

MYSQL MA'LUMOTLAR OMBORINI BOSHQARISH TIZIMIDA BANK SISTEMASINI MODELLASHTIRUVCHI JADVALLAR TUZISH. “XODIMLAR”, “HISOB”, “MIJOZ”, NOMLARI BILAN YARATISH.

Farg'ona davlat universiteti katta o'qituvchisi
Tojimamatov Israiljon Nurmamatovich
Farg'ona davlat universiteti 22.08-guruh talabasi
G'afforova Zaxrobonu Ixtiyorjon qizi

Annotatsiya: Maqolada MySQL tizimlarining mohiyati, ma'lumotlar omborining zamonaviy axborot saqlash va qidirishdagi o'rni tushuntiriladi. MySQL ning tezkor, ishonchli va mijoz-server modeli asosida ishlashi qayd etiladi. Jadval, yozuv va maydon tushunchalari hamda ular orasidagi 1:1, 1:N, N:M munosabatlari izohlanadi. Bank tizimini modellashtirish uchun “Xodimlar”, “Hisob”, “Mijoz” jadvallarini yaratish zarurligi ko'rsatiladi.

MySql ga kirish.

Hozirgi kunda ma'lumotlar ombori haqida tez-tez gapirilmoqda. Kompyuterlar zamonaviy jamiyatning ajralmas qismini tashkil qiladi, shu sababli quyidagiga o'xshash iboralarni ko'p eshitish mumkin «Men yozuvlarningnima'lumotlar omboridan qidirib ko'raman». Bu erda hujjatlar saqlanadigan katta qutilar emas, balki ma'lumotlarni tezkor qidirishga mo'ljallangan kompyuter tizimlari nazarda tutiladi.

Kompyuterlar hayotimizga chuqurroq kirib bormoqda, chunki ularni tez takrorlanadigan operatsiyalarni bajarish yoki kompyuterning hisoblash tezligi va ma'lumot tashish sig'imisiz echib bo'lmaydigan masalalarning echimini topishga dasturlash mumkin. Ma'lumotni qog'ozga joylashtirish va qog'ozlarni jild va kartotekalarda saqlash sxemasini ishlab chiqish – aniq ishlangan jarayon bo'lsada, qattiq diskdagi elektron hujjatlar jildlarining ko'chirilishi mumkinligi ko'pchilikka ma'qul keldi.

Ma'lumotlar omborining funktsiyalaridan biri ma'lumotni tartiblash va indeksatsiya qilish hisoblanadi. Bu kutubxona kartotekasi singari, faqat kerakli yozuvni topish uchun arxivning yarmini ko'rib chiqish shart emas. hammasi ancha tez bajariladi.

Barcha ma'lumotlar ombori ham bir xil printcip asosida yaratilmaydi, lekin an'anaviy tarzda ularda yozuvlar ko'rinishida ma'lumotlarni tashkil qilish g'oyasi qo'llaniladi. Har bir yozuv belgilangan maydonlar to'plamiga ega. Yozuvlar jadvalga joylashtiriladi, jadvallar yig'indisi esa ma'lumotlar omborini tashkil qiladi.

Ma'lumotlar ombori bilan ishlaganda MOBT (ma'lumotlar omborini boshqarish tizimlari), ya'ni ma'lumotlarga kirish bilan bog'liq barcha ishlarni o'z zimmasiga oluvchi dastur zarur. Unda jadvallar yaratishga imkon beruvchi buyruqlar mavjud,

jadvallarga yozuvlar kiritish, qidirish va hatto yozuvlarni o'chirishi xam mumkin.

MySql – bu tezkor, ishonchli, ochiq tarqatiladigan MOBT hisoblanadi. **MySql** boshqa ko'pchilik MOBT lar singari, «mijoz/server» modeli asosida ishlaydi. Bunda kompyuterlar mijoz yoki server rolini o'ynaydigan tarmoqli arxitektura tushuniladi. 1.1 rasmda mijoz kompyuteri va serverning qattiq diski o'rtasida ma'lumot uzatish sxemasi ko'rsatilgan.

MOBT bir yoki bir nechta ma'lumotlar omborini boshqaradi. Ma'lumotlar ombori ko'plik shaklida tashkil qilingan axborot yig'indisidir. Har bir ko'plik o'zida unifikirlangan yozuvni saqlaydi. Yozuvlar esa maydonlardan tashkil topgan. Odatda ko'pliklar jadvallar deb yuritiladi, yozuvlar esa – jadval satrlari.

Ma'lumotlarning mantiqiy modeli shunday tashkili qilingan. Barcha ma'lumotlar ombori qattiq diskda bitta faylda saqlanishi mumkin. **MySql** da harbir ma'lumotlar ombori uchun alohida katalog yaratiladi, har jadvalga esa uchta fayl to'g'ri keladi. Boshqa MOBT larda ma'lumotlarni jismoniy saqlashning boshqacha printsiplari qo'llanilishi mumkin.

Jadval satrlari o'zaro quyidagi uch usuldan biri bilan bog'lanishi mumkin. Eng oddiy munosabat – -birga birll. Bunday holda birinchi jadval satri ikkinchi jadvalning bitta yagona satriga mos keladi. Diagrammalarda bunday munosabat 1:1 yozuvi bilan ifodalanadi.

-Ko'pga birll munosabati bir jadvalning satri boshqa jadvalning bir nechta satriga mos keluvchi vaziyatni anglatadi. Bu munosabatlarning eng ko'p tarqalgan turidir. Diagrammalarda 1:N yozuvi bilan ifodalanadi.

Nihoyat, -ko'plikka ko'plikll munosabatida birinchi jadval satrlari ikkinchi jadvaldagi ixtiyoriy sonli satrlari bilan bo'lanishi mumkin. Bunday munosabat N:M ko'rinishida yoziladi.

Asosiy qism

Tarmoqli ma'lumotlar ombori

Tarmoqli model ko'plik yozuvlar orasidagi aloqani guruhlashga imkon beruvchi ierarxik modelni kengaytiradi. Mantiqiy jihatdan olib qaralganda, aloqa – bu yozuvning o'zi emas. Aloqalar faqatgina yozuvlar orasidagi munosabatni bildiradi. Ierarxik modeldagi singari, aloqalar ona yozuvdan qiz yozuvga olibboradi, lekin bu gal ko'plik qo'llab quvvatlanadi.

CODASYL spetsifikatsiyasiga mos holda, tarmoqli model DDL (Data Definition Language — ma'lumotlarni aniqlash tili) va DML (Data Manipulation Language — ma'lumotlarni qayta ishlash tili) ni qo'llaydi. Bu maxsus tillar bo'lib, ma'lumotlar ombori tuzilishini aniqlash va so'rovlar tuzish uchun mo'ljallangan. Ularning mavjudligiga qaramasdan, dasturchi avvalgidek ma'lumotlar ombori tuzilishini bilishi lozim.

Tarmoqli modelda –ko'plikka ko'plik munosabati ruxsat etiladi, yozuvlar esa bir biriga bog'liq emas. O'chirilganda yozuvlar va uning barcha aloqalari o'chiriladi, biroq bog'liq yozuvlarning o'zi o'chirilmaydi.

Tarmoqli modelda butunlikni saqlash va nusxalashga yo'l qo'ymaslik maqsadida aloqalarning mavjud yozuvlar orasida o'rnatilishi talab qilinadi. Ma'lumotlarni mos jadvallarda izolyatciya qilishva boshqa jadvallardagi yozuvlar bilan bog'lash mumkin.

Dasturchi ma'lumotlarni diskda qanday saqlanishi haqida qayg'urishi shart emas. Bu ilovalar va ma'lumotlar tobeligini susaytiradi. Ammo tarmoqli modelda

dasturchi so'rovlarni shakllantirayotganda ma'lumotlar strukturasi yodda tutishi talab qilinadi.

Ma'lumotlar omborining optimal strukturasi shakllantirish oson emas, tayyor tuzilmani almashtirish esa qiyin. Agar jadval ko'rinishi ko'p o'zgartirilsa, boshqa jadval bilan barcha bog'liqliklar ma'lumotlar butunligini buzilmasligi uchun qaytadan o'rnatilishi kerak. Ushbu masalaning murakkabligi shunga olib keladiki, dasturchilar ilovalarni soddalashtirish maqsadida butunlikning ba'zi cheklovlarini inkor etadilar.

Ma'lumotlar omborini yaratish, jadval bilan ishlash asoslari

Ushbu ma'ruzada ma'lumotlar omborini yaratish usullari, jadval yaratish vaularni o'chirish buyruqlari ko'rib chiqiladi.

Ushbu ma'ruzada ma'lumotlar omborini yaratishni o'rganamiz.

Windows va Linux da ma'lumotlar omborini yaratish buyruqlari bir xil.

Ammo Linux da oldindan bajariladigan buyruqlar murakkabroq.

BigFoot nomli biror kompaniyaning ishchilari haqidagi ma'lumotlardan tashkil topgan employees nomli ma'lumotlar omborini yaratamiz. Ushbu ma'lumotlar omborida ishchilarning ismi, familiyasi, maoshi, yoshi, manzili, e-mail, tug'ilgan sanasi, qiziqishlari, telefon raqamlari va boshqalarni saqlash ko'zdatutiladi.

Windows da ma'lumotlar omborini yaratish

1. **MySQL** serverini ishga tushiring, buning uchun c:\MySQL\bin katalogidatklif satridagi **MySQLd-shareware** –standalone buyrug'ini tanlang.

2. So'ngra taklif satrida **MySQL** ni kiritish orqali **MySQL** mijoz dasturini chaqiring.

3. Taklif **MySQL>** ga o'zgaradi. Buyruqni kiriting:
create database employees;

(Izoh: Buyruq nuqtali vergul bilan tugaydi).

4. **MySQL** serveri taxminan 3.1 rasmdagidek javob berishi kerak

```
Query OK, 1 row affected (0.00 sec)
```

1.Rasm. Jadval yaratish buyrug'i ishining natijasi[So'rov qayta ishlandi, 1 satr

o'zgardi (0.00 sek)]

5. Bu ma'lumotlar ombori muvaffaqiyatli yaratilganligini anglatadi. Endi tizimda nechta ma'lumotlar ombori mavjudligini ko'ramiz. Quyidagi buyruqni bajaring:

`show databases;`

Server 2 rasmda ko'rsatilgandek, ma'lumotlar omborlari ro'yxati bilan javob beradi.

```
+-----+
| Data base |
+-----+
| employees |
| mysql     |
| test      |
+-----+
2 rows in set (0.00 sec)
```

2.Rasm. Ma'lumotlar omborlarini ko'rish.

Bu erda uchta ma'lumotlar ombori ko'rsatilgan, ulardan ikkitasi **MySql** tomonidan o'rnatish vaqtida yaratilgan hamda yangi yaratilgan employees ma'lumotlar ombori.

6. Yana DOS taklifiga qaytish uchun **MySql** taklifidan quit buyrug'ini tanlang.

Linux da ma'lumotlar ombori yaratish

1. Foydalanuvchi o'zining hisob yozuvi ustidan o'z ishlagani ma'qul, superfoydalanuvchi root kabi emas. Terminal seansni ishga tushirish zarur va superfoydalanuvchi bo'lishingiz mumkin (Buning uchun su buyrug'ini bajaring va superfoydalanuvchi root parolini kiriting).

2. **MySql** serverini ishga tushiramiz. Kiritamiz:

MySql -u root -p

Tizim **MySql** ni Linux ga o'rnatganda berilgan root **MySql** foydalanuvchi parolini kiritishni taklif qiladi. (Izoh: Bu Linux tizimining root foydalanuvchining emas, **MySql** tizimining root foydalanuvchi paroli). Ekranda xavfsiz parolni kiriting.

Muvaffaqiyatli registratsiyadan so'ng, tizim 3 rasmda ko'rsatilganidek salomlashish va **MySql** ga taklif matnini chiqaradi.

```
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1 to server version: 5.01.01

Type 'help' for help.

mysql>
```

3.Rasm. Tizim bilan salomlashish.

(Siz **MySql** monitoriga xush kelibsiz. Buyruqlar ; yoki \g. id simvollari bilan tugallanadi, **MySql** bilan bog'lanish 5.01.01 versiyadagi server uchun 1 ga teng. Ma'lumotnoma olish uchun 'help' ni kiriting).

3. Endi employees ma'lumotlar omborini yaratsak bo'ladi. Quyidagi

buyruqni bajaring:

create database employees;

(Izoh: buyruqlar nuqtali vergul bilan tugallanadi)

4. Shuni ta'kidlash muhimki, ushbu ma'lumotlar ombori root foydalanuvchisi tomonidan yaratiladi va shuning uchun unga kirish faqat rootruksat bergan foydalanuvchilargagina mumkin bo'ladi. Ushbu ma'lumotlar omboridan boshqa, deylik, misha nomli hisob yozuvi bilan foydalanish uchun quyidagi buyruqlarni bajarish lozim:

GRANT ALL ON employees.* TO freak@localhost IDENTIFIED BY "pass"

Ushbu buyruq freak@localhost hisob yozuviga employees ma'lumotlar ombori uchun barcha vakolatlarni taqdim qiladi va pass parolini beradi. Ixtiyoriy boshqa foydalanuvchi uchun freak ni foydalanuvchining biror boshqa nomiga o'zgartirish va mos parolni tanlash kerak.

5. **MySQL** seansini quit buyrug'ini taklifida GRANT ALL ON employees.* TO freak@localhost IDENTIFIED BY "pass" ni kiritish orqali yoping. Superfoydalanuvchi rejimidan chiqing va o'zingizning hisob yozuvingizga o'ting (exit ni kiriting).

6. **MySQL** bilan oddiy hisob yozuvi yordamida bog'lanish uchun quyidagilarni kiriting:

MySQL –va foydalanuvchi nomi –p

Keyin taklifdan so'ng parolni kiriting (Ushbu parol yuqorida GRANTS ALL... buyrug'i bilan berilgan edi). **MySQL** da muvaffaqiyatli registratsiyadanso'ng, tizim salomlashish xabarini chiqaradi. Foydalanuvchi seansi 4 rasmda ko'rsatilganidek ko'rinishi kerak.

```
[misha@localhost misha]$ mysql -u misha -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 3 to server version: 3.22.32

Type 'help' for help.

mysql>
```

1.Rasm. MySQL tizimida salomlashish oynasi.

7. SHOW DATABASES buyrug'ini kiritish tizimdagi barcha kirish mumkinbo'lgan ma'lumotlar omborlarining ro'yxatini chiqaradi.

MySQL> SHOW DATABASES;

Ekranda 2 rasmdagiga o'xshash oyna paydo bo'lishi kerak.

MySQL mijoz dasturidan chiqish uchun **MySQL>** taklif satrida quit ni kiriting.

CREATE DATABASE buyrug'i

CREATE DATABASE buyrug'i sintaksisi quyidagi ko'rinishga ega: CREATE DATABASE [IF NOT EXISTS] ma'lumotlar ombori nomi[create_spetsifikatsiyasi [,create_spetsifikatsiyasi]...]

CREATE DATABASE buyrug'i ma'lumotlar omborini ko'rsatilgan nom bilan yaratadi. Buyruqdan foydalanish uchun CREATE imtiyoziga ma'lumotlar ombori uchun ega bo'lish kerak. Agar shu nomli ma'lumotlar ombori mavjud bo'lsa, xato generatsiyalanadi.

create_spetsifikatsiyasi:

[DEFAULT] CHARACTER SET simvollar_to'plami_nomi [DEFAULT] COLLATE taqqoslash_tartibi_nomi

create_spetsifikatsiyasi optsiyasi ma'lumotlar ombori xarakteristikalarini aniqlash maqsadida ko'rsatilishi mumkin. Ma'lumotlar ombori xarakteristikalari ma'lumotlar katalogida joylashgan db.opt faylida saqlanadi. CHARACTER SET konstruktsiyasi jimlik qoidasiga asosan, ma'lumotlar ombori uchun simvollar to'plamini aniqlaydi. COLLATION konstruktsiyasi jimlik qoidasiga asosan, taqqoslash tartibini beradi.

MySql da ma'lumotlar omborlari katalog shaklida amal qiladi, kataloglar ma'lumotlar ombori jadvallariga mos keluvchi fayllardan tashkil topgan. Omborda boshidan hech qanday jadval bo'lmaganligi bois, CREATE DATABASE operatori faqat **MySql** ma'lumotlar katalogida kichik katalog yaratadi.

Jadvallar bilan ishlash

Endi **MySql** ning ma'lumotlar omborini tanlash va jadvallar yaratish uchun mo'ljallangan buyruqlarini ko'rib chiqamiz.

Yuqorida keltirilgan jadval to'rt ustundan iborat, har bir ustunda ism, yosh, mamlakat va e-mail manzili saqlanadi. Har bir satr bir kishining ma'lumotlarini saqlaydi. Ushbu satr yozuv deyiladi. Snejananing e-mail manzili va mamlakatini topish uchun avval birinchi ustundan ismni tanlash hamda ushbu satrdagi uchinchi va to'rtinchi ustunlar tarkibini ko'rish zarur.

Ma'lumotlar ombori tarkibida ko'p jadvallarni saqlashi mumkin, aynan jadvallar haqiqiy ma'lumotlarni saqlaydi.

O'z navbatida o'zaro bog'liq (yoki bog'liq bo'lmagan) ma'lumotlarni turli jadvallarga ajratish mumkin. Employees ma'lumotlar ombori uchun kompaniyaning ishchilari haqidagi ma'lumotlarni bir jadvalda saqlaydi, boshqa jadvalda esa shaxsiy ma'lumotlar saqlanadi. Birinchi jadvalni yaratamiz.

Bunday jadvalni yaratish uchun SQL buyrug'i quyidagi ko'rinishda bo'ladi:

```
CREATE TABLE employee_data(
emp_id int unsigned not null auto_increment primary key,f_name varchar(20),
l_name varchar(20),title varchar(30), age int,
```



```
yos int, salary int, perks int,  
email varchar(60)  
);
```

Izoh: **MySQL** da buyruqlar va ustun nomlari simvol registrklarini farqlay olmaydi, lekin jadvallar va ma'lumotlar ombori nomlari, ular qo'llaniladigan platformaga qarab registrdan bog'liq bo'lishi mumkin (Linux dagi kabi). Shuning uchun CREATE TABLE o'rniga create table dan foydalanish mumkin.

CREATE TABLE kalit so'zlari ortidan yaratilayotgan jadval nomi employee_data keladi. Qavs ichidagi har bir satr bitta ustunni anglatadi. Bu ustunlar har bir xodimning identifikatsion raqami (emp_id), familiyasi (f_name), ismi (l_name), lavozimi (title), yoshi (age), kompaniyada ish staji (yos), maoshi (salary), qo'shimcha (perks) va e-mail manzili (email) ni saqlaydi.

Har bir ustun nomi ortidan ustun tipi keladi. Ustunlar tipi ulardagi ma'lumotlar tipini aniqlaydi. Ushbu misolda f_name, l_name, title va email ustunlari matnli satrlardan tashkil topgan bo'ladi, shuning uchun ustun tipi varchar, ya'ni simvollarning o'zgaruvchi miqdori kabi berilgan. Varchar ustunlari uchun simvollarning maksimal soni ustun nomidan so'ng qavs ichida yoziladigan son bilan aniqlanadi. Age, yos, salary i perks ustunlari sonlar (butun) dan tashkil topganligi uchun ustun tipi int kabi beriladi. Birinchi ustun (emp_id) xodimning identifikatsion raqami (id) ni saqlaydi. Uning ustuni tipi birmuncha murakkab ko'ringanligi uchun uni qismlarga bo'lib ko'rib chiqamiz:

- int: ustun tipini butun son sifatida aniqlaydi.
- unsigned: son belgisiz bo'lishini belgilaydi (musbat son).
- not null: qiymat null (bo'sh) bo'la olmasligini, ya'ni ushbu ustundagi har bir satr qiymatga ega bo'lishini belgilaydi.
- auto_increment: **MySQL** auto_increment atributli ustunga duch kelganda, ustundagi eng katta qiymatdan bir miqdorga katta yangi qiymat paydo bo'ladi. Shuning uchun biz ushbu ustun uchun qiymat berishimiz kerak, **MySQL** uni mustaqil generatsiyalaydi. Bundan kelib chiqadiki, ushbu ustundagi har bir qiymat unikal bo'ladi.
- primary key: ustunni indeksatsiyalashda yordam beradi, bu esa qiymatlarni qidirishni tezlashtiradi. Har bir qiymat unikal bo'lishi kerak. Kalit ustun ma'lumotlar takrorlanishini oldini olish uchun zarur. Misol uchun, ikki xodim bir xil ismga ega bo'lishi mumkin, bunda agar ularga unikal identifikatsion raqam berilmasa, ularni farqlash muammosi yuzaga keladi. Agar unikal qiymatli ustun mavjud bo'lsa, ikki yozuvni oson ajratish mumkin. Yaxshisi, unikal qiymatlarni o'zlashtirishni **MySQL** tizimining o'ziga topshirish kerak.

Ma'lumotlar omboridan foydalanish

Employees ma'lumotlar ombori yaratildi. U bilan ishlash uchun uni

-aktivlashtirish yoki -tanlash zarur. **MySql** taklifida quyidagi buyruqni bajaring:

SELECT DATABASE();

Ekranida 3.5 rasmda ko'rsatilganidek, tizim javobini ko'ramiz

```
mysql> SELECT DATABASE();
+-----+
| DATABASE |
+-----+
|          |
+-----+
1 rows in set (0.00 sec)
```

1.Rasm. Ma'lumotlar omborini tanlash

Bu shuni anglatadiki, birorta ma'lumotlar ombori tanlanmadi. Aslida **MySql** mijozi bilan har marotaba ishlaganda, qaysi ma'lumotlar omboridan foydalanilishni aniqlash zarur.

Joriy ma'lumotlar omborini bir necha usul bilan aniqlash mumkin:

- ishga tushirgan vaqtda ma'lumotlar ombori nomini aniqlash

Tizim taklifida quyidagilarni kiriting:

MySql employees (Windows da)

MySql employees -u manish -p (Linux da) Ma'lumotlar omborini **MySql** taklifidagi **USE** operatoriyordamida aniqlash

MySql>USE employees;

Ma'lumotlar omborini **MySql** taklifidagi u yordamida aniqlash

MySql>\u employees;

MySql bilan ishlaganda foydalaniladigan ma'lumotlar omborini aniqlab olish muhim, aks holda **MySql** xatoni keltirib chiqaradi.

Jadval yaratish

Employees ma'lumotlar ombori tanlangandan so'ng, **MySql** taklifida **CREATE TABLE** buyrug'ini tanlang.

```
CREATE TABLE employee_data(
emp_id int unsigned not null auto_increment primary key,f_name varchar(20),
l_name varchar(20),title varchar(30),
age int, yos int, salary int,perks int,
email varchar(60)
```


);

Izoh: Birinchi satrdan so'ng, Enter tugmasini bosish **MySQL** taklifini -> ga o'zgartiradi. Bu shuni anglatadiki, **MySQL** buyruq tugamaganligini tushunadi va qo'shimcha operatorlarni kiritishni taklif etadi. Unutmang, har bir **MySQL** buyrug'i nuqtali vergul bilan tugaydi, ustundagi har bir e'lon vergul bilan ajratiladi. Agar istasangiz, hamma buyruqni bitta satrda kiritish mumkin.

Ekran 6 rasmdagi ko'rinishga ega bo'lishi kerak.

```
mysql> CREATE TABLE employee data
-> (
-> emp_id int unsigned not null auto_increment primary key,
-> f_name varchar(20),
-> l_name varchar(20),
-> title varchar(30),
-> age int,
-> yos int,
-> salary int,
-> perks int,
-> email varchar(60)
-> );

Query OK, 0 rows affected (0.01 sec)
```

2.Rasm. Jadval yaratish

CREATE TABLE buyrug'ining sintaksisi

CREATE TABLE instruktsiyasining umumiy formati quyidagicha: CREATE [TEMPORARY] TABLE [IF NOT EXISTS] nom

[(spetsifikatsiya, ...)]

[optsiya, ...]

[[IGNORE | REPLACE] so'rov]

TEMPORARY bayrog'i joriy seans davomida amal qiladigan vaqtinchalik jadvalni yaratadi. Seans tugashi bilan jadval o'chiriladi. Vaqtincha amal qiladigan jadvallarga boshqa jadvallar nomini berish mumkin, bunda shu jadvallarga

vaqtincha kirib bo'lmaydi. IF NOT EXIST spetsifikatori agar ko'rsatilgan nomli jadval mavjud bo'lsa, xatolar to'g'risida xabar chiqarishni kamaytiradi. Jadvallar nomlari ma'lumotlar omborining nuqta bilan ajratilgan nomidan keyin kelishi mumkin. Agar bunday qilinmagan bo'lsa, jadval jimlik qoidasiga binoan o'rnatilgan ma'lumotlar omborida yaratiladi.

Jadvalga probelli nom berish uchun uni teskari qavsga, masalan, 'courses list'ga joylashtirish lozim. Xuddi shu ishni jadvalga barcha murojaatlar uchun qilish kerak, chunki probellar identifikatorlarni ajratish uchun qo'llaniladi.

Ustunsiz jadvallar yaratishga ruxsat beriladi, ammo ko'p hollarda hech bo'lmaganda bitta ustun spetsifikatsiyasi mavjud bo'ladi. Ustunlar va indekslar spetsifikatsiyasi aylana qavslar va vergullar bilan ajratiladi. Spetsifikatsiyalar formati quyidagicha:

nom tip

[NOT NULL | NULL]

[DEFAULT qiymat] [AUTO_INCREMENT][KEY]

[murojaat]

Tip spetsifikatsiyasi tipning nomlanishi va uning o'lchamlanishini o'z ichiga oladi. Jimlik qoidasiga asosan, ustunlar NULL qiymatini oladi. NOT NULL spetsifikatori bunday holatni man qiladi.

Har bir ustunda jimlik bo'yicha qiymati mavjud. Agar u ko'rsatilmagan bo'lsa, **MySQL** dasturi uni mustaqil tanlaydi. NULL qiymatini qabul qiluvchi ustunlar uchun jimlik bo'yicha qiymat NULL bo'ladi, satri ustunlar uchun – bo'shsatr, sonli ustunlar uchun – nul. Bunday o'rnatishni o'zgartirish uchun DEFAULT taklifidan foydalanish zarur.

AUTO_INCREMENT bayrog'i yordamida yaratiladigan hisoblagich –maydonlar jimlik bo'yicha berilgan qiymatlarni qabul qilmaydi, chunki ulardatartib raqamlari yoziladi. Hisoblagich tipi belgisiz butun bo'lishi zarur. Jadvalda faqat bitta hisoblagich – maydon bo'lishi mumkin. U birinchi kalit bo'lishi shartemas.

Jadvalni o'chirish

Jadvalni o'chirishdan oldin uning mavjudligiga ishonch hosil qilamiz. Buni

2 rasmda ko'rsatilganidek, SHOW TABLES buyrug'i yordamida tekshirish mumkin.

```
mysql> SHOW TABLES;
+-----+
| Tables in employees |
+-----+
| employee data      |
+-----+
1 rows in set (0.00 sec)
```

7.Rasm. Ma'lumotlar omborida jadvalni ko'rish

Jadvalni o'chirish uchun 8 rasmda ko'rsatilganidek, DROP TABLE buyrug'idan foydalaniladi.

```
mysql> DROP TABLE employee_data;
Query OK, 0 rows affected (0.01 sec)
```

7.Rasm. Jadvalni o'chirish

Endi SHOW TABLES buyrug'i bu jadvalni boshqa ko'rsatmaydi.

DROP TABLE buyrug'i sintaksisi

DROP TABLE instruktciyasi quyidagi sintaksisga ega:

DROP TABLE [IF EXISTS] jadval [RESTRICT | CASCADE]

IF EXISTS spetsifikatsiyasi berilgan jadval mavjud bo'lmagan holatda xato haqidagi xabarlarini chiqarishni kamaytiradi. Bir necha jadval nomlarini, ularni vergul bilan ajratib ko'rsatish mumkin.

RESTRICT i CASCADE bayroqlari boshqa MOBT da yaratilgan stsenariylarni bajarish uchun mo'ljallangan

XULOSA

SQL, Strukturalangan so'rovlar tili, MySQL kabi relyatsion ma'lumotlar bazalari bilan bog'lanish uchun oddiy, standartlashtirilgan til. SQL yordamida ma'lumotlar bazalari va jadvallarni yaratish, ma'lumotlar bazalari jadvallarida ma'lumotlarni saqlash, maxsus yozuvlar uchun ma'lumotlar bazasini so'rash, ma'lumotlar bazalarida ma'lumotlarni o'chirish va yangilash kabi har qanday ma'lumotlar bazasiga tegishli vazifalarni bajarishingiz mumkin

Ma'lumotlar bazasi mantiqiy bog'liq ma'lumotlarni saqlash uchun ishlatiladi. Ma'lumotlar bazasining har xil turlari mavjud. Matn va raqamlarni saqlaydigan ma'lumotlar bazalari an'anaviy ma'lumotlar bazasi hisoblanadi. Rasmlarni saqlaydigan ma'lumotlar bazalari Multimedia ma'lumotlar bazalari deb nomlanadi. Ba'zi tashkilotlar geografik tasvirlarni saqlash uchun geografik axborot tizimlaridan foydalanadilar. Ma'lumotlar bazasining umumiy turlaridan biri bu o'zaro bog'liq ma'lumotlar bazalari. Ushbu maqolada SQL va MySQL o'rtasidagi farq muhokama qilindi. SQL va MySQL o'rtasidagi farq shundaki, SQL bu ma'lumotlar bazasidagi ma'lumotlarni boshqarish uchun so'rovlar tili, MySQL esa SQL yordamida ma'lumotlar bazalarini boshqarish uchun ochiq manbali ma'lumotlar bazasini boshqarish tizimi .

MySql dasturida Bank tizimining modellashtiruvchi bir guruh jadvallar yaratish
jadvallar

```
mysql> SHOW TABLES;
+-----+
| Tables_in_bank |
+-----+
| bolimlar        |
| mijoz           |
| tranzaksiya     |
| xisob           |
| xizmatlar       |
| xodimlar        |
+-----+
6 rows in set (0.01 sec)
```

1. "Mijoz" jadvali

```
1 CREATE TABLE `mijoz` (
2   `id_raqami` int NOT NULL,
3   `xisob raqami` varchar(45) NOT NULL,
4   `ism` varchar(45) NOT NULL,
5   `familiya` varchar(45) NOT NULL,
6   PRIMARY KEY (`id_raqami`))
```

```
mysql> DESCRIBE MIJOZ;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id_raqami      | int           | NO   | PRI | NULL    |       |
| xisob raqami   | varchar(45)   | NO   |     | NULL    |       |
| ism            | varchar(45)   | NO   |     | NULL    |       |
| familiya       | varchar(45)   | NO   |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

SELECT * FROM MIJOZ

Result Grid				
Filter Rows:				
	id_raqami	xisob raqami	ism	familiya
▶	211111	5255889922	Dilorom	Xushvaqtova
	222222	5255889925	Muxlisa	Abdulhaqova
	234564	5255889923	Nozima	Kamolova
	555555	5889924	Kamoliddin	Norboyev
	788777	5255889921	Behruzбек	Temirov
●	NULL	NULL	NULL	NULL

2."Xisob" jadvali

```

1 CREATE TABLE `xisob` (
2   `T/r` int NOT NULL,
3   `xisob raqami` varchar(45) NOT NULL,
4   `amal qilish muddati` varchar(45) NOT NULL,
5   `mablag` varchar(45) NOT NULL,
6   `valyuta` varchar(45) NOT NULL,
7   PRIMARY KEY (`T/r`)

```

```
mysql> DESCRIBE XISOB;
```

Field	Type	Null	Key	Default	Extra
T/r	int	NO	PRI	NULL	
xisob raqami	varchar(45)	NO		NULL	
amal qilish muddati	varchar(45)	NO		NULL	
mablag	varchar(45)	NO		NULL	
valyuta	varchar(45)	NO		NULL	

5 rows in set (0.00 sec)

```
SELECT*FROM XISOB
```

T/r	xisob raqami	amal qilish muddati	mablag	valyuta
1	5255889925	2025 yil	100	yevro
2	7255889925	2024 yil	150	yevro
3	5955889928	2024 yil	1500	yevro
4	8255889929	2025 yil	500	yevro
5	2255889927	2023 yil	1000	yevro
6	5255889929	2025 yil	500	yevro
NULL	NULL	NULL	NULL	NULL

3."Xodimlar" jadvali

```

1 CREATE TABLE `xodimlar` (
2   `id_raqami` int NOT NULL,
3   `ismi` char(10) NOT NULL,
4   `familiya` char(10) NOT NULL,
5   `malumoti` char(10) NOT NULL,
6   PRIMARY KEY (`id_raqami`)

```



```
mysql> DESCRIBE XODIMLAR;
```

Field	Type	Null	Key	Default	Extra
id_raqami	int	NO	PRI	NULL	
ismi	char(10)	NO		NULL	
familiya	char(10)	NO		NULL	
malumoti	char(10)	NO		NULL	

4 rows in set (0.00 sec)

```
SELECT*FROM XODIMLAR
```

id_raqami	ismi	familiya	malumoti
1	Shohruh	Abdullayev	oliy
2	Inomjon	Ummatov	oliy
3	Nigora	Abdakimova	oliy
4	Ozoda	Xoliqulova	oliy
NULL	NULL	NULL	NULL

FOYDALANILGAN ADABIYOTLAR:

1. Ратшиллер Т., Геркен Т. PHP4: разработка Web-приложений. Питер,2001. - 384 с.
2. Д.Котерев, А.Котерев. PHP5. наиболее полное руководство./Санкт Петербург «БХВ-Петербург» - 2005.- 1120 с.
3. Уилтон П. JAVASCRIPT. Основы. Символ-плюс. 2002. 1056 с.
4. Кингли-Хью Э., Кингли-Хью К. JAVASCRIPT 1.5: Учебный курс.Питер. 1-е издание. 2002.
5. Дронов В. JavaScript в Web дизайне. Питер. 2005.
6. Мазуркевич М.М., Мазуркевич И.М. PHP4. Настольная книгапрограммиста. СПб. 2003.
7. Бранденбау. JAVASCRIPT. Сборник рецептов для профессионалов.СПб. 2001.
8. Томсон Л., Веллинг Л. Разработка Web-приложений на PHP иMySQL. - К.: "ДиаСофт", 2008. - 672 с.
9. Ahmedov A. Tayloqov N. Informatika. –Т.: -2001 у.
10. Hayitov A.G. Informatikadan mashqlarni kompyuterda bajarishningnazariy asoslari. –Т.: A.Qodiriy. -2003у. -211 b.

11. Tojimamatov, I., & Adxamova, C. (2025). AMALIY TIZIMLARDA BERILGANLAR BAZASINI BOSHQARISH TIZIMLARI O'RNI. *Академические исследования в современной науке*, 4(21), 77-82.
12. Tojimamatov, I. (2025). ADO-NET TEXNOLOGIYASI YORDAMIDA HISOBOTLAR VA FORMALARNI SHAKLLANTIRISH. *Академические исследования в современной науке*, 4(25), 122-126.
13. Tojimamatov, I., & Fazliddinov, X. (2025). Berilganlar bazasi administratori va uning xususyatlar. *Академические исследования в современной науке*, 4(21), 90-95.
14. Tojimamatov, I. N., & Iminova, G. I. (2025). BERILGANLARNI MARKAZLASHGAN TARZDA BOSHQARISH PRINSIPLARI. *ОБРАЗОВАНИЕ И НАУКА В XXI ВЕКЕ*, (62-4 (том 1)).
15. Tojimamatov, I. (2025). BERILGANLAR BAZASIDA AXBOROT ALMASHINISH SXEMASI HAMDA TURLARI. *Академические исследования в современной науке*, 4(21), 71-76.