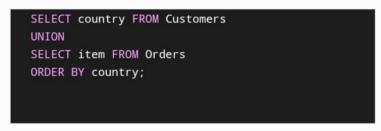
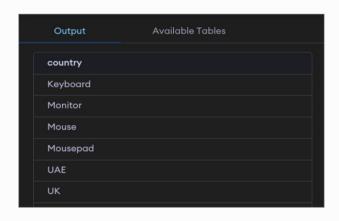
عملگر UNION دو یا چند دستور SELECT را با یکدیگر ترکیب می کند

SELECT age FROM Customers UNION SELECT item FROM Orders ORDER BY item





دستور Group by

عبارت GROUP BY در ترکیب با توابع aggregate استفاده می شود تا نتیجه را بر اساس یک یا چند ستون گروه بندی کند.

SELECT COUNT(customer_id), country FROM Customers GROUP BY country;





دستور CREATE DATABASE برای ایجاد یک پایگاه داده استفاده می شود.

CREATE DATABASE mr;

دستور DROP DATABASE برای حذف یک پایگاه داده استفاده می شود.

DROP DATABASE mr;

دستور CREATE TABLE برای ایجاد جدول در یک پایگاه داده استفاده می شود.

```
CREATE TABLE Persons (
PersonID int,
LastName varchar(255),
FirstName varchar(255),
Address varchar(255),
City varchar(255)
```

CREATE TABLE Persons (
PersonID int,	
LastName varchar(255),	-
<pre>FirstName varchar(255),</pre>	
Address varchar(255),	F
City varchar(255)	-
);	-
	€



دستور DROP TABLE برای حذف یک جدول در یک پایگاه داده استفاده می شود.

DROP TABLE Orders;

SQL query successfully executed. However, the result set is empty.

دستور ALTER TABLE برای اضافه کردن(Add) ، حذف کردن(Drop Column)، یا تغییر ستون ها (Alter Column)در جدول موجود استفاده می شود.

(ADD)

ALTER TABLE Customers ADD Email varchar(255);





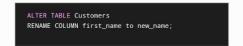
(DROP)

ALTER TABLE Customers DROP COLUMN age;





(RENAME) ALTER TABLE *Customers*RENAME COLUMN *old name* to *new name*;

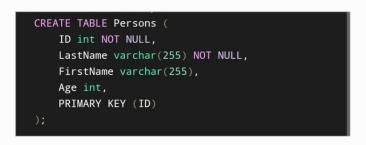


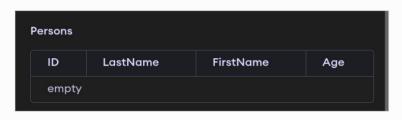


PRIMARY KEY

هر رکورد در پایگاه داده را به طور منحصر به فردی مشخص می کند.

```
CREATE TABLE Persons (
ID int NOT NULL PRIMARY KEY,
LastName varchar(255) NOT NULL,
FirstName varchar(255),
Age int
);
```



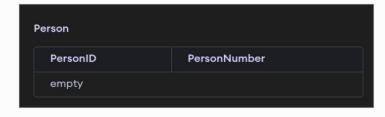


یک FOREIGN KEY (کلید خارجی) در یک جدول اشاره به PRIMARY KEY (کلید اصلی) در جدول دیگر دارد.

```
CREATE TABLE Person (
PersonID int NOT NULL,
PersonNumber int NOT NULL,
PRIMARY KEY (PersonID),
FOREIGN KEY (PersonID) REFERENCES Orders(amount)

nt)
);
```

```
CREATE TABLE Person (
PersonID int NOT NULL,
PersonNumber int NOT NULL,
PRIMARY KEY (PersonID),
FOREIGN KEY (PersonID) REFERENCES
Orders(amount)
);
```



Comments

برای توضیح بخش هایی از دستورات SQL یا جلوگیری از اجرای دستورات SQL استفاده می شود.

--Select all:

SELECT * FROM Customers;

```
--Select all:
--SELECT * FROM Customers;
```

SELECT * FROM Customers -- WHERE City='Berlin';

```
--Select all:
SELECT * FROM Customers;
```

دستور unique

محدودیت UNIQUE تعیین می کند هر رکوردی در جدول یکتاست.

```
CREATE TABLE Persons (
ID int NOT NULL UNIQUE,
LastName varchar(255) NOT NULL,
FirstName varchar(255),
Age int
);
```

```
CREATE TABLE Persons (
    ID int NOT NULL UNIQUE,
    LastName varchar(255) NOT NULL,
    FirstName varchar(255),
    Age int
);
INSERT INTO Persons(ID, LastName, FirstName, Age)
VALUES (1, 'Rasooli', 'Mohammad Javad', 21)
INSERT INTO Persons(ID, LastName, FirstName, Age)
VALUES (1, 'Rasooli', 'Javad', 21)
```

Output Available Tables

Error: UNIQUE constraint failed: Persons.ID

دستور NOT NULL یک ستون را مجبور می کند که مقدار خالی را قبول نکند.

```
CREATE TABLE myperson (
myID int NOT NULL,
myLastName varchar(255) NOT NULL,
myFirstName varchar(255) NOT NULL,
myAge int
);
```

```
CREATE TABLE myperson (

myID int NOT NULL,

myLastName varchar(255) NOT NULL,

myFirstName varchar(255) NOT NULL,

myAge int
);
```

