MySQL – CREATE, DESCRIBE, INSERT, and, SELECT

• Consider the following schema. Primary keys are underlined. Foreign keys are in red color.

- employee (emp-id, frst-name, last-name, address, manager-id)
    - manager-id is referencing to employee (emp-id)
- team (team-id, name, manager-id, project-working-on)
    - manager-id is referencing to employee (emp-id),
    - project-working-on is referencing to project (project-id).
- employee-team (emp-id, team-id)
    - emp-id is referencing to employee (emp-id)
    - team-id is referencing to team(team-id).
- project (project-id, name, start-date, fnish-date, produced-product)
    - produced-product is referencing to product (product-id).
- product (product-id, name, cost)
- customer (cust-id, frst-name, last-name, address, phone)
- customer-product (cust-id, product-id)
    - cust-id is referencing to customer (cust-id)
    - product-id is referencing to product (product-id)

• Perform the following tasks. When creating tables remember to follow an order: first create the parent then the child tables.

1. Create and run DDL statements for above schema. Use the lowercase-dash naming convention when creating table and columns e.g. emp-id, first-name, address, etc. Choose an appropriate data type for each field. Remember setting the primary and foreign keys. While creating the foreign keys set the following referential actions:
    a. When defining a foreign key on a normal table, use the options ON DELETE SET NULL and ON UPDATE CASCADE.
    b. When defining a foreign key on a junction table, use the options ON DELETE CASCADE and ON UPDATE CASCADE.
2. Using the DESCRIBE table statement to show the description of each table created in above step e.g. DESCRIBE TABLE employee;
3. Using the INSERT statement (today's lecture), insert at least 5 rows in each table. When inserting data, try incorrect entries (into the foreign key columns) and see what error you get when violating the referential integrity constraint.
4. Using the SELECT statement (today's lecture), write one SELECT query for each table. Use different styles of SELECT statement.
5. Using the DELETE statement (today's lecture), delete the middle row from each table.

- What do I submit?

  – **Create a new word document and save it with your name and student number**. Take **screen shot after executing each SQL statement.** Place all the screen shots in the document (no explanation needed for this lab, include only the screen shots). Submit the pdf document on moodle.

PART 1 Create and Run DDL statements schema of employee, customer, product, project, team, employee_team and customer_product.

- Created and using Database Company.

```
mysql> CREATE DATABASE company;
Query OK, 1 row affected (0.00 sec)

mysql> USE company;
Database changed
```

- employee (emp-id, frst-name, last-name, address, manager-id)
    - manager-id is referencing to employee (emp-id)

```
mysql> CREATE TABLE employee (
    -> emp_id INT NOT NULL
    -> ,frst_name VARCHAR(25)
    -> ,last_name VARCHAR(25)
    -> ,address VARCHAR(255)
    -> ,manager_id INT
    -> ,CONSTRAINT emp_id PRIMARY KEY(emp_id)
    -> ,CONSTRAINT manager_id FOREIGN KEY(manager_id)
    -> REFERENCES employee(emp_id)
    -> ON DELETE SET NULL
    -> ON UPDATE CASCADE
    -> );
Query OK, 0 rows affected (0.02 sec)
```

- customer (cust-id, frst-name, last-name, address, phone)

```
mysql> CREATE TABLE customer (
    -> cust_id INT NOT NULL
    -> ,frst_name VARCHAR(25)
    -> ,last_name VARCHAR(25)
    -> ,address VARCHAR(255)
    -> ,phone VARCHAR(15)
    -> ,CONSTRAINT cust_id PRIMARY KEY(cust_id)
    -> );
Query OK, 0 rows affected (0.02 sec)
```

- product (product-id, name, cost)

```
mysql> CREATE TABLE product (
    -> product_id INT NOT NULL
    -> , name VARCHAR(25)
    -> , cost INT
    -> , CONSTRAINT product_id PRIMARY KEY(product_id)
    -> );
Query OK, 0 rows affected (0.02 sec)
```

- project (project-id, name, start-date, fnish-date, produced-product)
    - produced-product is referencing to product (product-id).

```
mysql> CREATE TABLE project (
    -> project_id INT NOT NULL
    -> , name VARCHAR(25)
    -> , start_date DATE
    -> , fnish_date DATE
    -> , produced_product INT
    -> , CONSTRAINT product_id PRIMARY KEY(project_id)
    -> , CONSTRAINT produced_product FOREIGN KEY(produced_product)
    -> REFERENCES product (product_id)
    -> ON DELETE SET NULL
    -> ON UPDATE CASCADE
    -> );
Query OK, 0 rows affected (0.02 sec)
```

- team (team-id, name, manager-id, project-working-on)
    - manager-id is referencing to employee (emp-id),
    - project-working-on is referencing to project (project-id).

```
mysql> CREATE TABLE team (
    -> team_id INT NOT NULL
    -> ,name VARCHAR(25)
    -> ,manager_id INT
    -> ,project_working_on INT
    -> ,CONSTRAINT team_id PRIMARY KEY(team_id)
    -> ,FOREIGN KEY(manager_id)
    -> REFERENCES employee (emp_id)
    -> ON DELETE SET NULL
    -> ON UPDATE CASCADE
    -> ,FOREIGN KEY(project_working_on)
    -> REFERENCES project (project_id)
    -> ON DELETE SET NULL
    -> ON UPDATE CASCADE
    -> );
Query OK, 0 rows affected (0.20 sec)
```

- employee-team (emp-id, team-id)
    - emp-id is referencing to employee (emp-id)
    - team-id is referencing to team(team-id).

```
mysql> CREATE TABLE employee_team (
    -> emp_id INT NOT NULL
    -> ,team_id INT NOT NULL
    -> ,FOREIGN KEY(emp_id)
    -> REFERENCES employee (emp_id)
    -> ON DELETE CASCADE
    -> ON UPDATE CASCADE
    -> ,FOREIGN KEY(team_id)
    -> REFERENCES team(team_id)
    -> ON DELETE CASCADE
    -> ON UPDATE CASCADE
    -> );
Query OK, 0 rows affected (0.07 sec)
```

- customer-product (cust-id, product-id)
    - cust-id is referencing to customer (cust-id)
    - product-id is referencing to product (product-id)

```
mysql> CREATE TABLE customer_product (
    -> cust_id INT NOT NULL
    -> ,product_id INT NOT NULL
    -> ,FOREIGN KEY(cust_id)
    -> REFERENCES customer(cust_id)
    -> ON DELETE CASCADE
    -> ON UPDATE CASCADE
    -> ,FOREIGN KEY(product_id)
    -> REFERENCES product(product_id)
    -> ON DELETE CASCADE
    -> ON UPDATE CASCADE
    -> );
Query OK, 0 rows affected (0.02 sec)
```

- SHOW TABLES;

```
mysql> SHOW TABLES;
+------------------+
| Tables_in_company |
+------------------+
| customer          |
| customer_product  |
| employee          |
| employee_team     |
| product           |
| project           |
| team              |
+------------------+
7 rows in set (0.00 sec)
```

PART 2 Using the DESCRIBE table statement to show the description of each table created

- DESCRIBE employee

```
mysql> DESCRIBE employee;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| emp_id     | int(11)     | NO   | PRI | NULL    |       |
| frst_name  | varchar(25) | YES  |     | NULL    |       |
| last_name  | varchar(25) | YES  |     | NULL    |       |
| address    | varchar(255)| YES  |     | NULL    |       |
| manager_id | int(11)     | YES  | MUL | NULL    |       |
+------------+-------------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```

- DESCRIBE customer;

```
mysql> DESCRIBE customer;
+-----------+-------------+------+-----+---------+-------+
| Field     | Type        | Null | Key | Default | Extra |
+-----------+-------------+------+-----+---------+-------+
| cust_id   | int(11)     | NO   | PRI | NULL    |       |
| frst_name | varchar(25) | YES  |     | NULL    |       |
| last_name | varchar(25) | YES  |     | NULL    |       |
| address   | varchar(255)| YES  |     | NULL    |       |
| phone     | varchar(15) | YES  |     | NULL    |       |
+-----------+-------------+------+-----+---------+-------+
5 rows in set (0.01 sec)
```

- DESCRIBE product;

```
mysql> DESCRIBE project;
+------------------+-------------+------+-----+---------+-------+
| Field            | Type        | Null | Key | Default | Extra |
+------------------+-------------+------+-----+---------+-------+
| project_id       | int(11)     | NO   | PRI | NULL    |       |
| name             | varchar(25) | YES  |     | NULL    |       |
| start_date       | date        | YES  |     | NULL    |       |
| fnish_date       | date        | YES  |     | NULL    |       |
| produced_product | int(11)     | YES  | MUL | NULL    |       |
+------------------+-------------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```

- DESCRIBE project;

```
mysql> DESCRIBE project;
+------------------+-------------+------+-----+---------+-------+
| Field            | Type        | Null | Key | Default | Extra |
+------------------+-------------+------+-----+---------+-------+
| project_id       | int(11)     | NO   | PRI | NULL    |       |
| name             | varchar(25) | YES  |     | NULL    |       |
| start_date       | date        | YES  |     | NULL    |       |
| fnish_date       | date        | YES  |     | NULL    |       |
| produced_product | int(11)     | YES  | MUL | NULL    |       |
+------------------+-------------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```

- DESCRIBE team;

```
mysql> DESCRIBE employee_team;
+---------+---------+------+-----+---------+-------+
| Field   | Type    | Null | Key | Default | Extra |
+---------+---------+------+-----+---------+-------+
| emp_id  | int(11) | NO   | MUL | NULL    |       |
| team_id | int(11) | NO   | MUL | NULL    |       |
+---------+---------+------+-----+---------+-------+
2 rows in set (0.00 sec)
```

- DESCRIBE employee_team;

```
mysql> DESCRIBE team;
+--------------------+-------------+------+-----+---------+-------+
| Field              | Type        | Null | Key | Default | Extra |
+--------------------+-------------+------+-----+---------+-------+
| team_id            | int(11)     | NO   | PRI | NULL    |       |
| name               | varchar(25) | YES  |     | NULL    |       |
| manager_id         | int(11)     | YES  | MUL | NULL    |       |
| project_working_on | int(11)     | YES  | MUL | NULL    |       |
+--------------------+-------------+------+-----+---------+-------+
4 rows in set (0.01 sec)
```

- DESCRIBE customer_product;

```
mysql> DESCRIBE customer_product;
+------------+---------+------+-----+---------+-------+
| Field      | Type    | Null | Key | Default | Extra |
+------------+---------+------+-----+---------+-------+
| cust_id    | int(11) | NO   | MUL | NULL    |       |
| product_id | int(11) | NO   | MUL | NULL    |       |
+------------+---------+------+-----+---------+-------+
2 rows in set (0.00 sec)
```

PART 3 Using the INSERT statement, insert at least 5 rows in each table. When inserting data, try incorrect entries (into the foreign key columns) and see what error you get when violating the referential integrity constraint.

Employee

- INSERT 5 rows statement into employee.

```
mysql> INSERT INTO employee VALUES
    -> (100001,'Camron','Valdez','8 SW Wagon Lane Milwaukee, WI 53204', 100001),
    -> (100002,'Eliza','Oneil','174 South Peachtree Street Merrick, NY 11566', 100002),
    -> (100003,'Adeline','Tanner','9259 Prospect St. Vista, CA 92083', 100003),
    -> (100004,'Presley','Mcconnell','828 West High Ridge Lane Lutherville, MD 21093',100004),
    -> (100005,'Presley','Mcconnell','115 N. Lookout Lane Long Branch, NJ 07740', 100005);
Query OK, 5 rows affected (0.00 sec)
Records: 5  Duplicates: 0  Warnings: 0
```

- Creating an error input test for employee table.

```
mysql> INSERT INTO employee VALUES
    -> (100006,'Trent','9684 6th Court Macon, GA 31204', 100005);
ERROR 1136 (21S01): Column count doesn't match value count at row 1
```

Customer

- INSERT 5 rows statement into customer.

```
mysql> INSERT INTO customer VALUES
    -> (200005,'Horace','Harper','916 Tunnel Court Portland, ME 04103', '(460) 963-8219');
ERROR 1062 (23000): Duplicate entry '200005' for key 'PRIMARY'
mysql> INSERT INTO product VALUES
    -> (300001,'Hardrive 1.0', 100),
    -> (300002, 'Hard drive 2.0', 125),
    -> (300003, 'Hard drive 3.0', 150),
    -> (300004, 'Hard drive 4.0', 175),
    -> (300005, 'Hard drive 5.0', 200);
Query OK, 5 rows affected (0.01 sec)
Records: 5  Duplicates: 0  Warnings: 0
```

- Creating an error input test for customer table.

```
mysql> INSERT INTO customer VALUES
    -> (200005,'Horace','Harper','916 Tunnel Court Portland, ME 04103', '(460) 963-8219');
ERROR 1062 (23000): Duplicate entry '200005' for key 'PRIMARY'
```

Product

- INSERT 5 rows statement into product.

```
mysql> INSERT INTO product VALUES
    -> (300001,'Hardrive 1.0', 100),
    -> (300002, 'Hard drive 2.0', 125),
    -> (300003, 'Hard drive 3.0', 150),
    -> (300004, 'Hard drive 4.0', 175),
    -> (300005, 'Hard drive 5.0', 200);
Query OK, 5 rows affected (0.01 sec)
Records: 5  Duplicates: 0  Warnings: 0
```

- Creating an error input test for product table.

```
mysql> INSERT INTO product VALUES
    -> (300003, 'Hard drive 6.0', 500);
ERROR 1062 (23000): Duplicate entry '300003' for key 'PRIMARY'
```

Project

- INSERT 5 rows statement into project.

```
mysql> INSERT INTO project VALUES
    -> (400001,'Project 1.0', '2009-01-05', '2009-06-05',300001),
    -> (400002,'Project 2.0', '2010-06-06', '2010-01-05',300002),
    -> (400003,'Project 2.0', '2011-01-06', '2010-06-05',300003),
    -> (400004,'Project 4.0', '2012-06-06', '2012-01-05',300004),
    -> (400005,'Project 5.0', '2013-01-06', '2013-06-05',300005);
Query OK, 5 rows affected (0.01 sec)
Records: 5  Duplicates: 0  Warnings: 0
```

- Creating an error input test for project table.

```
mysql> INSERT INTO project VALUES
    -> (400006,'Project 6.0', 2011-01-05, 2011-06-05, 300003);
Query OK, 1 row affected, 2 warnings (0.01 sec)
```

Team

- INSERT 5 rows statement into team.

```
mysql> INSERT INTO team VALUES
    -> (500001,'Team 1.0', 100001, 400001),
    -> (500002,'Team 2.0', 100002, 400002),
    -> (500003,'Team 3.0', 100003, 400003),
    -> (500004,'Team 4.0', 100004, 400004),
    -> (500005,'Team 5.0', 100005, 400005);
Query OK, 5 rows affected (0.01 sec)
Records: 5  Duplicates: 0  Warnings: 0
```

- Creating an error input test for team table.

```
mysql> INSERT INTO team VALUES
    -> (500006,'Team 1.0', 200001, 300001);
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails (`company
`.`team`, CONSTRAINT `team_ibfk_1` FOREIGN KEY (`manager_id`) REFERENCES `employee` (`emp_id`)
 ON DELETE SET NULL ON UPDATE CASCADE)
```

Employee_team

- INSERT 5 rows statement into employee_team.

```
mysql> INSERT INTO employee_team VALUES
    -> (100001, 500001),
    -> (100002, 500002),
    -> (100003, 500003),
    -> (100004, 500004),
    -> (100005, 500005);
Query OK, 5 rows affected (0.01 sec)
Records: 5  Duplicates: 0  Warnings: 0
```

- Creating an error input test for employee_team table.

```
mysql> INSERT INTO employee_team VALUES
    -> (400003, 600003);
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails (`company
`.`employee_team`, CONSTRAINT `employee_team_ibfk_1` FOREIGN KEY (`emp_id`) REFERENCES `employ
ee` (`emp_id`) ON DELETE CASCADE ON UPDATE CASCADE)
```

Customer_product

- INSERT 5 rows statement into customer_product.

```
mysql> INSERT INTO customer_product VALUES
    -> (200001, 300001),
    -> (200002, 300002),
    -> (200003, 300003),
    -> (200004, 300004),
    -> (200005, 300005);
Query OK, 5 rows affected (0.01 sec)
Records: 5  Duplicates: 0  Warnings: 0
```

- Creating an error input test for customer_product table.

```
mysql> INSERT INTO customer_product VALUES
    -> (300003, 600003);
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails (`company
`.`customer_product`, CONSTRAINT `customer_product_ibfk_1` FOREIGN KEY (`cust_id`) REFERENCES
`customer` (`cust_id`) ON DELETE CASCADE ON UPDATE CASCADE)
```

PART 4 Using the SELECT statement, write one SELECT query for each table. Use different styles of SELECT statement.

- SELECT * FROM employee;

```
mysql> SELECT * FROM employee;
+--------+-----------+-----------+-----------------------------------------------+------------+
| emp_id | frst_name | last_name | address                                       | manager_id |
+--------+-----------+-----------+-----------------------------------------------+------------+
| 100001 | Camron    | Valdez    | 8 SW Wagon Lane Milwaukee, WI 53204           |     100001 |
| 100002 | Eliza     | Oneil     | 174 South Peachtree Street Merrick, NY 11566  |     100002 |
| 100003 | Adeline   | Tanner    | 9259 Prospect St. Vista, CA 92083             |     100003 |
| 100004 | Presley   | Mcconnell | 828 West High Ridge Lane Lutherville, MD 21093 |     100004 |
| 100005 | Presley   | Mcconnell | 115 N. Lookout Lane Long Branch, NJ 07740     |     100005 |
+--------+-----------+-----------+-----------------------------------------------+------------+
5 rows in set (0.00 sec)
```

- SELECT cust_id, frst_name, last_name FROM customer;

```
mysql> SELECT cust_id, frst_name, last_name FROM customer;
+---------+-----------+------------+
| cust_id | frst_name | last_name  |
+---------+-----------+------------+
|  200001 | Daisy     | Best       |
|  200002 | Gurpreet  | Lara       |
|  200003 | Rubie     | Childs     |
|  200004 | Isabell   | Wainwright |
|  200005 | Levi      | Begum      |
+---------+-----------+------------+
5 rows in set (0.00 sec)
```

- SELECT * FROM product ORDER BY name;

```
mysql> SELECT * FROM product ORDER BY name;
+------------+----------------+------+
| product_id | name           | cost |
+------------+----------------+------+
|     300002 | Hard drive 2.0 |  125 |
|     300003 | Hard drive 3.0 |  150 |
|     300004 | Hard drive 4.0 |  175 |
|     300005 | Hard drive 5.0 |  200 |
|     300001 | Hardrive 1.0   |  100 |
+------------+----------------+------+
5 rows in set (0.00 sec)
```

- SELECT * FROM project WHERE project_id = 400001;

```
mysql> SELECT * FROM project WHERE project_id = 400001;
+------------+-------------+------------+------------+------------------+
| project_id | name        | start_date | fnish_date | produced_product |
+------------+-------------+------------+------------+------------------+
|     400001 | Project 1.0 | 2009-01-05 | 2009-06-05 |           300001 |
+------------+-------------+------------+------------+------------------+
1 row in set (0.00 sec)
```

- SELECT team_id, name, manager_id FROM team where team_id < 500003;

```
mysql> SELECT team_id, name, manager_id FROM team where team_id < 500003;
+---------+----------+------------+
| team_id | name     | manager_id |
+---------+----------+------------+
|  500001 | Team 1.0 |     100001 |
|  500002 | Team 2.0 |     100002 |
+---------+----------+------------+
2 rows in set (0.00 sec)
```

- SELECT * FROM employee_team where team_id = 500005;

```
mysql> SELECT * FROM employee_team where team_id = 500005;
+--------+---------+
| emp_id | team_id |
+--------+---------+
| 100005 |  500005 |
+--------+---------+
1 row in set (0.00 sec)
```

- SELECT * FROM customer_product where cust_id = 200001;

```
mysql> SELECT * FROM customer_product where cust_id = 200001;
+---------+------------+
| cust_id | product_Id |
+---------+------------+
|  200001 |     300001 |
+---------+------------+
1 row in set (0.01 sec)
```

PART 5 Using the DELETE statement (today's lecture), delete the middle row from each table.

- SELECT * FROM employee;

```
mysql> SELECT * FROM employee;
+--------+-----------+-----------+-----------------------------------------------+------------+
| emp_id | frst_name | last_name | address                                       | manager_id |
+--------+-----------+-----------+-----------------------------------------------+------------+
| 100001 | Camron    | Valdez    | 8 SW Wagon Lane Milwaukee, WI 53204           |     100001 |
| 100002 | Eliza     | Oneil     | 174 South Peachtree Street Merrick, NY 11566  |     100002 |
| 100003 | Adeline   | Tanner    | 9259 Prospect St. Vista, CA 92083             |     100003 |
| 100004 | Presley   | Mcconnell | 828 West High Ridge Lane Lutherville, MD 21093 |     100004 |
| 100005 | Presley   | Mcconnell | 115 N. Lookout Lane Long Branch, NJ 07740     |     100005 |
+--------+-----------+-----------+-----------------------------------------------+------------+
5 rows in set (0.00 sec)
```

- DELETE FROM employee WHERE emp_id = 100003;

```
mysql> DELETE FROM employee WHERE emp_id = 100003;
Query OK, 1 row affected (0.01 sec)
```

- SELECT * FROM customer;

```
mysql> SELECT * FROM customer;
+---------+-----------+------------+-----------------------------------------+-----------------+
| cust_id | frst_name | last_name  | address                                 | phone           |
+---------+-----------+------------+-----------------------------------------+-----------------+
|  200001 | Daisy     | Best       | 8086 Pine St.Mokena, IL 60448           | (411) 671-7982  |
|  200002 | Gurpreet  | Lara       | 80 San Juan Rd. Dayton, OH 45420        | (417) 909-5098  |
|  200003 | Rubie     | Childs     | 260 Saxton Ave. Natick, MA 01760        | (430) 523-3729  |
|  200004 | Isabell   | Wainwright | 949 Liberty Drive New Rochelle, NY 10801 |  (459) 457-3361 |
|  200005 | Levi      | Begum      | 7822 West Rose St. Fond Du Lac, WI 54935 | (264) 580-1232  |
+---------+-----------+------------+-----------------------------------------+-----------------+
5 rows in set (0.00 sec)
```

- DELETE FROM customer WHERE cust_id = 200003;

```
mysql> DELETE FROM customer WHERE cust_id = 200003;
Query OK, 1 row affected (0.01 sec)
```

- SELECT * FROM product;

```
mysql> SELECT * FROM product;
+------------+----------------+------+
| product_id | name           | cost |
+------------+----------------+------+
|     300001 | Hardrive 1.0   |  100 |
|     300002 | Hard drive 2.0 |  125 |
|     300003 | Hard drive 3.0 |  150 |
|     300004 | Hard drive 4.0 |  175 |
|     300005 | Hard drive 5.0 |  200 |
+------------+----------------+------+
5 rows in set (0.00 sec)
```

- DELETE FROM product WHERE product_id = 300003;

```
mysql> DELETE FROM product WHERE product_id = 300003;
Query OK, 1 row affected (0.00 sec)
```

- SELECT * FROM project;

```
mysql> SELECT * FROM project;
+------------+-------------+------------+------------+-----------------+
| project_id | name        | start_date | fnish_date | produced_product |
+------------+-------------+------------+------------+-----------------+
|     400001 | Project 1.0 | 2009-01-05 | 2009-06-05 |          300001 |
|     400002 | Project 2.0 | 2010-06-06 | 2010-01-05 |          300002 |
|     400003 | Project 2.0 | 2011-01-06 | 2010-06-05 |          300003 |
|     400004 | Project 4.0 | 2012-06-06 | 2012-01-05 |          300004 |
|     400005 | Project 5.0 | 2013-01-06 | 2013-06-05 |          300005 |
+------------+-------------+------------+------------+-----------------+
5 rows in set (0.00 sec)
```

- DELETE FROM project WHERE project_id = 400003;

```
mysql> DELETE FROM project WHERE project_id = 400003;
Query OK, 1 row affected (0.01 sec)
```

- SELECT * FROM team;

```
mysql> SELECT * FROM team;
+---------+----------+------------+--------------------+
| team_id | name     | manager_id | project_working_on |
+---------+----------+------------+--------------------+
|  500001 | Team 1.0 |     100001 |             400001 |
|  500002 | Team 2.0 |     100002 |             400002 |
|  500003 | Team 3.0 |     100003 |             400003 |
|  500004 | Team 4.0 |     100004 |             400004 |
|  500005 | Team 5.0 |     100005 |             400005 |
+---------+----------+------------+--------------------+
5 rows in set (0.00 sec)
```

- DELETE FROM team WHERE team_id = 500003;

```
mysql> DELETE FROM team WHERE team_id = 500003;
Query OK, 1 row affected (0.00 sec)
```

- SELECT * FROM employee_team;

```
mysql> SELECT * FROM customer_product;
+---------+------------+
| cust_id | product_Id |
+---------+------------+
|  200001 |     300001 |
|  200002 |     300002 |
|  200004 |     300004 |
|  200005 |     300005 |
+---------+------------+
4 rows in set (0.00 sec)
```

- SELECT * FROM customer_product;

```
mysql> SELECT * FROM employee_team;
+--------+---------+
| emp_id | team_id |
+--------+---------+
| 100001 |  500001 |
| 100002 |  500002 |
| 100004 |  500004 |
| 100005 |  500005 |
+--------+---------+
4 rows in set (0.00 sec)
```