

FINAL SEMESTER EXAMINATION /
Academic Year 2020 – 2021 / 2nd Semester
Subject : Database Systems
Lecturer : Zain Saifullah
Study Program : Information Technology / IT4
Date of Exam : May 3, 2021

Instructions to Students

1. This examination consist of 5 pages and 2 questions
 2. Due date of this examination is **Monday May 10, 2021 23.59 PM**
 3. Sanctions will be given to those students who are not following the examination rules
 4. All answers to be written directly following the questions. The number to the corresponding question must be written correctly
 5. This is a take home examination
 6. Students are not allowed to communicate or to cooperate each other or copy someone's work while the examination is going on
 7. You must submit your handwritten answer by screenshot it and inserted into word or pdf files. You also must submit screenshot all answers (from cmd) in the same file. Your file name is YourName_YourStudentID.doc(x) or pdf
-

Questions:

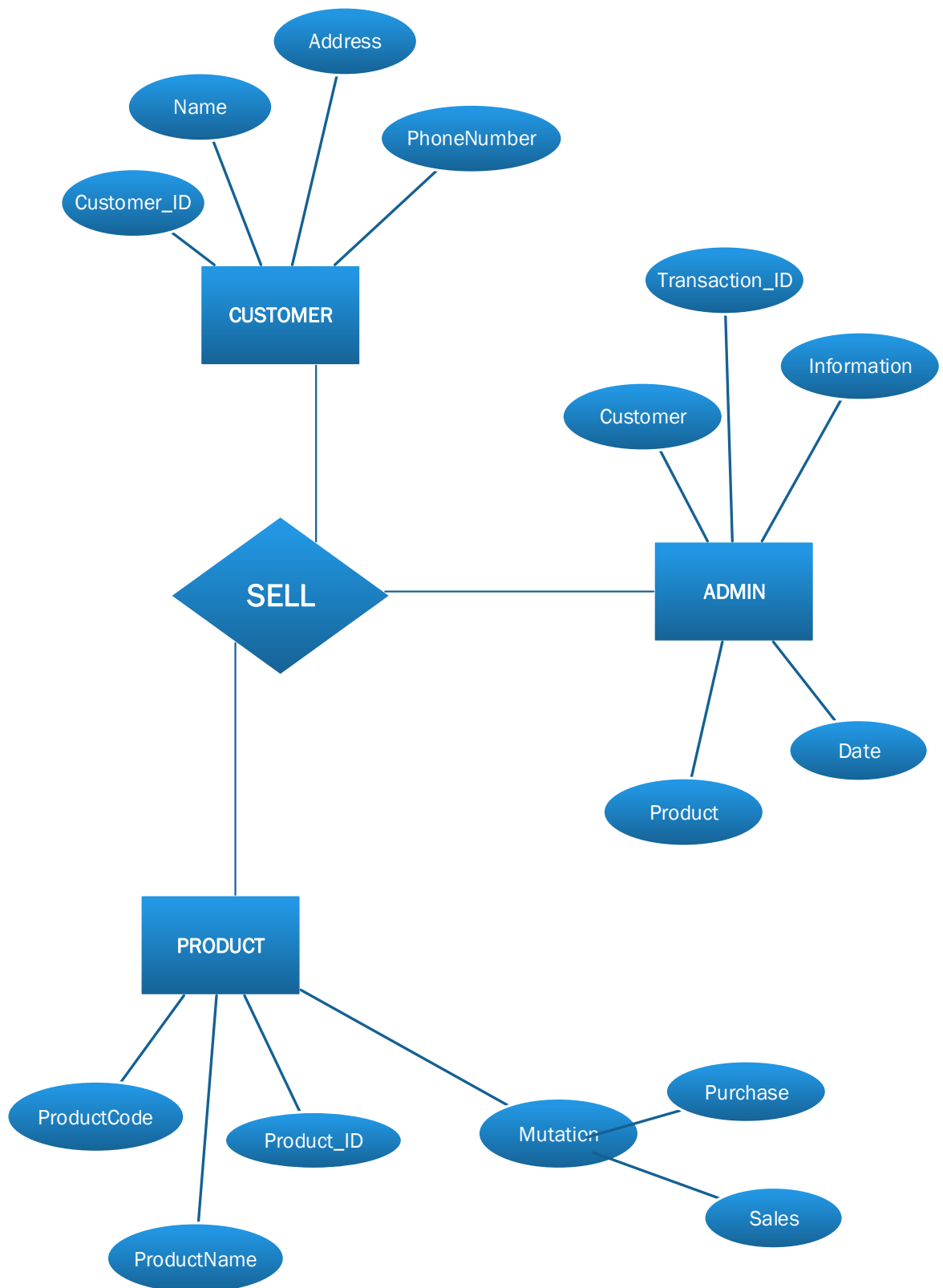
1. (50 Marks)
Make ERD based on the attachment at the end of the file.
If you think the number of the entities less than 3, you can add other entities related to the problem.
You must give explanation if you think there is multi interpretation on your ERD

ANSWERS:

My Student ID is 001202000067 and the last digit of my id is 7. So I created an Online Sales System to be used as a reference for making Entity-Relationship Diagrams. I thought that the entities in the reference table below were less than 3, so I added related entities and attributes to my ERD.

The diagram image that I made is on the next page. In the diagram, there are 3 tables in the database that was formed. There is a **Customer table** that functions to manage or manage purchases. Then there is the Product table, in this **Product table**, there is a table that contains the management of the products being sold. And then there is the **Admin table**, which functions to evaluate and analyze the system for selling goods at any given time.

ONLINE SALES SYSTEM



Explanations:

1. ENTITY,

The entity which is in the ERD above are as follows:

- Customer (Master entity)
- Admin (Associative entity)
- Product (Master entity)

2. ATTRIBUTE,

The attribute which is in the ERD above are as follows:

- Customer : Customer_ID (primary key), Name, Address, PhoneNumber
- Admin : Customer (Foreign Key), Transaction_ID (Primary_Key), Information, Date, Product (Foreign Key).
- Product : ProductCode (Unique), ProductID (Primary Key), ProductName, Mutation Purchase, Mutation Sales.

3. RELATIONSHIPS AND CARDINALITY,

Relationships and cardinality which is in the ERD above are as follows:

- Customer with Product (many to many)
A customer can buy many types of goods, one type of item can be purchased by many customers. The results of many to many produce an associative entity, namely the Admin entity.
- Many To Many Implementations
The solution that is usually done if there is a many to many relationship is, usually we will add 1 table in the middle. This table serves as a bridge to combine many to many relationships
 1. The contents of this table will contain the id of the first table and the second table, in this case the products and customers
 2. Thus, we can add some data to this relation table, so it means that one product can be sold several times, and one customer can buy more than one product.

2. (50 Marks) Create SQL Command which

- a) Creating the database and tables (at least 4 and has connections among the tables) with the following topics (must be not the same with question no.1 for all students)

Last Digit of your ID	Topics
0	Sports
1	Politics
2	Finance
3	Culinary
4	Plant Production
5	Health
6	Education
7	Transportation
8	Culture
9	Property

Question b) until g) will be done using no a) and also make explanations on each question

- b) Demonstrate you can insert, update, and delete the data and make explanations
You must insert some data for demonstrate other questions
- c) Demonstrate you can implement **JOIN, LEFT JOIN, and RIGHT JOIN**
- d) Demonstrate you can use 3 out of 5 aggregate functions
- e) Demonstrate you can use sub-query (nested query) and sub-sub-query
- f) Demonstrate you can implement UNION and INTERSECTION
- g) Demonstrate you can implement modify/change the column

ANSWERS (HAND WRITTEN):

A. Transportation

```
MariaDB [finalexam]> SHOW TABLES;
+-----+
| Tables_in_finalexam |
+-----+
| car_makers           |
| car_names            |
| continents           |
| countries             |
| model_details        |
+-----+
5 rows in set (0.029 sec)

MariaDB [finalexam]>
```

>>> Details:

```

Create database finalexam;
Use finalexam;

Create table continents (
  ContID INT auto-increment,
  Continent VARCHAR (25),
  Constraint continent_id-pk Primary key (ContID)
);

Create table Countries (
  CountryID INT auto-increment,
  CountryName VARCHAR (25),
  Continent INT,
  Constraint country-continent-id-fk foreign key (Continent)
  References continents (ContID),
  Constraint country-id-pk Primary key (CountryID)
);

Create table Car-makers (
  ID INT, Maker VARCHAR (25), Full Name VARCHAR (25),
  Country INT,
  Constraint car-maker-country-id-fk Foreign key (Country)
  References (CountryID),
  Constraint car-maker-id Primary key (ID)
);

Create table Model-Details (
  ModelID INT, maker INT, Model VARCHAR (25),
  Constraint model-unique UNIQUE (Model),
  Constraint model-details-id-pk Primary Key (ModelID),
  Constraint model-maker-fk Foreign Key (Maker)
  References Car-makers (ID)
);

Create table car-names (
  ID INT, Model VARCHAR (25), Descr VARCHAR (25),
  Constraint car-names-model-fk Foreign Key (Model)
  References Model-Details (ModelID)
);
```

```

MariaDB [finalexam]> DESC countries;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra      |
+-----+-----+-----+-----+-----+-----+
| CountryID  | int(11)   | NO   | PRI | NULL    | auto_increment |
| CountryName | varchar(25) | YES  |     | NULL    |              |
| Continent  | int(11)   | YES  | MUL | NULL    |              |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.159 sec)

MariaDB [finalexam]> DESC continents;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra      |
+-----+-----+-----+-----+-----+-----+
| ContID     | int(11)   | NO   | PRI | NULL    | auto_increment |
| Continent  | varchar(25) | YES  |     | NULL    |              |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.090 sec)

MariaDB [finalexam]> DESC car_makers;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra      |
+-----+-----+-----+-----+-----+-----+
| ID         | int(11)   | NO   | PRI | NULL    |              |
| Maker      | varchar(25) | YES  |     | NULL    |              |
| FullName   | varchar(25) | YES  |     | NULL    |              |
| Country    | int(11)   | YES  | MUL | NULL    |              |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.087 sec)

MariaDB [finalexam]> DESC car_names;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra      |
+-----+-----+-----+-----+-----+-----+
| ID         | int(11)   | YES  |     | NULL    |              |
| Model      | varchar(25) | YES  | MUL | NULL    |              |
| Descr      | varchar(25) | YES  |     | NULL    |              |
| prices_$  | int(11)   | YES  |     | NULL    |              |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.092 sec)

MariaDB [finalexam]> DESC model_details;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra      |
+-----+-----+-----+-----+-----+-----+
| ModelID    | int(11)   | NO   | PRI | NULL    |              |
| Maker      | int(11)   | YES  | MUL | NULL    |              |
| Model      | varchar(25) | YES  | UNI | NULL    |              |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.123 sec)

MariaDB [finalexam]> 

```

>>> Explanations:

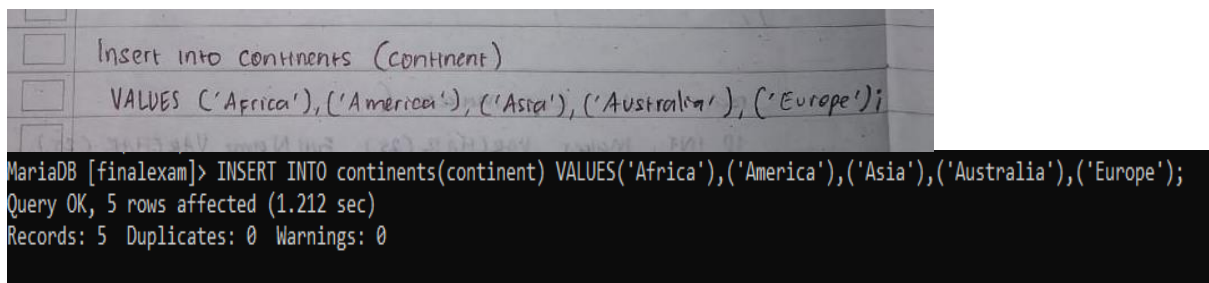
In the picture above, I created several tables with the theme Transportation. There are five tables, namely, countries, continent, model_details, car_makers, and car_names.

In the tables above, there is an AUTO_INCREMENT query where the numbers entered into the column will be added automatically, so there is no need to write them manually, and also the ContID is unique, there can be no twin numbers in it.

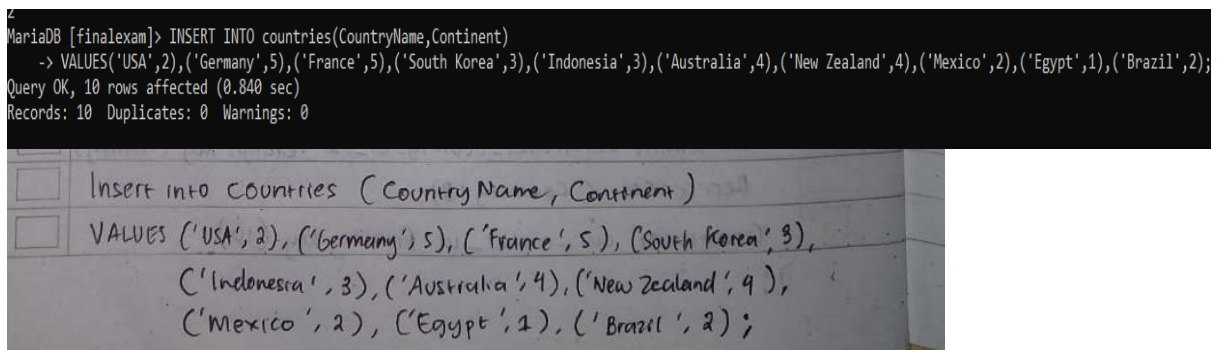
In the tables above, there is also a CONSTRAINT query, namely, to protect against incorrect validation in our program, so that the data that enters the database will be maintained. Relations between tables can use FOREIGN KEY and I also create multiple tables using the same primary key.

In the table above the relationship that I use between tables is ONE TO ONE. At ONE TO ONE the way to make it is quite easy. We can create a foreign key column, then set the column using UNIQUE KEY, this can prevent duplicate data from occurring in the column or we can create a table with the same primary key, so we no longer need columns for FOREIGN KEY.

B. Insert data into Continent



Insert data into Countries



Insert data into Car Makers

```
Insert into car_makers (ID, Maker, FullName, Country)
VALUES (1, 'amc', 'American Motor Company', 1),
       (2, 'volkswagen', 'Volkswagen', 2),
       (3, 'citroen', 'Citroen', 3),
       (4, 'hyundai', 'Hyundai', 4),
       (5, 'viar', 'Viar', 5),
       (6, 'hi', 'hi', null),
       (7, 'kia', 'Kia Motor', 4);
```

```
MariaDB [finalexam]>
MariaDB [finalexam]> INSERT INTO car_makers(ID,Maker,FullName,Country)
-> VALUES(1,'amc','American Motor Company',1),
-> (2,'volkswagen','Volkswagen',2),
-> (3,'citroen','Citroen',3),
-> (4,'hyundai','Hyundai',4),
-> (5,'viar','viar',5),
-> (6,'hi','hi',null),
-> (7,'kia','Kia Motors',4);
Query OK, 7 rows affected (0.513 sec)
Records: 7 Duplicates: 0 Warnings: 0
```

Insert data into Model Details

```
Insert into model_details ( Model ID, Maker, Model)
VALUES (1,2,'amc'), (2,2,'audi'), (3,3,'bmw'), (4,4,'buick'),
       (5,4,'cadillac'), (6,3,'citroen'), (8,7,'chrysler'),
       (7,4,'datsun'), (9,6,'dodge');
```

```
MariaDB [finalexam]>
MariaDB [finalexam]>
MariaDB [finalexam]> INSERT INTO MODEL_DETAILS (ModelID,Maker,Model)
-> VALUES(1,1,'amc'),(2,2,'audi'),(3,3,'bmw'),(4,4,'buick'),(5,4,'cadillac'),(6,3,'citroen'),(8,7,'chrysler'),(7,4,'datsun'),(9,6,'dodge');
Query OK, 9 rows affected (0.106 sec)
Records: 9 Duplicates: 0 Warnings: 0
```


Insert data into Car Names

```

Insert into car_names (ID, Model, Descr)
VALUES (1, 'chrysler', 'chrysler newport royal'),
(2, 'bmw', 'bmw 2002'),
(3, 'chrysler', 'chrysler new yorker brougham'),
(4, 'buick', 'buick electra 225 custom'),
(5, 'audi', 'audi 100ls'),
(6, 'buick', 'buick skylark 320'),
(7, 'citroen', 'citroen ds-21 pallas'),
(8, 'dodge', 'dodge challenger'),
(9, 'buick', 'buick estate wagon'),
(10, 'amc', 'amc hornet'),
(11, 'datson', 'datson p1510'),
(12, 'audi', 'audi 100 ls'),
(13, 'dodge', 'dodge d200'),
(14, 'datson', 'datson p1510'),
(15, 'chrysler', 'chrysler newport royal');

```

```

MariaDB [finalexam]> INSERT INTO car_names(ID,Model,Descr)
-> VALUES (1,'chrysler','chrysler newport royal'),
-> (2,'bmw','bmw 2002'),
-> (3,'chrysler','chrysler new yorker brougham'),
-> (4,'buick','buick electra 225 custom'),
-> (5,'audi','audi 100ls'),
-> (6,'buick','buick skylark 320'),
-> (7,'citroen','citroen ds-21 pallas'),
-> (8,'dodge','dodge challenger'),
-> (9,'buick','buick estate wagon (sw)'),
-> (10,'amc','amc hornet'),
-> (11,'datson','datson p1510'),
-> (12,'audi','audi 100 ls'),
-> (13,'dodge','dodge d200'),
-> (14,'datson','datson p1510'),
-> (15,'chrysler','chrysler newport royal');
Query OK, 15 rows affected, 1 warning (0.136 sec)
Records: 15 Duplicates: 0 Warnings: 1

```

Display Table of Continents and Countries

```
MariaDB [finalexam]> SELECT * FROM continents;
+-----+-----+
| ContID | Continent |
+-----+-----+
| 1 | Africa |
| 2 | America |
| 3 | Asia |
| 4 | Australia |
| 5 | Europe |
+-----+-----+
5 rows in set (0.002 sec)

MariaDB [finalexam]> SELECT * FROM countries;
+-----+-----+-----+
| CountryID | CountryName | Continent |
+-----+-----+-----+
| 1 | USA | 2 |
| 2 | Germany | 5 |
| 3 | France | 5 |
| 4 | South Korea | 3 |
| 5 | Indonesia | 3 |
| 6 | Australia | 4 |
| 7 | New Zealand | 4 |
| 8 | Mexico | 2 |
| 9 | Egypt | 1 |
| 10 | Brazil | 2 |
+-----+-----+-----+
10 rows in set (0.001 sec)
```

Display Table of Model Details and Car Makers

```
MariaDB [finalexam]> SELECT * FROM model_details;
+-----+-----+-----+
| ModelID | Maker | Model |
+-----+-----+-----+
| 1 | 1 | amc |
| 2 | 2 | audi |
| 3 | 3 | bmw |
| 4 | 4 | buick |
| 5 | 4 | cadillac |
| 6 | 3 | citroen |
| 7 | 4 | datsun |
| 8 | 7 | chrysler |
| 9 | 6 | dodge |
+-----+-----+-----+
9 rows in set (0.001 sec)

MariaDB [finalexam]> SELECT * FROM car_makers;
+-----+-----+-----+-----+
| ID | Maker | FullName | Country |
+-----+-----+-----+-----+
| 1 | amc | American Motor Company | 1 |
| 2 | volkswagen | Volkswagen | 2 |
| 3 | citroen | Citroen | 3 |
| 4 | hyundai | Hyundai | 4 |
| 5 | viar | viar | 5 |
| 6 | hi | hi | NULL |
| 7 | kia | Kia Motors | 4 |
+-----+-----+-----+-----+
7 rows in set (0.002 sec)
```

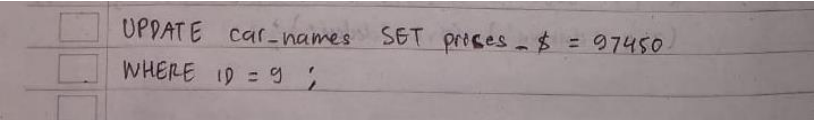
Display Table of Car Names:

```
MariaDB [finalexam]> SELECT * FROM car_names;
```

ID	Model	Descr
1	chrysler	chrysler newport royal
2	bmw	bmw 2002
3	chrysler	chrysler new yorker broug
4	buick	buick electra 225 custom
5	audi	audi 100ls
6	buick	buick skylark 320
7	citroen	citroen ds-21 pallas
8	dodge	dodge challenger
9	buick	buick estate wagon (sw)
10	amc	amc hornet
11	datsum	datsum pl510
12	audi	audi 100 ls
13	dodge	dodge d200
14	datsum	datsum pl510
15	chrysler	chrysler newport royal

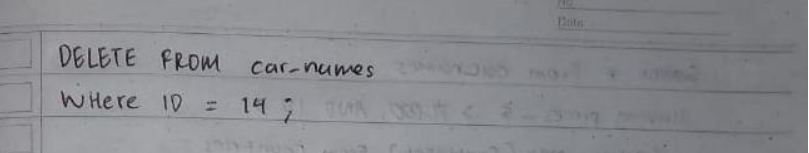
```
15 rows in set (0.001 sec)
```

Update data of Car Names:



```
MariaDB [finalexam]> UPDATE car_names SET prices_$ = 97450 WHERE ID = 9;
Query OK, 1 row affected (0.077 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

Delete data of Car Names:



```
MariaDB [finalexam]> DELETE FROM car_names WHERE ID = 14;
Query OK, 1 row affected (0.077 sec)
```

>>> Explanations:

Insert data is used to enter various kinds of data or content into a table.

An **update** is used to change the value of a column with a specific id. In the picture above I changed the prices_\$ at ID = 9 to 97450 dollars. **Delete** is used to delete one of the contents in the table. In the picture above I deleted the contents at ID = 14.

C. JOIN

```
Select * From model_details
Join car_makers ON (model_details.Maker = car_makers.ID);
```

```
MariaDB [finalexam]> SELECT * FROM model_details
-> JOIN car_makers ON (model_details.Maker = car_makers.ID);
MariaDB [finalexam]> SELECT * FROM model_details
-> JOIN car_makers ON (model_details.Maker = car_makers.ID);
```

ModelID	Maker	Model	ID	Maker	FullName	Country
1	1	amc	1	amc	American Motor Company	1
2	2	audi	2	volkswagen	Volkswagen	2
3	3	bmw	3	citroen	Citroen	3
4	4	buick	4	hyundai	Hyundai	4
5	4	cadillac	4	hyundai	Hyundai	4
6	3	citroen	3	citroen	Citroen	3
7	4	datsun	4	hyundai	Hyundai	4
8	7	chrysler	7	kia	Kia Motors	4
9	6	dodge	6	hi	hi	NULL

9 rows in set (0.288 sec)

LEFT JOIN

```
Select * from countries
Left join continents ON (countries.Continent = continents.ContID);
```

```
MariaDB [finalexam]> SELECT * FROM countries
-> LEFT JOIN continents ON (countries.Continent = continents.ContID);
MariaDB [finalexam]> SELECT * FROM countries
-> LEFT JOIN continents ON (countries.Continent = continents.ContID);
```

CountryID	CountryName	Continent	ContID	Continent
1	USA	2	2	America
2	Germany	5	5	Europe
3	France	5	5	Europe
4	South Korea	3	3	Asia
5	Indonesia	3	3	Asia
6	Australia	4	4	Australia
7	New Zealand	4	4	Australia
8	Mexico	2	2	America
9	Egypt	1	1	Africa
10	Brazil	2	2	America

10 rows in set (0.067 sec)

RIGHT JOIN

```
Select * From car_names
Right join model_details ON (car_names.Model = model_details.Model);
```

```
MariaDB [finalexam]> SELECT * FROM car_names
-> RIGHT JOIN model_details ON (car_names.Model = model_details.Model);
MariaDB [finalexam]> SELECT * FROM car_names
-> RIGHT JOIN model_details ON (car_names.Model = model_details.Model);
```

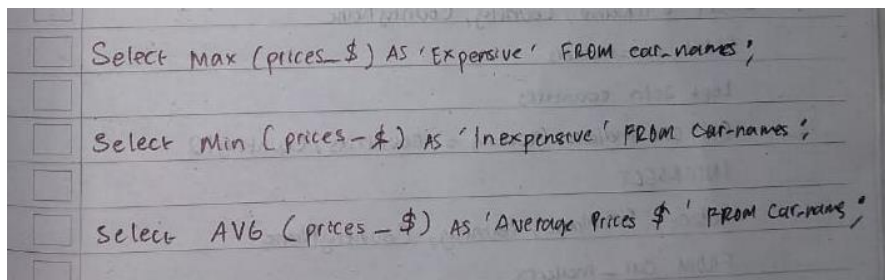
ID	Model	Descr	prices_\$	ModelID	Maker	Model
10	amc	amc hornet	45600	1	1	amc
5	audi	audi 100ls	57900	2	2	audi
12	audi	audi 100 ls	45600	2	2	audi
2	bmw	bmw 2002	87900	3	3	bmw
4	buick	buick electra 225 custom	57900	4	4	buick
6	buick	buick skylark 320	57900	4	4	buick
9	buick	buick estate wagon (sw)	97450	4	4	buick
NULL	NULL	NULL	NULL	5	4	cadillac
7	citroen	citroen ds-21 pallas	98600	6	3	citroen
11	datson	datson pl510	45600	7	4	datson
1	chrysler	chrysler newport royal	87900	8	7	chrysler
3	chrysler	chrysler new yorker broug	57900	8	7	chrysler
15	chrysler	chrysler newport royal	76000	8	7	chrysler
8	dodge	dodge challenger	89500	9	6	dodge
13	dodge	dodge d200	76900	9	6	dodge

15 rows in set (0.079 sec)

>>> Explanations:

In the image above I used the JOIN function to concatenate all the columns from the model_details table and the car_makers table. I use LEFT JOIN to combine all the columns in the Countries table and the Continents table which are combined starting from the left. I use RIGHT JOIN to combine from the table car_names and model_details which are combined starting from the right.

D. 3 OUT OF 5 AGGREGATE FUNCTIONS



```
MariaDB [finalexam]> SELECT MAX(prices_$) AS 'Expensive' FROM car_names;
+-----+
| Expensive |
+-----+
|      98600 |
+-----+
1 row in set (0.162 sec)

MariaDB [finalexam]> SELECT MIN(prices_$) AS 'Inexpensive' FROM car_names;
+-----+
| Inexpensive |
+-----+
|      45600 |
+-----+
1 row in set (0.001 sec)

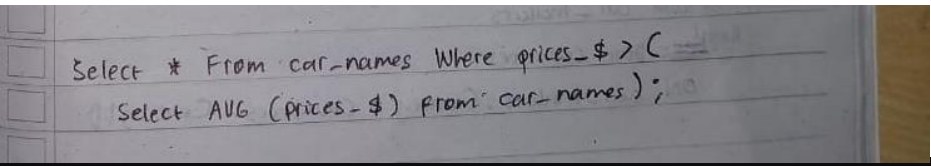
MariaDB [finalexam]> SELECT AVG(prices_$) AS 'Average Prices $' FROM car_names;
+-----+
| Average Prices $ |
+-----+
|      70253.5714 |
+-----+
1 row in set (0.079 sec)

MariaDB [finalexam]> _
```

>>> Explanations:

In the picture above you can see that I am using three of the five aggregate functions. I use the Max function to display the maximum price in the car_names table, the Min function to display the minimum price, and the AVG function to display the average price from the car_names table.

E. SUB-QUERY (NESTED QUERY)



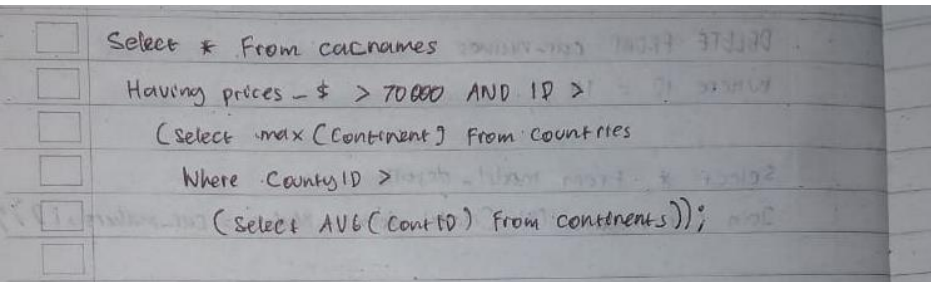
```
Select * From car_names Where prices_$ > (
Select AVG (prices_$) From car_names);
```

MariaDB [finalexam]> SELECT * FROM car_names
-> WHERE prices_\$ > (
-> SELECT AVG(prices_\$) FROM car_names
->);

ID	Model	Descr	prices_\$
1	chrysler	chrysler newport royal	87900
2	bmw	bmw 2002	87900
7	citroen	citroen ds-21 pallas	98600
8	dodge	dodge challenger	89500
9	buick	buick estate wagon (sw)	97450
13	dodge	dodge d200	76900
15	chrysler	chrysler newport royal	76900

7 rows in set (0.980 sec)

SUB-SUB-QUERY



```
Select * From carnames
Having prices_$ > 70000 AND ID >
(select max(Continent) From countries
Where CountryID >
(select AVG(ContID) From continents));
```

MariaDB [finalexam]> SELECT * FROM car_names
-> HAVING prices_\$ > 70000 AND ID >
-> (SELECT MAX(Continent) FROM countries
-> WHERE CountryID >
-> (SELECT AVG(ContID) FROM continents));

ID	Model	Descr	prices_\$
7	citroen	citroen ds-21 pallas	98600
8	dodge	dodge challenger	89500
9	buick	buick estate wagon (sw)	97450
13	dodge	dodge d200	76900
15	chrysler	chrysler newport royal	76900

5 rows in set (0.208 sec)

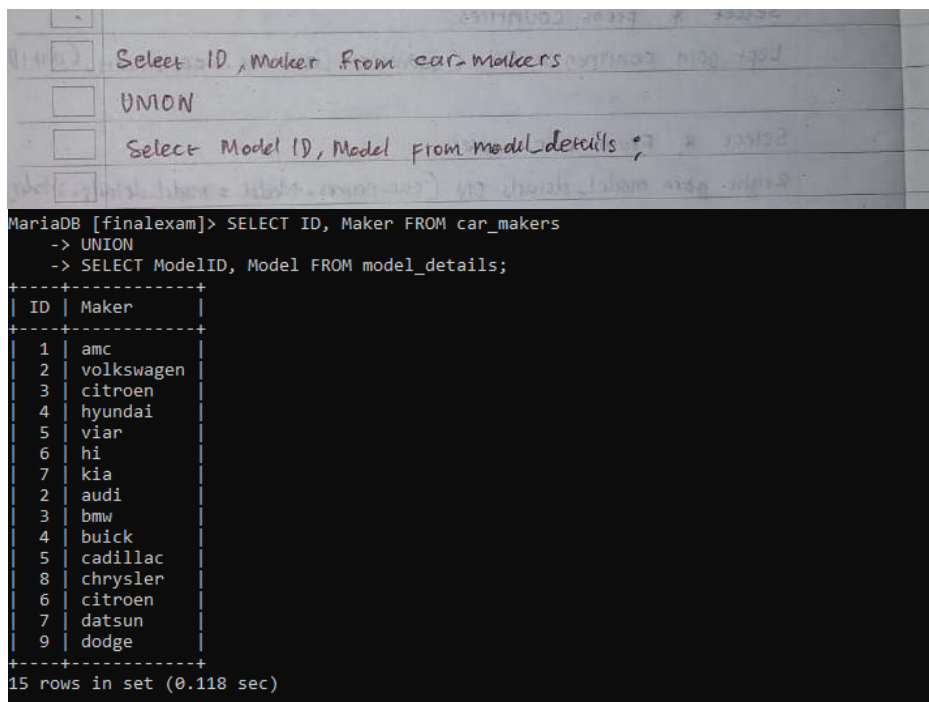
MariaDB [finalexam]>

>>> Explanations:

In the picture above, it can be explained that a subquery is a query that is in the query. The subquery above selects the price from the car_names table, where the price is greater than the average price in the table.

Sub nested queries are queries that are inside a nested query. Used to return data that will be used in the main query as a condition to further limit the data to be retrieved. In a nested sub-query or sub-query, it can be explained that it will display the car_names table by selecting it where the price is greater than 70000 dollars and the ID is greater than the CountryID value in the Countries table where the CountryID value must be greater than the average ID value. on the Continent table.

F. UNION AND



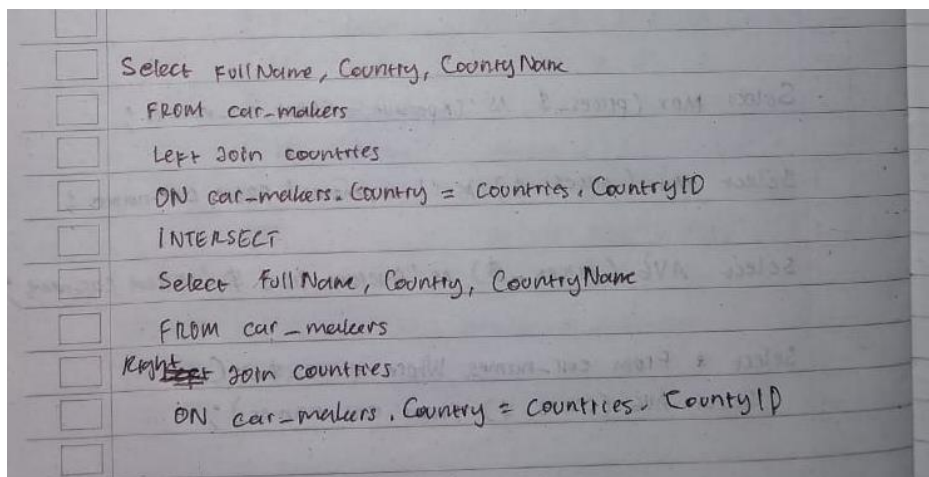
The image shows handwritten notes on lined paper and a screenshot of a MariaDB terminal. The notes describe a UNION query combining data from 'car_makers' and 'model_details' tables. The terminal shows the execution of this query, resulting in a table with 15 rows.

```
SELECT ID, Maker FROM car_makers
UNION
SELECT ModelID, Model FROM model_details;
```

ID	Maker
1	amc
2	volkswagen
3	citroen
4	hyundai
5	viar
6	hi
7	kia
2	audi
3	bmw
4	buick
5	cadillac
8	chrysler
6	citroen
7	datsum
9	dodge

15 rows in set (0.118 sec)

INTERSECTION



```
MariaDB [finalexam]> SELECT FullName, Country, CountryName
-> FROM car_makers
-> LEFT JOIN countries
-> ON car_makers.Country = countries.CountryID
-> INTERSECT
-> SELECT FullName, Country, CountryName
-> FROM car_makers
-> LEFT JOIN countries
-> ON car_makers.Country = countries.CountryID;
```

FullName	Country	CountryName
American Motor Company	1	USA
Volkswagen	2	Germany
Citroen	3	France
Hyundai	4	South Korea
viar	5	Indonesia
hi	NULL	NULL
Kia Motors	4	South Korea

```
7 rows in set (0.102 sec)

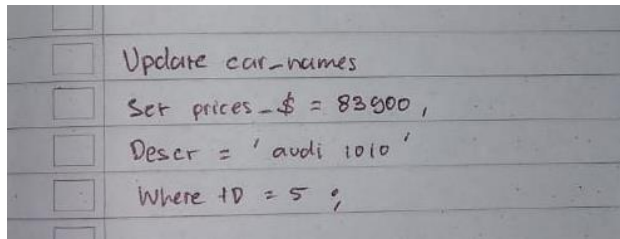
MariaDB [finalexam]>
```

>>> Explanations:

From the picture above, it can be explained that UNION functions to display the combined days of the contents of the two tables. Namely on the car_makers table and the model_details table with columns ID and Maker

Meanwhile, INTERSECTION is useful for displaying slices of two tables, namely the car_makers table and the countries table. In the FullName, Country, and CountryName columns.

G. IMPLEMENT MODIFY/CHANGE THE COLUMN



```
MariaDB [finalexam]> UPDATE car_names  
-> SET prices_$ = 83900,  
-> Descr = 'audi 1010'  
-> WHERE ID = 5;  
Query OK, 1 row affected (0.356 sec)  
Rows matched: 1 Changed: 1 Warnings: 0
```

```
MariaDB [finalexam]> SELECT * FROM car_names;
```

ID	Model	Descr	prices_\$
1	chrysler	chrysler newport royal	87900
2	bmw	bmw 2002	87900
3	chrysler	chrysler new yorker broug	57900
4	buick	buick electra 225 custom	57900
5	audi	audi 1010	83900
6	buick	buick skylark 320	57900
7	citroen	citroen ds-21 pallas	98600
8	dodge	dodge challenger	89500
9	buick	buick estate wagon (sw)	97450
10	amc	amc hornet	45600
11	datson	datson pl510	45600
12	audi	audi 100 ls	45600
13	dodge	dodge d200	76900
15	chrysler	chrysler newport royal	76900

```
14 rows in set (0.004 sec)
```

>>> Explanations:

From the picture above we can see that there is a change in the 5th ID, which changes the price to 83900 dollars and changes the description sentence to "audi 1010"

LAST DIGIT OF YOUR ID: 0, 5



PEROLEHAN KURSI PARTAI POLITIK
PEMILIHAN UMUM **ANGGOTA DPR RI** TAHUN 2019
PROVINSI SUMATERA UTARA



DAERAH PEMILIHAN

SUMUT I 10 KURSI				SUMUT II 10 KURSI				SUMUT III 10 KURSI			
PARTAI	JUMLAH KURSI	NAMA CALON TERPILIH	SUARA SAH	PARTAI	JUMLAH KURSI	NAMA CALON TERPILIH	SUARA SAH	PARTAI	JUMLAH KURSI	NAMA CALON TERPILIH	SUARA SAH
	2	RONO H.R. MUHAMMAD SYARI, SH., M.Hum.	125,169		1	MARWAN DASOPANG	60,762		1	Prof. DR. Ir. DZOHAR ARIFIN HUSIN	41,705
	2	M. HUSNI, SE	57,330		1	H. GUS IRAWAN PASARIBU, SE., AK., MM., CA	168,342		3	Dr. TEGAR SAPTA HENDRIK M.S., S.JUNJUNATT GIBANG, S.H., M.B.A., M.H. BOE ANGBA HANANA STEPU, S.H.	92,390 35,490 56,890
	1	YASONNA H. LAOLY, Ph.D	124,848		2	TRIMEDYA PANDAITAN, S.H., M.H.	106,103		2	H. AHMAD DOLI KURNIA TANDJUNG	63,771
	1	Dr. SOFYAN TAN	158,495		1	SIHAR P.H SITORUS	185,918		1	DELIA PRATIWI BR. SITEPU, S.H.	121,690
	1	MEUTYA HAFID	87,139		1	LAMHOT SINAGA	53,398		1	H. RUDI HARTONO BANGUN, S.E., MAG	45,814
	1	PRANANDA SURYA PALOH	95,676		2	MARTIN MANURUNG, SE, MA	63,755		1	H. ANSORY SIREGAR	71,333
	2	IR. H. TIFATUL SEMBIRING	127,223		1	DELMERIA	91,096		1	NASRIL BAHAR, SE	95,557
	2	H. HIDAYATULLAH, SE.	31,067		1	H. ISKAN QOLBA LUBIS	62,877		1	DR. HINCA IP PANDJAITAN XII, S.H., M.H., ACCS.	27,277
	1	MULFIACHRI HARAHAP, SH., MH	59,641		1	Dr. SALEH PARTAONAN DAULAY, M.Ag., M.Hum., MA.	142,683				
	1	HABDUL WAHAB DALIMUNTJE, SH	35,574		1	Dh. JHONNI ALLEN MARIKUN, MM	49,381				

LAST DIGIT OF YOUR ID: 1, 6

PT. JAVA ANIMA DARMAJA

One Stop Business and IT Solution

Jl. Cempaka Blok C3 No.24 Perum Beringin Raya, Kemiling. Bandar Lampung Telp 0721 271411

SLIP GAJI KARYAWAN

Periode 1 Juni 2013 – 30 Juni 2013

NIK : JAD-003
Nama : DENI SETIAWAN, S.Kom
Jabatan : Marketing Manager
Status : Karyawan Tetap

PENGHASILAN

Gaji Pokok = 1,800,000
Tj. Jabatan = 900,000
Tj. Konsumsi = 455,000
Tj. Harian = 520,000
Bonus Target = 752,000
Total (A) Rp 4,427,000

POTONGAN

PPh 21 = 180,000
Asuransi = -
Total (B) Rp 180,000

PENERIMAAN BERSIH (A – B) = Rp 4,247,000

Terbilang: # empat juta empat ratus dua puluh tujuh ribu rupiah #

B. Lampung, 30 Juni 2013

Manajer Operasional

Yulitiawati, SE.

LAST DIGIT OF YOUR ID: 2, 7

	A	B	C	D	E	F	G	H
3								
4	No	Tanggal	Kode Barang	Nomor Bukti	Nama Barang	Keterangan	Mutasi	
5							Pembelian	Penjualan
6	1	1-Aug-18	K-002	IV-001	Monitor 19"	Penjualan	-	2
7	2	5-Aug-18	K-004	IV-002	Laptop Toshiba	Penjualan	-	2
8	3	5-Aug-18	K-005	IV-002	Notebook Advan	Penjualan	-	1
9	4	5-Aug-18	K-007	IV-002	Flashdisk SANDISK	Penjualan	-	1
10	5	5-Aug-18	K-002	IV-003	Monitor 19"	Penjualan	-	1
11	6	6-Aug-18	K-004	IV-004	Laptop Toshiba	Penjualan	-	1
12	7	10-Aug-18	K-009	IV-005	DVD External	Penjualan	-	1
13	8	16-Aug-18	K-002	PB-001	Monitor 19"	Pembelian	6	-
14	9	16-Aug-18	K-005	IV-006	Notebook Advan	Penjualan	-	2
15	10	16-Aug-18	K-002	IV-006	Monitor 19"	Penjualan	-	1
16	11	16-Aug-18	K-005	IV-006	Notebook Advan	Penjualan	-	4
17	12	25-Aug-18	K-004	IV-007	Laptop Toshiba	Penjualan	-	1
18	13	26-Aug-18	K-005	PB-002	Notebook Advan	Pembelian	1	-
19	14	26-Aug-18	K-003	IV-008	Monitor 15"	Penjualan	-	1
20	15	26-Aug-18	K-002	IV-008	Monitor 19"	Penjualan	-	1
21	16	28-Aug-18	K-003	IV-009	Monitor 15"	Penjualan	-	2
22	17	29-Aug-18	K-004	IV-010	Laptop Toshiba	Penjualan	-	2
23	18	29-Aug-18	K-003	PB-003	Monitor 15"	Pembelian	3	-
24	19	31-Aug-18	K-008	IV-011	Keyboard Logitech	Penjualan	-	1
25								

LAST DIGIT OF YOUR ID: 3, 8

Daftar Hadir Siswa																												cara-excel.blogspot.com																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																
Semester Genap Tahun Pembelajaran 2014-2015																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																												
Kelas : 2A					Jumlah Siswa : 5 orang					Laki-Laki : 2 orang					Perempuan : 3 orang					Wali Kelas :																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																								
FEBRUARI 2015																												Rekap																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																
No.	No. Induk	No. NISN	Nama	L/P	Kelas	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J	S	M	S	R	k	J

LAST DIGIT OF YOUR ID: 4, 9

ORDER DETAILS

WEDNESDAY, 14 APRIL 2021

ORDER ID: RB-196709-2695666



 **PICK UP • 11:33**

Smk Iptek

Jl. Raya Sengkol, Muncul, Setu, Kota Tangerang Selatan, Banten 15314, Indonesia

 **DESTINATION • 11:44**

Serpong Train Station

Serpong, Kecamatan Serpong, Banten, 15310 Indonesia



Usnaeni Usnaeni

DISTANCE
4.8 km

| Travel time
00:10:37

Fare (4.8 km)	Rp14.000
Platform Fee	Rp1.000
Voucher	-Rp2.000
TOTAL PAYMENT	Rp13.000

Paid with GoPay	Rp13.000
-----------------	-----------------