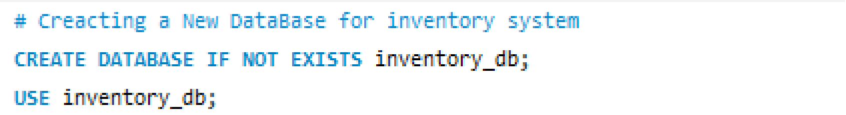
OUTPUT-AGUIRRE, PAUL VINCENT S.
COMPROG-CP102

PART 1: SQL – DDL and DML Tasks

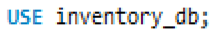
Create and execute an SQL script that performs the following:

1. ***Create a new MySQL database.***
    - Attached at the bottom is a screen shot taken of my coded from my MySQL Workbench and here is the first step into creating a database.



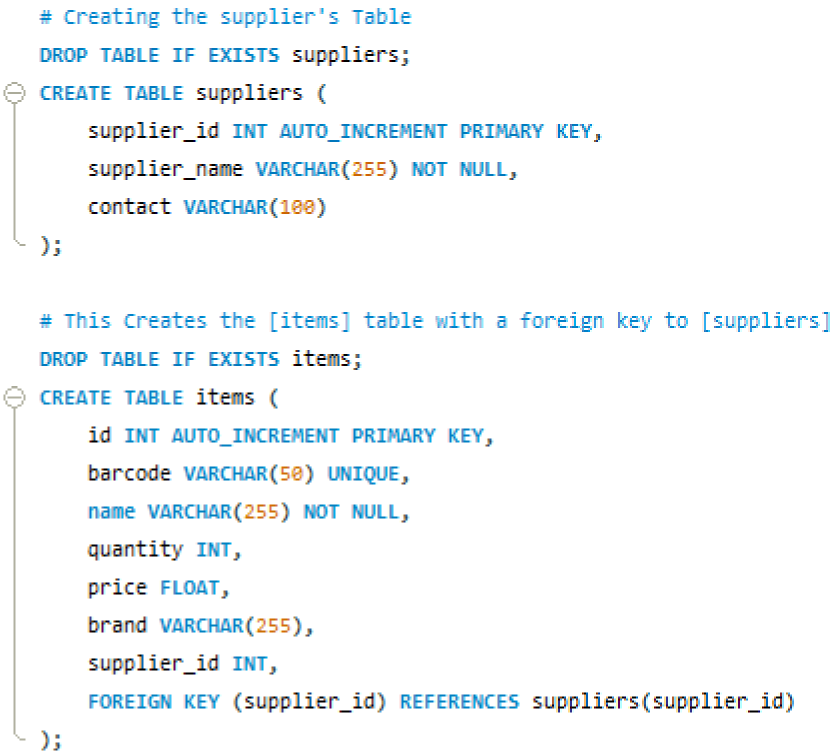
```
# Creacting a New DataBase for inventory system
CREATE DATABASE IF NOT EXISTS inventory_db;
USE inventory_db;
```

2. ***Select the database for use.***
    - Using USE to use the data base we just created.



```
USE inventory_db;
```

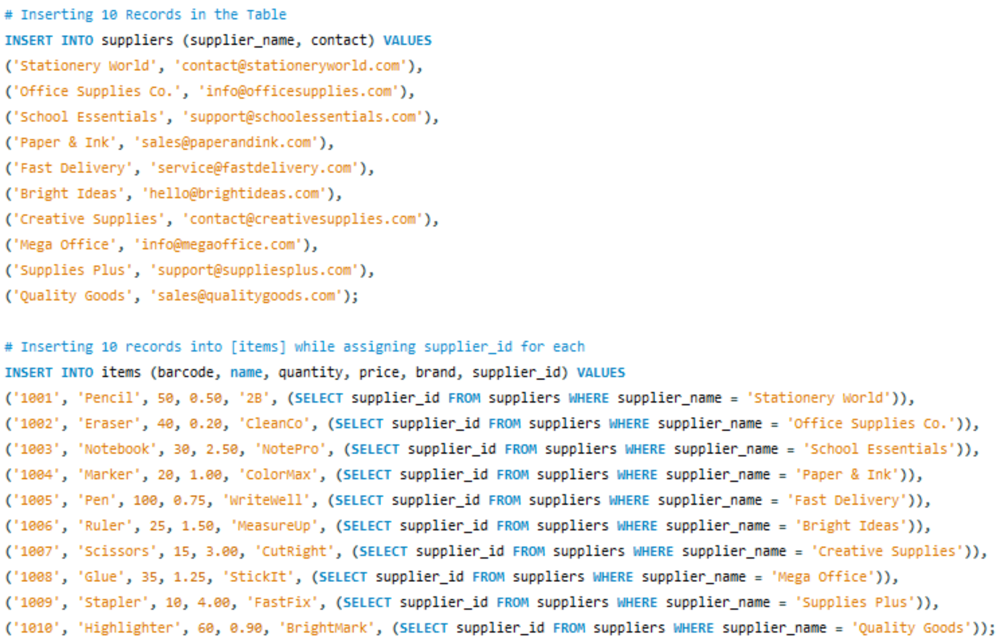
3. ***Create at least two tables with appropriate fields, data types, and primary key.***
    - We now try to make 2 TABLE using CREATE TABLE with the names of [items] and [suppliers].



```
# Creating the supplier's Table
DROP TABLE IF EXISTS suppliers;
CREATE TABLE suppliers (
    supplier_id INT AUTO_INCREMENT PRIMARY KEY,
    supplier_name VARCHAR(255) NOT NULL,
    contact VARCHAR(100)
);

# This Creates the [items] table with a foreign key to [suppliers]
DROP TABLE IF EXISTS items;
CREATE TABLE items (
    id INT AUTO_INCREMENT PRIMARY KEY,
    barcode VARCHAR(50) UNIQUE,
    name VARCHAR(255) NOT NULL,
    quantity INT,
    price FLOAT,
    brand VARCHAR(255),
    supplier_id INT,
    FOREIGN KEY (supplier_id) REFERENCES suppliers(supplier_id)
);
```

4. ***Insert a minimum of 10 records into each table.***
    - Inputting a sample data.

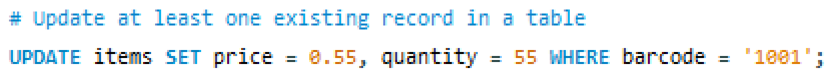


```
# Inserting 10 Records in the Table
INSERT INTO suppliers (supplier_name, contact) VALUES
('Stationery World', 'contact@stationeryworld.com'),
('Office Supplies Co.', 'info@officesupplies.com'),
('School Essentials', 'support@schoolessentials.com'),
('Paper & Ink', 'sales@paperandink.com'),
('Fast Delivery', 'service@fastdelivery.com'),
('Bright Ideas', 'hello@brightideas.com'),
('Creative Supplies', 'contact@creativesupplies.com'),
('Mega Office', 'info@megaoffice.com'),
('Supplies Plus', 'support@suppliesplus.com'),
('Quality Goods', 'sales@qualitygoods.com');

# Inserting 10 records into [items] while assigning supplier_id for each
INSERT INTO items (barcode, name, quantity, price, brand, supplier_id) VALUES
('1001', 'Pencil', 50, 0.50, '2B', (SELECT supplier_id FROM suppliers WHERE supplier_name = 'Stationery World')),
('1002', 'Eraser', 40, 0.20, 'CleanCo', (SELECT supplier_id FROM suppliers WHERE supplier_name = 'Office Supplies Co.')),
('1003', 'Notebook', 30, 2.50, 'NotePro', (SELECT supplier_id FROM suppliers WHERE supplier_name = 'School Essentials')),
('1004', 'Marker', 20, 1.00, 'ColorMax', (SELECT supplier_id FROM suppliers WHERE supplier_name = 'Paper & Ink')),
('1005', 'Pen', 100, 0.75, 'WriteWell', (SELECT supplier_id FROM suppliers WHERE supplier_name = 'Fast Delivery')),
('1006', 'Ruler', 25, 1.50, 'MeasureUp', (SELECT supplier_id FROM suppliers WHERE supplier_name = 'Bright Ideas')),
('1007', 'Scissors', 15, 3.00, 'CutRight', (SELECT supplier_id FROM suppliers WHERE supplier_name = 'Creative Supplies')),
('1008', 'Glue', 35, 1.25, 'StickIt', (SELECT supplier_id FROM suppliers WHERE supplier_name = 'Mega Office')),
('1009', 'Stapler', 10, 4.00, 'FastFix', (SELECT supplier_id FROM suppliers WHERE supplier_name = 'Supplies Plus')),
('1010', 'Highlighter', 60, 0.90, 'BrightMark', (SELECT supplier_id FROM suppliers WHERE supplier_name = 'Quality Goods'));
```

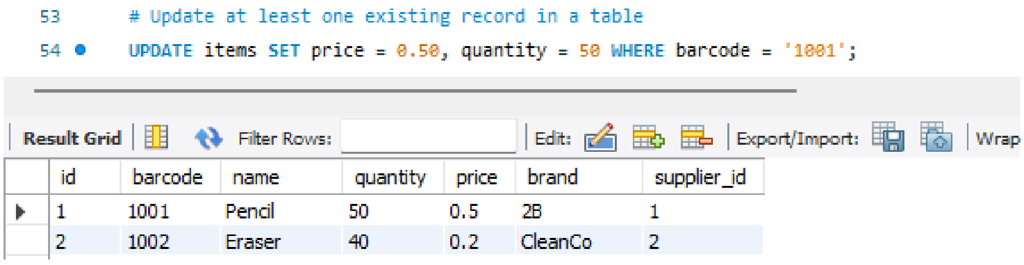
5. ***Update at least one existing record in a table.***

- CODE;



```
# Update at least one existing record in a table
UPDATE items SET price = 0.55, quantity = 55 WHERE barcode = '1001';
```

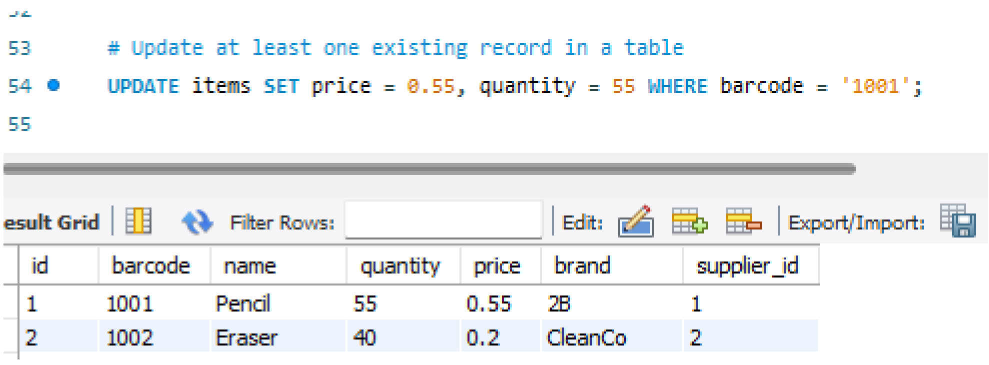
- Before ITEM;



```
# Update at least one existing record in a table
UPDATE items SET price = 0.50, quantity = 50 WHERE barcode = '1001';
```

id	barcode	name	quantity	price	brand	supplier_id
1	1001	Pencil	50	0.5	2B	1
2	1002	Eraser	40	0.2	CleanCo	2

- After ITEM;

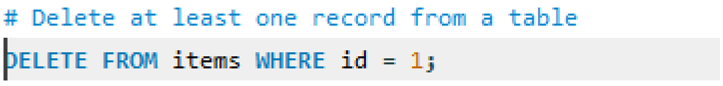


```
# Update at least one existing record in a table
UPDATE items SET price = 0.55, quantity = 55 WHERE barcode = '1001';
```

id	barcode	name	quantity	price	brand	supplier_id
1	1001	Pencil	55	0.55	2B	1
2	1002	Eraser	40	0.2	CleanCo	2

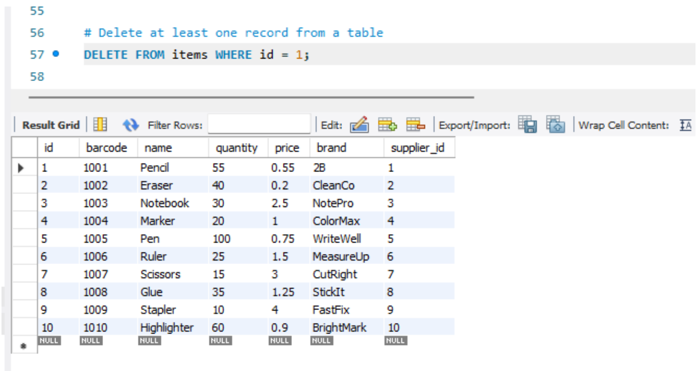
6. ***Delete at least one record from a table.***

- CODE;



```
# Delete at least one record from a table
DELETE FROM items WHERE id = 1;
```

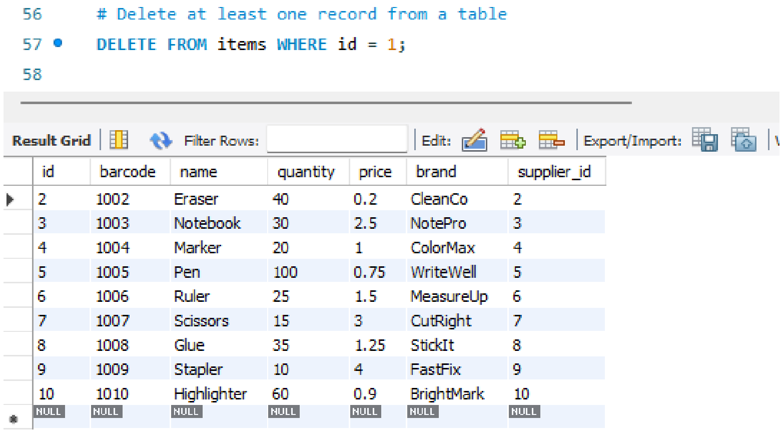
- In Items (Before);



```
# Delete at least one record from a table
DELETE FROM items WHERE id = 1;
```

id	barcode	name	quantity	price	brand	supplier_id
1	1001	Pencil	55	0.55	2B	1
2	1002	Eraser	40	0.2	CleanCo	2
3	1003	Notebook	30	2.5	NotePro	3
4	1004	Marker	20	1	ColorMax	4
5	1005	Pen	100	0.75	WriteWell	5
6	1006	Ruler	25	1.5	MeasureUp	6
7	1007	Scissors	15	3	CutRight	7
8	1008	Glue	35	1.25	StickIt	8
9	1009	Stapler	10	4	FastFix	9
10	1010	Highlighter	60	0.9	BrightMark	10
NULL	NULL	NULL	NULL	NULL	NULL	NULL

- In Items (After);



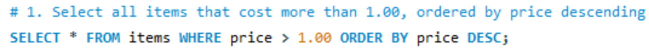
```
# Delete at least one record from a table
DELETE FROM items WHERE id = 1;
```

id	barcode	name	quantity	price	brand	supplier_id
2	1002	Eraser	40	0.2	CleanCo	2
3	1003	Notebook	30	2.5	NotePro	3
4	1004	Marker	20	1	ColorMax	4
5	1005	Pen	100	0.75	WriteWell	5
6	1006	Ruler	25	1.5	MeasureUp	6
7	1007	Scissors	15	3	CutRight	7
8	1008	Glue	35	1.25	StickIt	8
9	1009	Stapler	10	4	FastFix	9
10	1010	Highlighter	60	0.9	BrightMark	10
NULL	NULL	NULL	NULL	NULL	NULL	NULL

OUTPUT-AGUIRRE, PAUL VINCENT S.
COMPROG-CP102

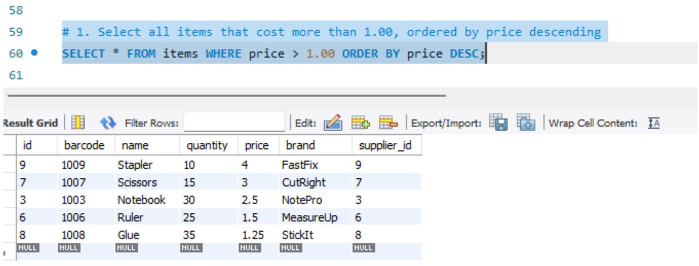
7. ***Write at least five SELECT statements that retrieve specific records based on different criteria (e.g., filtering, ordering, etc.).***

- CODE;



```
# 1. Select all items that cost more than 1.00, ordered by price descending
SELECT * FROM items WHERE price > 1.00 ORDER BY price DESC;
```

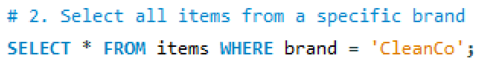
- Output;



```
# 1. Select all items that cost more than 1.00, ordered by price descending
SELECT * FROM items WHERE price > 1.00 ORDER BY price DESC;
```

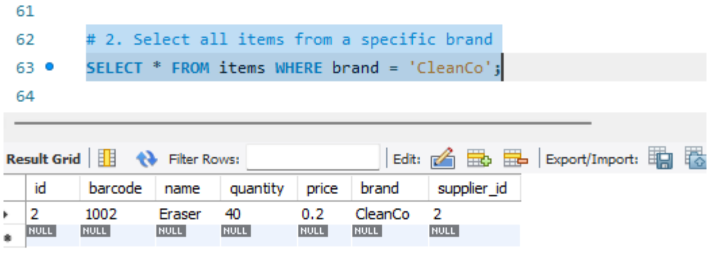
id	barcode	name	quantity	price	brand	supplier_id
9	1009	Stapler	10	4	FastFix	9
7	1007	Scissors	15	3	CutRight	7
3	1003	Notebook	30	2.5	NotePro	3
6	1006	Ruler	25	1.5	MeasureUp	6
8	1008	Glue	35	1.25	StickIt	8
NULL	NULL	NULL	NULL	NULL	NULL	NULL

- CODE (2);



```
# 2. Select all items from a specific brand
SELECT * FROM items WHERE brand = 'CleanCo';
```

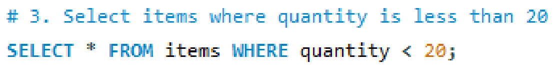
- Output (2);



```
# 2. Select all items from a specific brand
SELECT * FROM items WHERE brand = 'CleanCo';
```

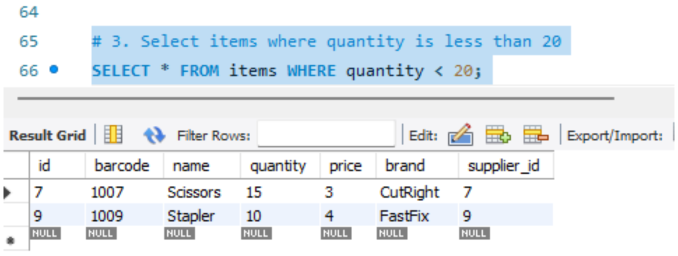
id	barcode	name	quantity	price	brand	supplier_id
2	1002	Eraser	40	0.2	CleanCo	2
NULL	NULL	NULL	NULL	NULL	NULL	NULL

- CODE (3);



```
# 3. Select items where quantity is less than 20
SELECT * FROM items WHERE quantity < 20;
```

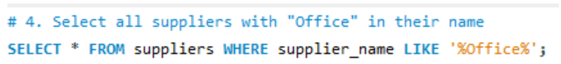
- Output (3);



```
# 3. Select items where quantity is less than 20
SELECT * FROM items WHERE quantity < 20;
```

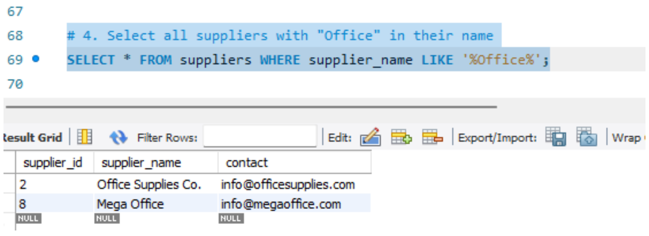
id	barcode	name	quantity	price	brand	supplier_id
7	1007	Scissors	15	3	CutRight	7
9	1009	Stapler	10	4	FastFix	9
NULL	NULL	NULL	NULL	NULL	NULL	NULL

- CODE (4);



```
# 4. Select all suppliers with "Office" in their name
SELECT * FROM suppliers WHERE supplier_name LIKE '%Office%';
```

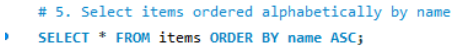
- Output (4);



```
# 4. Select all suppliers with "Office" in their name
SELECT * FROM suppliers WHERE supplier_name LIKE '%Office%';
```

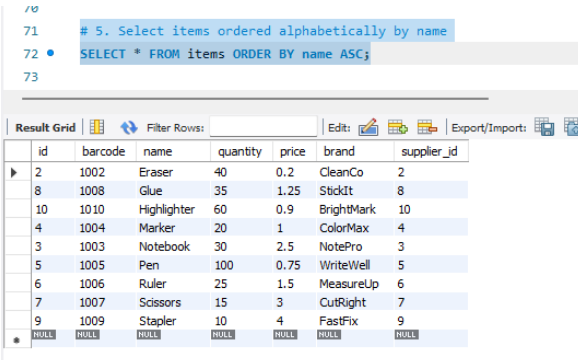
supplier_id	supplier_name	contact
2	Office Supplies Co.	info@officesupplies.com
8	Mega Office	info@megaoffice.com
NULL	NULL	NULL

- CODE (5);



```
# 5. Select items ordered alphabetically by name
SELECT * FROM items ORDER BY name ASC;
```

- Output (5);

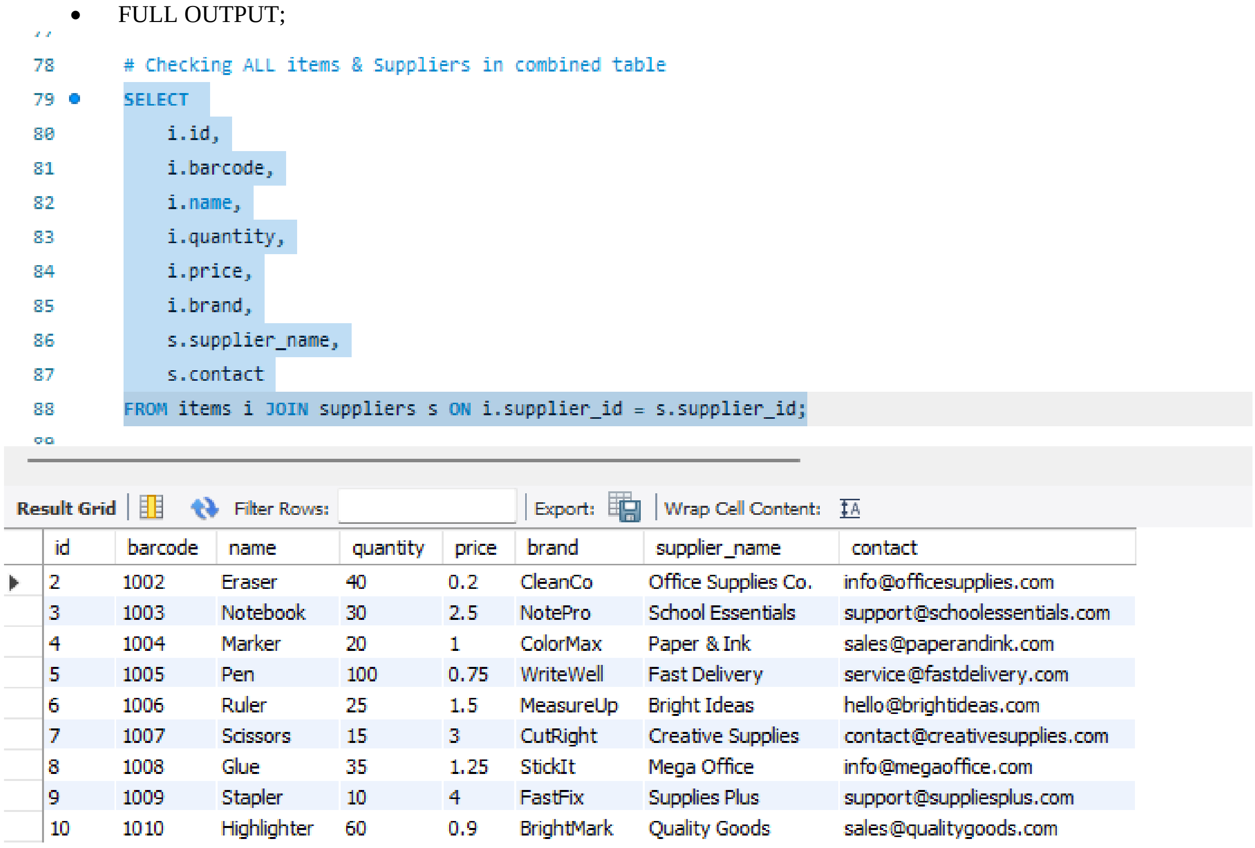


```
# 5. Select items ordered alphabetically by name
SELECT * FROM items ORDER BY name ASC;
```

id	barcode	name	quantity	price	brand	supplier_id
2	1002	Eraser	40	0.2	CleanCo	2
8	1008	Glue	35	1.25	StickIt	8
10	1010	Highlighter	60	0.9	BrightMark	10
4	1004	Marker	20	1	ColorMax	4
3	1003	Notebook	30	2.5	NotePro	3
5	1005	Pen	100	0.75	WriteWell	5
6	1006	Ruler	25	1.5	MeasureUp	6
7	1007	Scissors	15	3	CutRight	7
9	1009	Stapler	10	4	FastFix	9
NULL	NULL	NULL	NULL	NULL	NULL	NULL

OUTPUT-AGUIRRE, PAUL VINCENT S.
COMPROG-CP102

- FULL OUTPUT;



```
# Checking ALL items & Suppliers in combined table
SELECT
    i.id,
    i.barcode,
    i.name,
    i.quantity,
    i.price,
    i.brand,
    s.supplier_name,
    s.contact
FROM items i JOIN suppliers s ON i.supplier_id = s.supplier_id;
```

id	barcode	name	quantity	price	brand	supplier_name	contact
2	1002	Eraser	40	0.2	CleanCo	Office Supplies Co.	info@officesupplies.com
3	1003	Notebook	30	2.5	NotePro	School Essentials	support@schoolessentials.com
4	1004	Marker	20	1	ColorMax	Paper & Ink	sales@paperandink.com
5	1005	Pen	100	0.75	WriteWell	Fast Delivery	service@fastdelivery.com
6	1006	Ruler	25	1.5	MeasureUp	Bright Ideas	hello@brightideas.com
7	1007	Scissors	15	3	CutRight	Creative Supplies	contact@creativesupplies.com
8	1008	Glue	35	1.25	StickIt	Mega Office	info@megaoffice.com
9	1009	Stapler	10	4	FastFix	Supplies Plus	support@suppliesplus.com
10	1010	Highlighter	60	0.9	BrightMark	Quality Goods	sales@qualitygoods.com

[CODE]

```
# Creacting a New DataBase for inventory system

CREATE DATABASE IF NOT EXISTS inventory_db;

USE inventory_db;


# Creating the supplier's Table

DROP TABLE IF EXISTS suppliers;

CREATE TABLE suppliers (

   supplier_id INT AUTO_INCREMENT PRIMARY KEY,

   supplier_name VARCHAR(255) NOT NULL,

   contact VARCHAR(100)

);


# This Creates the [items] table with a foreign key to [suppliers]

DROP TABLE IF EXISTS items;

CREATE TABLE items (

   id INT AUTO_INCREMENT PRIMARY KEY,

   barcode VARCHAR(50) UNIQUE,

   name VARCHAR(255) NOT NULL,

   quantity INT,

   price FLOAT,

   brand VARCHAR(255),

   supplier_id INT,

   FOREIGN KEY (supplier_id) REFERENCES suppliers(supplier_id)

);
```

```
# Inserting 10 Records in the Table

INSERT INTO suppliers (supplier_name, contact) VALUES

('Stationery World', 'contact@stationeryworld.com'),

('Office Supplies Co.', 'info@officesupplies.com'),

('School Essentials', 'support@schoolessentials.com'),

('Paper & Ink', 'sales@paperandink.com'),

('Fast Delivery', 'service@fastdelivery.com'),

('Bright Ideas', 'hello@brightideas.com'),

('Creative Supplies', 'contact@creativesupplies.com'),

('Mega Office', 'info@megaoffice.com'),

('Supplies Plus', 'support@suppliesplus.com'),

('Quality Goods', 'sales@qualitygoods.com');


# Insert 10 records into [items] while assigning supplier_id for each

INSERT INTO items (barcode, name, quantity, price, brand, supplier_id) VALUES

('1001', 'Pencil', 50, 0.50, '2B', (SELECT supplier_id FROM suppliers WHERE supplier_name = 'Stationery World')),

('1002', 'Eraser', 40, 0.20, 'CleanCo', (SELECT supplier_id FROM suppliers WHERE supplier_name = 'Office Supplies Co.')),

('1003', 'Notebook', 30, 2.50, 'NotePro', (SELECT supplier_id FROM suppliers WHERE supplier_name = 'School Essentials')),

('1004', 'Marker', 20, 1.00, 'ColorMax', (SELECT supplier_id FROM suppliers WHERE supplier_name = 'Paper & Ink')),

('1005', 'Pen', 100, 0.75, 'WriteWell', (SELECT supplier_id FROM suppliers WHERE supplier_name = 'Fast Delivery')),

('1006', 'Ruler', 25, 1.50, 'MeasureUp', (SELECT supplier_id FROM suppliers WHERE supplier_name = 'Bright Ideas')),
('1007', 'Scissors', 15, 3.00, 'CutRight', (SELECT supplier_id FROM suppliers WHERE supplier_name = 'Creative Supplies')),

('1008', 'Glue', 35, 1.25, 'StickIt', (SELECT supplier_id FROM suppliers WHERE supplier_name = 'Mega Office')),

('1009', 'Stapler', 10, 4.00, 'FastFix', (SELECT supplier_id FROM suppliers WHERE supplier_name = 'Supplies Plus')),

('1010', 'Highlighter', 60, 0.90, 'BrightMark', (SELECT supplier_id FROM suppliers WHERE supplier_name = 'Quality Goods'));



# Update at least one existing record in a table

UPDATE items SET price = 0.55, quantity = 55 WHERE barcode = '1001';


# Delete at least one record from a table

DELETE FROM items WHERE id = 1;


# 1. Select all items that cost more than 1.00, ordered by price descending

SELECT * FROM items WHERE price > 1.00 ORDER BY price DESC;


# 2. Select all items from a specific brand

SELECT * FROM items WHERE brand = 'CleanCo';


# 3. Select items where quantity is less than 20

SELECT * FROM items WHERE quantity < 20;


# 4. Select all suppliers with "Office" in their name

SELECT * FROM suppliers WHERE supplier_name LIKE '%Office%';


# 5. Select items ordered alphabetically by name

SELECT * FROM items ORDER BY name ASC;


# Checking ALL items & Suppliers in seperate TABLE

SELECT * FROM items;

SELECT * FROM suppliers;


# Checking ALL items & Suppliers in combined table

SELECT

          i.id,

   i.barcode,

   i.name,

   i.quantity,

   i.price,

   i.brand,

   s.supplier_name,

   s.contact

FROM items i JOIN suppliers s ON i.supplier_id = s.supplier_id;
```