PDP ACADEMY

Data types and Constraints in PostgreSQL.

PDP ACADEMY

Reja:

1. Connect To a PostgreSQL Database
2. Server
3. Data types
4. Constraints => NOT NULL, UNIQUE, PK, FK, CHECK

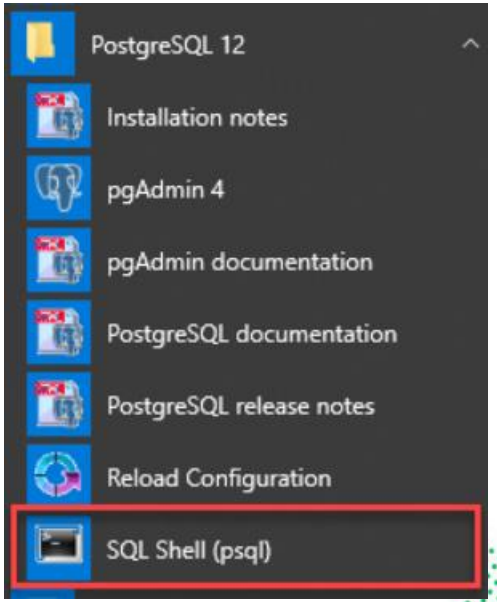
Connect To a PostgreSQL Database

Agar PostgreSQL ma`lumotlar bazasi kompyuteringizga o`rnatilgan bo'lsa, PostgreSQL ma'lumotlar bazasi serveriga quyidagi vositalar orqali ulanishingiz mumkin:

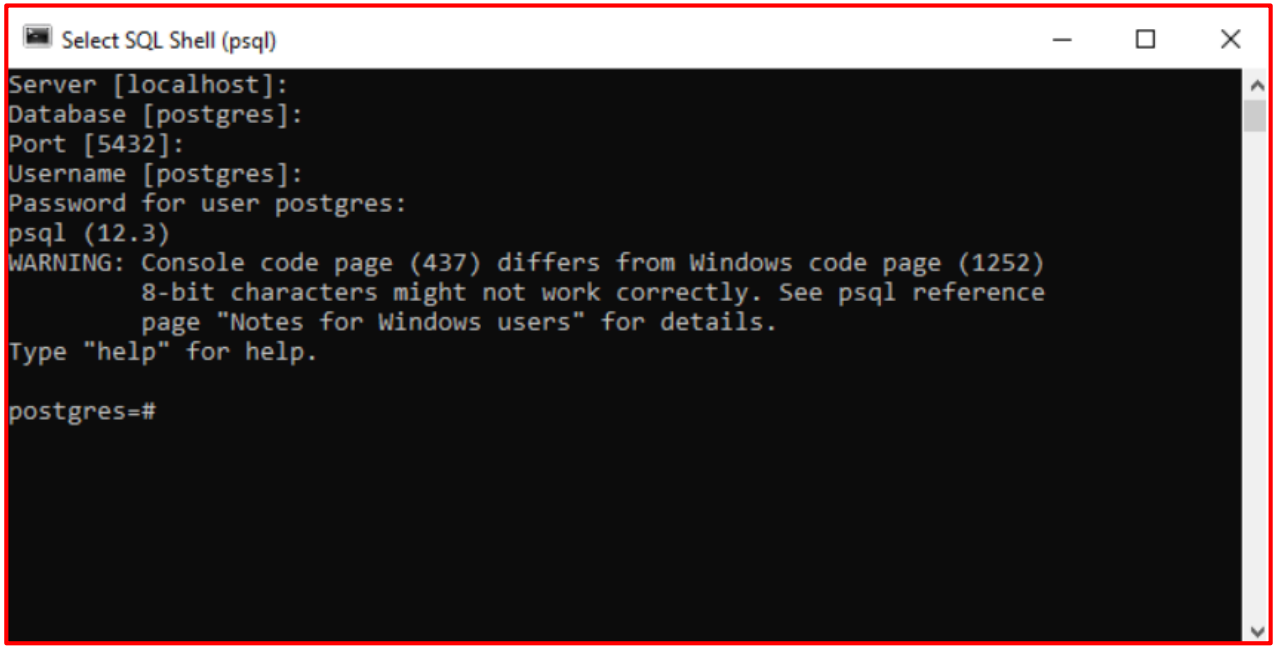
- **psql** - PostgreSQL ma`lumotlar bazasi serverining terminalga asoslangan old qismi.
- **pgAdmin** - PostgreSQL ma`lumotlar bazasi serverining veb-ga asoslangan front-end qismi.

psql yordamida PostgreSQL ma'lumotlar bazasi serveriga ulanish

psql - bu PostgreSQL tomonidan taqdim etilgan interaktiv terminal dasturi. Bu sizga PostgreSQL ma`lumotlar bazasi serveri bilan ishlashga imkon beradi, masalan, SQL bayonotlarini bajarish va ma`lumotlar bazasi ob`ektlarini boshqarish v.h.



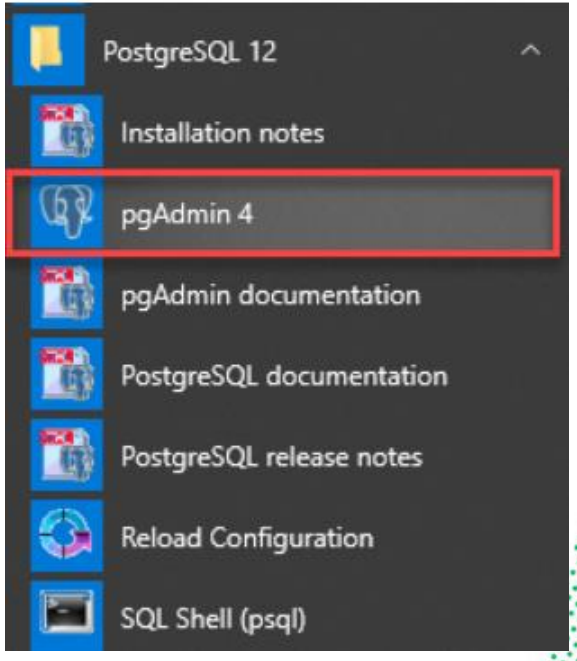
Server, Ma'lumotlar bazasi, Port, Foydalanuvchi nomi va Parol kabi barcha ma'lumotlarni kiriting. Agar siz Enter tugmasini bossangiz, dastur kvadrat qavsda ko'rsatilgan standart qiymatdan foydalanadi [] va kursorni yangi qatorga o'tkazadi . Masalan, **localhost** standart ma`lumotlar bazasi serveri. Postgres foydalanuvchisi parolini kiritish bosqichida siz PostgreSQL o'rnatish vaqtida tanlagan foydalanuvchi postgres parolini kiritishingiz kerak .



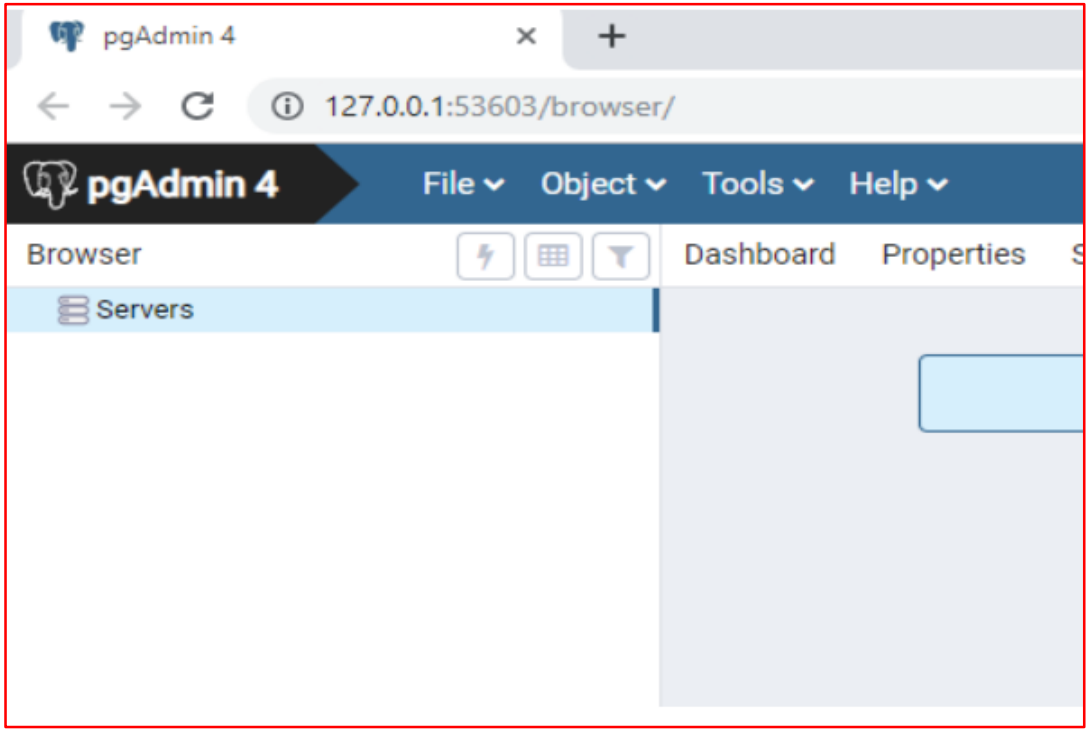
pgAdmin yordamida PostgreSQL ma'lumotlar bazasi serveriga ulanish

Ma`lumotlar bazasiga ulanishning ikkinchi usuli **pgAdmin** ilovasidan foydalanishdir. **pgAdmin** ilovasi intuitiv foydalanuvchi interfeysi orqali **PostgreSQL** ma'lumotlar bazasi serveri bilan o'zaro ishlash imkonini beradi.

Birinchi, pgAdmin dasturini ishga tushiring.

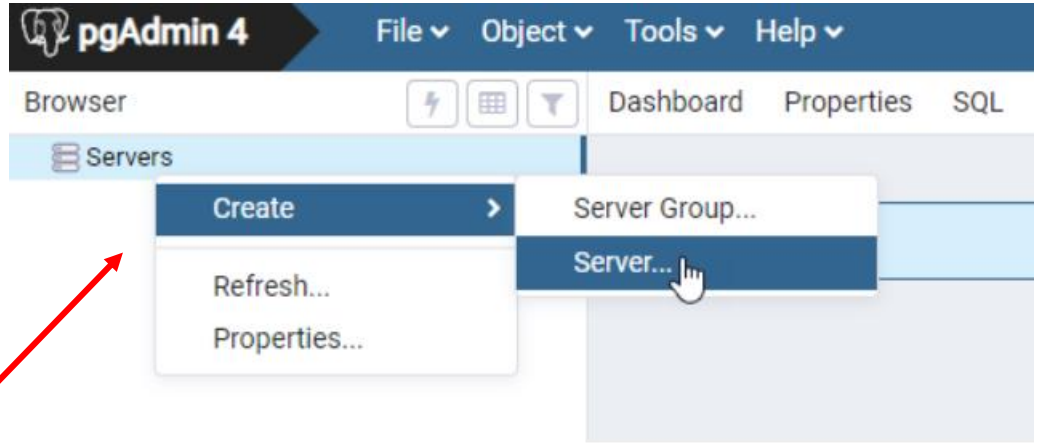


pgAdmin ilovasi quyidagi rasmda ko'rsatilganidek, veb-brauzerda ishga tushadi:



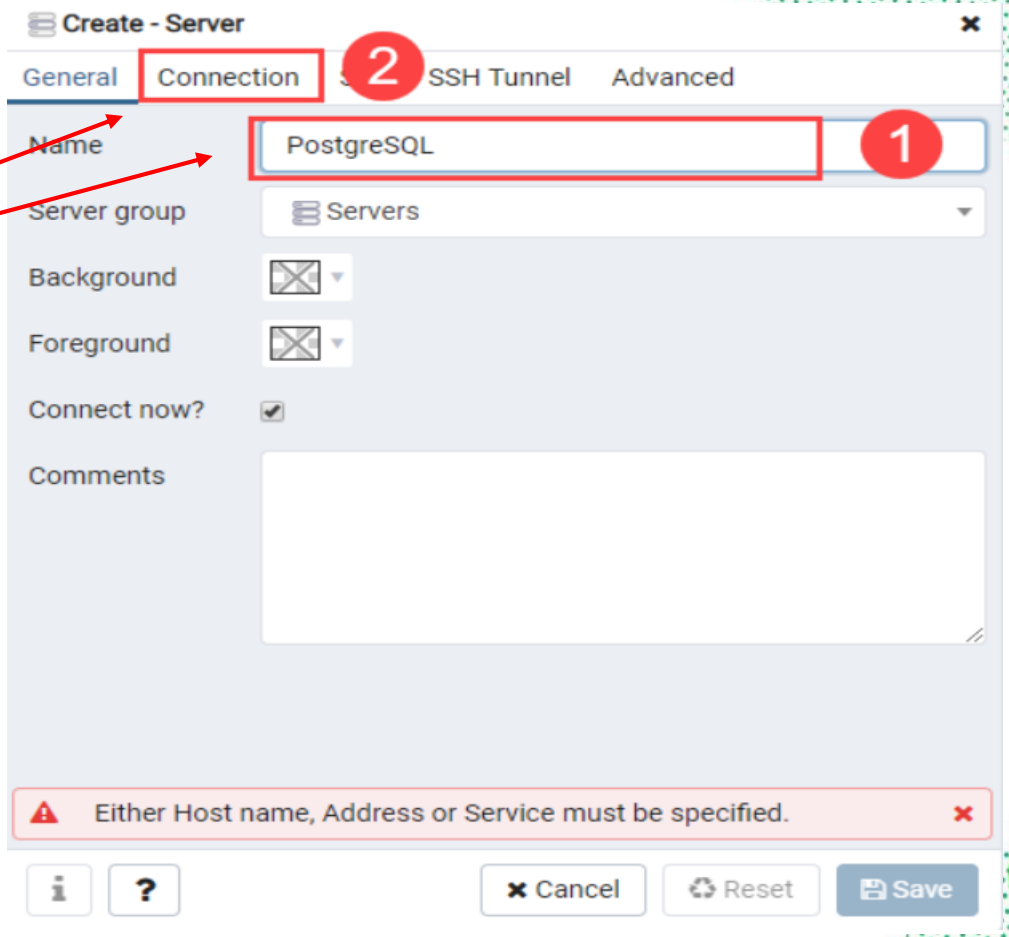
PostgreSQL Server

PostgreSQL dasturini o`rnatganingizda, sizda mos keladigan PostgreSQL server xizmati bo`ladi. PostgreSQL server xizmati PostgreSQL serveri sifatida ham tanilgan. Siz turli xil portlar va ma`lumotlarni saqlash uchun turli joylarga ega bo'lgan jismoniyA serverga bir nechta PostgreSQL serverlarini o`rnatishingiz mumkin. **Server yaratish** uchun Serverlar tugmasi ustida sichqonchaning o`ng tugmasi bilan bosing va **Yaratish > Server…** menyusini tanlang



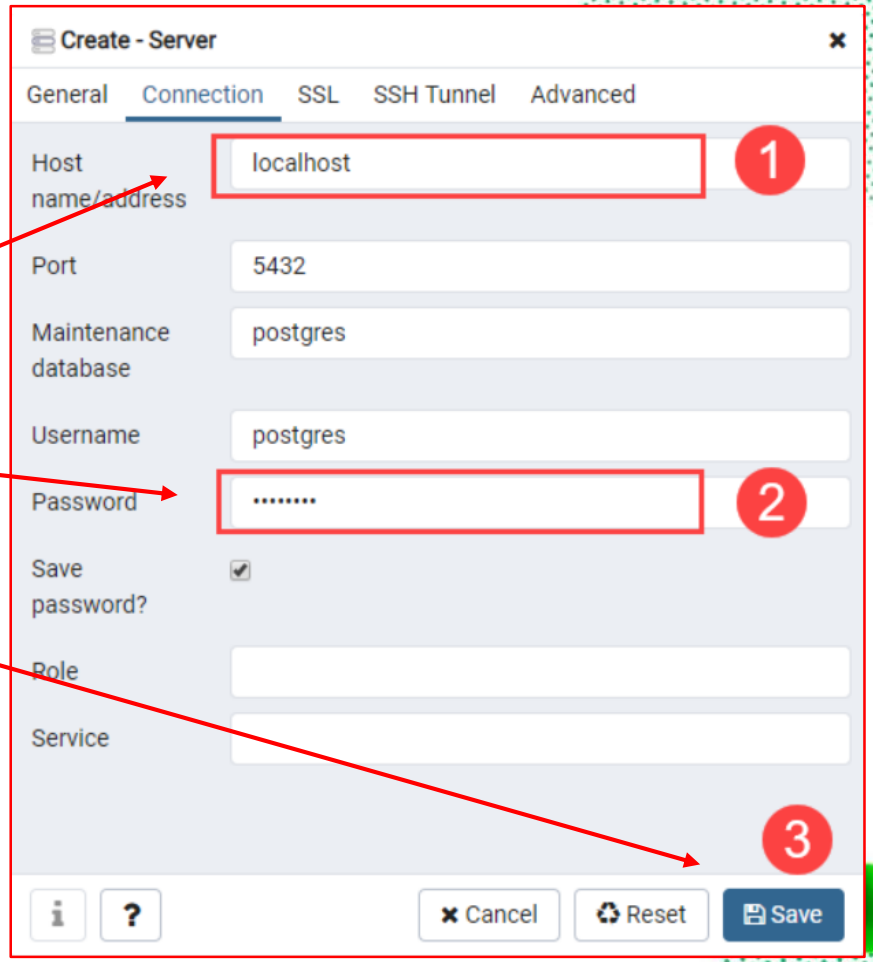
PDP ACADEMY

Server nomini kiriting, masalan, **PostgreSQL** va **Connection** tugmasini bosing:



PDP ACADEMY

postgres useri uchun **host** va **parol**ni kiriting va **Save** tugmasini bosing:



PDP ACADEMY

PostgreSql da ma`lumotlar jadval ko`rinishda taqdim etiladi.

Har qanday jadval, nomlangan qatorlardan tashkil topadi. Jadvaldagi hamma qatorlar bir xil nomlangan ustunlardan tashkil topadi. Bu holatda har bitta ustun uchun qandaydir bir tipni ko`rsatish mumkin. Ya`ni bu ustunga tegishli ma`lumotlar qaysi tipda bo`lishi mumkinligini aniqlab olishga imkon beradi.

Jadvallar ma`lumotlar omboriga birlashtiriladi, aniq bir PostgreSQL serverdagi ma`lumotlar ombori esa ma`lumotlar ombori klasterini tashkil etadi.

Shaxsiy ish N	Familiyasi	Ismi	Otasining ismi	Tug`ilgan sana
16493	Aliev	Karim	Ergashevich	1 yanvar 1979 yil
16498	Bokiev	Dilmurod	Raxmatullaevich	5 dekabr 1985 yil
16595	Zokirov	Anvar	Rashidovich	15 may 1984 yil

Ma`lumot turlari

1. Numeric Types
2. Character Types
3. Binary Data Types
4. Boolean Type
5. Date/time Types
6. Geometric Types
7. Network Address Types
8. UUID
9. XML
10. JSON

Sonli toifalar

Имя	Размер	Диапазон
smallint	2 byte	-32768 .. +32767
integer	4 byte	-2147483648 .. +2147483647
bigint	8 byte	-9223372036854775808 .. 9223372036854775807
decimal	O’zgaruvchan	butun qismi 131072 va kasr qismi 16383 raqamgacha
numeric	O’zgaruvchan	butun qismi 131072 va kasr qismi 16383 raqamgacha
real	4 byte	kasr qismi 6 raqamgacha
double precision	8 byte	kasr qismi 15 raqamgacha
smallserial	2 byte	1 .. 32767
serial	4 byte	1 .. 2147483647
bigserial	8 byte	1 .. 9223372036854775807

Belgili toifalar

character varying(n), varchar(n)	Uzunligi o’zgaruvchan va uni cheklash mumkin
character(n), char(n)	Uzunligi o’zgarmas (fiksirlangan)
text	Uzunligi o’zgaruvchan va cheklanmagan

Ikkilik(binary) toifalar

bytea	Uzunligi o’zgaruvchan ikkilik(binary) qator
bit [(n)]	O’zgarmas uzunlikdagi ikkilik(binary) qator
bit varying [(n)]	Uzunligi o’zgaruvchan ikkilik(binary) qator va uning uzunligini cheklash mumkin

Boolean toifasi

boolean	TRUE yoki FALSE

Sana va vaqt toifalari

Nomi	**Hajmi**
timestamp [(*p*)] [without time zone]	8 byte
timestamp [(*p*)] with time zone	8 byte
date	4 byte
time [(*p*)] [without time zone]	8 byte
time [(*p*)] with time zone	12 byte
interval [*fields*] [(*p*)]	16 byte

Geometrik toifalar

point	Nuqta (x,y)
lseg	Kesma((x1,y1),(x2,y2))
box	To’rtburchak ((x1,y1),(x2,y2))
path	Ko’pburchak ((x1,y1),...)
path	Siniq kesma[(x1,y1),...]
circle	Aylana (x,y),r (markaz varadius)

PDP ACADEMY

Boshqa toifalar

cidr
inet
macaddr
uuid
xml
json

Constraints
(cheklovlar)

Cheklovlar - bu jadvaldagi ma`lumotlar ustunlarida qo`llaniladigan qoidalar. Ular ma`lumotlar bazasiga noto`g`ri ma`lumotlar kiritilishining oldini olish uchun ishlatiladi. Bu ma`lumotlar bazasidagi ma`lumotlarning aniqligi va ishonchliligini ta`minlaydi.

Cheklovlar ustun darajasi yoki jadval darajasi bo`lishi mumkin. Ustun darajasidagi cheklovlar faqat bitta ustunga qo`llaniladi, jadval darajasidagi cheklovlar esa butun jadvalga qo`llaniladi. Ustun uchun ma`lumotlar turini belgilashning o`zi cheklovdir. Masalan, DATE turidagi ustun ustunni haqiqiy sanalar bilan cheklaydi.

Quyida PostgreSQL-da mavjud bo`lgan keng tarqalgan cheklovlar mavjud:

- **NOT NULL Constraint** - ustun NULL qiymatiga ega bo`lmasligini ta`minlaydi.
- **UNIQUE Constraint -** ustundagi barcha qiymatlar takrorlanmas bo`lishini ta`minlaydi.
- **PRIMARY Key –** ma`lumotlar bazasi jadvalidagi har bir qatorni/yozuvni o`ziga xos tarzda aniqlaydi.
- **FOREIGN Key -** boshqa jadvallardagi ustunlar asosidagi ma`lumotlarni cheklaydi.
- **CHECK Constraint -** CHECK cheklovi ustundagi barcha qiymatlar ma`lum shartlarga javob berishini ta`minlaydi.

NOT NULL cheklovi

Default holatda, ustun NULL qiymatlarga ega bo`lishi mumkin. Agar siz ustunning NULL qiymatiga ega bo`lishini xohlamasangiz, u holda ushbu ustun uchun NULLga ruxsat berilmaganligini ko`rsatib, ushbu ustunda bunday cheklovni belgilashingiz kerak. NOT NULL cheklovi har doim ustun cheklovi sifatida yoziladi.

Masalan, quyidagi PostgreSQL bayonoti COMPANY deb nomlangan yangi jadval yaratadi va beshta ustun qo`shadi, ulardan uchtasi ID va NAME va AGE NULL qiymatlarni qabul qilmaslikni belgilaydi -

```
CREATE TABLE COMPANY(
ID INT PRIMARY KEY NOT NULL,
NAME TEXT NOT NULL,
AGE INT NOT NULL,
ADDRESS CHAR(50),
SALARY REAL );
```

UNIQUE cheklovi

UNIQUE cheklovi ikkita yozuvning ma`lum bir ustunda bir xil qiymatlarga ega bo`lishiga yo`l qo`ymaydi. Masalan, COMPANY jadvalida siz ikki yoki undan ortiq odamning yoshi bir xil bo`lishining oldini olishni xohlashingiz mumkin.

Masalan, quyidagi PostgreSQL bayonoti COMPANY deb nomlangan yangi jadval yaratadi va beshta ustun qo`shadi. Bu yerda AGE ustuni UNIQUE qilib o`rnatiladi, shuning uchun siz bir xil yoshdagi ikkita yozuvni jadvalga qo`sha olmaysiz-

```
CREATE TABLE COMPANY(
ID INT PRIMARY KEY NOT NULL,
NAME TEXT NOT NULL,
AGE INT NOT NULL UNIQUE,
ADDRESS CHAR(50),
SALARY REAL DEFAULT 50000.00 );
```

PRIMARY KEY cheklovi

PRIMARY KEY cheklovi ma`lumotlar bazasi jadvalidagi har bir yozuvni noyob tarzda aniqlaydi. Jadvalda ko'proq UNIKAL ustunlar bo`lishi mumkin, lekin faqat bitta asosiy kalit. Ma`lumotlar bazasi jadvallarini loyihalashda primary keylar muhim ahamiyatga ega. Primary keylar noyob identifikatorlardir.

Biz ulardan jadval qatorlariga murojaat qilish uchun foydalanamiz. Primary keylar jadvallar o`rtasida munosabatlarni yaratishda boshqa jadvallarda tashqi kalitlarga aylanadi. Primary key - jadvaldagi maydon bo`lib, u ma`lumotlar bazasi jadvalidagi har bir qatorni/yozuvni yagona aniqlaydi. Birlamchi kalitlar noyob qiymatlarni o`z ichiga olishi kerak. Birlamchi kalit ustunida NULL qiymatlari bo'lishi mumkin emas.

Jadvalda faqat bitta primary key bo`lishi mumkin.

Masalan:

Siz allaqachon yuqorida birlamchi kalit sifatida ID-ga ega COMAPNY jadvalini yaratgan turli misollarni ko'rgansiz -

```
CREATE TABLE COMPANY(
ID INT PRIMARY KEY NOT NULL,
NAME TEXT NOT NULL,
AGE INT NOT NULL,
ADDRESS CHAR(50),
SALARY REAL );
```

FOREIGN KEY cheklovi

Foreign key cheklovi ustundagi (yoki ustunlar guruhidagi) qiymatlar boshqa jadvalning qaysidir qatorida ko`rsatilgan qiymatlarga mos kelishi kerakligini bildiradi. Bu ikkita bog`liq jadval o`rtasidagi mos yozuvlar yaxlitligini saqlaydi.

Masalan, quyidagi PostgreSQL bayonoti COMPANY deb nomlangan yangi jadval yaratadi va beshta ustun qo`shadi.

```
CREATE TABLE COMPANY(
ID INT PRIMARY KEY NOT NULL,
NAME TEXT NOT NULL,
AGE INT NOT NULL,
ADDRESS CHAR(50),
SALARY REAL );
```

Quyidagi PostgreSQL bayonoti uchta ustun qo`shadigan DEPARTMENT deb nomlangan yang jadvalni yaratadi. EMP_ID ustuni foreign key bo`lib, COMPANY jadvalining ID maydoniga havola qiladi.

```
CREATE TABLE DEPARTMENT(
ID INT PRIMARY KEY NOT NULL,
DEPT CHAR(50) NOT NULL,
EMP_ID INT references COMPANY(ID) );
```

CHECK cheklovi

CHECK cheklovi yozuvga kiritilayotgan qiymatni tekshirish shartini beradi. Agar shart noto`g`ri deb baholansa, yozuv cheklovni buzadi va jadvalga kiritilmaydi.

Masalan, quyidagi PostgreSQL bayonoti COMPANY deb nomlangan yangi jadval yaratadi va beshta ustun qo`shadi. Bu yerda biz CHECK SALARY ustunini qo`shamiz, shunda sizda hech qanday SALARY nolga teng bo`lmaydi.

```
CREATE TABLE COMPANY(
ID INT PRIMARY KEY NOT NULL,
NAME TEXT NOT NULL,
AGE INT NOT NULL,
ADDRESS CHAR(50),
SALARY REAL CHECK(SALARY > 0) );
```

PDP ACADEMY

E`TIBORINGIZ UCHUN RAHMAT!