

Worksheet 4

Working with Foreign Keys

Before we begin, here are some example commands that are useful for using foreign keys. The worksheet exercises begin on the next page.

1. Create a foreign key when creating a table¹:

```
CREATE TABLE a (x INT, y INT, FOREIGN KEY (y)
  REFERENCES b(y) ON DELETE CASCADE ON UPDATE NO
  ACTION);
```

2. Find out the name of a foreign key:
`SHOW CREATE TABLE a;`

This shows something like:

```
| a      | CREATE TABLE `a` (
  `x` int(11) DEFAULT NULL,
  `y` int(11) DEFAULT NULL,
  KEY `y` (`y`),
  CONSTRAINT `a_ibfk_1` FOREIGN KEY (`y`) REFERENCES
  `b` (`y`) ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=latin1 |
```

Here, 'a_ibfk_1' is the name of the foreign key constraint.

3. Drop a foreign key from a table:

```
ALTER TABLE a DROP FOREIGN KEY a_ibfk_1;
```

4. Add a foreign key to a table that already exists. You will need to give it a name. Here, I use 'fk1' as the example:

```
ALTER TABLE a ADD CONSTRAINT fk1 FOREIGN KEY (y)
REFERENCES b(y) ON DELETE SET NULL ON UPDATE
CASCADE;
```

¹ In MySQL, creating the FOREIGN KEY constraint at the end of the CREATE TABLE command means that the constraint will be enforced properly. If you use REFERENCES at the same time you declare the field, it will not be enforced.

Worksheet Exercises

1. Create a new database, named 'pets'. We will use this database for the following exercises.
2. In this database, create the following tables:
 - a. A table named 'owners', with the following attributes:
 - **id**: Integer value to identify each owner. Should be the primary key.
 - **name**: Variable length character string, with a maximum length of 30. Stores the name of each owner.
 - b. A table named 'pet_types', with the following attributes:
 - **id**: Integer value to identify each pet type. Should be the primary key.
 - **name**: Variable length character string, with a maximum length of 10. Stores the name of each type of pet.
 - c. A table named 'pets', with the following attributes and constraints:

Attributes:

 - **id**: Integer value to identify each pet. Should be the primary key.
 - **name**: Variable length character string to store the pet's name.
 - **type**: Integer value that refers to the pet's type.
 - **owner**: Integer value that refers to the pet's owner.
 - **dob**: Date field to store the pet's date of birth.

Constraints:

 - The 'owner' field is a foreign key that refers to the 'id' field in the 'owners' table.
 - The 'type' field is a foreign key that refers to the 'id' field in the 'pet_types' table.
3. Download the file "inserts.sql" from Moodle. This contains the queries to insert data into this database. You can run these queries by using the .sql file in the way that you have used before to import databases, or you can copy/paste the queries into your MySQL terminal window.
4. Insert the following data into the 'pets' table:

11, 'Lucy', 1, 6, '2016-04-08'

What happens? Why?
5. In the 'pets' table, remove the foreign key constraint for 'type' and replace it with another one that uses ON DELETE NO ACTION.

Now try to delete the 'budgie' row from 'pet_types'. What happens? Why?
5. Repeat question 5, but use ON DELETE CASCADE.

6. In the 'pets' table, remove the foreign key constraint for 'owner' and replace it with another one that uses ON UPDATE CASCADE ON DELETE SET NULL.

Now try to delete 'Henry' row from 'owners'. What happens? Why?

Now try to change the id of 'Sean' to be 7. What happens? Why?

7. Write SQL queries to find the following information:
 - a. List the names of all pets born in the year 2013.
 - b. Name all the owners who have no pets.
 - c. Name all the owners who own a dog.
 - d. Add a new pet type: "elephant".
 - e. Add a new pet: an elephant named 'tiny' who was born today and is owned by Eleni.
 - f. Add the correct type of foreign key constraint so that whenever an owner is deleted from the database, all of their pets are also deleted.