Update Queries

Update Queries

Update queries can be used to perform any of the following actions:

- Modify a field value in a table.
- Add a new value. (insert into)
- Remove a value.

insert Into

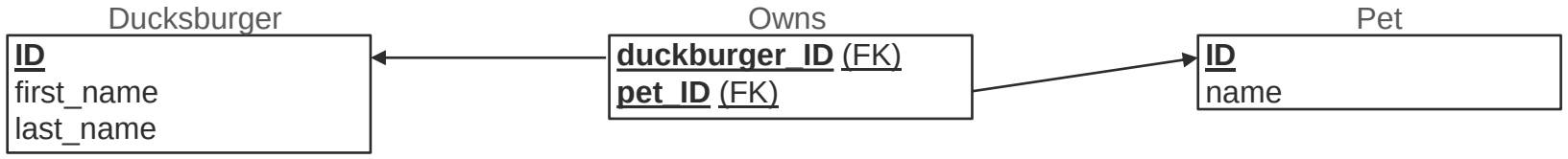
- Let's add a new citizen to out duckburger table:

```
MariaDB [duckburg]> insert into duckburger(first_name, last_name)
    -> values("Grandma", "Duck");
Query OK, 1 row affected (0.009 sec)
```

```
MariaDB [duckburg]> select * from duckburger;
+----+------------+-----------+
| ID | first_name | last_name |
+----+------------+-----------+
|  1 | Donald     | Duck      |
|  2 | Scrooge    | McDuck    |
|  3 | Huey       | Duck      |
|  4 | Magica     | De Spell  |
|  5 | Mickey     | Mouse     |
|  6 | Grandma    | Duck      |
+----+------------+-----------+
```

update

- Magica De Spell has been testing some evil spells on Ratface. Ratface decides to escape to Grandma Duck's farm.



- Let's make a query that assigns Grandma Duck as the new owner of poor Ratface.

update

- The update must be done in the owns table.
- The ownership information in the owns table uses foreign keys that point the primary keys owner id and pet id.
- We now that the name is Ratface and the new owner is Grandma Duck.
- This is where we need a **subquery**.

update

- Before we start, let's look at the current state of our tables. You can see that the IDs for Ratface and Magica De Spell are 3 and 4, respectively. It is not possible to accomodate these separately aqcuired data into the update query. However, we can compare this information with our results later.

```
MariaDB [duckburg]> select * from
duckburger;
+----+------------+-----------+
| ID | first_name | last_name |
+----+------------+-----------+
|  1 | Donald     | Duck      |
|  2 | Scrooge    | McDuck    |
|  3 | Huey       | Duck      |
|  4 | Magica     | De Spell  |
|  5 | Mickey     | Mouse     |
|  6 | Grandma    | Duck      |
+----+------------+-----------+
6 rows in set (0.000 sec)

MariaDB [duckburg]> select * from owns;
+--------+--------------+
| pet_ID | duckburger_ID |
+--------+--------------+
|      1 |            1 |
|      1 |            3 |
|      2 |            5 |
|      3 |            4 |
+--------+--------------+
4 rows in set (0.001 sec)
```

```
MariaDB [duckburg]> select * from pet;
+----+------------+
| ID | name       |
+----+------------+
|  1 | Bolivar    |
|  2 | Pluto      |
|  3 | Ratface    |
+----+------------+
```

update

```
MariaDB [duckburg]> update owns
    -> set duckburger_id = (select id from duckburger where last_name = "Duck" and first_name = "Grandma")
    -> where pet_id in (select id from pet where name = "Ratface");
Query OK, 1 row affected (0.004 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

- Notice that both the owner and pet id have been fetched with subqueries.
- The set operation sets a value. It cannot be replaced with the in operation by set theory.

update: results

- Only the owns table was changed:

```
MariaDB [duckburg]> select * from owns;
+--------+---------------+
| pet_ID | duckburger_ID |
+--------+---------------+
|      1 |             1 |
|      1 |             3 |
|      2 |             5 |
|      3 |             6 |
+--------+---------------+
```

- The id of the owner of Ratface (3) has changed from 4 to 6, which corresponds to Grandma Duck.

Referential Integrity

- Can we remove Grandma Duck?
  - No, because there is a reference to Grandma Duck in the owns table.
  - The database engine behind MariaDB, InnoDB, notifies us if that there is a problem with referential integrity if we try. The operation fails.

delete

- We could, however, remove Magica De Spell as there are no references to her in any of the other tables. However, let's not remove her now in case Ratface still wants to return home.
- Let's remove Scrooge McDuck instead.
- Note! MariaDB removes records permanently without asking, so be sure what and when to delete.

delete

```
MariaDB [duckburg]> delete from duckburger
    -> where first_name = "Scrooge" and last_name = "Duck";
Query OK, 1 row affected (0.004 sec)

MariaDB [duckburg] select * from duckburger;
+----+------------+-----------+
| ID | first_name | last_name |
+----+------------+-----------+
|  1 | Donald     | Duck      |
|  3 | Huey       | Duck      |
|  4 | Magica     | De Spell  |
|  5 | Mickey     | Mouse     |
|  6 | Grandma    | Duck      |
+----+------------+-----------+
```

- Bye, Scrooge!

Thank you for the course!