

4.5 Primary and foreign key constraints

PRIMARY KEY constraint

A **constraint** is a rule that applies to table data. Constraints are specified in a CREATE TABLE statement or may be added to a preexisting table with an ALTER TABLE statement.

The **PRIMARY KEY** constraint in a CREATE TABLE statement names the table's primary key, the column(s) that uniquely identify each row.

PARTICIPATION ACTIVITY

4.5.1: Adding primary key constraints to tables.



- 1
- 2
- 3
- 4



2x speed

```
CREATE TABLE Employee (
    ID      SMALLINT UNSIGNED,
    Name    VARCHAR(60),
    Salary   DECIMAL(7, 2),
    PRIMARY KEY (ID)
);
```

Employee

ID	Name	Salary
2538	Lisa Ellison	45000
5384	Sam Snead	30400
6381	Maria Rodriguez	92300

```
CREATE TABLE Family (
    ID      SMALLINT UNSIGNED,
    Number  SMALLINT UNSIGNED,
    Relationship  VARCHAR(20),
    Name    VARCHAR(60),
    PRIMARY KEY(ID, Number)
);
```

Family

ID	Number	Relationship	Name
2538	1	Spouse	Henry Ellison
2538	2	Son	Edward Ellison
6381	1	Spouse	Jose Rodriguez
6381	2	Daughter	Gina Rodriguez
6381	3	Daughter	Clara Rodriguez

All rows added to the Family table must have a unique combination of ID and Number.

Captions

1. The CREATE TABLE statement uses the keywords PRIMARY KEY to indicate the ID column is the table's primary key.
2. All rows added to the Employee table must have a unique ID.
3. The PRIMARY KEY constraint identifies the ID and Number columns as the Family table's composite primary key.
4. All rows added to the Family table must have a unique combination of ID and Number.

PARTICIPATION ACTIVITY

4.5.2: PRIMARY KEY constraints.



Refer to the animation above.

- 1) Lisa, Sam, and Maria must have unique IDs and names.

True
 False

- 2) Only one column may be listed in a PRIMARY KEY constraint.

True
 False

- 3) Assuming the Family table has the five rows shown above, a new row with values (2538, 2, 'Daughter', 'Ella Ellison') may be added to the Family table.

True
 False

Correct

The ID column is the Employee table's primary key, so only the ID must be unique. Ex: An employee with ID 7654 and name Lisa Ellison may be added to the Employee table.

**Correct**

Multiple columns may be listed in the PRIMARY KEY constraint. In the animation above, two columns are listed to create the composite primary key ID and Number for the Family table.

**Correct**

The ID and Number are the Family table's composite primary key, so only rows with unique ID and Number values may be added. Edward Ellison already has the ID 2538 and Number 2, so Ella must use a different Number.

[Feedback?](#)

Auto-increment columns

A primary key on an ID column is commonly implemented as an auto-increment column. An **auto-increment column** is a column that is assigned an automatically incrementing value. Ex: A database system may assign an auto-incrementing column values 1, 2, 3, etc. for each row that is inserted into the table.

The **AUTO_INCREMENT** keyword defines an auto-increment column in MySQL.

Figure 4.5.1: ID is an AUTO_INCREMENT

COLUMN.

```
CREATE TABLE Employee (
    ID      SMALLINT UNSIGNED AUTO_INCREMENT,
    Name    VARCHAR(60),
    BirthDate DATE,
    Salary   DECIMAL(7,2),
    PRIMARY KEY (ID)
);
```

[Feedback?](#)**PARTICIPATION
ACTIVITY**

4.5.3: Auto-increment column.



Four employees were added to the Employee table above, which has an auto-incrementing ID column. Place the employee rows in the order in which the employees were most likely added to the Employee table.

**1, 'Susan Miller', '2000-10-01',
45300****1st****Correct**

The ID value 1 is likely assigned to the first employee added to the Employment table.

**2, 'Juan Lopez', '2001-12-11',
39200****2nd****Correct**

The ID value 2 usually follows 1.

3, 'Adib Rida', '1999-08-21', 52000**3rd****Correct**

The ID value 3 usually follows 2.

**4, 'Tye Nguyen', '2001-05-02',
49500****4th****Correct**

The ID value 4 usually follows 3.

Reset[Feedback?](#)

FOREIGN KEY constraint

A foreign key constraint is added to a CREATE TABLE statement with the **FOREIGN KEY** and **REFERENCES** keywords. The CREATE statement in the animation below indicates the foreign key ManagerID column refers to the primary key ID in the Employee table. When a foreign key constraint is specified, the database will not allow an insert or update that violates referential integrity. Ex: Inserting a row in Department with ManagerID 9999 is rejected if Employee ID 9999 does not exist.

PARTICIPATION
ACTIVITY

4.5.4: Foreign key constraint on the Department table.



- 1 2 3 ← ✓ 2x speed

Department			Employee			
● Code	Name	○ ManagerID	● ID	Name	BirthDate	Salary
44	Engineering	2538	2538	Lisa Ellison	1993-10-02	45000
82	Sales	6381	5384	Sam Snead	1995-03-15	30500
12	Marketing	9999 6381	6381	Maria Rodriguez	2001-12-21	92300
99	Technical support	7343	7343	Gary Smith	1984-09-22	85000

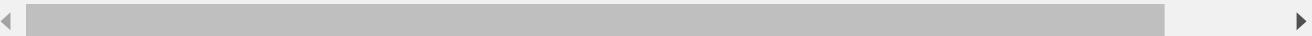
```
CREATE TABLE Department (
    Code      TINYINT UNSIGNED,
    Name      VARCHAR(20),
    ManagerID SMALLINT UNSIGNED,
    PRIMARY KEY (Code),
    FOREIGN KEY (ManagerID) REFERENCES Employee(ID)
);
```

When rows are added to Department, the ManagerID value must exist in Employee ID. ManagerID 9999 is rejected because 9999 does not exist in Employee ID.

Captions ^

1. The Employee table has primary key ID and several rows.
2. The Department table is created with a FOREIGN KEY constraint that REFERENCES the Employee ID column.
3. When rows are added to Department, the ManagerID value must exist in Employee ID. ManagerID 9999 is rejected because 9999 does not exist in Employee ID.

[Feedback?](#)



PARTICIPATION
ACTIVITY

4.5.5: Add primary and foreign key constraints.



The given SQL creates an Album and Song tables. The SHOW COLUMNS queries show information about the Album and Song table columns.

Modify the two CREATE TABLE statements. Add a primary key constraint to the Album table's ID column. Add a primary key constraint to the Song table's ID column, and add a foreign key constraint so the AlbumID column refers to the Album table's ID column.

```

1 -- Add a primary key
2 CREATE TABLE Album (
3   ID INT,
4   Title VARCHAR(60),
5   ReleaseYear INT
6 );
7
8 -- Add primary and foreign keys
9 CREATE TABLE Song (
10   ID INT,
11   Title VARCHAR(60),
12   Artist VARCHAR(60),
13   AlbumID INT
14 );
15
16 SHOW COLUMNS
17 FROM Album;
18
19 SHOW COLUMNS
20 FROM Song;
```

Run

Reset code

Feedback?

PARTICIPATION ACTIVITY

4.5.6: Foreign key constraint.



- 1) A FOREIGN KEY constraint must REFERENCE a primary key.

True

False

Correct



Foreign keys must refer to a primary key in the same table or another table.

- 2) Adding a FOREIGN KEY constraint to a table only affects inserting new rows into the table.

True

False

Correct



Inserting and updating rows are affected. Ex: Updating the ManagerID 2538 to 9999 is rejected if Employee ID 9999 does not exist.

ON DELETE and ON UPDATE actions

Actions can be specified on the FOREIGN KEY constraint with ON DELETE and ON UPDATE keywords:

- **ON DELETE** responds to an invalid primary key deletion. Ex: Deleting a primary key 1234 that is used in a foreign key.
- **ON UPDATE** responds to an invalid primary key update. Ex: Updating a primary key 1234 to 5555 when 1234 is used in a foreign key.

ON DELETE and ON UPDATE must be followed by a response:

- **RESTRICT** rejects an insert, update, or delete that violates referential integrity. RESTRICT is applied by default when no action is specified.
- **SET NULL** sets an invalid foreign key value to NULL.
- **SET DEFAULT** sets invalid foreign keys to a default primary key value.
- **CASCADE** propagates primary key changes to foreign keys. If a primary key is deleted, rows containing matching foreign keys are deleted. If a primary key is updated, matching foreign keys are updated to the same value.

PARTICIPATION
ACTIVITY

4.5.7: Foreign key constraints with ON DELETE and ON UPDATE actions.



1 2 3 ← ✓ 2x speed

Department			Employee			
Code	Name	ManagerID	ID	Name	BirthDate	Salary
44	Engineering	NULL	8754	Lisa Ellison	1993-10-02	45000
82	Sales	6381	5384	Sam Snead	1995-03-15	30500
12	Marketing	6381	6381	Maria Rodriguez	2001-12-21	92300
99	Technical support	7343	7343	Gary Smith	1984-09-22	85000

```
CREATE TABLE Department (
    Code      TINYINT UNSIGNED,
    Name      VARCHAR(20),
    ManagerID SMALLINT UNSIGNED,
    PRIMARY KEY (Code),
    FOREIGN KEY (ManagerID) REFERENCES Employee(ID)
        ON DELETE CASCADE
        ON UPDATE SET NULL
);
```

ON UPDATE SET NULL causes the database to set ManagerID 2538 to NULL when the Employee ID changed to 8754.

Captions ^

1. ManagerID is a foreign key that references the Employee ID column.
2. ON DELETE CASCADE causes the database to delete the row with ManagerID 7343 when the employee with ID 7343 is deleted.
3. ON UPDATE SET NULL causes the database to set ManagerID 2538 to NULL when the Employee ID 2538 is changed to 8754.

Feedback?**PARTICIPATION ACTIVITY**

4.5.8: ON DELETE and ON UPDATE keywords.



Refer to the Department and Employee tables and the CREATE TABLE statement below that created the Department table.

Department

Code	Name	ManagerID
44	Engineering	2538
82	Sales	6381
12	Marketing	6381
99	Technical support	NULL
49	Administration	7343

Employee

ID	Name	Salary
2538	Lisa Ellison	60000
5384	Sam Snead	30500
6381	Maria Rodriguez	92300
7343	Gary Smith	85000

```
CREATE TABLE Department (
    Code TINYINT UNSIGNED,
    Name VARCHAR(20),
    ManagerID SMALLINT UNSIGNED,
    PRIMARY KEY (Code),
    FOREIGN KEY (ManagerID) REFERENCES Employee(ID)
        ON DELETE SET NULL
        ON UPDATE CASCADE
);
```

What is the result of each operation?

- 1) Delete Lisa Ellison.

- Lisa Ellison is deleted.
- Lisa Ellison is deleted, and the Engineering

Correct

ON DELETE SET NULL sets the foreign key to NULL when the matching primary key is deleted.

- ManagerID is set to NULL. rejected.

2) Update Lisa Ellison's ID to 1000.

- Lisa Ellison's ID is set to 1000.
- Lisa Ellison's ID and the Engineering ManagerID are set to 1000.
- The update is rejected.

3) Update the Engineering ManagerID to 9999.

- The Engineering ManagerID is set to 9999.
- Lisa Ellison's ID and the Engineering ManagerID are set to 9999.
- The update is rejected.

4) Delete Engineering.

- Engineering is deleted.
- Engineering is deleted, and Lisa Ellison's ID is set to NULL.
- The delete is rejected.

Correct



ON UPDATE CASCADE updates the foreign key to the same value used to update the primary key.

Correct



ON UPDATE only applies to updating the primary key, not the foreign key. 9999 is an invalid foreign key since 9999 does not exist in Employee ID. The database does not allow invalid foreign keys.

Correct

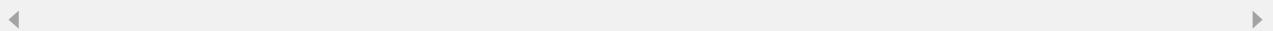


Deleting a foreign key has no effect on the primary key.

Feedback?

DROP TABLE with foreign key constraints

When a table with a foreign key constraint references a table's primary key, the table with primary key cannot be deleted with a `DROP TABLE` statement unless the table with the foreign key constraint is deleted first. Ex: The `Employee` table cannot be deleted without first deleting the `Department` table.



Exploring further:

- [CREATE TABLE Statement](#) from MySQL.com
- [Using FOREIGN KEY Constraints](#) from MySQL.com

**CHALLENGE
ACTIVITY**

4.5.1: Primary and foreign key constraints.



379958.2369558.qx3zqy7

Start



1



2



3



4

Country

• ISOCode2	ISOCode3	Population	Area
RU	RUS	144480000	6323100
GE	GEO	3730000	26800
JM	JAM	2930000	4200

Complete the statement that creates the table.

```
CREATE TABLE Country (
    ISOCode2 CHAR(2),
    ISOCode3 CHAR(3),
    Population INTEGER,
    Area DECIMAL(9, 2),
    /* Your code goes here */
);
```

1 2 3 4

Check Next

◀ ▶

Feedback?

How
was this   **Provide feedback**
section?