SQL - Foreign Key

A foreign key is a key used to link two tables together. This is sometimes also called as a referencing key.

A Foreign Key is a column or a combination of columns whose values match a Primary Key in a different table.

**The relationship between 2 tables matches the Primary Key in one of the tables with a Foreign Key in the second table.**

If a table has a primary key defined on any field(s), then you cannot have two records having the same value of that field(s).

Example

Consider the structure of the following two tables.

**CUSTOMERS table**

CREATE TABLE CUSTOMERS(

ID INT NOT NULL,

NAME VARCHAR (20) NOT NULL,

AGE INT NOT NULL,

ADDRESS CHAR (25) ,

SALARY DECIMAL (18, 2),

PRIMARY KEY (ID)

);

**ORDERS table**

CREATE TABLE ORDERS (

ID INT NOT NULL,

DATE DATETIME,

CUSTOMER\_ID INT,

AMOUNT double,

PRIMARY KEY (customer\_ID)

);

If the ORDERS table has already been created and the foreign key has not yet been set, the use the syntax for specifying a foreign key by altering a table.

ALTER TABLE ORDERS

ADD FOREIGN KEY (ID) REFERENCES CUSTOMERS (ID);

DROP a FOREIGN KEY Constraint

To drop a FOREIGN KEY constraint, use the following SQL syntax.

ALTER TABLE ORDERS

DROP FOREIGN KEY;

**SQL join two tables related by a single column primary key or foreign key pair using where clause**

Description

In this page, we are going to discuss the usage of two or more tables in a joining with single column PRIMARY KEY and FOREIGN KEY.

**Example:**

**Sample table: foods**

**| ITEM\_ID | ITEM\_NAME | ITEM\_UNIT | COMPANY\_ID |**

**+---------+--------------+-----------+------------+**

**| 1 | Chex Mix | Pcs | 16 |**

**| 6 | Cheez-It | Pcs | 15 |**

**| 2 | BN Biscuit | Pcs | 15 |**

**| 3 | Mighty Munch | Pcs | 17 |**

**| 4 | Pot Rice | Pcs | 15 |**

**| 5 | Jaffa Cakes | Pcs | 18 |**

**| 7 | Salt n Shake | Pcs | |**

**+---------+--------------+-----------+------------+**

**Sample table: company**

**------------+---------------+--------------+**

**| COMPANY\_ID | COMPANY\_NAME | COMPANY\_CITY |**

**+------------+---------------+--------------+**

**| 18 | Order All | Boston |**

**| 15 | Jack Hill Ltd | London |**

**| 16 | Akas Foods | Delhi |**

**| 17 | Foodies. | London |**

**| 19 | sip-n-Bite. | New York |**

**+------------+---------------+--------------+**

To get 'item\_name' and 'item\_unit' from 'foods' table and 'company\_name' and 'company\_city' form 'company' table after a joining with following conditions-

**1.** 'company\_id' is primary key in 'company' table,

**2.** 'company\_id' is foreign key in 'foods' table which is referencing to the, primary key of 'company' table,

**3.** 'company\_id' of 'company' and 'foods' must be same,

**4.** 'company\_city' of 'company' column must be 'London',

the following SQL statement can be used :

**SQL Code:**

**SELECT foods.item\_name,foods.item\_unit,**

**company.company\_name, company.company\_city**

**FROM foods ,company**

**WHERE foods.company\_id =company.company\_id**

**AND company.company\_city='London';**

Copy

Output

ITEM\_NAME ITEM\_UNIT COMPANY\_NAME COMPANY\_CITY

------------------------- ---------- ------------------------- -------------

Cheez-It Pcs Jack Hill Ltd London

BN Biscuit Pcs Jack Hill Ltd London

Mighty Munch Pcs Foodies. London

Pot Rice Pcs Jack Hill Ltd London