

Q1 create table students (primary key) and exam (foreign key)

```
CREATE TABLE students (id integer PRIMARY KEY AUTO_INCREMENT, name text, branch text);
INSERT INTO students (Name, Branch) VALUES('Jay', 'Computer Science'),
('Suhani', 'Electronic And Com'),
('Kriti', 'Electronic And Com');
```

The screenshot shows the phpMyAdmin interface for a database named 'assignment'. The left sidebar lists databases and tables, including 'assignment' which contains 'students', 'exam', 'customer', 'employee', 'incentive', 'item_mast', 'orders', 'salespeople', 'salesperson', and 'New'. The main panel displays the 'students' table with the following data:

		id	name	branch
<input type="checkbox"/>	Edit	1	Jay	Computer Science
<input type="checkbox"/>	Edit	2	Suhani	Electronic And Com
<input type="checkbox"/>	Edit	3	Kriti	Electronic And Com

Below the table, there are buttons for 'Check all', 'With selected:', 'Edit', 'Copy', 'Delete', and 'Export'. At the bottom, there are links for 'Print', 'Copy to clipboard', 'Export', 'Display chart', and 'Create view'. A 'Bookmark this SQL query' button is also present.

```
CREATE TABLE exam (rollno integer, FOREIGN KEY(rollno)REFERENCES students(id),s_code text,marks integer, p_code text);
```

```
INSERT INTO exam(Rollno, S_code, Marks, P_code) VALUES (1, 'CS11', 50, 'CS'),  
(1, 'CS12', 60, 'CS'),  
(2, 'EC101', 66, 'EC'),  
(2, 'EC102', 70, 'EC'),  
(3, 'EC101', 45, 'EC'),  
(3, 'EC102', 50, 'EC');
```

The screenshot shows the phpMyAdmin interface for the 'exam' table in the 'assignment' database. The table has four columns: rollno (int(11)), s_code (text), marks (int(11)), and p_code (text). A primary key constraint is defined on the rollno column. An index named 'rollno' is present on the rollno column.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	rollno	int(11)			Yes	NULL		
2	s_code	text	utf8mb4_general_ci		Yes	NULL		
3	marks	int(11)			Yes	NULL		
4	p_code	text	utf8mb4_general_ci		Yes	NULL		

Indexes:

Action	Keyname	Type	Unique	Packed	Column	Cardinality
Edit	rollno	BTREE	No	No	rollno	6

Partitions:

No partitioning defined!

Q2. Create table given below Employee and incentive ::

```
CREATE TABLE employee (employee_id integer PRIMARY KEY AUTO_INCREMENT,fisrst_name
text,last_name text,salary double,joining_date date,department text);
```

```
INSERT INTO employee (employee_id, fisrst_name, last_name, salary, joining_date, department)
VALUES (1, 'John', 'Abraham', 1000000, '2013-01-01','Banking'),
(2, 'Michael', 'Clarke', 800000, '2013-01-01','Insurance'),
(3, 'Roy', 'Thomas', 700000, '2013-02-01','Banking'),
(4, 'Tom', 'Jose', 600000, '2013-02-01','Insurance'),
(5, 'Jerry', 'Pinto', 650000, '2013-02-01','Insurance'),
(6, 'Philip', 'Mathew', 750000, '2013-01-01','Services'),
(7, 'TestName1', '123', 650000, '2013-01-01','Services')
(8, 'TestName2', 'Lname%', 600000, '2013-02-01','Insurance');
```

The screenshot shows the phpMyAdmin interface for a MySQL database named 'assignment'. The left sidebar lists various databases and tables, including 'assessment1', 'assessment2', 'assignment', 'customer', 'employee', 'exam', 'incentive', 'item_mast', 'orders', 'salespeople', 'salesperson', 'students', 'database1', 'information_schema', 'mysql', 'performance_schema', 'phpmyadmin', 'product master', 'question-2', and 'test'. The 'employee' table is selected in the main workspace. The table structure is displayed with columns: employee_id, fisrst_name, last_name, salary, joining_date, and department. Below the table, there is a list of 8 rows of data, each with edit, copy, and delete options. At the bottom of the table area, there are buttons for 'Check all', 'With selected:', and 'Export'. The status bar at the bottom right shows the date and time as 4/17/2024 6:03 PM.

	employee_id	fisrst_name	last_name	salary	joining_date	department
1	1	John	Abraham	1000000	2013-01-01	Banking
2	2	Michael	Clarke	800000	2013-01-01	Insurance
3	3	Roy	Thomas	700000	2013-02-01	Banking
4	4	Tom	Jose	600000	2013-02-01	Insurance
5	5	Jerry	Pinto	650000	2013-02-01	Insurance
6	6	Philip	Mathew	750000	2013-01-01	Services
7	7	TestName1	123	650000	2013-01-01	Services
8	8	TestName2	Lname%	600000	2013-02-01	Insurance

```
CREATE TABLE incentive (id integer primary key,incentive_date date,incentive_amount integer,  
employee_ref_id integer,FOREIGN KEY(employee_ref_id) REFERENCES  
employee(employee_id));
```

```
INSERT INTO incentive ( incentive_date, incentive_amount, employee_ref_id) VALUES  
( '2013-02-01', 5000,1),  
( '2013-02-01', 3000,2),  
( '2013-02-01', 4000,3),  
( '2013-01-01', 4500,1),  
( '2013-01-01', 3500,2)
```

The screenshot shows the phpMyAdmin interface for a MySQL database named 'assignment'. The left sidebar lists databases and tables, including 'assignment' which contains 'incentive', 'customer', 'employee', 'exam', 'item_mast', 'orders', 'salesperson', 'students', and 'viewable'. The main area displays the 'incentive' table with the following data:

incentive_date	incentive_amount	employee_ref_id
2013-02-01	5000	1
2013-02-01	3000	2
2013-02-01	4000	3
2013-01-01	4500	1
2013-01-01	3500	2

Below the table, there are 'Query results operations' buttons for Print, Copy to clipboard, Export, Display chart, and Create view. A 'Bookmark this SQL query' button is also present.

3 . SELECT first_name FROM employee WHERE employee_id=1;

The screenshot shows the phpMyAdmin interface running on a Windows desktop. The left sidebar lists databases and tables, with 'assignment' selected. The main area displays the results of the SQL query: 'SELECT first_name FROM employee WHERE employee_id=1;'. The results show one row: 'John'. Below the results, there are sections for 'Query results operations' (Print, Copy to clipboard, Export, Display chart, Create view) and 'Bookmark this SQL query' (Label input field, Let every user access this bookmark checkbox). The bottom taskbar shows various pinned icons and the system clock.

4 Get first_name,joining_date, and salary from employee table::

```
SELECT first_name,joining_date,salary FROM employee WHERE employye_id=1;
```

The screenshot shows the phpMyAdmin interface for a database named 'assignment'. The 'employee' table is selected. A query has been run: `SELECT first_name,joining_date,salary FROM employee WHERE employye_id=1;`. The result set contains one row: John, 2013-01-01, 1000000.

first_name	joining_date	salary
John	2013-01-01	1000000

5 Get all employee detail from the employee table order by first_name ascending and salary descending::

```
SELECT * FROM employee ORDER BY first_name ASC;
```

The screenshot shows the phpMyAdmin interface for a database named 'assignment'. The 'employee' table is selected. A query has been run: `SELECT * FROM employee ORDER BY first_name ASC;`. The result set contains 8 rows of employee details, ordered by first name.

employee_id	first_name	last_name	salary	joining_date	department
5	Jerry	Pinto	650000	2013-02-01	Insurance
1	John	Abraham	1000000	2013-01-01	Banking
2	Michael	Clarke	800000	2013-01-01	Insurance
6	Philip	Mathew	750000	2013-01-01	Services
3	Roy	Thomas	700000	2013-02-01	Banking
7	TestName1	123	650000	2013-01-01	Services
8	TestName2	Lname%	600000	2013-02-01	Insurance
4	Tom	Jose	600000	2013-02-01	Insurance

`SELECT * FROM employee ORDER BY salary DESC;`

The screenshot shows the phpMyAdmin interface for a database named 'assignment'. The 'employee' table is selected. A query has been run: `SELECT * FROM employee ORDER BY salary DESC;`. The results show 8 rows of data:

employee_id	firsrt_name	last_name	salary	joining_date	department
1	John	Abraham	1000000	2013-01-01	Banking
2	Michael	Clarke	800000	2013-01-01	Insurance
6	Philip	Mathew	750000	2013-01-01	Services
3	Roy	Thomas	700000	2013-02-01	Banking
5	Jerry	Pinto	650000	2013-02-01	Insurance
7	TestName1	123	650000	2013-01-01	Services
4	Tom	Jose	600000	2013-02-01	Insurance
8	TestName2	Lname%	600000	2013-02-01	Insurance

6 Get employee details from employee table whose first name contains 'j' ::

`SELECT * FROM employee WHERE firsrt_name LIKE '%j%';`

The screenshot shows the phpMyAdmin interface for a database named 'assignment'. The 'employee' table is selected. A query has been run: `SELECT * FROM employee WHERE firsrt_name LIKE '%j%';`. The results show 2 rows of data:

employee_id	firsrt_name	last_name	salary	joining_date	department
1	John	Abraham	1000000	2013-01-01	Banking
5	Jerry	Pinto	650000	2013-02-01	Insurance

7 Get department wise maximum salary from employee table order by salary ascending::

```
SELECT * ,MAX(salary) AS Maxsalary FROM employee GROUP BY department ORDER BY salary;
```

The screenshot shows the phpMyAdmin interface for a database named 'assignment'. The left sidebar lists various databases and tables. The main area displays a query result for the 'employee' table. The query was: `SELECT * ,MAX(salary) AS Maxsalary FROM employee GROUP BY department ORDER BY salary;`. The results show three rows of data:

employee_id	first_name	last_name	salary	joining_date	department	Maxsalary
6	Philip	Mathew	750000	2013-01-01	Services	750000
2	Michael	Clarke	800000	2013-01-01	Insurance	800000
1	John	Abraham	1000000	2013-01-01	Banking	1000000

8 select first_name, incentive amount from employee and incentives table for those employees who have incentives and incentive amount greater than 3000::

```
SELECT employee.first_name, incentive.incentive_amount FROM employee JOIN incentive ON employee_id= employee_ref_id WHERE incentive_amount > 3000;
```

The screenshot shows the phpMyAdmin interface with the following details:

- Left Panel (Database Structure):** Shows the database structure with various tables like assessment1, assessment2, assignment, customer, employee, exam, incentive, item_mast, orders, salespeople, salesperson, students, and viewable.
- Top Bar:** Shows the URL as `localhost/phpmyadmin/index.php?route=/table/sql&db=assignment&table=incentive`.
- Table Structure:** The "incentive" table has columns `firsrt_name` and `incentive_amount`.
- Query Result:** A query was run: `SELECT employee.firsrt_name, incentive.incentive_amount FROM employee JOIN incentive ON employee.employee_id = incentive.employee_ref_id WHERE incentive.incentive_amount > 3000;`. The results show four rows:

firsrt_name	incentive_amount
John	5000
Roy	4000
John	4500
Michael	3500

 A note at the top says: "Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.".
- Bottom Bar:** Shows the Windows taskbar with various pinned icons and the date/time as 6/29/2024 12:13 PM.

9 create after insert trigger on employee table which insert records in viewtable::

```
CREATE TABLE viewtable(employee_id int ,firsrt_name varchar(20),last_name varchar(20),salary int ,joining_date date,department varchar(20));
```

```
DELIMITER //
```

```
CREATE TRIGGER emplog
```

```
AFTER INSERT
```

```
ON employee
```

```
FOR EACH ROW
```

```
BEGIN
```

```
INSERT INTO viewtable(employee_id,firsrt_name,last_name,salary,joining_date,department)VALUES  
(NEW.employee_id,NEW.firsrt_name,NEW.last_name,NEW.salary,NEW.joining_date,NEW.department);
```

```
END //
```

```
DELIMITER//
```

The screenshot shows the phpMyAdmin interface for the 'assignment' database. The left sidebar lists various databases and tables. The main panel is titled 'Triggers' for the 'employee' table. It displays a single trigger named 'employ' with the event 'AFTER INSERT'. A 'Create new trigger' button is visible in the top right. The bottom status bar shows system information: 31°C Haze, 9:56 PM, 7/4/2024.

```
INSERT INTO viewable(employee_id,fisrst_name,last_name,salary,joining_date,department)VALUES  
(9,'aarohi','sharma',34000,2020-07-02,'IT');
```

The screenshot shows the phpMyAdmin interface for the 'assignment' database. The left sidebar lists various databases and tables. The main panel is titled 'viewable' and shows a single row of data: employee_id 9, first_name aarohi, last_name sharma, salary 34000, joining_date 2020-07-02, and department IT. The status bar at the bottom shows system information: 31°C Haze, 9:56 PM, 7/4/2024.

Q.11 Create table Salesperson and Customer ::

```
CREATE TABLE salesperson (sno int PRIMARY KEY,sname text,city text, comm REAL);

INSERT INTO salesperson VALUES (1001, 'Peel', 'London', .12),
(1002, 'Serres', 'San Jose', .13),
(1004, 'Motika', 'London', .11),
(1007, 'Rafkin', 'Barcelona', .15),
(1003, 'Axelrod', 'New York', .1);
```

The screenshot shows the phpMyAdmin interface for the 'salesperson' table. The table structure is as follows:

	sno	sname	city	comm
<input type="checkbox"/>	1001	Peel	London	0.12
<input type="checkbox"/>	1002	Serres	San Jose	0.13
<input type="checkbox"/>	1003	Axelrod	New York	0.1
<input type="checkbox"/>	1004	Motika	London	0.11
<input type="checkbox"/>	1007	Rafkin	Barcelona	0.15

Below the table, there are buttons for 'Edit', 'Copy', 'Delete', 'Check all', and 'With selected:'. At the bottom, there are links for 'Print', 'Copy to clipboard', 'Export', 'Display chart', 'Create view', and 'Bookmark this SQL query'.

```
CREATE TABLE customer( cmn INT PRIMARY KEY, cname VARCHAR(20), city VARCHAR(20), rating INT,  
sno INT, FOREIGN KEY(sno) REFERENCES salesperson(sno));
```

```
INSERT INTO customer(cnm, cname, city, rating, sno) VALUES (201, 'Hoffman', 'London', 100, 1001),  
(202, 'Giovanne', 'Roe', 200, 1003),  
(203, 'Liu', 'San Jose', 300, 1002),  
(204, 'Grass', 'Barcelona', 100, 1002),  
(206, 'Clemens', 'London', 300, 1007),  
(207, 'Pereira', 'Roe', 100, 1004);
```

The screenshot shows the phpMyAdmin interface for a MySQL database named 'assignment'. The left sidebar lists various databases and tables. The 'customer' table under the 'assignment' database is selected. The main panel displays the 'Table structure' tab, showing the columns: cmn (INT, Primary Key, Null No, Default None), cname (VARCHAR(20), Collation utf8mb4_general_ci, Null Yes, Default NULL), city (VARCHAR(20), Collation utf8mb4_general_ci, Null Yes, Default NULL), rating (INT(11), Null Yes, Default NULL), and sno (INT, Null Yes, Default NULL). Below the table structure, there are sections for 'Indexes' and 'Partitions'. The 'Indexes' section shows two indexes: 'PRIMARY' (BTREE, Unique Yes, Column cmn, Cardinality 6, Collation A, Null No) and 'sno' (BTREE, Unique No, Column sno, Cardinality 6, Collation A, Null Yes). The 'Partitions' section is currently empty.

The screenshot shows the phpMyAdmin interface for the 'customer' table in the 'assignment' database. The table has columns: cnum, cname, city, rating, sno. The data is as follows:

cnum	cname	city	rating	sno
201	Hoffman	London	100	1001
202	Giovanna	Roe	200	1003
203	Liu	San Jose	300	1002
204	Grass	Barcelona	100	1002
206	Clemens	London	300	1007
207	Pereira	Roe	100	1004

Q 14 names and cities of all salesperson in london with commission above 0.12::

SELECT sname,city FROM salesperson WHERE city='London' AND comm>0.12;

The screenshot shows the phpMyAdmin interface for the 'salesperson' table in the 'assignment' database. The table has columns: sname, city. The data is as follows:

sname	city
-------	------

A SQL query is shown in the query box:

```
SELECT sname,city FROM salesperson WHERE city='London' AND comm>0.12;
```

Q.15 All salespeople either in Barcelona or in London::

```
SELECT * FROM salesperson WHERE city='Barcelona' OR city='London';
```

The screenshot shows the phpMyAdmin interface for a MySQL database named 'assignment'. The left sidebar lists various databases and tables, including 'salesperson'. The main area displays the results of the SQL query: 'SELECT * FROM salesperson WHERE city='Barcelona' OR city='London';'. The results table shows three rows of data:

sno	sname	city	comm
1001	Peel	London	0.12
1004	Motika	London	0.11
1007	Rafkin	Barcelona	0.15

Below the table are standard MySQL operation buttons: Edit, Copy, Delete, and Export.

Q.16 All salespeople with commission between 0.10 and 0.12. (Boundary values should be excluded)::

SELECT * FROM salesperson WHERE comm BETWEEN 0.10 AND 0.12;

The screenshot shows the phpMyAdmin interface for a database named 'assignment'. The left sidebar lists various tables and databases. The main area displays the results of a query: 'Showing rows 0 - 2 (3 total, Query took 0.0003 seconds.)'. The query is 'SELECT * FROM salesperson WHERE comm BETWEEN 0.10 AND 0.12;'. The results table has columns sno, sname, city, and comm. The data is:

sno	sname	city	comm
1001	Peel	London	0.12
1003	Axelrod	New York	0.1
1004	Motika	London	0.11

Q.17 All customers excluding those with rating <= 100 unless they are located in Rome:::

SELECT * FROM customer WHERE rating <=100 AND city != 'Rome';

The screenshot shows the phpMyAdmin interface for a database named 'assignment'. The left sidebar lists various tables and databases. The main area displays the results of a query: 'Showing rows 0 - 2 (3 total, Query took 0.0003 seconds.)'. The query is 'SELECT * FROM customer WHERE rating <=100 AND city != 'Rome';'. The results table has columns cnum, cname, city, rating, and sno. The data is:

cnum	cname	city	rating	sno
201	Hoffman	London	100	1001
204	Grass	Barcelona	100	1002
207	Pereira	Roe	100	1004

Q 18 Write a sql statement that displays all the information about all salespeople ::

```
CREATE TABLE salespeople(salesman_id int PRIMARY KEY, name text, city varchar(20), commission REAL);
```

```
INSERT INTO salespeople VALUES (5001, 'James Hoog', 'New York', 0.15),
(5002, 'Nail Knite', 'Paris', 0.13),
(5005, 'Pit Alex', 'London', 0.11),
(5006, 'Mc Lyon', 'Paris', 0.14),
(5007, 'Paul Adam', 'Rome', 0.13),
(5003, 'Lauson Hen', 'San Jose', 0.12)
```

```
SELECT * FROM salespeople;
```

The screenshot shows the phpMyAdmin interface on a Windows desktop. The left sidebar lists databases and tables, with 'assignment' selected. The main area shows the 'salespeople' table with 6 rows of data. The table has columns: salesman_id, name, city, and commission. The data is as follows:

	salesman_id	name	city	commission
1	5001	James Hoog	New York	0.15
2	5002	Nail Knite	Paris	0.13
3	5003	Lauson Hen	San Jose	0.12
4	5005	Pit Alex	London	0.11
5	5006	Mc Lyon	Paris	0.14
6	5007	Paul Adam	Rome	0.13

Below the table, there are buttons for Edit, Copy, Delete, and other operations. The status bar at the bottom shows the date and time as 4/17/2024 7:07 PM.

Q.19 From the following table, write a SQL query to find orders that are delivered by a salesperson with ID. 5001. Return ord_no, ord_date, purch_amt ::

```
CREATE TABLE orders (ord_no int PRIMARY KEY,purch_amt REAL, ord_date date, customer_id int,
salesman_id int,
FOREIGN KEY(salesman_id)REFERENCES salespeople(salesman_id));
INSERT INTO orders VALUES (70001, 150.5, '2012-10-05', 3005, 5002),
(70009, 270.65, '2012-09-10', 3001, 5005),
(70002, 65.26, '2012-10-05', 3002, 5001),
(70004, 110.5, '2012-08-17', 3009, 5003),
(70007, 948.5, '2012-09-10', 3005, 5002),
(70005, 2400.6, '2012-07-27', 3007, 5001),
(70008, 5760, '2012-09-10', 3002, 5001),
(70010, 1983.43, '2012-10-10', 3004, 5006),
(70003, 2480.4, '2012-10-10', 3009, 5003),
(70012, 250.45, '2012-06-27', 3008, 5002),
(70011, 75.29, '2012-08-17', 3003, 5007),
(70013, 3045.6, '2012-04-25', 3002, 5001)
```

localhost / 127.0.0.1 / assignment

localhost/phpmyadmin/index.php?route=/sql&pos=0&db=assignment&table=orders

phpMyAdmin

Server: 127.0.0.1 > Database: assignment > Table: orders

Browse Structure SQL Search Insert Export Privileges Operations Tracking Triggers

Showing rows 0 - 11 (12 total. Query took 0.0006 seconds.)

SELECT * FROM `orders`

Profile Edit inline Explain SQL Create PHP code Refresh

Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

	ord_no	purch_amt	ord_date	customer_id	salesman_id
<input type="checkbox"/>	70001	150.5	2012-10-05	3005	5002
<input type="checkbox"/>	70002	65.26	2012-10-05	3002	5001
<input type="checkbox"/>	70003	2480.4	2012-10-10	3009	5003
<input type="checkbox"/>	70004	110.5	2012-08-17	3009	5003
<input type="checkbox"/>	70005	2400.6	2012-07-27	3007	5001
<input type="checkbox"/>	70007	948.5	2012-09-10	3005	5002
<input type="checkbox"/>	70008	5760	2012-09-10	3002	5001
<input type="checkbox"/>	70009	270.65	2012-09-10	3001	5005
<input type="checkbox"/>	70010	1983.43	2012-10-10	3004	5006
<input type="checkbox"/>	70011	75.29	2012-08-17	3003	5007
<input type="checkbox"/>	70012	250.45	2012-06-27	3008	5002
<input type="checkbox"/>	70013	3045.6	2012-04-25	3002	5001

Check all With selected: Edit Copy Delete Export

Console

40°C Sunny 7:10 PM 4/17/2024

localhost / 127.0.0.1 / assignment

localhost/phpmyadmin/index.php?route=/table/structure&db=assignment&table=orders

phpMyAdmin

Server: 127.0.0.1 > Database: assignment > Table: orders

Browse Structure SQL Search Insert Export Privileges Operations Tracking Triggers

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	ord_no	int(11)			No	None			<input type="button"/> Change <input type="button"/> Drop <input type="button"/> More
2	purch_amt	double			Yes	NULL			<input type="button"/> Change <input type="button"/> Drop <input type="button"/> More
3	ord_date	date			Yes	NULL			<input type="button"/> Change <input type="button"/> Drop <input type="button"/> More
4	customer_id	int(11)			Yes	NULL			<input type="button"/> Change <input type="button"/> Drop <input type="button"/> More
5	salesman_id	int(11)			Yes	NULL			<input type="button"/> Change <input type="button"/> Drop <input type="button"/> More

Check all With selected: Browse Change Drop Primary Unique Index Spatial Fulltext Add to central columns Remove from central columns

Print Propose table structure Track table Move columns Normalize

Add 1 column(s) after salesman_id Go

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
<input type="button"/> Edit <input type="button"/> Rename <input type="button"/> Drop	PRIMARY	BTREE	Yes	No	ord_no	12	A	No	
<input type="button"/> Edit <input type="button"/> Rename <input type="button"/> Drop	salesman_id	BTREE	No	No	salesman_id	12	A	Yes	

Create an index on 1 columns Go

Partitions

Console

40°C Sunny 7:11 PM 4/17/2024

The screenshot shows the phpMyAdmin interface with the 'assignment' database selected. The 'orders' table is open, displaying four rows of data. The columns are labeled 'ord_no', 'ord_date', and 'purch_amt'. The data is as follows:

	ord_no	ord_date	purch_amt
<input type="checkbox"/>	70002	2012-10-05	65.26
<input type="checkbox"/>	70005	2012-07-27	2400.6
<input type="checkbox"/>	70008	2012-09-10	5760
<input type="checkbox"/>	70013	2012-04-25	3045.6

Q.20 From the following table, write a SQL query to select a range of products whose price is in the range Rs.200 to Rs.600. Begin and end values are included. Return pro_id, pro_name, pro_price, and pro_com ::

```

CREATE TABLE item_mast(pro_id int PRIMARY KEY,pro_name varchar(20),pro_price REAL,pro_com int);

INSERT INTO item_mast VALUES
(101,'Mother Board', 3200.00, 15),
(102,'Key Board', 450.00, 16),
(103,'ZIP drive', 250.00, 14),
(104,'Speaker', 550.00, 16),
(105,'Monitor', 5000.00, 11),
(106,'DVD drive', 900.00, 12),
(107,'CD drive', 800.00, 12),
(108,'Printer', 2600.00, 13),
(109,'Refill cartridge', 350.00, 13),
(110,'Mouse', 250.00, 12)

```

The screenshot shows the phpMyAdmin interface for the 'assignment' database. The 'item_mast' table is selected, displaying 10 rows of data. The columns are pro_id, pro_name, pro_price, and pro_com. The data includes items like Mother Board, Key Board, ZIP drive, Speaker, Monitor, DVD drive, CD drive, Printer, Refill cartridge, and Mouse.

pro_id	pro_name	pro_price	pro_com
101	Mother Board	3200	15
102	Key Board	450	16
103	ZIP drive	250	14
104	Speaker	550	16
105	Monitor	5000	11
106	DVD drive	900	12
107	CD drive	800	12
108	Printer	2600	13
109	Refill cartridge	350	13
110	Mouse	250	12

```
SELECT pro_id,pro_name,pro_price,pro_com FROM item_mast WHERE pro_price BETWEEN 200.00 AND 600.00;
```

The screenshot shows the phpMyAdmin interface after executing the query. The results show 5 rows of data from the 'item_mast' table where the price is between 200.00 and 600.00. The columns are pro_id, pro_name, pro_price, and pro_com. The data includes items like Key Board, ZIP drive, Speaker, Refill cartridge, and Mouse.

pro_id	pro_name	pro_price	pro_com
102	Key Board	450	16
103	ZIP drive	250	14
104	Speaker	550	16
109	Refill cartridge	350	13
110	Mouse	250	12

Q.21 From the following table, write a SQL query to calculate the average price for a manufacturer code of 16. Return avg ::

```
SELECT AVG(Pro_Price) FROM item_mast WHERE Pro_Code=16
```

The screenshot shows the phpMyAdmin interface on a Windows desktop. The left sidebar lists databases and tables, with 'item_mast' selected. The main area displays the results of the executed SQL query:

```
SELECT AVG(Pro_Price) FROM item_mast WHERE pro_com=16;
```

The results show a single row with the value 500.

Below the results, there are buttons for 'Print', 'Copy to clipboard', 'Export', 'Display chart', and 'Create view'. There is also a 'Bookmark this SQL query' button and a 'Label:' input field.

The system tray at the bottom shows the date and time as 4/17/2024 7:20 PM, the weather as 40°C Sunny, and other icons.

Q.22 From the following table, write a SQL query to display the pro_name as 'Item Name' and pro_priceas 'Price in Rs.'::

```
SELECT pro_name AS item_name, pro_price AS price_in_rs FROM item_mast;
```

The screenshot shows the phpMyAdmin interface on a Windows desktop. The left sidebar lists databases and tables, with 'item_mast' selected. The main area displays the results of the SQL query:

```
SELECT pro_name AS item_name, pro_price AS price_in_rs FROM item_mast;
```

The results table shows the following data:

	item_name	price_in_rs
<input type="checkbox"/> <input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	Mother Board	3200
<input type="checkbox"/> <input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	Key Board	450
<input type="checkbox"/> <input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	ZIP drive	250
<input type="checkbox"/> <input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	Speaker	550
<input type="checkbox"/> <input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	Monitor	5000
<input type="checkbox"/> <input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	DVD drive	900
<input type="checkbox"/> <input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	CD drive	800
<input type="checkbox"/> <input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	Printer	2600
<input type="checkbox"/> <input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	Refill cartridge	350
<input type="checkbox"/> <input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	Mouse	250

Q.23 From the following table, write a SQL query to find the items whose prices are higher than or equal to \$250. Order the result by product price in descending, then product name in ascending. Return pro_name and pro_price:::

```
SELECT pro_name,pro_price FROM item_mast WHERE pro_price>=250 ORDER BY pro_price DESC;
```

The screenshot shows the phpMyAdmin interface on a Windows desktop. The left sidebar lists databases and tables, with 'item_mast' selected. The main area shows the SQL query: `SELECT pro_name,pro_price FROM item_mast WHERE pro_price>=250 ORDER BY pro_price DESC;`. Below the query, the results are displayed in a table:

pro_name	pro_price
Monitor	5000
Mother Board	3200
Printer	2600
DVD drive	900
CD drive	800
Speaker	550
Key Board	450
Refill cartridge	350
ZIP drive	250
Mouse	250

The status bar at the bottom indicates it's 7:25 PM on 4/17/2024, the temperature is 40°C, and the weather is sunny.

Q.24 From the following table, write a SQL query to calculate average price of the items for each company. Return average price and company code.:::

```
SELECT AVG (pro_price),pro_com FROM item_mast GROUP BY pro_com;
```

The screenshot shows the phpMyAdmin interface for a database named 'assignment'. The left sidebar lists various tables and databases. The main area is titled 'Table: item_mast' and displays the results of the executed SQL query:

```
SELECT AVG (pro_price),pro_com FROM item_mast GROUP BY pro_com;
```

The results table shows the following data:

	Avg (pro_price)	pro_com
<input type="checkbox"/>	5000	11
<input type="checkbox"/>	650	12
<input type="checkbox"/>	1475	13
<input type="checkbox"/>	250	14
<input type="checkbox"/>	3200	15
<input type="checkbox"/>	500	16

At the bottom of the results table, there are buttons for 'Check all', 'With selected:', and 'Edit', 'Copy', 'Delete', 'Export'.

The status bar at the bottom right indicates: 40°C Sunny, 7:27 PM, 4/17/2024.

