OUTPUT

```
mysql> CREATE DATABASE bankDatabase;
mysql> USE bankDatabase;
Database changed
mysql> CREATE TABLE bank(name VARCHAR(25) NOT NULL, code VARCHAR(10) PRIMARY KEY,
    -> address VARCHAR(50) NOT NULL);
mysql> DESC bank;
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| name    | varchar(25) | NO   |     | NULL    |       |
| code    | varchar(10) | NO   | PRI | NULL    |       |
| address | varchar(50) | NO   |     | NULL    |       |
+---------+-------------+------+-----+---------+-------+
mysql> CREATE TABLE branch(branch_no INT PRIMARY KEY,name VARCHAR(20) NOT NULL,
    -> address VARCHAR(30) NOT NULL,bank_code VARCHAR(10) NOT NULL,
    -> FOREIGN KEY(bank_code) REFERENCES bank(code));
mysql> desc branch;
+-----------+-------------+------+-----+---------+-------+
| Field     | Type        | Null | Key | Default | Extra |
+-----------+-------------+------+-----+---------+-------+
| branch_no | int         | NO   | PRI | NULL    |       |
| name      | varchar(20) | NO   |     | NULL    |       |
| address   | varchar(30) | NO   |     | NULL    |       |
| bank_code | varchar(10) | NO   | MUL | NULL    |       |
+-----------+-------------+------+-----+---------+-------+
mysql> CREATE TABLE loan(loan_id INT PRIMARY KEY, loan_type VARCHAR(10) ,
    -> amount INT NOT NULL, branch_no INT NOT NULL,
    -> FOREIGN KEY(branch_no) REFERENCES branch(branch_no));
mysql> desc loan;
+-----------+-------------+------+-----+---------+-------+
| Field     | Type        | Null | Key | Default | Extra |
+-----------+-------------+------+-----+---------+-------+
| loan_id   | int         | NO   | PRI | NULL    |       |
| loan_type | varchar(10) | YES  |     | NULL    |       |
| amount    | int         | NO   |     | NULL    |       |
| branch_no | int         | NO   | MUL | NULL    |       |
+-----------+-------------+------+-----+---------+-------+
mysql> CREATE TABLE loan_installment(installment_no INT NOT NULL,
    ->loan_id INT NOT NULL, amount INT NOT NULL,
    -> PRIMARY KEY(loan_id,installment_no),
    -> FOREIGN KEY(loan_id) REFERENCES loan(loan_id));
mysql> desc loan_installment;
+----------------+------+------+-----+---------+-------+
| Field          | Type | Null | Key | Default | Extra |
+----------------+------+------+-----+---------+-------+
| installment_no | int  | NO   | PRI | NULL    |       |
| loan_id        | int  | NO   | PRI | NULL    |       |
| amount         | int  | NO   |     | NULL    |       |
+----------------+------+------+-----+---------+-------+


mysql> CREATE TABLE account ( account_no INT PRIMARY KEY,account_type VARCHAR(10),
```

```
    -> balance DECIMAL(10,3) NOT NULL,branch_no INT NOT NULL,
    -> FOREIGN KEY(branch_no) REFERENCES branch(branch_no));
mysql> desc account;
+--------------+---------------+------+-----+---------+-------+
| Field        | Type          | Null | Key | Default | Extra |
+--------------+---------------+------+-----+---------+-------+
| account_no   | int           | NO   | PRI | NULL    |       |
| account_type | varchar(10)   | YES  |     | NULL    |       |
| balance      | decimal(10,3) | NO   |     | NULL    |       |
| branch_no    | int           | NO   | MUL | NULL    |       |
+--------------+---------------+------+-----+---------+-------+
mysql> CREATE TABLE customer(customer_id INT PRIMARY KEY,
    -> name VARCHAR(15) NOT NULL,address VARCHAR(30) NOT NULL);
mysql> DESC customer;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| customer_id | int         | NO   | PRI | NULL    |       |
| name        | varchar(15) | NO   |     | NULL    |       |
| address     | varchar(30) | NO   |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
mysql> CREATE TABLE customer_loan(customer_id INT NOT NULL,
    -> loan_id INT NOT NULL,FOREIGN KEY(loan_id) REFERENCES loan(loan_id),
    ->FOREIGN KEY(customer_id) REFERENCES customer(customer_id));
mysql> desc customer_loan;
+-------------+------+------+-----+---------+-------+
| Field       | Type | Null | Key | Default | Extra |
+-------------+------+------+-----+---------+-------+
| customer_id | int  | NO   | MUL | NULL    |       |
| loan_id     | int  | NO   | MUL | NULL    |       |
+-------------+------+------+-----+---------+-------+
mysql> CREATE TABLE customer_phone(customer_id INT NOT NULL,
    -> phone VARCHAR(10) NOT NULL,PRIMARY KEY(customer_id,phone),
    -> FOREIGN KEY(customer_id) REFERENCES customer(customer_id));
mysql> desc customer_phone;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| customer_id | int         | NO   | PRI | NULL    |       |
| phone       | varchar(10) | NO   | PRI | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
mysql> CREATE TABLE customer_account(
    -> customer_id INT NOT NULL,account_no INT NOT NULL,
    -> FOREIGN KEY(account_no) REFERENCES account(account_no),
    -> FOREIGN KEY(account_no) REFERENCES account(account_no));
mysql> desc customer_account;
+-------------+------+------+-----+---------+-------+
| Field       | Type | Null | Key | Default | Extra |
+-------------+------+------+-----+---------+-------+
| customer_id | int  | NO   |     | NULL    |       |
| account_no  | int  | NO   | MUL | NULL    |       |
+-------------+------+------+-----+---------+-------+
```

1)NOT NULL

```
mysql> CREATE TABLE students(
    -> id INT NOT NULL,
    -> name varchar(15),
    -> address varchar(30));
Query OK, 0 rows affected (0.01 sec)

mysql> INSERT INTO students VALUES(NULL,'Gill','gujarat');
ERROR 1048 (23000): Column 'id' cannot be null
```

2) UNIQUE

```
mysql> ALTER TABLE students
    -> ADD UNIQUE(name);
Query OK, 0 rows affected (0.04 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> INSERT INTO students VALUES(1,'Gill','gujarat');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO students VALUES(2,'Gill','mumbai');
ERROR 1062 (23000): Duplicate entry 'Gill' for key 'students.name'
```

3)PRIMARY KEY

```
mysql> ALTER TABLE students
    -> ADD PRIMARY KEY(id);
Query OK, 0 rows affected (0.07 sec)
```

4)FOREIGN KEY

```
mysql> CREATE TABLE course(
    -> student_id INT NOT NULL,
    -> course_name VARCHAR(15) NOT NULL,
    -> CONSTRAINT fk_course_students_id
    -> FOREIGN KEY(student_id) REFERENCES students(id));
Query OK, 0 rows affected (0.02 sec)

mysql> INSERT INTO course VALUES(1,'CSE');
Query OK, 1 row affected (0.00 sec)
mysql> INSERT INTO course VALUES(4,'ECE');
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint
fails (`constraints`.`course`, CONSTRAINT `fk_course_students_id` FOREIGN KEY
(`student_id`) REFERENCES `students` (`id`))
```

5)CHECK

```
mysql> ALTER TABLE students ADD COLUMN age INT CHECK(age>17);
Query OK, 1 row affected (0.05 sec)
Records: 1  Duplicates: 0  Warnings: 0
mysql> INSERT INTO students(id,name,address,age) VALUES(3,'siraj','hyderabad',15);
ERROR 3819 (HY000): Check constraint 'students_chk_1' is violated.
```

6) ENUM

```
mysql> ALTER TABLE students ADD COLUMN
    -> gender ENUM('M','F','T');
Query OK, 0 rows affected (0.02 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> INSERT INTO students(id,name,address,age,gender) VALUES(3, 'Jasprit
Bumrah','hyderabad',18,'Q');
ERROR 1265 (01000): Data truncated for column 'gender' at row 1
mysql> INSERT INTO students(id,name,address,age,gender) VALUES(3, 'Jasprit
Bumrah','hyderabad',18,'M');
Query OK, 1 row affected (0.01 sec)

mysql> select constraint_name , constraint_type
    -> from information_schema.table_constraints
    -> where table_name = 'students';
+-----------------+-----------------+
| CONSTRAINT_NAME | CONSTRAINT_TYPE |
+-----------------+-----------------+
| name            | UNIQUE          |
| PRIMARY         | PRIMARY KEY     |
| students_chk_1  | CHECK           |
+-----------------+-----------------+
3 rows in set (0.01 sec)

mysql> select constraint_name , constraint_type
    -> from information_schema.table_constraints
    -> where table_name = 'course';
+----------------------+-----------------+
| CONSTRAINT_NAME      | CONSTRAINT_TYPE |
+----------------------+-----------------+
| fk_course_students_id | FOREIGN KEY     |
+----------------------+-----------------+
1 row in set (0.00 sec)
```

1]CREATE COMMAND

```
mysql> CREATE TABLE cricket_players(
    -> player_id INT PRIMARY KEY,
    -> name VARCHAR(15) NOT NULL,
    -> country VARCHAR(10),
    -> age VARCHAR(2));
Query OK, 0 rows affected (0.01 sec)
```

2]ALTER COMMAND

```
mysql> ALTER TABLE cricket_players
    -> ADD COLUMN team_name VARCHAR(20),
    -> MODIFY COLUMN age INT,
    -> RENAME COLUMN name TO player_name,
    -> DROP COLUMN country;
Query OK, 0 rows affected (0.04 sec)
Records: 0  Duplicates: 0  Warnings: 0
mysql> desc cricket_players;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| player_id   | int         | NO   | PRI | NULL    |       |
| player_name | varchar(15) | NO   |     | NULL    |       |
| age         | int         | YES  |     | NULL    |       |
| team_name   | varchar(20) | YES  |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

3]RENAME COMMAND

```
mysql> RENAME TABLE cricket_players TO players_of_rcb_ipl;
Query OK, 0 rows affected (0.02 sec)
```

4]TRUNCATE COMMAND

```
mysql> TRUNCATE TABLE players_of_rcb_ipl;
Query OK, 0 rows affected (0.02 sec)

mysql> show tables;
+--------------------+
| Tables_in_cricket  |
+--------------------+
| players_of_rcb_ipl |
+--------------------+
1 row in set (0.00 sec)
```

5]DROP COMMAND

```
mysql> DROP TABLE players_of_rcb_ipl;
Query OK, 0 rows affected (0.01 sec)

mysql> show tables;
Empty set (0.01 sec)
```

```
mysql> CREATE TABLE cricket_players(
    -> player_id INT PRIMARY KEY,
    -> player_name VARCHAR(15) NOT NULL,
    -> age INT );

1)INSERT COMMAND

INSERT INTO cricket_players(player_id,player_name,age) VALUES
    -> (3,'Ravindra Jadeja',34),
    -> (6,'Jasprit Bumrah',29),
    -> (9,'Kuldeep Yadav',28),
    -> (18,'Virat Kohli',35),
    -> (21,'Mohammed Siraj',29);
Query OK, 5 rows affected (0.01 sec)
Records: 5  Duplicates: 0  Warnings: 0
mysql> SELECT * FROM cricket_players;
+-----------+-----------------+------+
| player_id | player_name     | age  |
+-----------+-----------------+------+
|         3 | Ravindra Jadeja |   34 |
|         6 | Jasprit Bumrah  |   29 |
|         9 | Kuldeep Yadav   |   28 |
|        18 | Virat Kohli     |   35 |
|        21 | Mohammed Siraj  |   29 |
+-----------+-----------------+------+

2)UPDATE COMMAND

mysql> UPDATE cricket_players
    -> SET age = 36
    -> WHERE player_id = 18;
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> SELECT * FROM cricket_players where player_id = 18;
+-----------+-------------+------+
| player_id | player_name | age  |
+-----------+-------------+------+
|        18 | Virat Kohli |   36 |
+-----------+-------------+------+
3) DELETE COMMAND

mysql> DELETE FROM cricket_players
    -> WHERE player_id = 3;
Query OK, 1 row affected (0.01 sec)
mysql> SELECT * FROM cricket_players ;
+-----------+----------------+------+
| player_id | player_name    | age  |
+-----------+----------------+------+
|         6 | Jasprit Bumrah |   29 |
|         9 | Kuldeep Yadav  |   28 |
|        18 | Virat Kohli    |   36 |
|        21 | Mohammed Siraj |   29 |
+-----------+----------------+------+
```

```
mysql> CREATE TABLE students(
    -> roll_no INT PRIMARY KEY,
    -> fname varchar(10) NOT NULL, lname varchar(10) NOT NULL,
    -> mark INT ,fee INT);
+---------+----------+--------+------+-------+
| roll_no | fname    | lname  | mark | fee   |
+---------+----------+--------+------+-------+
|       1 | Jasprit  | Bumrah |   99 | 35000 |
|       2 | Kuldeep  | Yadav  |   85 | 75000 |
|       3 | Mohammed | Siraj  |   90 |  8000 |
|       4 | Virat    | Kohli  |  100 |  7000 |
|       5 | Travis   | Head   |   80 | 75000 |
+---------+----------+--------+------+-------+

a)
mysql> SELECT COUNT(*) AS total_students
    -> FROM students;
+----------------+
| total_students |
+----------------+
|              5 |
+----------------+
1 row in set (0.01 sec)
b)
mysql> SELECT MIN(mark) as minimum_mark,MAX(mark) as maximum_mark,
    -> AVG(mark) as average_mark
    -> FROM students;
+--------------+--------------+--------------+
| minimum_mark | maximum_mark | average_mark |
+--------------+--------------+--------------+
|           80 |          100 |      90.8000 |
+--------------+--------------+--------------+
1 row in set (0.00 sec)
c)
mysql> SELECT fname,lname,mark FROM students
    -> WHERE mark = (SELECT MAX(mark) from students)
    -> OR mark = (SELECT MIN(mark) from students);
+--------+-------+------+
| fname  | lname | mark |
+--------+-------+------+
| Virat  | Kohli |  100 |
| Travis | Head  |   80 |
+--------+-------+------+
2 rows in set (0.00 sec)
d)
mysql> SELECT SUM(fee) as total_fee
    -> FROM students;
+-----------+
| total_fee |
+-----------+
|    200000 |
+-----------+
1 row in set (0.00 sec)
```

e)

```
mysql> SELECT UPPER(CONCAT(fname,' ',lname)) AS FIRST_RANK
    -> FROM students
    -> WHERE mark = (
    -> SELECT MAX(mark)
    -> from students);
+-------------+
| FIRST_RANK  |
+-------------+
| VIRAT KOHLI |
+-------------+
1 row in set (0.00 sec)
```

f)

```
mysql> SELECT DATE_FORMAT(CURDATE(),'%M-%d-%Y') AS DATE;
+------------------+
| DATE             |
+------------------+
| December-03-2023 |
+------------------+
1 row in set (0.00 sec)
```

g)

```
mysql> SELECT POWER(3,5) AS 5th_power_of_3;
+-----------------+
| 5th_power_of_3 |
+-----------------+
|             243 |
+-----------------+
1 row in set (0.01 sec)
```

OUTPUT

```
mysql> CREATE TABLE communicable_diseases(
    -> serial_no INT AUTO_INCREMENT, state VARCHAR(20) NOT NULL,
    -> year INT , month INT CHECK (month >=1 AND month <= 12),
    -> no_of_deaths INT, no_of_infections INT,
    -> PRIMARY KEY(serial_no));

mysql> SELECT * FROM communicable_diseases;
+-----------+---------+------+-------+--------------+------------------+
| serial_no | state   | year | month | no_of_deaths | no_of_infections |
+-----------+---------+------+-------+--------------+------------------+
|         1 | Goa     | 2020 |     6 |            9 |              150 |
|         2 | Goa     | 2021 |    12 |            5 |               20 |
|         3 | Gujarat | 2020 |     3 |           20 |              500 |
|         4 | Gujarat | 2020 |     4 |           15 |              700 |
|         5 | Kerala  | 2020 |     3 |           10 |              200 |
|         6 | Kerala  | 2020 |     5 |           20 |              300 |
|         7 | Kerala  | 2021 |     1 |           18 |              150 |
+-----------+---------+------+-------+--------------+------------------+
```

a)

```
mysql> SELECT state , AVG(no_of_deaths) AS average_deaths
    -> FROM communicable_diseases
    -> WHERE year = 2020
    -> GROUP BY state;
+---------+----------------+
| state   | average_deaths |
+---------+----------------+
| Goa     |         9.0000 |
| Gujarat |        17.5000 |
| Kerala  |        15.0000 |
+---------+----------------+
3 rows in set (0.00 sec)
```

b)

```
mysql> SELECT state, SUM(no_of_deaths) AS total_deaths
    -> FROM communicable_diseases
    -> GROUP BY state
    -> HAVING total_deaths > 10;
+---------+--------------+
| state   | total_deaths |
+---------+--------------+
| Goa     |           14 |
| Gujarat |           35 |
| Kerala  |           48 |
+---------+--------------+
3 rows in set (0.01 sec)
```

C)

```
mysql> SELECT t1.state, t1.year, max_deaths, t1.month
    -> FROM communicable_diseases t1 JOIN
    -> (SELECT state, MAX(no_of_deaths) AS max_deaths
    -> FROM communicable_diseases
    -> GROUP BY state
    -> HAVING max_deaths > 10 ) t2
    -> ON t1.state = t2.state AND t1.no_of_deaths = t2.max_deaths;
+---------+------+------------+-------+
| state   | year | max_deaths | month |
+---------+------+------------+-------+
| Gujarat | 2020 |         20 |     3 |
| Kerala  | 2020 |         20 |     5 |
+---------+------+------------+-------+
```

d)

```
mysql> SELECT * FROM communicable_diseases
    -> ORDER BY state DESC;
+-----------+---------+------+-------+--------------+------------------+
| serial_no | state   | year | month | no_of_deaths | no_of_infections |
+-----------+---------+------+-------+--------------+------------------+
|         5 | Kerala  | 2020 |     3 |           10 |              200 |
|         6 | Kerala  | 2020 |     5 |           20 |              300 |
|         7 | Kerala  | 2021 |     1 |           18 |              150 |
|         3 | Gujarat | 2020 |     3 |           20 |              500 |
|         4 | Gujarat | 2020 |     4 |           15 |              700 |
|         1 | Goa     | 2020 |     6 |            9 |              150 |
|         2 | Goa     | 2021 |    12 |            5 |               20 |
+-----------+---------+------+-------+--------------+------------------+
7 rows in set (0.00 sec)
```

OUTPUT

```
mysql> CREATE TABLE arts(
    -> serial_no INT AUTO_INCREMENT, name VARCHAR(15),
    -> student_id INT, event VARCHAR(10),
    -> grade ENUM('A','B','C'), PRIMARY KEY(serial_no));

mysql> CREATE TABLE sports(
    -> serial_no INT AUTO_INCREMENT, student_id INT,
    -> name VARCHAR(15), grade ENUM('A','B','C'),
    -> item VARCHAR(10), PRIMARY KEY(serial_no));
```

sports

serial_no	student_id	name	grade	item
1	33	Jobin	A	cricket
2	45	Jaya	C	cricket
3	59	Sujith	A	cricket

arts

serial_no	name	student_id	event	grade
1	Jelan	32	music	A
2	Jobin	33	dance	B
3	Joel	34	painting	C
4	Sujith	59	painting	A

a)

```
mysql> SELECT student_id, name FROM arts
    -> UNION
    -> SELECT student_id, name FROM sports;
```

student_id	name
32	Jelan
33	Jobin
34	Joel
59	Sujith
45	Jaya

```
5 rows in set (1.16 sec)
```

b)

```
mysql> SELECT student_id, name FROM sports
    -> INTERSECT
    -> SELECT student_id, name FROM arts;
```

student_id	name
33	Jobin
59	Sujith

c)

```
mysql> SELECT student_id, name FROM sports
    -> EXCEPT
    -> SELECT student_id, name FROM arts;
+------------+------+
| student_id | name |
+------------+------+
|         45 | Jaya |
+------------+------+
1 row in set (0.00 sec)
```

d)

```
mysql> CREATE TABLE project(
    -> student_name VARCHAR(15),
    -> project_title VARCHAR(20),
    -> expense INT);
project
+--------------+--------------------+---------+
| student_name | project_title      | expense |
+--------------+--------------------+---------+
| Sujith       | social media       |   50000 |
| Jelan        | e commerse website |   75000 |
| Joel         | ai powered chatbot |   25000 |
+--------------+--------------------+---------+

mysql> SELECT * FROM project
    -> WHERE expense = (
    -> SELECT MAX(expense)
    -> FROM project);
+--------------+--------------------+---------+
| student_name | project_title      | expense |
+--------------+--------------------+---------+
| Jelan        | e commerse website |   75000 |
+--------------+--------------------+---------+
1 row in set (0.60 sec)
```

OUTPUT

a)

```
mysql> CREATE TABLE shop (
    -> orderid INT PRIMARY KEY,
    -> item VARCHAR(20),
    -> price DECIMAL(10,2),
    -> quantity INT,
    -> discount DECIMAL(4,2));
+---------+--------+--------+----------+----------+
| orderid | item   | price  | quantity | discount |
+---------+--------+--------+----------+----------+
|       1 | Apple  |  50.00 |        5 |     1.50 |
|       2 | Banana |  40.00 |        3 |     5.00 |
|       3 | Cherry |  60.00 |        2 |     3.00 |
|       4 | Date   | 120.00 |        1 |     1.60 |
+---------+--------+--------+----------+----------+
```

b)

```
mysql> CREATE VIEW shop_items_and_price AS
    -> SELECT item,price
    -> FROM shop;
mysql> SELECT * FROM shop_items_and_price;
+--------+--------+
| item   | price  |
+--------+--------+
| Apple  |  50.00 |
| Banana |  40.00 |
| Cherry |  60.00 |
| Date   | 120.00 |
+--------+--------+
```

c)

```
mysql> CREATE VIEW shop_items_with_quantity AS
    -> SELECT item, quantity
    -> FROM shop
    -> WHERE quantity > 0;
mysql> SELECT * FROM shop_items_with_quantity ;
+--------+----------+
| item   | quantity |
+--------+----------+
| Apple  |        5 |
| Banana |        3 |
| Cherry |        2 |
| Date   |        1 |
+--------+----------+
```

d)

```
mysql> CREATE VIEW shop_items_with_discount_gt2 AS
    -> SELECT item, price, discount
    -> FROM shop
    -> WHERE discount > 2;
```

```
mysql> SELECT * FROM shop_items_with_discount_gt2;
+--------+-------+----------+
| item   | price | discount |
+--------+-------+----------+
| Banana | 40.00 |     5.00 |
| Cherry | 60.00 |     3.00 |
+--------+-------+----------+

e)
mysql> show tables;
+------------------------------+
| Tables_in_experiment         |
+------------------------------+
| shop                         |
| shop_items_and_price         |
| shop_items_with_discount_gt2 |
| shop_items_with_quantity     |
+------------------------------+

mysql> DROP VIEW shop_items_with_discount_gt2;
Query OK, 0 rows affected (0.01 sec)

mysql> DROP VIEW shop_items_and_price;
Query OK, 0 rows affected (0.00 sec)

mysql> DROP VIEW shop_items_with_quantity;
Query OK, 0 rows affected (0.01 sec)

mysql> show tables;
+---------------------+
| Tables_in_experiment|
+---------------------+
| shop                |
+---------------------+
1 row in set (0.00 sec)
```

OUTPUT

a)

```
mysql> CREATE TABLE customer ( customer_id INT PRIMARY KEY,
    -> name VARCHAR(15), phone VARCHAR(10), address VARCHAR(100));
+-------------+----------+------------+--------------------------------+
| customer_id | name     | phone      | address                        |
+-------------+----------+------------+--------------------------------+
|           1 | John Doe | 1234567890 | 123 Main St, Mumbai, India     |
|           2 | Jane Doe | 0987654321 | 456 Park St, Delhi, India      |
|           3 | Alice    | 1112223333 | 789 Market St, Chennai, India  |
|           4 | Bob      | 4445556666 | 321 Broadway, Bangalore, India |
|           5 | Charlie  | 7778889999 | 654 Broadway, Kolkata, India   |
+-------------+----------+------------+--------------------------------+
```

b)

```
mysql> CREATE TABLE accounts( customer_id INT NOT NULL,
    -> bank_code VARCHAR(15), account_no VARCHAR(15),
    -> account_type VARCHAR(20), balance DECIMAL(10,2),
    -> PRIMARY KEY(account_no),
    -> FOREIGN KEY(customer_id) REFERENCES customer(customer_id));
+-------------+-----------+------------+--------------+----------+
| customer_id | bank_code | account_no | account_type | balance  |
+-------------+-----------+------------+--------------+----------+
|           1 | SBI123    | ACC123456  | Savings      | 10000.00 |
|           3 | SBI789    | ACC345678  | Savings      | 30000.00 |
|           5 | SBI345    | ACC678901  | Savings      | 50000.00 |
|           2 | SBI456    | ACC789012  | Current      | 20000.00 |
|           4 | SBI012    | ACC901234  | Current      | 40000.00 |
+-------------+-----------+------------+--------------+----------+
```

c)

```
mysql> CREATE TABLE loan ( loan_id INT PRIMARY KEY, loan_type VARCHAR(20),
    -> loan_amount DECIMAL(10,2), customer_id INT NOT NULL,
    -> FOREIGN KEY(customer_id) REFERENCES customer(customer_id));
+---------+----------------+-------------+-------------+
| loan_id | loan_type      | loan_amount | customer_id |
+---------+----------------+-------------+-------------+
|     101 | Home Loan      |   500000.00 |           1 |
|     102 | Car Loan       |   200000.00 |           2 |
|     103 | Education Loan |   100000.00 |           3 |
|     104 | Personal Loan  |    30000.00 |           4 |
|     105 | Business Loan  |   400000.00 |           5 |
+---------+----------------+-------------+-------------+
```

d)

```
mysql> CREATE TABLE loan_installment ( loan_id INT,
    -> installment_no INT,
    -> installment_amount DECIMAL(10,2),
```

```
    -> PRIMARY KEY(loan_id,installment_no),
    -> FOREIGN KEY(loan_id) REFERENCES loan(loan_id));


mysql> SELECT * FROM loan_installment;
+---------+----------------+--------------------+
| loan_id | installment_no | installment_amount |
+---------+----------------+--------------------+
|     101 |              1 |           20000.00 |
|     101 |              2 |           20000.00 |
|     102 |              1 |           10000.00 |
|     103 |              1 |           15000.00 |
|     104 |              1 |           10000.00 |
|     105 |              1 |            5000.00 |
|     105 |              2 |            5000.00 |
+---------+----------------+--------------------+

e)
mysql> SELECT customer.customer_id,name,address,account_no
    -> FROM customer
    -> JOIN accounts
    -> ON customer.customer_id = accounts.customer_id;
+-------------+----------+--------------------------------+------------+
| customer_id | name     | address                        | account_no |
+-------------+----------+--------------------------------+------------+
|           1 | John Doe | 123 Main St, Mumbai, India     | ACC123456  |
|           2 | Jane Doe | 456 Park St, Delhi, India      | ACC789012  |
|           3 | Alice    | 789 Market St, Chennai, India  | ACC345678  |
|           4 | Bob      | 321 Broadway, Bangalore, India | ACC901234  |
|           5 | Charlie  | 654 Broadway, Kolkata, India   | ACC678901  |
+-------------+----------+--------------------------------+------------+

f)
mysql> SELECT loan.loan_id,loan_type,total_amount_paid
    -> FROM loan
    -> JOIN (
    -> SELECT loan_id,SUM(installment_amount) AS total_amount_paid
    -> FROM loan_installment
    -> GROUP BY loan_id) paid
    -> ON loan.loan_id = paid.loan_id;
+---------+----------------+--------------------+
| loan_id | loan_type      | total_amount_paid |
+---------+----------------+--------------------+
|     101 | Home Loan      |          40000.00 |
|     102 | Car Loan       |          10000.00 |
|     103 | Education Loan |          15000.00 |
|     104 | Personal Loan  |          10000.00 |
|     105 | Business Loan  |          10000.00 |
+---------+----------------+--------------------+
```

OUTPUT

a)

```
mysql> CREATE TABLE customer(
    -> customer_id INT PRIMARY KEY,
    -> name VARCHAR(15),
    -> city VARCHAR(20),
    -> pin INT,
    -> phone_number VARCHAR(10));
Query OK, 0 rows affected (0.02 sec)

mysql> SELECT * FROM customer;
+-------------+--------+----------------+--------+--------------+
| customer_id | name   | city           | pin    | phone_number |
+-------------+--------+----------------+--------+--------------+
|          32 | Jelan  | Ezhamkulam     | 691543 | 9988124321   |
|          53 | Sabari | kollam         | 691001 | 8934325612   |
|          59 | Sujith | Pathanamthitts | 689656 | 9496755712   |
+-------------+--------+----------------+--------+--------------+
3 rows in set (0.00 sec)
```

b)

```
mysql> DELIMITER //
mysql> CREATE PROCEDURE display_customers()
    -> BEGIN
    ->  SELECT name,city FROM customer;
    -> END //
Query OK, 0 rows affected (0.04 sec)

mysql> DELIMITER ;
mysql> CALL display_customers();
+--------+----------------+
| name   | city           |
+--------+----------------+
| Jelan  | Ezhamkulam     |
| Sabari | kollam         |
| Sujith | Pathanamthitts |
+--------+----------------+
3 rows in set (0.01 sec)

Query OK, 0 rows affected (0.02 sec)
```

c)

```
mysql> DELIMITER //
mysql> CREATE PROCEDURE display_customers_from_city(IN city_name
VARCHAR(15))
    -> BEGIN
    ->  SELECT * FROM customer WHERE city = city_name;
    -> END //
```

```
Query OK, 0 rows affected (0.01 sec)

mysql> DELIMITER ;
mysql> call display_customers_from_city('pathanamthitts');
+-------------+--------+----------------+--------+--------------+
| customer_id | name   | city           | pin    | phone_number |
+-------------+--------+----------------+--------+--------------+
|          59 | Sujith | Pathanamthitts | 689656 | 9496755712   |
+-------------+--------+----------------+--------+--------------+
1 row in set (0.01 sec)

Query OK, 0 rows affected (0.01 sec)

d)
mysql> DELIMITER //
mysql> CREATE PROCEDURE get_customer_phone(IN customer_name VARCHAR(15)
,OUT customer_phone VARCHAR(10))
    -> BEGIN
    ->  SELECT phone_number INTO customer_phone
    ->  FROM customer
    ->  WHERE name = customer_name;
    -> END
    -> //
Query OK, 0 rows affected (0.00 sec)

mysql> DELIMITER ;
mysql> CALL get_customer_phone('sujith',@phone);
Query OK, 1 row affected (0.01 sec)

mysql> SELECT @phone;
+------------+
| @phone     |
+------------+
| 9496755712 |
+------------+
```

OUTPUT

a)

```
CREATE TABLE account (
  -> account_no int NOT NULL,
  -> customer_name varchar(15),
  -> balance decimal(10,2) DEFAULT 0,
  -> PRIMARY KEY (account_no));
+------------+---------------+----------+
| account_no | customer_name | balance  |
+------------+---------------+----------+
|          1 | Sujith        |  5000.00 |
|          2 | James         |  6000.00 |
|          3 | Jelan         |  8500.00 |
|          4 | Bob           |  8000.00 |
|          5 | Charlie       |  9000.00 |
+------------+---------------+----------+
mysql> START TRANSACTION;
Query OK, 0 rows affected (0.00 sec)

mysql> UPDATE account
    -> SET balance = balance + 3000
    -> WHERE customer_name = 'Sujith';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> SELECT * FROM account;
+------------+---------------+---------+
| account_no | customer_name | balance |
+------------+---------------+---------+
|          1 | Sujith        | 8000.00 |
|          2 | James         | 6000.00 |
|          3 | Jelan         | 8500.00 |
|          4 | Bob           | 8000.00 |
|          5 | Charlie       | 9000.00 |
+------------+---------------+---------+
5 rows in set (0.00 sec)

mysql> SAVEPOINT save1;
Query OK, 0 rows affected (0.00 sec)
```

b)

```
mysql> UPDATE account
    -> SET balance = balance + 1000
    -> WHERE customer_name = 'Jelan';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> SELECT * FROM account;
```

```
+------------+---------------+---------+
| account_no | customer_name | balance |
+------------+---------------+---------+
|          1 | Sujith        | 8000.00 |
|          2 | James         | 6000.00 |
|          3 | Jelan         | 9500.00 |
|          4 | Bob           | 8000.00 |
|          5 | Charlie       | 9000.00 |
+------------+---------------+---------+
5 rows in set (0.00 sec)

mysql> ROLLBACK TO save1;
Query OK, 0 rows affected (0.00 sec)

mysql> SELECT * FROM account;
+------------+---------------+---------+
| account_no | customer_name | balance |
+------------+---------------+---------+
|          1 | Sujith        | 8000.00 |
|          2 | James         | 6000.00 |
|          3 | Jelan         | 8500.00 |
|          4 | Bob           | 8000.00 |
|          5 | Charlie       | 9000.00 |
+------------+---------------+---------+
5 rows in set (0.00 sec)

mysql> COMMIT;
Query OK, 0 rows affected (0.00 sec)
```