Table of Contents

1. TASK 2

2. DATABASE CREATION 3

3. TABLES CREATION 4

3.1. "Rozetka" table creation 4

3.2. "Allo" table creation 4

3.3. "Company" table creation 5

4. ESTABLISHING A RELATION BETWEEN TABLES 6

4.1. Between “Rozetka” and “Company” tables 6

4.2. Between “Allo” and “Company” tables 6

4.3. “Shops” database schema 6

5. FILLING TABLES WITH VALUES 7

5.1. Filling the “Rozetka” table 7

5.2. Filling the “Allo” table 7

5.3. Filling the “Company” table 7

6. TASK 2. SELECT QUERIES 8

6.1. Make a simple selection for each table 8

6.2. Sort each table by company 9

6.3. Group table by name 9

6.4. Select the maximum price value for each table 10

6.5. Calculate the total cost of goods in each table 11

6.6. Make a selection of all smartphones from all tables (product name, price, company) 11

6.7. Make a selection of all Bosch products for all products table 12

6.8. Select Samsung products from all tables where the price is more than 600 12

6.9. Select smartphones worth more than 15,000 13

1. TASK

1) Create a database on **MySQL**, create several tables, create, modify, delete the records.

Create database: **shops**

Tables:

Rozetka (id_r, name_r, id_com, price_r)
Allo (id_a, name_a, id_com, price_a)
Company (id_c, name_c)

Rozetka:

Kettle, Bosch, 1500
TV, Samsung, 5000
Smartphone, Samsung, 15,000
Smartphone, Apple, 20,000
Blender, Bosch, 500

Allo:

TV, Samsung, 10000
TV, Apple, 20,000
Kettle, Samsung, 250
Blender, Samsung, 600
Smartphone, Apple, 15,000

Company:

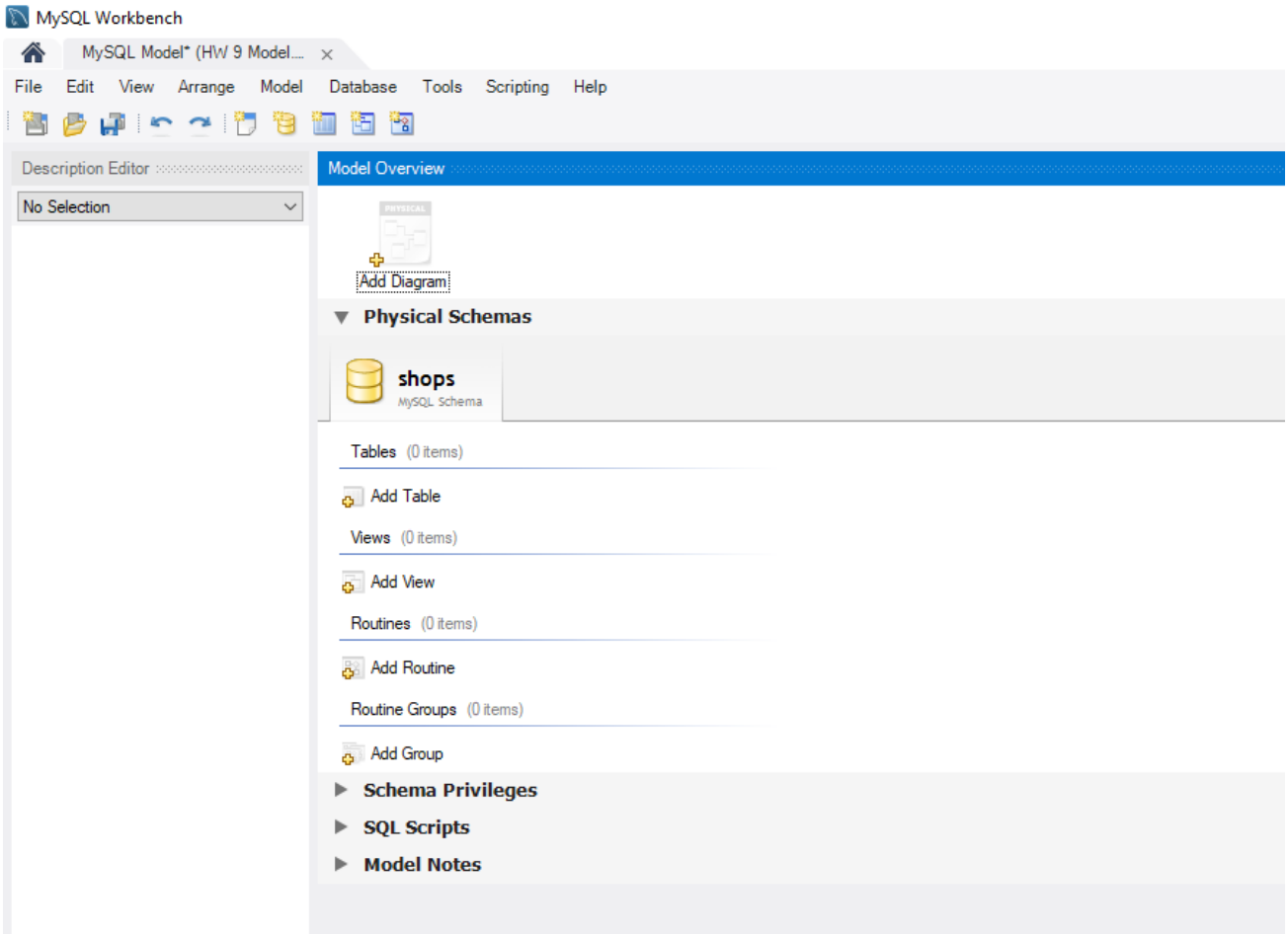
Bosch
Samsung
Magic
Apple

2) Create simple and complex queries.

1. Make a simple selection for each table
2. Sort each table by company
3. Group table by name
4. Select the maximum price value for each table
5. Calculate the total cost of goods in each table
6. Make a selection of all smartphones from all tables (product name, price, company)
7. Make a selection of all Bosch products for all products table
8. Select Samsung products from all tables where price more than 600
9. Select smartphones worth more than 15,000

2. DATABASE CREATION

```
CREATE DATABASE shops;
```



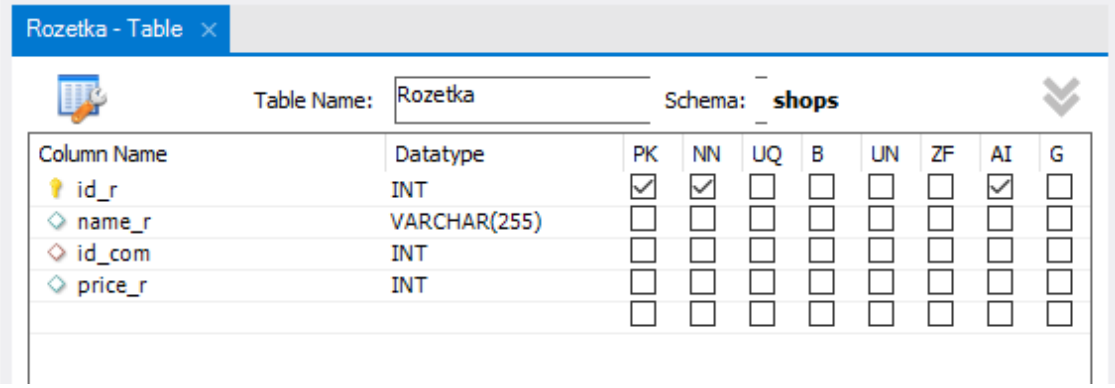

```
SHOW databases;
```

3. TABLES CREATION

```
USE shops;
```

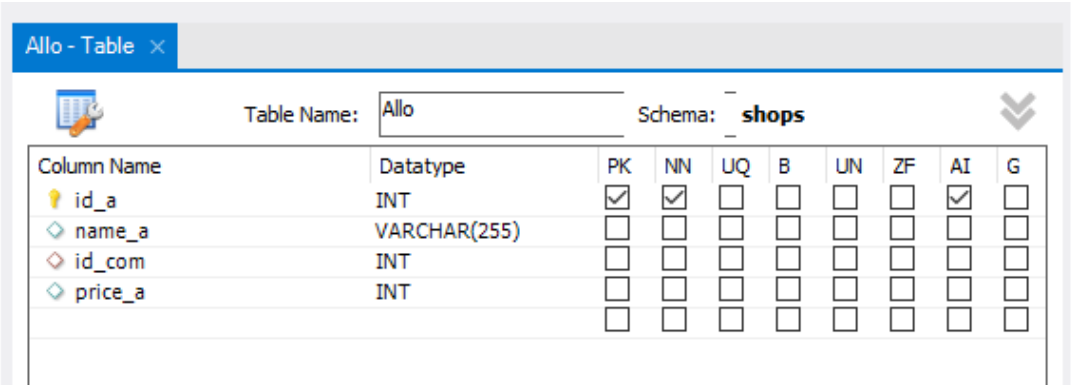
3.1. "Rozetka" table creation

```
CREATE TABLE Rozetka (
    id_r int PRIMARY KEY AUTO_INKREMENT UNIQUE,
    name_r varchar(255),
    id_com int,
    price_r int
);
```



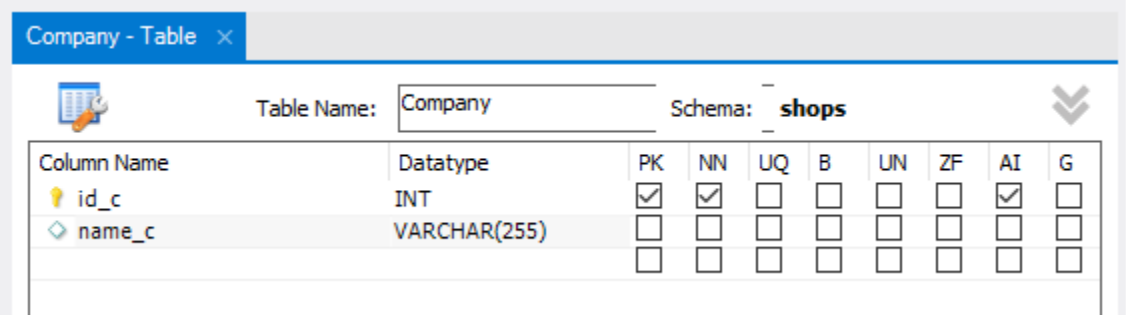
3.2. "Allo" table creation

```
CREATE TABLE Allo(
    id_a int PRIMARY KEY AUTO_INKREMENT UNIQUE,
    name_a varchar(255),
    id_com int,
    price_a int
);
```



3.3. "Company" table creation

```
CREATE TABLE Company(
    id_c int PRIMARY KEY AUTO_INKREMENT UNIQUE,
    name_c varchar(255),
);
```



4. ESTABLISHING A RELATIONSHIP BETWEEN TABLES

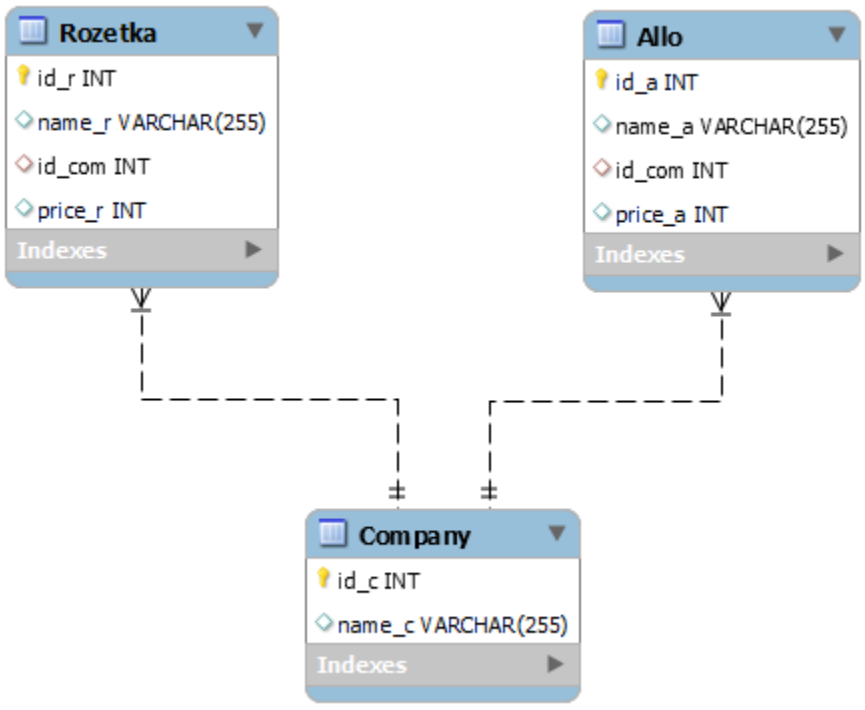
4.1. Between “Rozetka” and “Company” tables

```
ALTER TABLE Rozetka
ADD CONSTRAINT Rozetka_Company_FK
FOREIGN KEY (id_com) REFERENCES Company(id_c) ON DELETE CASCADE;
```

4.2. Between “Allo” and “Company” tables

```
ALTER TABLE Allo
ADD CONSTRAINT Allo_Company_FK
FOREIGN KEY (id_com) REFERENCES Company(id_c) ON DELETE CASCADE;
```

4.3. “Shops” database schema



5. FILLING TABLES WITH VALUES

5.1. Filling the “Rozetka” table

```
INSERT INTO Rozetka
VALUES
(1, 'Kettle', 1, 1500),
(2, 'TV', 2, 5000),
(3, 'Smartphone', 2, 15000),
(4, 'Smartphone', 4, 20000),
(5, 'Blender', 1, 500);
```

5.2. Filling the “Allo” table

```
INSERT INTO Allo
VALUES
(1, 'TV', 2, 10000),
(2, 'TV', 4, 20000),
(3, 'Kettle', 2, 250),
(4, 'Blender', 2, 600),
(5, 'Smartphone', 4, 15000);
```

5.3. Filling the “Company” table

```
INSERT INTO Company
VALUES
(1, 'Bosch'),
(2, 'Samsung'),
(3, 'Magic'),
(4, 'Apple');
```

6. TASK 2. SELECT QUERIES

6.1. Make a simple selection for each table

- “Rozetka” request:

```
SELECT * FROM Rozetka;
```

“Rozetka” table

id_r	name_r	id_com	price_r
1	Kettle	1	1500
2	TV	2	5000
3	Smartphone	2	15000
4	Smartphone	4	20000
5	Blender	1	500
NULL	NULL	NULL	NULL

- “Allo” request:

```
SELECT * FROM Allo;
```

“Allo” table

id_a	name_a	id_com	price_a
1	TV	2	10000
2	TV	4	20000
3	Kettle	2	250
4	Blender	2	600
5	Smartphone	4	15000
NULL	NULL	NULL	NULL

- “Company” request:

```
SELECT * FROM Company;
```

“Company” table

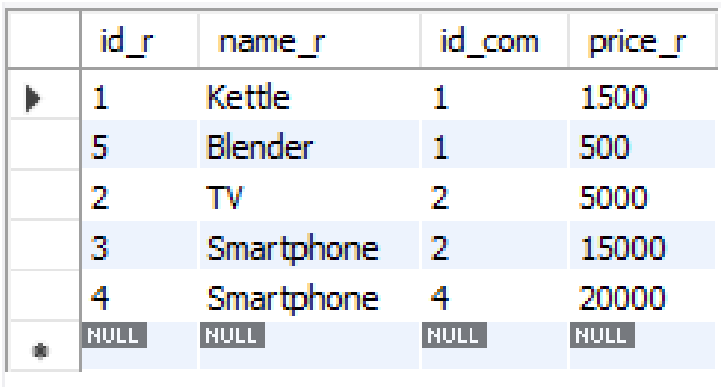
id_c	name_c
1	Bosch
2	Samsung
3	Magic
4	Apple
NULL	NULL

6.2. Sort each table by company

- “Rozetka” request:

```
SELECT *
FROM Rozetka
ORDER BY Rozetka.id_com;
```

Response:

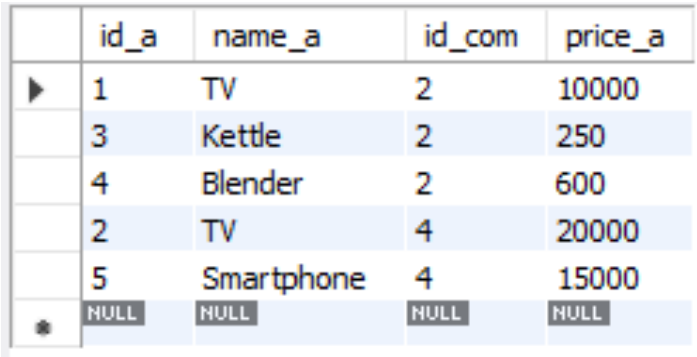


id_r	name_r	id_com	price_r
1	Kettle	1	1500
5	Blender	1	500
2	TV	2	5000
3	Smartphone	2	15000
4	Smartphone	4	20000
NULL	NULL	NULL	NULL

- “Allo” request:

```
SELECT *
FROM Allo
ORDER BY Allo.id_com;
```

Response:



id_a	name_a	id_com	price_a
1	TV	2	10000
3	Kettle	2	250
4	Blender	2	600
2	TV	4	20000
5	Smartphone	4	15000
NULL	NULL	NULL	NULL

6.3. Group table by name

- “Rozetka” request:

```
SELECT name_r, id_com, count(name_r) AS Quantity, SUM(price_r)
AS Total_price
FROM Rozetka
GROUP BY name_r;
```

Response:

	name_r	id_com	Quantity	Total_price
▶	Kettle	1	1	1500
	TV	2	1	5000
	Smartphone	2	2	35000
	Blender	1	1	500

- “Allo” request:

```
SELECT name_a, id_com, count(name_a) AS Quantity, SUM(price_a)
AS Total_price
FROM Allo
GROUP BY name_a;
```

Response:

	name_a	id_com	Quantity	Total_price
▶	TV	2	2	30000
	Kettle	2	1	250
	Blender	2	1	600
	Smartphone	4	1	15000

6.4. Select the maximum price value for each table

- “Rozetka” request:

```
SELECT MAX(price_r) AS Max_price
FROM Rozetka;
```

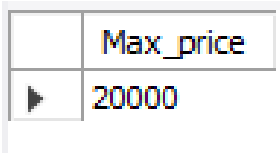
Response:

	Max_price
▶	20000

- “Allo” request:

```
SELECT MAX(price_a) AS Max_price
FROM Allo;
```

Response:



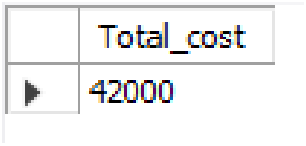
	Max_price
▶	20000

6.5. Calculate the total cost of goods in each table

- “Rozetka” request:

```
SELECT SUM(price_r) AS Total_cost
FROM Rozetka;
```

Response:

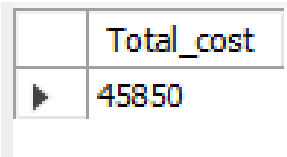


	Total_cost
▶	42000

- “Allo” request:

```
SELECT SUM(price_a) AS Total_cost
FROM Allo;
```

Response:



	Total_cost
▶	45850

6.6. Make a selection of all smartphones from all tables (product name, price, company)

- Request:

```
SELECT name_r AS Product_name, price_r AS Price, name_c AS
Company
FROM Rozetka
INNER JOIN Company
ON Rozetka.id_com = Company.id_c
WHERE name_r = 'Smartphone'
UNION
SELECT name_a, price_a, name_c
FROM Allo
INNER JOIN Company
```

```
ON Allo.id_com = Company.id_c
WHERE name_a = 'Smartphone';
```

Response:

	Product_name	Price	Company
▶	Smartphone	15000	Samsung
	Smartphone	20000	Apple
	Smartphone	15000	Apple

6.7. Make a selection of all Bosch products for all products table

- Request:

```
SELECT name_r AS Product_name, price_r AS Price, name_c AS
Company
FROM Rozetka
INNER JOIN Company
ON Rozetka.id_com = Company.id_c
WHERE id_com = 1
UNION
SELECT name_a AS Product_name, price_a AS Price, name_c AS
Company
FROM Allo
INNER JOIN Company
ON Allo.id_com = Company.id_c
WHERE id_com = 1;
```

Response:

	Product_name	Price	Company
▶	Kettle	1500	Bosch
	Blender	500	Bosch

6.8. Select Samsung products from all tables where the price is more than 600

- Request:

```
SELECT name_r AS Product_name, price_r AS Price, name_c AS
Company
```

```
FROM Rozetka
INNER JOIN Company
ON Rozetka.id_com = Company.id_c
WHERE id_com = 2 AND price_r > 600
UNION
SELECT name_a AS Product_name, price_a AS Price, name_c AS
Company
FROM Allo
INNER JOIN Company
ON Allo.id_com = Company.id_c
WHERE id_com = 2 AND price_a > 600;
```

Response:

	Product_name	Price	Company
▶	TV	5000	Samsung
	Smartphone	15000	Samsung
	TV	10000	Samsung

6.9. Select smartphones worth more than 15,000

- Request:

```
SELECT name_r AS Product_name, price_r AS Price, name_c AS
Company
FROM Rozetka
INNER JOIN Company
ON Rozetka.id_com = Company.id_c
WHERE name_r = 'Smartphone' AND price_r > 15000
UNION
SELECT name_a, price_a, name_c
FROM Allo
INNER JOIN Company
ON Allo.id_com = Company.id_c
WHERE name_a = 'Smartphone' AND price_a > 15000 ;
```

Response:

	Product_name	Price	Company
▶	Smartphone	20000	Apple