# **SQL FOREIGN KEY Constraint**

**SQL FOREIGN KEY Constraint:**

1. The FOREIGN KEY constraint is used to prevent actions that would destroy links between tables.
2. A FOREIGN KEY is a field (or collection of fields) in one table, that refers to the PRIMARY KEY in another table.
3. The table with the foreign key is called the child table, and the table with the primary key is called the referenced or parent table.
4. Look at the following two tables:  
   Persons Table:

|  |  |  |  |
| --- | --- | --- | --- |
| **PersonID** | **LastName** | **FirstName** | **Age** |
| 1 | Hansen | Ola | 30 |
| 2 | Svendson | Tove | 23 |
| 3 | Pettersen | Kari | 20 |

Orders Table:

|  |  |  |
| --- | --- | --- |
| **OrderID** | **OrderNumber** | **PersonID** |
| 1 | 77895 | 3 |
| 2 | 44678 | 3 |
| 3 | 22456 | 2 |
| 4 | 24562 | 1 |

Notice that the "PersonID" column in the "Orders" table points to the "PersonID" column in the "Persons" table.  
  
The "PersonID" column in the "Persons" table is the PRIMARY KEY in the "Persons" table.

The "PersonID" column in the "Orders" table is a FOREIGN KEY in the "Orders" table.  
  
The FOREIGN KEY constraint prevents invalid data from being inserted into the foreign key column, because it has to be one of the values contained in the parent table.

**SQL FOREIGN KEY on CREATE TABLE**

1. The following SQL creates a FOREIGN KEY on the "PersonID" column when the "Orders" table is created:  
   MySQL:

|  |
| --- |
| CREATE TABLE Orders (  OrderID int NOT NULL,  OrderNumber int NOT NULL,  PersonID int,  PRIMARY KEY (OrderID),  FOREIGN KEY (PersonID) REFERENCES Persons(PersonID)  ); |

SQL Server / Oracle / MS Access:

|  |
| --- |
| CREATE TABLE Orders (  OrderID int NOT NULL PRIMARY KEY,  OrderNumber int NOT NULL,  PersonID int FOREIGN KEY REFERENCES Persons(PersonID)  ); |

To allow naming of a FOREIGN KEY constraint, and for defining a FOREIGN KEY constraint on multiple columns, use the following SQL syntax:  
  
MySQL / SQL Server / Oracle / MS Access:

|  |
| --- |
| CREATE TABLE Orders (  OrderID int NOT NULL,  OrderNumber int NOT NULL,  PersonID int,  PRIMARY KEY (OrderID),  CONSTRAINT FK\_PersonOrder FOREIGN KEY (PersonID)  REFERENCES Persons(PersonID)  ); |

**SQL FOREIGN KEY on ALTER TABLE**

1. To create a FOREIGN KEY constraint on the "PersonID" column when the "Orders" table is already created, use the following SQL:MySQL / SQL Server / Oracle / MS Access:

|  |
| --- |
| ALTER TABLE Orders  ADD FOREIGN KEY (PersonID) REFERENCES Persons(PersonID); |

To allow naming of a FOREIGN KEY constraint, and for defining a FOREIGN KEY constraint on multiple columns, use the following SQL syntax:  
  
MySQL / SQL Server / Oracle / MS Access:

|  |
| --- |
| ALTER TABLE Orders  ADD CONSTRAINT FK\_PersonOrder  FOREIGN KEY (PersonID) REFERENCES Persons(PersonID); |

**DROP a FOREIGN KEY Constraint**

1. To drop a FOREIGN KEY constraint, use the following SQL:  
   MySQL:

|  |
| --- |
| ALTER TABLE Orders  DROP FOREIGN KEY FK\_PersonOrder; |

SQL Server / Oracle / MS Access:

|  |
| --- |
| ALTER TABLE Orders  DROP CONSTRAINT FK\_PersonOrder; |