A **Foreign Key** is a column (or a set of columns) in one table that is used to establish a link between the data in two tables. It acts as a constraint that ensures the values in one table correspond to values in another table, maintaining data integrity.

**Key Points:**

1. A **Foreign Key** in one table points to the **Primary Key** in another table.
2. It enforces referential integrity by ensuring that the value in the foreign key column matches an existing value in the referenced table.
3. Foreign keys help in establishing relationships between tables, such as **one-to-one**, **one-to-many**, and **many-to-many** relationships.

**Example of a Foreign Key**

Suppose we have two tables: Departments and Employees.

* **Departments** table:

| **DepartmentID (PK)** | **DepartmentName** |
| --- | --- |
| 1 | HR |
| 2 | IT |
| 3 | Finance |

* **Employees** table:

| **EmployeeID (PK)** | **Name** | **DepartmentID (FK)** |
| --- | --- | --- |
| 101 | Alice | 1 |
| 102 | Bob | 2 |
| 103 | Charlie | 3 |

Here, the DepartmentID column in the Employees table is a **foreign key** that references the DepartmentID in the Departments table. This ensures that each employee is associated with a valid department.

**SQL to Create Tables with Foreign Key**

1. **Creating the Departments Table (with Primary Key):**

CREATE TABLE Departments (

DepartmentID INT PRIMARY KEY,

DepartmentName VARCHAR(100)

);

1. **Creating the Employees Table (with Foreign Key):**

CREATE TABLE Employees (

EmployeeID INT PRIMARY KEY,

Name VARCHAR(100),

DepartmentID INT,

FOREIGN KEY (DepartmentID) REFERENCES Departments(DepartmentID)

);

In this example:

* The DepartmentID column in the Employees table is a foreign key.
* The DepartmentID in the Employees table refers to the DepartmentID in the Departments table, ensuring that each employee belongs to a valid department.

**Example: Inserting Data**

Now that the foreign key constraint is in place, let's try to insert some data:

1. **Inserting Data into the Departments Table:**

INSERT INTO Departments (DepartmentID, DepartmentName)

VALUES (1, 'HR'), (2, 'IT'), (3, 'Finance');

1. **Inserting Data into the Employees Table:**

INSERT INTO Employees (EmployeeID, Name, DepartmentID)

VALUES (101, 'Alice', 1), (102, 'Bob', 2), (103, 'Charlie', 3);

* This will work because 1, 2, and 3 are valid DepartmentIDs in the Departments table.

1. **Attempting to Insert Invalid Data:**

INSERT INTO Employees (EmployeeID, Name, DepartmentID)

VALUES (104, 'David', 5);

* This will **fail** because there is no DepartmentID with value 5 in the Departments table, violating the foreign key constraint.