به نام خدا

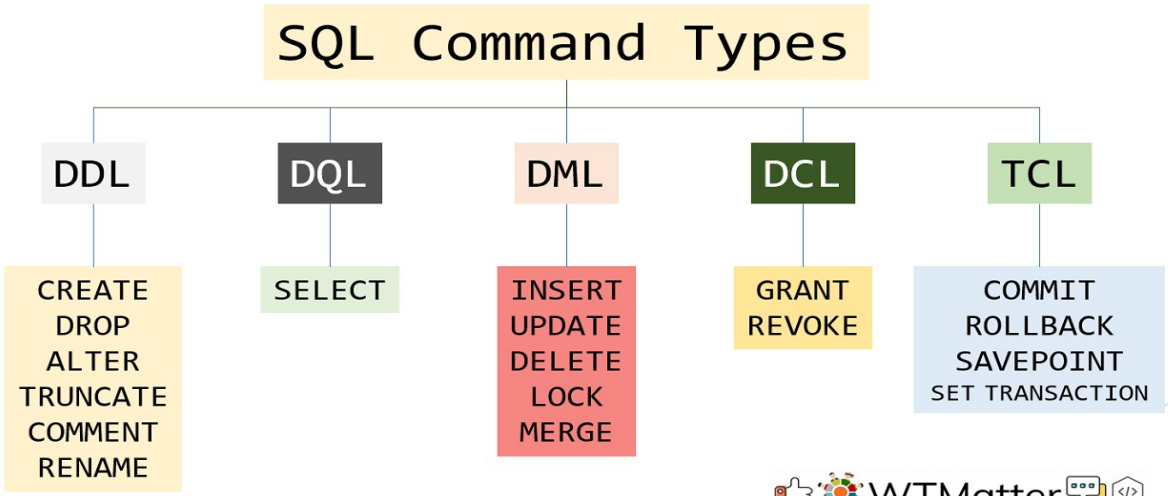
آزمایشگاه پایگاه داده

جلسه دوم

کار با جداول و کلیدها و نوشتن پرس‌وجوهای ساده

انواع دستورات در SQL

- Data Definition Language : DDL
- Data Manipulation Language : DML
- Data Query Language : DQL
- Data Control Language : DCL
- Transaction Control Language : TCL



برخی از دستورات ساده در PosgtreSQL

- **ایجاد و حذف پایگاه‌داده**
- `CREATE DATABASE DB_NAME;`
- برای حذف پایگاه‌داده مدنظر :
- `DROP DATABASE database_name;`
- `DROP DATABASE IF EXISTS some_database;`

برخی از دستورات ساده در PosgtreSQL

- برای ایجاد جدول جدید

```
CREATE [IF NOT EXISTS] TABLE table_name (
    column_name TYPE [column_constraint],
    [table_constraint,]
);
```

- مثال: ایجاد جدول معلمان با کلید اصلی id با نوع int و ستون های ویژگی First_name و last_name و Subject و Grade_level

```
CREATE TABLE teachers (
    id INT PRIMARY KEY,
    first_name VARCHAR,
    last_name VARCHAR,
    subject VARCHAR,
    grade_level int
);
```

برخی از دستورات ساده در PosgtreSQL

- برای حذف جدول

```
1. DROP TABLE table_name;
```

- اگر جدولی که می‌خواهید حذف کنید کلید خارجی داشته باشد

```
DROP TABLE table_name CASCADE;
```

- برای اضافه یا حذف کردن ستون و یا ایجاد تغییر در ستون و ساختار یک جدولی که قبلا ایجاد کردیم

```
ALTER TABLE table_name action;
ALTER TABLE table_name
ADD COLUMN column_name datatype column_constraint;
ALTER TABLE table_name
DROP COLUMN column_name ;
DROP COLUMN column_name ;
```

برخی از دستورات ساده در PosgtreSQL

- برای تغییرنام ستونی که قبلا ایجاد شده

```
ALTER TABLE table_name
RENAME COLUMN column_name
TO new_column_name;
```

- برای تغییر یک گزینه پیشفرض در ستون جدول و حذف قبلی

```
DROP TABLE table_name CASCADE;
```

- برای اضافه یا حذف کردن ستون و یا ایجاد تغییر در ستون و ساختار یک جدولی که قبلا ایجاد کردیم

```
ALTER TABLE table_name
ALTER COLUMN column_name
[SET DEFAULT value | DROP DEFAULT];
```

برخی از دستورات ساده در PosgtreSQL

- برای اضافه کردن شرایط و ویژگی:

```
ALTER TABLE table_name
ADD CONSTRAINT constraint_name constraint_definition;
```

- برای تغییر نام یک جدول به نام دیگر:

```
ALTER TABLE table_name
RENAME TO new_table_name;
```

- برای تغییر نام ستون به یک نام دیگر:

```
ALTER TABLE table_name
RENAME COLUMN title TO table_name_title;
```

- برای ساختن مقدار خالی برای پیشفرض برای ستون یک ستون

```
ALTER TABLE tabel_name
ALTER COLUMN new_tabel_name
SET DEFAULT '_blank';
```

تعریف کلید اصلی و خارجی و استفاده از آن‌ها

- کلید اصلی (PK) در هر جدول، یک ستون یا گروهی از ستون هاست که برای مشخص کردن یک سطر به صورت یکتا استفاده می‌شود

```
CREATE TABLE table (
    column_1 datatype PRIMARY KEY
    column_2 data_type
    ...
);
CREATE TABLE table (
    column_1 data_type ,
    column_2 data_type,
    ...
    PRIMARY KEY (column_1, column_2)
);
```

تعریف کلید اصلی و خارجی و استفاده از آن‌ها

- اگر میخواهید در جدول های ساخته شده قبلی خود یک ستون دیگررا هم به PK اضافه کنید:

```
ALTER TABLE table_name ADD PRIMARY KEY (column_1, column_2);
```

- اگر قبلا برای جدولتان PK نداشته‌اید با دستور زیر PK بگذارید:

```
ALTER TABLE table_name
ADD PRIMARY KEY (table_name_pk);
```

- اگر می‌خواهید یک ستون شمارنده در جدولتان داشته باشید، فیلد AUTO_INCREMENT برای شما این کار را میکند

```
CREATE TABLE table_name(
    ID int NOT NULL AUTO_INCREMENT,
    ...
    PRIMARY KEY (ID)
    );
```

- می‌توانید با دستور زیر عدد شروع شمارش را تغییر دهید:

```
ALTER TABLE Persons AUTO_INCREMENT=100;
```

- اگر میخواهید ویژگی Auto_increment را تغییر دهید از IDENTITY به صورت زیر استفاده کنید:

```
ID int IDENTITY(1,1)  PRIMARY KEY;
```

تعریف کلید اصلی و خارجی و استفاده از آن‌ها

- کلید خارجی(FK) در هر جدول به یک یا چند ستون گفته می‌شود که به یک PK از یک جدول دیگر ارجاع می‌دهد.
- جدولی که شامل کلید خارجی است، جدول ارجاع یا جدول فرزند نامیده می شود. و جداول ارجاع شده توسط کلید خارجی ، جدول ارجاع شده یا جدول والدین نامیده می شود. یک جدول باتوجه به روابط آن با جداول دیگر می تواند چندین کلید خارجی داشته باشد که با Constraints شرایط آن را مشخص می‌کنیم:

- *CONSTRAINT [fk_name]*
- *FOREIGN KEY(fk_columns)*
- *REFERENCES parent_table(parent_key_columns)*
- *ON DELETE [delete_action]*
- *ON UPDATE [update_action]*

تعریف کلید اصلی و خارجی و استفاده از آن‌ها

- شرایطی که برای کلید خارجی می‌توان انجام داد:
  - SET NULL
  - SET DEFAULT
  - RESTRICT
  - NO ACTION
  - CASCADE

تعریف کلید اصلی و خارجی و استفاده از آن‌ها

- SET NULL : در زمان پاک کردن به صورت آبشاری به صورت پیشفرض کلید خارجی ستون در سطری که ارجاع به فرزند در زمان ارجاع دادن به آن سطر توسط جدول والدین پاک شده راNULL می‌کند.
- SET DEFAULT : مقدار را روی پیشفرض تنظیم کند.
- RESTRICT و NO ACTION : عملیات حذف و یا بروزرسانی را رد میکند.
- CASCADE: وقتی که می‌خواهیم تمام سطرهای جداول فرزند به تبع سطهای والدین به صورت آبشاری حذف یا بروزرسانی کنیم از این عبارت استفاده می‌کنیم.

پرس‌وجوهای ساده

- برای تعریف قید و ویژگی از دستور زیر استفاده می‌کنیم

```
CREATE TABLE table_name (
    column1 datatype constraint,
    column2 datatype constraint,
    column3 datatype constraint,
    ....
);
```

پرس‌وجوهای ساده

- NOT NULL : اجازه نمی‌دهد که هیچ ستونی در آن جدول شامل مقدار خالی/پوچ باشد.
- UNIQUE: اجازه نمی‌دهد که هیچ خانه‌ای در یک ستون مقدار تکراری داشته باشد.
- PRIMARY KEY: این شرط یعنی باید هردو شرط قبلی یعنی NOTNULLو UNIQUE را داشته باشد تا به PK در جدول تبدیل شود.
- FOREIGN KEY: باید در جدول ارجاع داده شده یک صفت یکتا در سطر/خانه باشد.
- CHECK: در وارد کردن مقادیر ستون‌ها یک شرطی را چک می‌کند.
- DEFAULT: برای زمانی که مقداری را بعنوان ورودی مقداردهی ستون نمی‌دهیم، یک مقدار پیشفرض برای تعریف می‌کند.
- INDEX: برای ساختن و بازیابی داده از پایگاه‌داده با سرعت زیاد

دستورات پایه‌ای و تعریف View

- SELECT : این عبارت بیشترین استفاده و همزمان بیشترین پیچیدگی و تنوع را درمیان عبارات پرس‌وجو را دارد. به همین دلیل برای راحتی در یادگیری می‌تواند به پرس‌وجوهای کوچکتر بدل شود.
- عباراتی که می‌توان در ساختار SELECT استفاده کرد:
  - DISTINCT
  - ORDER BY
  - WHERE
  - GROUP BY
  - UNION
  - INTERSECT
  - JOIN
  - EXCEPT
  - HAVING

```
SELECT  select_list  FROM   table_name
```

دستورات پایه‌ای و تعریف View

- دستور INSERT INTO برای اضافه کردن مقادیر در ستون‌های خاص از جداول است.

```
INSERT INTO table_name (column1, column2, column3,...)
VALUES (value1, value2, value3, ...);
```

- اگر برای تمام ستون‌ها میخواهید مقادیر اضافه کنید:

```
INSERT INTO table_name
VALUES (value1, value2, value3, ...);
```

- UPDATE:
- برای بروزرسانی ساختار در جدول یا ستون یا یک خانه‌ی خاص به‌کار میرود.

```
 UPDATE table_name
SET column1 = value1, column2 = value2, ...
WHERE condition
```

دستورات پایه‌ای و تعریف View

- VIEW : چیزی بیش از یک عبارت SQL نیست که در یک پایگاه داده با یک اسم مرتبط ذخیره می شود.
- یک VIEW در واقع نمایش مجازی یک جدول از پیش تعریف شده در SQL است.یک VIEW می تواند شامل تمام ردیف های یک جدول باشد یا سطرها را از یک جدول انتخاب کنید.
- VIEW را می توان از یک یا بسیاری از جداول ایجاد کرد که بستگی به سؤال نوشتاری SQL برای ایجاد یک VIEW دارد
- VIEW ها را می‌توانید ایجاد یا حذف یا شامل تغییر کنید. همچنین مانند SQL از JOINیا WHERE برای آن استفاده کنید
- VIEW ، که نوعی جداول مجازی است به کاربران امکان می دهد موارد زیر را انجام دهند:
  - ساختار داده ها به روشی که کاربران یا طبقات کاربران به صورت طبیعی یا بصری بیابند.
  - دسترسی به داده ها را به گونه ای محدود کنید که کاربر بتواند دقیقاً مورد نیاز خود را ببیند و (گاهی) دقیقاً مورد نیاز خود را تغییر دهد.
  - خلاصه داده ها از جداول مختلف که می تواند برای تولید گزارش استفاده شود.

```
CREATE VIEW view_name AS
SELECT column1, column2.....
FROM table_name
WHERE [condition];
```

انواع عملگرها در SQL

- عملگرهای حسابی(Arithmetic operators)
- عملگرهای مقایسه‌ای(Comparison operators)
- عملگرهای منطقی(Logical operators)
- عملگرهای نفی(used to negate conditions)

انواع عملگرها در SQL

عملگرهای منطقی	عملگرهای مقایسه‌ای	عملگرهای حسابی
ALL	=	+(جمع)
AND	!=	-(تفریق)
ANY	<>	*(ضرب)
BETWEEN	>	/(تقسیم)
EXISTS	<	%(باقیمانده)
IN	>=	
LIKE	<=	
NOT	!<	
OR	!>	
IS NULL		
UNIQUE		

تمرین

- ۱-جداول ساخته شده جلسه قبل را با دستور ALTER ویرایش کنید و یک دستور دلخواه اضافه کنید.
- ۲-یک مثال برای یک مجموعه و چگونگی تعریف کلید اصلی و کلید خارجی شرح دهید و بگویید چرا این ویژگی را برای آن مجموعه مناسب کلید اصلی می‌دانید.
- ۳-سه دستور ساده با استفاده از INSERT و SELECT و UPDATE بزنید و بفرستید.
- ۴-یک VIEW از چند ستون از جدول ساخته شده‌ی خود بسازید و آن را نمایش دهید.
- ۵-به انتخاب خود از هرکدام از انواع عملگرها(حسابی،منطقی،مقایسه‌ای) ۲ عملگر را انتخاب و کاربرد آن را به صورت خلاصه توضیح دهید.