LABORATORY WORK 2. BASICS OF DATA INTEGRITY CONTROL MECHANISMS

Goal: learn how to use the referential integrity control mechanisms using the MySQL database.

Progress

Warning! It is recommended to create the temporary database using the queries shown in the laboratory work 2. Use this temporary database in this laboratory work.

1. Learn the features of the referential integrity control mechanism NO ACTION

Let's consider the features of the referential integrity mechanism NO ACTION on the example of the relationship between supplier and contract tables, supplier and supplier_person, supplier and supplier_org. These tables are linked by the supplier_id field. In this regard, the supplier table is parent, and the tables contract, supplier_org, supplier_person are child tables. In order to learn the features of the mechanism of referential integrity, the following sequence of statements must be executed.

Set the ON DELETE and ON UPDATE parameters that determine the behavior when deleting and updating entries from the parent table.

```
ALTER TABLE contract
DROP FOREIGN KEY contract_ibfk_1;

ALTER TABLE contract
ADD CONSTRAINT contract_ibfk_1 FOREIGN KEY (supplier_id) REFERENCES supplier(supplier_id) ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE supplier_org
DROP FOREIGN KEY supplier_org_ibfk_1;

ALTER TABLE supplier_org
ADD CONSTRAINT supplier_org_ibfk_1 FOREIGN KEY (supplier_id) REFERENCES supplier(supplier_id) ON DELETE NO ACTION ON UPDATE NO ACTION;

ALTER TABLE supplier_person
DROP FOREIGN KEY supplier_person_ibfk_1;

ALTER TABLE supplier_person
ADD CONSTRAINT supplier_person_ibfk_1 FOREIGN KEY (supplier_id) REFERENCES supplier(supplier_id) ON DELETE NO ACTION ON UPDATE NO ACTION;
```

Assume that due to certain reasons, it is required to remove the supplier with code 4 (figure 2.1).

```
DELETE FROM supplier WHERE supplier_id = 4;
```

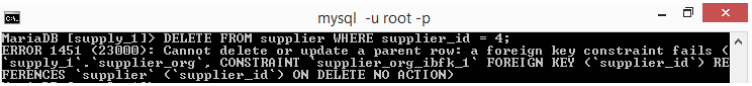

Figure 2.1

Therefore, in order to remove this supplier, you must first delete all the data associated with it. To do this, delete the corresponding entry from the supplier_org table and check the availability of contracts with this supplier in the contract table. If there are such contracts, they should also be deleted (it should be kept in mind that there may be a need to remove and a content of these contracts). After that, you need to try to remove the suppliers with code 4 again. If there is no data associated with it, the supplier will be deleted.

Suppose that for some reason there was a need for a supplier with code 5 to change the code to 7 (figure 2.2).

```
UPDATE supplier SET supplier_id = 7 WHERE supplier_id = 5;
```

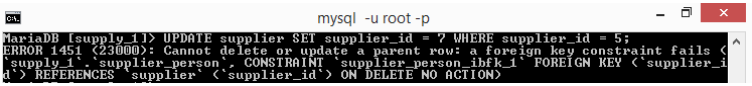

Figure 6.2

Since the contracts with this supplier are not available, the link to it is only in the supplier_person table. After deleting this entry, you must repeat the supplier code change from 5 to 7. Now, this operation must be successful. After that, you need to check the contents of the tables.

2. Learn the features of the referential integrity control mechanism CASCADE

Change the referential integrity mechanisms for links between all the above tables to the CASCADE.

```
ALTER TABLE contract
DROP FOREIGN KEY contract_ibfk_1;

ALTER TABLE contract
ADD CONSTRAINT contract_ibfk_1 FOREIGN KEY (supplier_id) REFERENCES supplier(supplier_id) ON DELETE CASCADE ON UPDATE CASCADE;

ALTER TABLE supplier_org
DROP FOREIGN KEY supplier_org_ibfk_1;

ALTER TABLE supplier_org
ADD CONSTRAINT supplier_org_ibfk_1 FOREIGN KEY (supplier_id) REFERENCES supplier(supplier_id) ON DELETE CASCADE ON UPDATE CASCADE;

ALTER TABLE supplier_person
DROP FOREIGN KEY supplier_person_ibfk_1;

ALTER TABLE supplier_person
ADD CONSTRAINT supplier_person_ibfk_1 FOREIGN KEY (supplier_id) REFERENCES supplier(supplier_id) ON DELETE CASCADE ON UPDATE CASCADE;
```

Suppose that for some reason there was a need for a supplier with code 2 to change the code to 8 (figure 2.3).

```
UPDATE supplier SET supplier_id = 8 WHERE supplier_id = 2;
```

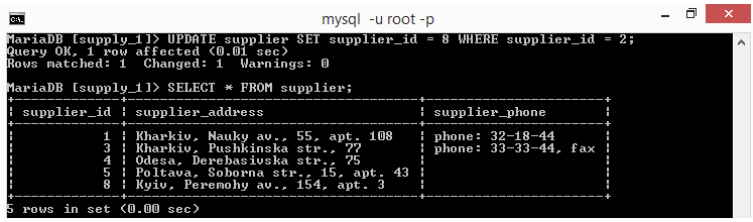

Figure 2.3

Check for the appropriate changes in the supplier_org table.

Now assume that this supplier (which now has code 8) must be removed (figure 2.4).

```
DELETE FROM supplier WHERE supplier_id = 8;
```

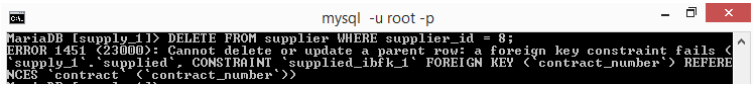

Figure 2.4

Determine the reason why entries were not deleted. Make the necessary changes in the referential integrity mechanisms of the required tables in order to ensure that the necessary data has still been deleted.

3. Learn the features of the referential integrity control mechanism SET NULL

Consider the features of the SET NULL referential integrity mechanism, e.g., for the supplier and contract tables.

Change the referential integrity mechanisms for links between all the above tables to the SET NULL.

```
ALTER TABLE contract
DROP FOREIGN KEY contract_ibfk_1;

ALTER TABLE contract
MODIFY supplier_id INT NULL;

ALTER TABLE contract
ADD CONSTRAINT contract_ibfk_1 FOREIGN KEY (supplier_id) REFERENCES supplier(supplier_id) ON DELETE SET NULL ON UPDATE SET NULL;
```

In the supplier table, change supplier code 3 to 10. Check the data in the contract table (figure 2.5).

```
UPDATE supplier SET supplier_id = 10 WHERE supplier_id = 3;
```

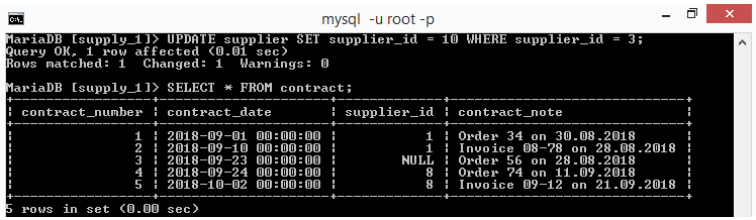

Figure 2.5

Instead of NULL set the value of the supplier code 10 for the contract with number 3.

4. Make a report for the laboratory work

The report should include the main stages of laboratory work and screenshots that demonstrate them.

5. Questions

1. Are the ON DELETE and ON UPDATE commands are necessary for the CRATE TABLE or ALTER TABLE commands?
2. What behavior of a database the ON DELETE command defines?
3. What behavior of a database the ON UPDATE command defines?
4. Which parameters might be defined after the ON DELETE and ON UPDATE statements?
5. Name features of the referential integrity mode CASCADE.
6. Name features of the referential integrity mode SET NULL.
7. Name features of the referential integrity mode NO ACTION.
8. Name features of the referential integrity mode SET DEFAULT.
9. Name features of the referential integrity mode RESTRICT.
10. Why the referential integrity mechanism SET DEFAULT has not been considered in this laboratory work?
11. How to set a certain referential integrity mechanism for a foreign key of a table?
12. Why the supplier_id field of the contract table was modified before the SET NULL mode is set?

13. In which cases it is not recommended to used the referential integrity mechanism CASCADE?

14. Which referential integrity mode is always used by default in the MySQL database in case if it was not defined using the ON DELETE and ON UPDATE statements?