlash.
DELETE FROM waterfalls WHERE falls_name='Tannery
alls';
LOB qidiruvi qo'ng'irog'i bilan yangi qatorni
ylashtiring EMPTY_CLOB()
```

```
INSERT INTO waterfalls
(falls_name, falls_directions)
VALUES ('Tannery Falls', EMPTY_CLOB());
Oldingi INSERT buyrug'i tomonidan yaratilgan LOB
qidiruvchisini o'qish
SELECT falls_directions
INTO directions
FROM waterfalls
WHERE falls_name='Tannery Falls';
CLOB obyektini va asl BFILE obyektini ochish
DBMS_LOB.OPEN(directions,
DBMS_LOB.LOB_READWRITE);
DBMS_LOB.OPEN(Tannery_Falls_Directions);
BFILE tarkibini CLOB ustuniga yuklash
DBMS_LOB.LOADCLOBFROMFILE
(directions, Tannery_Falls_Directions,
DBMS_LOB.LOBMAXSIZE,
destination_offset, source_offset,
NLS_CHARSET_ID('US7ASCII'),
language_context, warning_message);
Faqatgina mumkin bo'lgan ogohlantirishni tekshirish
IF warning_message =
DBMS_LOB.WARN_INCONVERTIBLE_CHAR THEN
DBMS_OUTPUT.PUT_LINE ('Warning! Some characters
couldn''t be converted.');
```

Kodning ushbu qismidagi asosiy ish DBMS_LOB.LOADCLOBFROMFILEga qo'ng'iroq qilish orqali amalga oshiriladi. Ushbu protsedura tashqi fayldan ma'lumotlarni o'qiydi, barcha kerakli belgilar to'plamini o'zgartiradi va CLOB ustuniga ma'lumotlarni yozadi. Men yuklangan ma'lumotlar miqdorini aniqlash uchun DBMS_LOB.LOBMAXSIZE doimiy qiymatidan foydalanaman. Menga tashqi fayldagi barcha ma'lumotlar kerak va

DBMS_LOB.LOBMAXSIZE CLOB-da saqlanishi mumkin bo'lgan maksimal miqdorni aniqlaydi.

Siqish

Siqish rejimida SecureFiles ma'lumotlar bazasi LOB ma'lumotlarini ham diskda, ham xotirada siqadi. Qo'llabquvvatlanadigan rejimlar MEDIUM (o'rtacha siqish, odatdagidek ishlatiladi) va HIGH (yuqori siqish). Yuqori siqishni ko'proq vaqt talab etadi, ammo LOB ma'lumotlarining hajmini kamaytiradi. PDF fayllar bilan oddiy sinovlar shuni ko'rsatdiki, HIGH rejimida yuklash ma'lumotlari MEDIUM rejimiga qaraganda taxminan 25% ko'proq vaqtni oladi. Siqish bilan bir vaqtda takroriy o'chirishni yoqish uchun LOB ta'rifida tegishli parametrlarni ko'rsating:

```
TABLE waterfalls
(
falls_name VARCHAR2 (80),
falls_photo BLOB,
falls_directions CLOB,
falls_description NCLOB,
falls_web_page BFILE )
LOB (falls_photo)
STORE AS SECUREFILE (COMPRESS DEDUPLICATE)
LOB (falls_directions) STORE AS SECUREFILE (COMPRESS
DEDUPLICATE)
LOB (falls_description) STORE AS SECUREFILE
(COMPRESS DEDUPLICATE)
```

Nazorat uchun savol va topshiriqlar

1. Katta hajmli obyektlarga ta'rif bering.
2. MapReduce nima?
3. NoSQL texnologiyasini tushuntirib bering.
4. Katta hajmli obyektlarni boshqarish usul va vositalarini aytib bering.
5. Hadoop texnologiyasini tushuntirib bering
6. R dasturlash tili asosiy vazifalarini aytib bering.

FOYDALANILGAN ADABIYOTLAR

1. Mirziyoyev Sh.M. "Erkin va farovon, demokratik O'zbekiston davlatini birgalikda barpo etamiz" - Toshkent "O'zbekiston" 2016 56 b.
2. 5. Tom White. Hadoop: The Definitive Guide., 3th Edition. O'REILLY, 2013, — p.672.
3. 6. Токмаков Г.П. Базы данных. Концепция баз данных, реляционная модель данных, языки SQL и XML : учебное пособие / Г. П. Токмаков. – Ульяновск : УлГТУ, 2010. – 192 с.
4. 7. Фаулер Мартин, Салаладж Прамодекумар Дж. NoSQL: новая методология разработки нереляционных баз данных. : Пер. с англ. — М.: ООО «И.Д.Вильямс», 2013. – 192с.
5. Codd E. F., "A Relational Model of Data for Large Shared Data Banks", Communications of the ACM, vol. 13, issue 6, pp. 377–387, June 1970.
6. (For Java Programmers) "JDBC Basics", Java Online Tutorial <http://download.oracle.com/javase/tutorial/jdbc/basics/index.html>.
7. Paul DuBois, "MySQL Developer's Library", 4th ed, 2009 (5th ed is probably available).
8. Russell Dyer, "MySQL in a Nutshell", 2nd ed, O'reilly, 2008.
9. MySQL Mother Site @ www.mysql.com.
10. MySQL 5.7 "Reference Manual" @ <http://dev.mysql.com/doc/>.
11. MySQL 5.7 "SQL Statement Syntax" <http://dev.mysql.com/doc/refman/5.7/en/sql-syntax.html>.
12. "ISO/IEC 9075 Information Technology - Database Languages SQL": 1989 (SQL-89), 1992 (SQL-92 or SQL2), 1999 (SQL-99 or SQL3), 2003 (SQL-2003), 2006 (SQL-2006) and 2011 (SQL-2011).
13. MySQL Employees Sample Database <http://dev.mysql.com/doc/employee/en/index.html>.
14. MySQL Sakila Sample Database <http://dev.mysql.com/doc/sakila/en/index.html>.
15. Microsoft Northwind Sample database @ <http://www.microsoft.com/en-us/download/details.aspx?id=23654>; MySQL port <http://code.google.com/p/northwindextended>.
16. The "Classic Models" Retailer database <http://www.mysqltutorial.org>.

MUNDARIJA	
KIRISH.....	3
1-BOB. MULTIMEDIA MA'LUMOTLAR BAZASI FANIGA KIRISH.....	4
1.1. Ma'lumotlar bazasi asoslari. Multimedia MBBT.....	4
1.2. Multimediali ma'lumotlar turlari.....	10
Nazorat savollari.....	16
2-BOB. MULTIMEDIA MA'LUMOTLAR MODEL TUSHUNCHASI.....	17
2.1. Ierarxik, taqsimlangan va relyatsion ma'lumotlar modeli.....	17
2.2. Taqsimlangan ma'lumotlar bazasi.....	18
Nazorat savollari.....	21
3-BOB. MULTIMEDIA MB LOYIHALASHDA RELYATSION YONDOSHUV.....	22
3.1. MMB konseptual va mantiqiy loyahasiga relyatsion yondoshuv.....	22
3.2. Relyatsion algebraga.....	25
3.3. Bog'lanish turlari va xususiyatlari: Atributlar va kalitlar.....	29
Nazorat savollari.....	35
4-BOB. MULTIMEDIA MB SO'ROVLARI.....	36
4.1. SQL tili haqida tushuncha.....	36
4.2. SQL tilida ma'lumotlar turlari.....	37
4.3. Jadvallar yaratish va ularga cheklovlar kiritish.....	38
4.4. Maydon qiymatlarini tekshirish (CHECK cheklovi).....	40
4.5. Oddiy SELECT so'rovlar va mantiqiy operatorlar.....	43
4.6. SQL turlari va strukturasi.....	48
Nazorat savollari.....	67
5-BOB. MULTIMEDIALI MA'LUMOTLAR BAZASINI TASHKIL QILISH.....	68
5.1. Oracle ma'lumotlar bazasini boshqarish tizimi.....	68
5.2. Oracle ma'lumotlar bazasiga ulanish.....	75
5.3. Indekslash va ma'lumot turlari.....	78
Nazorat uchun savol va topshiriqlar.....	80
6-BOB. MULTIMEDIALI MA'LUMOTLARNI IZLASH.....	81
6.1. Ma'lumotlarni izlash.....	81
6.2. Izlashning ikkilik daraxtlari.....	84
6.3. Oracle da ma'lumotlarni Order By orqali tartiblash.....	90
6.4. Ma'lumotlarni filtrlash.....	100
Nazorat uchun savol va topshiriqlar.....	112

7-BOB. SURATLARNI MA'LUMOTLAR BAZASIDA SAQLASH.....	113
7.1. Tasvirli ma'lumotlar shakllari.....	113
7.2. Ma'lumotlar bazasida tasvirlarni saqlash.....	117
7.3. Suratlar MBda izlash.....	120
7.4. Suratlar uchun ma'lumotlar bazasi strukturasi.....	121
Nazorat uchun savol va topshiriqlar.....	131
8-BOB. VIDEO VA AUDIO MA'LUMOTLARNI MA'LUMOTLAR BAZASIDA SAQLASH.....	132
8.1. Video va audio ma'lumotlar shakllari.....	132
8.2. Ma'lumotlar bazasida video va audio ma'lumotlarni qidirish.....	133
Nazorat uchun savol va topshiriqlar.....	138
9-BOB. XML VA XML-MA'LUMOTLAR BAZASI.....	139
9.1. XML multimedia ma'lumotlar bazasi.....	139
9.2. XML so'rovlari va qidiruvi.....	142
Nazorat uchun savol va topshiriqlar.....	146
10-BOB. KATTA HAJMLI OBYEKTLARNI TASHKIL QILISH VA BOSHQARISH.....	147
10.1. Katta hajmli multimedia obyektlar ularni boshqarish usul va vositalari.....	147
10.2. Katta multimedia obyektlarini boshqarish usullari va vositalari.....	148
Nazorat uchun savol va topshiriqlar.....	154
FOYDALANILGAN ADABIYOTLAR.....	155

E.Sh. Nazirova, Sh.B. Abidova, M.M. Mahmudova

MULTIMEDIALI MA'LUMOTLAR BAZASI

Muhammad Al-Xorazmiy nomidagi Toshkent axborot
texnologiyalari universiteti tomonidan o'quv qo'llanma sifatida
tavsiya etilgan

Toshkent - "METHODIST NASHRIYOTI" - 2024

Muharrir: Bakirov Nurmuhammad

Texnik muharrir: Tashatov Farrux

Musahhih: Hazratqulova Ruxshona

Dizayner: Ochilova Zarnigor

Boxishga 10.05.2024 da ruxsat etildi.

Bichimi 60x90. "Times New Roman" garniturasida.

Ofset bosma usulida bosildi.

Shartli bosma tabog'i 10. Nashr bosma tabog'i 9.87.

Adadi 300 nusxa.

*"METHODIST NASHRIYOTI" MCHJ matbaa bo'limida chop etildi.
Manzil: Toshkent shahri, Shota Rustaveli 2-vagon tor ko'chasi, 1-uy.*



+99893 552-11-21

Nashriyot rozilgisiz chop etish ta'qiqlanadi.

