PRACTICAL NO : 1

For the given Library database
BOOK (Book_ISBN [PK], Title[Not Null], Publisher_ Name, price[Check Price>0], Date_Of_Publication,Book_Copy),
BOOK_AUTHORS (Book_ISBN [PK,FK]Author_Name [PK], Author_City)
Solve the following
a) Create view BOOK_AUTHOR_INFO consisting Book_ISBN, Title from BOOK Table and Author_Name from BOOK_AUTHORS Table in ascending order of ISBN no.
b) Create an index on Book_Author on table on attribute “Author_Name”.
c) Create table Book_Auto_Increment (BookID int Auto_increament=100, Book Name) insert five records in table.
d) Delete the book from Book table written by Author ‘Korth’.
e) Select Book Names from table Book whose copies are in between 10 to 15.

1)CREATE BOOK TABLE

mysql> create table book(book_isbn int primary key,title varchar(50),publisher_name varchar(50),price int check(price>0),date_of_publication date,book_copy int);
Query OK, 0 rows affected (0.02 sec)

DESCRIBE BOOK TABLE

```
mysql> describe book;
+--------------------+-------------+------+-----+---------+-------+
| Field              | Type        | Null | Key | Default | Extra |
+--------------------+-------------+------+-----+---------+-------+
| book_isbn          | int         | NO   | PRI | NULL    |       |
| title              | varchar(50) | YES  |     | NULL    |       |
| publisher_name     | varchar(50) | YES  |     | NULL    |       |
| price              | int         | YES  |     | NULL    |       |
| date_of_publication | date        | YES  |     | NULL    |       |
| book_copy          | int         | YES  |     | NULL    |       |
+--------------------+-------------+------+-----+---------+-------+
6 rows in set (0.01 sec)
```

INSERT 5 RECORDS IN BOOK TABLE

```
mysql> insert into book(book_isbn,title,publisher_name,price,date_of_publica
tion,book_copy)values
    -> (1, 'Book1', 'Publisher1', 25.00, '2023-01-15', 50),
    -> (2,'Book2', 'Publisher2',40,'2022-07-20',40),
```

```
    -> (3, 'Book3', 'Publisher3', 20.00, '2021-05-10', 60),
    -> (4, 'Book4', 'Publisher1', 150, '2020-11-30', 35),
    -> (5, 'Book5', 'Publisher4', 180, '2019-09-25', 45);
Query OK, 5 rows affected (0.01 sec)
```

DISPLAY THE INSERTED RECORDS OF THE BOOK TABLE

mysql> select * from book;

```
+-----------+-------+---------------+-------+--------------------+-----------+
| book_isbn | title | publisher_name | price | date_of_publication | book_copy |
+-----------+-------+---------------+-------+--------------------+-----------+
|         1 | Book1 | Publisher1     |    25 | 2023-01-15         |        50 |
|         2 | Book2 | Publisher2     |    40 | 2022-07-20         |        40 |
|         3 | Book3 | Publisher3     |    20 | 2021-05-10         |        60 |
|         4 | Book4 | Publisher1     |   150 | 2020-11-30         |        35 |
|         5 | Book5 | Publisher4     |   180 | 2019-09-25         |        45 |
+-----------+-------+---------------+-------+--------------------+-----------+
5 rows in set (0.00 sec)
```

2)CREATE BOOK_AUTHOR TABLE

```
mysql> create table book_author(book_isbn int ,author_name varchar(50),autho
r_city varchar(50),primary key(book_isbn,author_name),foreign key(book_isbn)
references book(book_isbn));
Query OK, 0 rows affected (0.02 sec)
Records: 5  Duplicates: 0  Warnings: 0
```

DESCRIBE BOOK_AUTHOR TABLE

```
mysql> describe book_author;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| book_isbn   | int         | NO   | PRI | NULL    |       |
| author_name | varchar(50) | NO   | PRI | NULL    |       |
| author_city | varchar(50) | YES  |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
3 rows in set (0.00 sec)
```

INSERT 5 RECORDS IN BOOK_AUTHOR TABLE

```
mysql> insert into book_author(book_isbn,author_name,author_city)values
    -> (1,'author1','city1'),
    -> (2,'author2','city2'),
```

```
   -> (3,'author3','city3'),
   -> (4,'author4','city4'),
   -> (5,'author5','city5');
Query OK, 5 rows affected (0.00 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql> INSERT INTO BOOK_AUTHOR (Book_ISBN, Author_Name, Author_City)values
   ->   ('4', 'korth', 'City A');
Query OK, 1 row affected (0.01 sec)
```

DISPLAY THE INSERTED RECORDS OF THE BOOK_AUTHOR TABLE

mysql> select *from book_author;

```
+-----------+-------------+-------------+
| book_isbn | author_name | author_city |
+-----------+-------------+-------------+
|         1 | author1     | city1       |
|         2 | author2     | city2       |
|         3 | author3     | city3       |
|         4 | author4     | city4       |
|         4 | korth       | City A      |
|         5 | author5     | city5       |
+-----------+-------------+-------------+
```

6 rows in set (0.00 sec)

a) Create view BOOK_AUTHOR_INFO consisting of Book_ISBN, Title from BOOK Table and Author_Name from BOOK_AUTHORS Table in ascending order of ISBN no.:

```
mysql> create view book_author_info as
   -> select B.book_isbn, B.title,BA.author_name
   -> from book B
   -> inner join book_author BA on B.book_isbn=BA.book_isbn
   -> order by B.book_isbn ASC;
Query OK, 0 rows affected (0.01 sec)


mysql> select * from book_author_info;
+-----------+-------+-------------+
| book_isbn | title | author_name |
+-----------+-------+-------------+
|         1 | Book1 | author1     |
|         2 | Book2 | author2     |
```

```
|         3 | Book3 | author3     |
|         4 | Book4 | author4     |
|         5 | Book5 | author5     |
+-----------+-------+-------------+
5 rows in set (0.00 sec)
```

b) Create an index on Book_Author on the table on attribute "Author_Name":

```
mysql> create index idx_author_name on book_author(author_name);
Query OK, 0 rows affected (0.05 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

mysql> show index from book_author;

```
+-------------+------------+-----------------+--------------+-------------+-----------+-------------+----------+--------
+------+------------+---------+---------------+---------+------------+
| Table       | Non_unique | Key_name        | Seq_in_index | Column_name | Collation |
Cardinality | Sub_part | Packed | Null | Index_type | Comment | Index_comment | Visible |
Expression |
+-------------+------------+-----------------+--------------+-------------+-----------+-------------+----------+--------
+------+------------+---------+---------------+---------+------------+
| book_author |          0 | PRIMARY         |            1 | book_isbn   | A         |           5 |     NULL |
NULL |      | BTREE      |         |               | YES     | NULL       |
| book_author |          0 | PRIMARY         |            2 | author_name | A         |           5 |     NULL |
NULL |      | BTREE      |         |               | YES     | NULL       |
| book_author |          1 | idx_author_name |            1 | author_name | A         |           5 |     NULL |
NULL |      | BTREE      |         |               | YES     | NULL       |
+-------------+------------+-----------------+--------------+-------------+-----------+-------------+----------+--------
+------+------------+---------+---------------+---------+------------+
3 rows in set (0.02 sec)
```

c) Create table Book_Auto_Increment (BookID int Auto_increament=100, Book Name) insert five records in the table:

d) Delete the book from Book table written by Author ‘korth’.

e) Select Book Names from table Book whose copies are in between 30 to 50.

```
mysql> select title as book_name from book where book_copy between 30 and 50
;
+-----------+
| book_name |
+-----------+
| Book1     |
| Book2     |
| Book4     |
| Book5     |
+-----------+
4 rows in set (0.00 sec)
```

PRACTICAL NO : 2

2 . For the given Library database
BOOK (Book_ISBN [PK], Title[Not Null], Publisher_ Name, price[Check Price>0], Date_Of_Publication,Book_Copy),
BOOK_AUTHORS (Book_ISBN [PK,FK]Author_Name [PK], Author_City)
Solve the following :

a) Select Book_ISBN, Title, Author_Name from relations Book and Book_Authors INNER JOIN on attribute Book_ISBN.

```
mysql> select B.book_isbn, B.title, BA.author_name from book B inner join bo
ok_author BA on B.book_isbn = BA.book_isbn;
+-----------+-------+-------------+
| book_isbn | title | author_name |
+-----------+-------+-------------+
|         1 | Book1 | author1     |
|         2 | Book2 | author2     |
|         3 | Book3 | author3     |
|         4 | Book4 | author4     |
|         5 | Book5 | author5     |
+-----------+-------+-------------+
5 rows in set (0.02 sec)
```

b) Select Book_ISBN, Title, Publisher, Author_Name from relations Book and Book_Authors LEFT OUTER JOIN on attribute Book_ISBN.

```
mysql> select B.book_isbn, B.title,B.publisher_name, BA.author_name from book B left outer
join book_author BA on B.book_isbn = BA.book_isbn;
```

```
+-----------+-------+---------------+-------------+
| book_isbn | title | publisher_name | author_name |
+-----------+-------+---------------+-------------+
|         1 | Book1 | Publisher1     | author1     |
|         2 | Book2 | Publisher2     | author2     |
|         3 | Book3 | Publisher3     | author3     |
|         4 | Book4 | Publisher1     | author4     |
|         5 | Book5 | Publisher4     | author5     |
+-----------+-------+---------------+-------------+
5 rows in set (0.00 sec)
```

c) Select Book_ISBN, Title, Publisher, Author_Name from relations Book and Book_Authors RIGHT OUTER JOIN on attribute Book_ISBN.

```
mysql> select B.book_isbn, B.title,B.publisher_name, BA.author_name from book B right outer
join book_author BA on B.book_isbn = BA.book_isbn;
+-----------+-------+---------------+-------------+
| book_isbn | title | publisher_name | author_name |
+-----------+-------+---------------+-------------+
|         1 | Book1 | Publisher1     | author1     |
|         2 | Book2 | Publisher2     | author2     |
|         3 | Book3 | Publisher3     | author3     |
|         4 | Book4 | Publisher1     | author4     |
|         5 | Book5 | Publisher4     | author5     |
+-----------+-------+---------------+-------------+
5 rows in set (0.00 sec)
```

d) Select Book_ISBN, Title from relation Book whose author is living in City ='Pune'.

```
mysql> select B.book_isbn,B.title
    -> from book B
    -> inner join book_author BA on B.book_isbn =BA.book_isbn
    -> where BA.author_city='city3';
+-----------+-------+
| book_isbn | title |
+-----------+-------+
|         3 | Book3 |
+-----------+-------+
1 row in set (0.00 sec)
```

e) Select Book_ISBN, Title from relation Book, which written by more than 2 Authors.

```
mysql> select b.book_isbn,b.title
    -> from book b
    -> inner join(
    -> select book_isbn
    -> from book_author
    -> group by book_isbn
    -> having count(*) > 2
    -> )ba on b.book_isbn = ba.book_isbn;
Empty set (0.02 sec)
```

PRACTICAL NO : 3

3 For the given Library database
BOOK (Book_ISBN [PK], Title[Not Null], Publisher_ Name, price[Check Price>0], Date_Of_Publication,Book_Copy),
BOOK_AUTHORS (Book_ISBN [PK,FK]Author_Name [PK], Author_City)
Solve the following

```
mysql> create table books (Book_ISBN varchar(20), Title varchar(50), Publisher_Name varchar(50), price float, Date_Of_Publication date, Book_Copy int)
    -> ;
Query OK, 0 rows affected (0.03 sec)

mysql> INSERT INTO BOOKs (Book_ISBN, Title, Publisher_Name, price, Date_Of_P ublication, Book_Copy)values
    -> ('ISBN1', 'Book Title 1', 'Publisher A', 15.99, '2023-01-01', 10),
    -> ('ISBN2', 'Book Title 2', 'Publisher B', 19.99, '2023-02-01', 12),
    -> ('ISBN3', 'Book Title 3', 'Publisher A', 12.99, '2023-03-01', 8),
    -> ('ISBN4', 'Book Title 4', 'Publisher C', 9.99, '2023-04-01', 14),
    -> ('ISBN5', 'Book Title 5', 'Publisher B', 14.99, '2023-05-01', 9);
Query OK, 5 rows affected (0.01 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql> describe table books;
+----+-------------+-------+------------+------+---------------+------+---------+------+------+----------+-------+
| id | select_type | table | partitions | type | possible_keys | key  | key_len | ref  | rows | filtered | Extra |
+----+-------------+-------+------------+------+---------------+------+---------+------+------+----------+-------+
|  1 | SIMPLE      | books | NULL       | ALL  | NULL          | NULL | NULL    | NULL |    5 |   100.00 | NULL  |
```

```
+----+-------------+-------+------------+------+---------------+------+---------+------+------+----------+-------+
1 row in set, 1 warning (0.00 sec)

mysql> select * from books;
+-----------+--------------+----------------+-------+--------------------+-----------+
| Book_ISBN | Title        | Publisher_Name | price | Date_Of_Publication | Book_Copy |
+-----------+--------------+----------------+-------+--------------------+-----------+
| ISBN1     | Book Title 1 | Publisher A    | 15.99 | 2023-01-01          |        10 |
| ISBN2     | Book Title 2 | Publisher B    | 19.99 | 2023-02-01          |        12 |
| ISBN3     | Book Title 3 | Publisher A    | 12.99 | 2023-03-01          |         8 |
| ISBN4     | Book Title 4 | Publisher C    |  9.99 | 2023-04-01          |        14 |
| ISBN5     | Book Title 5 | Publisher B    | 14.99 | 2023-05-01          |         9 |
+-----------+--------------+----------------+-------+--------------------+-----------+
5 rows in set (0.00 sec)

mysql> create table BOOK_AUTHORS (Book_ISBN varchar(20), Author_Name varchar(20),
Author_City varchar(20));
Query OK, 0 rows affected (0.02 sec)

mysql> INSERT INTO BOOK_AUTHORS (Book_ISBN, Author_Name, Author_City)values
    -> ('ISBN1', 'Author A', 'City A'),
    -> ('ISBN1', 'Author B', 'City B'),
    -> ('ISBN2', 'Author C', 'City C'),
    -> ('ISBN3', 'Author D', 'City A'),
    -> ('ISBN4', 'Author E', 'City D');

Query OK, 5 rows affected (0.02 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql> INSERT INTO BOOK_AUTHORS (Book_ISBN, Author_Name, Author_City)values
    ->  ('ISBN5', 'korth', 'City A');
Query OK, 1 row affected (0.01 sec)


mysql> describe table book_authors;
+----+-------------+--------------+------------+------+---------------+------+---------+------+------+----------+-----
--+
| id | select_type | table        | partitions | type | possible_keys | key  | key_len | ref  | rows |
filtered | Extra |
+----+-------------+--------------+------------+------+---------------+------+---------+------+------+----------+-----
--+
|  1 | SIMPLE      | book_authors | NULL       | ALL  | NULL          | NULL | NULL    | NULL |    5 |
100.00 | NULL  |
```

```
+----+-------------+--------------+------------+------+---------------+------+---------+------+------+----------+-----
--+
1 row in set, 1 warning (0.00 sec)

mysql>  select *from book_authors;
+-----------+-------------+-------------+
| Book_ISBN | Author_Name | Author_City |
+-----------+-------------+-------------+
| ISBN1     | Author A    | City A      |
| ISBN1     | Author B    | City B      |
| ISBN2     | Author C    | City C      |
| ISBN3     | Author D    | City A      |
| ISBN4     | Author E    | City D      |
| ISBN5     | korth       | City A      |
+-----------+-------------+-------------+
6 rows in set (0.00 sec)
```

i) Display name of publishers as per no of books published by them in ascending order.

```
mysql> SELECT Publisher_Name, COUNT(Book_ISBN) AS BooksPublished
    -> FROM BOOK
    -> GROUP BY Publisher_Name
    -> ORDER BY BooksPublished ASC;
+----------------+----------------+
| Publisher_Name | BooksPublished |
+----------------+----------------+
| Publisher2     |              1 |
| Publisher3     |              1 |
| Publisher4     |              1 |
| Publisher1     |              2 |
+----------------+----------------+
4 rows in set (0.01 sec)
```

ii) Get publisher names who published at least one book written by author name like “K%’’.

```
mysql> SELECT DISTINCT b.Publisher_Name
    -> FROM BOOK b
    -> INNER JOIN BOOK_AUTHORS ba ON b.Book_ISBN = ba.Book_ISBN
    -> WHERE ba.Author_Name LIKE 'k%';
```

iii) Get book name and Authors names where book written by maximum Authors.

```
mysql> SELECT b.Title, GROUP_CONCAT(ba.Author_Name) AS Authors
    -> FROM BOOK b
    -> INNER JOIN BOOK_AUTHORS ba ON b.Book_ISBN = ba.Book_ISBN
    -> GROUP BY b.Title
    -> HAVING COUNT(ba.Author_Name) = (
    ->  SELECT MAX(author_count)
    -> FROM (
    -> SELECT COUNT(Author_Name) AS author_count
    -> FROM BOOK_AUTHORS
    -> GROUP BY Book_ISBN
    -> ) AS counts
    -> );
Empty set (0.02 sec)
```

iv) Get publisher names accordingly books published alphabetically

```
mysql> SELECT DISTINCT Publisher_Name
    -> FROM BOOKs
    -> ORDER BY Publisher_Name ASC;
+----------------+
| Publisher_Name |
+----------------+
| Publisher A    |
| Publisher B    |
| Publisher C    |
+----------------+
3 rows in set (0.00 sec)
```

Find the no of books published in 01 Jan 2014 to till date.

```
mysql> SELECT COUNT(*) AS BooksPublished
    -> FROM BOOKs
    -> WHERE Date_Of_Publication >= '2014-01-01';
+----------------+
| BooksPublished |
```

```
+----------------+
|              5 |
+----------------+
1 row in set (0.00 sec)
```

PRACTICAL NO : 4

Consider insurance database with following schema :

person(driver-id, name, address)
car(license, model, year)
accident (report - no, date, location)
owns(driver-id,license)
participated(driver-id,car,report-no,damage-amount)
Write a query in SQL for following requirements :
i) Find the total no. of people who owned cars that were involved in accidents in 2016.
ii) Retrieve the name of person whose address contains Pune.
iii) Find the name of persons having more than two cars.

```
mysql> create table person(driver_id int primary key, name varchar(50), addr
ess varchar(50));
Query OK, 0 rows affected (0.03 sec)

mysql> insert into person(driver_id,name,address) values
    -> (1, 'ram','pune'),
    -> (2, 'sham','nashik'),
    -> (3, 'rutvik','supe'),
    -> (4, 'aniket','baramati'),
    -> (5, 'shivtej','malegan');
Query OK, 5 rows affected (0.01 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql> describe table person;
+----+-------------+--------+------------+------+---------------+------+---------+------+------+----------+-------+
| id | select_type | table  | partitions | type | possible_keys | key  | key_len | ref  | rows | filtered | Extra |
+----+-------------+--------+------------+------+---------------+------+---------+------+------+----------+-------+
|  1 | SIMPLE      | person | NULL       | ALL  | NULL          | NULL | NULL    | NULL |    5 |   100.00 | NULL  |
+----+-------------+--------+------------+------+---------------+------+---------+------+------+----------+-------+
1 row in set, 1 warning (0.00 sec)

mysql> select * from person;
+-----------+---------+----------+
| driver_id | name    | address  |
+-----------+---------+----------+
|         1 | ram     | pune     |
```

```
|         2 | sham    | nashik   |
|         3 | rutvik  | supe     |
|         4 | aniket  | baramati |
|         5 | shivtej | malegan  |
+-----------+---------+----------+
5 rows in set (0.00 sec)

mysql> create table car(license varchar(10) primary key, model varchar(20),year int);
Query OK, 0 rows affected (0.01 sec)

mysql> insert into car(license,model,year)values
    -> ('ABC123', 'Toyota', 2018),
    -> ('XYZ456', 'Honda', 2019),
    -> ('DEF789', 'Ford', 2017),
    -> ('GHI101', 'Chevrolet', 2020),
    -> ('JKL202', 'BMW', 2016);
Query OK, 5 rows affected (0.00 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql> describe table car;
+----+-------------+-------+------------+------+---------------+------+---------+------+------+----------+-------+
| id | select_type | table | partitions | type | possible_keys | key  | key_len | ref  | rows | filtered |
Extra |
+----+-------------+-------+------------+------+---------------+------+---------+------+------+----------+-------+
|  1 | SIMPLE      | car   | NULL       | ALL  | NULL          | NULL | NULL    | NULL |    5 |   100.00 |
NULL  |
+----+-------------+-------+------------+------+---------------+------+---------+------+------+----------+-------+
1 row in set, 1 warning (0.00 sec)

mysql> select * from car;
+---------+-----------+------+
| license | model     | year |
+---------+-----------+------+
| ABC123  | Toyota    | 2018 |
| DEF789  | Ford      | 2017 |
| GHI101  | Chevrolet | 2020 |
| JKL202  | BMW       | 2016 |
| XYZ456  | Honda     | 2019 |
+---------+-----------+------+
5 rows in set (0.00 sec)

mysql> create table accident(report_no int primary key, date date, location
varchar(50));
Query OK, 0 rows affected (0.03 sec)
```

```
mysql> insert into accident(report_no,date,location)values
    -> (1, '2016-05-10', 'Pune'),
    -> (2, '2016-08-20', 'Mumbai'),
    -> (3, '2016-11-15', 'Pune'),
    -> (4, '2017-01-25', 'Delhi'),
    -> (5, '2016-09-05', 'Pune');
Query OK, 5 rows affected (0.02 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql> describe table accident;
+----+-------------+----------+------------+------+---------------+------+---------+------+------+----------+-------+
| id | select_type | table    | partitions | type | possible_keys | key  | key_len | ref  | rows | filtered | Extra |
+----+-------------+----------+------------+------+---------------+------+---------+------+------+----------+-------+
|  1 | SIMPLE      | accident | NULL       | ALL  | NULL          | NULL | NULL    | NULL |    5 |   100.00 | NULL  |
+----+-------------+----------+------------+------+---------------+------+---------+------+------+----------+-------+
1 row in set, 1 warning (0.00 sec)

mysql> select * from accident;
+-----------+------------+----------+
| report_no | date       | location |
+-----------+------------+----------+
|         1 | 2016-05-10 | Pune     |
|         2 | 2016-08-20 | Mumbai   |
|         3 | 2016-11-15 | Pune     |
|         4 | 2017-01-25 | Delhi    |
|         5 | 2016-09-05 | Pune     |
+-----------+------------+----------+
5 rows in set (0.00 sec)

mysql> create table owns( driver_id int, license varchar(20), foreign key (driver_id) references person(driver_id),foreign key (license) references car(license), primary key (driver_id,license));
Query OK, 0 rows affected (0.05 sec)

mysql> insert into owns(driver_id,license) values
    -> (1, 'ABC123'),
    -> (2, 'XYZ456'),
    -> (3, 'DEF789'),
    -> (4, 'GHI101'),
    -> (5, 'JKL202');
Query OK, 5 rows affected (0.00 sec)
Records: 5  Duplicates: 0  Warnings: 0
```

```
mysql> describe table owns;
+----+-------------+-------+------------+-------+---------------+---------+---------+------+------+----------+-----------------+
| id | select_type | table | partitions | type  | possible_keys | key     | key_len | ref  | rows | filtered | Extra           |
+----+-------------+-------+------------+-------+---------------+---------+---------+------+------+----------+-----------------+
|  1 | SIMPLE      | owns  | NULL       | index | NULL          | license | 82      | NULL |    5 |   100.00 | Using index |
+----+-------------+-------+------------+-------+---------------+---------+---------+------+------+----------+-----------------+
1 row in set, 1 warning (0.01 sec)

mysql> select * from owns;
+-----------+---------+
| driver_id | license |
+-----------+---------+
|         1 | ABC123  |
|         3 | DEF789  |
|         4 | GHI101  |
|         5 | JKL202  |
|         2 | XYZ456  |
+-----------+---------+
5 rows in set (0.00 sec)


mysql> create table participated(driver_id int, car_license varchar(50),repo
rt_no int,damage_amount decimal(10,2),FOREIGN KEY (driver_id) REFERENCES
person(driver_id),FOREIGN KEY (car_license) REFERENCES car(license),FOREIGN KEY
(report_no) REFERENCES accident(report_no), PRIMARY KEY (driver_id, car_license,
report_no));
Query OK, 0 rows affected (0.02 sec)

mysql> insert into participated(driver_id, car_license, report_no, damage_amount) values
    -> (1, 'ABC123', 1, 5000),
    -> (2, 'XYZ456', 2, 7000),
    -> (3, 'DEF789', 3, 3000),
    -> (4, 'GHI101', 4, 10000),
    -> (5, 'JKL202', 5, 1500);
Query OK, 5 rows affected (0.01 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql> describe table  participated;
```

```
+----+-------------+-------------+------------+------+---------------+------+---------+------+------+----------+-----
--+
| id | select_type | table       | partitions | type | possible_keys | key  | key_len | ref  | rows |
filtered | Extra |
+----+-------------+-------------+------------+------+---------------+------+---------+------+------+----------+-----
--+
|  1 | SIMPLE      | participated | NULL       | ALL  | NULL          | NULL | NULL    | NULL |    5 |
100.00 | NULL  |
+----+-------------+-------------+------------+------+---------------+------+---------+------+------+----------+-----
--+
1 row in set, 1 warning (0.00 sec)

mysql> select * from  participated;
+-----------+-------------+-----------+---------------+
| driver_id | car_license | report_no | damage_amount |
+-----------+-------------+-----------+---------------+
|         1 | ABC123      |         1 |       5000.00 |
|         2 | XYZ456      |         2 |       7000.00 |
|         3 | DEF789      |         3 |       3000.00 |
|         4 | GHI101      |         4 |      10000.00 |
|         5 | JKL202      |         5 |       1500.00 |
+-----------+-------------+-----------+---------------+
5 rows in set (0.00 sec)
```

i) Find the total no. of people who owned cars that were involved in accidents in 2016.

```
mysql> SELECT COUNT(DISTINCT p.driver_id) AS TotalPeople
    -> FROM person p
    -> JOIN owns o ON p.driver_id = o.driver_id
    -> JOIN car c ON o.license = c.license
    -> JOIN participated pd ON c.license = pd.car_license
    -> JOIN accident a ON pd.report_no = a.report_no
    -> WHERE YEAR(a.date) = 2016;
+-------------+
| TotalPeople |
+-------------+
|           4 |
+-------------+
1 row in set (0.01 sec)
```

ii) Retrieve the name of person whose address contains Pune.

```
mysql> SELECT name
    -> FROM person
```

```
   -> WHERE address LIKE '%Pune%';
+------+
| name |
+------+
| ram  |
+------+
1 row in set (0.01 sec)
```

iii) Find the name of persons having more than two cars.

```
mysql> SELECT p.name
   -> FROM person p
   -> JOIN owns o ON p.driver_id = o.driver_id
   -> GROUP BY p.driver_id
   -> HAVING COUNT(o.license) > 2;
Empty set (0.01 sec)
```

PRACTICAL NO : 6

For the given Employee database
EmployeeInfo(EmpID[PK],EmpFname,EmpLname,Department,Project,Address,DOB,Ge
nder)
EmployeePosition(EmpID[FK],EmpPosition,DateOfJoining,Salary)

```
mysql>  CREATE TABLE EmployeeInfo (EmpID INT PRIMARY KEY,EmpFname
VARCHAR(50),EmpLname VARCHAR(50),Department VARCHAR(50),Project
VARCHAR(50),Address VARCHAR(100), DOB DATE,Gender VARCHAR(10));
Query OK, 0 rows affected (0.04 sec)

mysql> INSERT INTO EmployeeInfo (EmpID, EmpFname, EmpLname, Department, Project,
Address, DOB, Gender) VALUES
    -> (1, 'John', 'Doe', 'HR', 'Project A', ' Pune', '1990-05-15', 'Male'),
    -> (2, 'Alice', 'Smith', 'IT', 'Project B', ' Mumbai', '1988-09-20', 'Female'),
    -> (3, 'Bob', 'Johnson', 'HR', 'Project C', ' Pune', '1995-02-10', 'Male'),
    -> (4, 'Emily', 'Davis', 'Marketing', 'Project D', ' Delhi', '1992-11-30', 'Female'),
    -> (5, 'Michael', 'Wilson', 'IT', 'Project E', ' Pune', '1985-07-25', 'Male');
Query OK, 5 rows affected (0.01 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql> describe table  EmployeeInfo;
+----+-------------+--------------+------------+------+---------------+------+---------+------+------+----------+-----
--+
| id | select_type | table        | partitions | type | possible_keys | key  | key_len | ref  | rows |
filtered | Extra |
+----+-------------+--------------+------------+------+---------------+------+---------+------+------+----------+-----
--+
|  1 | SIMPLE      | EmployeeInfo | NULL       | ALL  | NULL          | NULL | NULL    | NULL |    5 |
100.00 | NULL  |
+----+-------------+--------------+------------+------+---------------+------+---------+------+------+----------+-----
--+
1 row in set, 1 warning (0.00 sec)

mysql> select *  from  EmployeeInfo;
+-------+----------+----------+------------+-----------+---------+------------+--------+
| EmpID | EmpFname | EmpLname | Department | Project   | Address | DOB        | Gender |
+-------+----------+----------+------------+-----------+---------+------------+--------+
|     1 | John     | Doe      | HR         | Project A |  Pune   | 1990-05-15 | Male   |
|     2 | Alice    | Smith    | IT         | Project B |  Mumbai | 1988-09-20 | Female |
|     3 | Bob      | Johnson  | HR         | Project C |  Pune   | 1995-02-10 | Male   |
|     4 | Emily    | Davis    | Marketing  | Project D |  Delhi  | 1992-11-30 | Female |
|     5 | Michael  | Wilson   | IT         | Project E |  Pune   | 1985-07-25 | Male   |
```

```
+-------+----------+----------+------------+-----------+---------+------------+--------+
5 rows in set (0.01 sec)

mysql> CREATE TABLE EmployeePosition (EmpID INT PRIMARY KEY,EmpPosition
VARCHAR(50),DateOfJoining DATE,Salary DECIMAL(10, 2), FOREIGN KEY (EmpID)
REFERENCES EmployeeInfo(EmpID));
Query OK, 0 rows affected (0.03 sec)

mysql> INSERT INTO EmployeePosition (EmpID, EmpPosition, DateOfJoining, Salary)
VALUES
    -> (1, 'Manager', '2015-03-10', 75000),
    -> (2, 'Developer', '2018-06-20', 65000),
    -> (3, 'HR Executive', '2019-01-15', 55000),
    -> (4, 'Marketing Specialist', '2017-08-05', 90000),
    -> (5, 'Software Engineer', '2016-02-28', 80000);
Query OK, 5 rows affected (0.02 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql> describe table  EmployeePosition;
+----+-------------+-----------------+------------+------+---------------+------+---------+------+------+----------+-
------+
| id | select_type | table           | partitions | type | possible_keys | key  | key_len | ref  | rows |
filtered | Extra |
+----+-------------+-----------------+------------+------+---------------+------+---------+------+------+----------+-
------+
|  1 | SIMPLE      | EmployeePosition | NULL       | ALL  | NULL          | NULL | NULL    | NULL |
5 |   100.00 | NULL  |
+----+-------------+-----------------+------------+------+---------------+------+---------+------+------+----------+-
------+
1 row in set, 1 warning (0.00 sec)

mysql> select * from  EmployeePosition;
+-------+---------------------+--------------+----------+
| EmpID | EmpPosition         | DateOfJoining | Salary   |
+-------+---------------------+--------------+----------+
|     1 | Manager             | 2015-03-10    | 75000.00 |
|     2 | Developer           | 2018-06-20    | 65000.00 |
|     3 | HR Executive        | 2019-01-15    | 55000.00 |
|     4 | Marketing Specialist | 2017-08-05    | 90000.00 |
|     5 | Software Engineer    | 2016-02-28    | 80000.00 |
+-------+---------------------+--------------+----------+
5 rows in set (0.00 sec)
```

i. Write a query to fetch the EmpFname from the EmployeeInfo table in the upper case and use the ALIAS name as EmpName.

```
mysql> SELECT UPPER(EmpFname) AS EmpName FROM EmployeeInfo;
+---------+
| EmpName |
+---------+
| JOHN    |
| ALICE   |
| BOB     |
| EMILY   |
| MICHAEL |
+---------+
```

ii. Write a query to fetch the number of employees working in the department ‘HR’.

```
mysql> SELECT COUNT(*) AS EmployeesInHR FROM EmployeeInfo WHERE Department =
'HR';
+---------------+
| EmployeesInHR |
+---------------+
|             2 |
+---------------+
1 row in set (0.00 sec)
```

iii. Write q query to find all the employees whose salary is between 50000 to 100000

```
mysql> SELECT EmpFname FROM EmployeeInfo ei JOIN EmployeePosition ep ON ei.EmpID
= ep.EmpID WHERE ep.Salary BETWEEN 50000 AND 100000;
+----------+
| EmpFname |
+----------+
| John     |
| Alice    |
| Bob      |
| Emily    |
| Michael  |
+----------+
5 rows in set (0.01 sec)
```

iv. Write a query to find the names of employees that begin with ‘b’

```
mysql> SELECT EmpFname FROM EmployeeInfo WHERE EmpFname LIKE 'b%';
```

```
+----------+
| EmpFname |
+----------+
| Bob      |
+----------+
1 row in set (0.00 sec)
```

v. Write a query to fetch top N records.

```
mysql> SELECT * FROM EmployeeInfo LIMIT 1;
+-------+----------+----------+------------+-----------+---------+------------+--------+
| EmpID | EmpFname | EmpLname | Department | Project   | Address | DOB        | Gender |
+-------+----------+----------+------------+-----------+---------+------------+--------+
|     1 | John     | Doe      | HR         | Project A |  Pune   | 1990-05-15 | Male   |
+-------+----------+----------+------------+-----------+---------+------------+--------+
1 row in set (0.00 sec)
```

PRACTICAL NO : 27

Create a table named STUDENT with the following fields : 20 (FIRST NAME, MIDDLE NAME, LAST

NAME, STUDENT_ENRLNO, DATE_OF_BIRTH, CLASS, SECTION, GENDER, YEAR_OF JOIN,
ADMISSION_NO, ADDRESS1, ADDRESS2, CITY, STATE, RESPHONE, PIN_CODE)
(a) Display all the list of students who are in class - 6, section - A.
(b) To display all the students list whose first name starts with "A".
(c) To display all the students list who are girls.
(d) To display all the students whose YEAR-OF-JOIN is 2000.
(e) Sort the records of students with respect to their ADMISSION_NO, in ascending order.

```
mysql> use dbms;
Database changed

mysql>  CREATE TABLE STUDENT( FIRST_NAME VARCHAR(50),MIDDLE_NAME
VARCHAR(50), LAST_NAME VARCHAR(50),STUDENT_ENRLNO INT, DATE_OF_BIRTH
DATE,CLASS INT,SECTION VARCHAR(10),GENDER VARCHAR(10),YEAR_OF_JOIN INT,
ADMISSION_NO INT, ADDRESS1 VARCHAR(100), ADDRESS2 VARCHAR(100), CITY
VARCHAR(50), STATE VARCHAR(50), RESPHONE VARCHAR(15),PIN_CODE
VARCHAR(10));
Query OK, 0 rows affected (0.04 sec)

mysql> INSERT INTO STUDENT (FIRST_NAME, MIDDLE_NAME, LAST_NAME,
STUDENT_ENRLNO, DATE_OF_BIRTH, CLASS, SECTION, GENDER, YEAR_OF_JOIN,
ADMISSION_NO, ADDRESS1, ADDRESS2, CITY, STATE, RESPHONE, PIN_CODE)VALUES
    -> ('Alice', 'Marie', 'Smith', 1, '2008-05-15', 6, 'A', 'Female', 2000, 1001, '123 Main St', 'Apt
101', 'City1', 'State1', '1234567890', '12345'),
    -> ('Bob', 'Alexander', 'Johnson', 2, '2007-08-20', 7, 'B', 'Male', 2001, 1002, '456 Elm St', 'Apt
202', 'City2', 'State2', '2345678901', '23456'),
    -> ('Anna', 'Grace', 'Williams', 3, '2009-01-10', 6, 'A', 'Female', 2000, 1003, '789 Oak St', 'Apt
303', 'City3', 'State3', '3456789012', '34567'),
    -> ('Alex', 'James', 'Brown', 4, '2008-11-28', 6, 'B', 'Male', 2000, 1004, '012 Pine St', 'Apt 404',
'City4', 'State4', '4567890123', '45678'),
    -> ('Ava', 'Rose', 'Miller', 5, '2009-03-05', 5, 'A', 'Female', 2002, 1005, '345 Cedar St', 'Apt 505',
'City5', 'State5', '5678901234', '56789');
Query OK, 5 rows affected (0.01 sec)
Records: 5  Duplicates: 0  Warnings: 0
```

mysql> describe table STUDENT;

```
+----+-------------+---------+------------+------+---------------+------+---------+------+------+----------+-------+
| id | select_type | table   | partitions | type | possible_keys | key  | key_len | ref  | rows | filtered | Extra |
+----+-------------+---------+------------+------+---------------+------+---------+------+------+----------+-------+
|  1 | SIMPLE      | STUDENT | NULL       | ALL  | NULL          | NULL | NULL    | NULL |    5 |   100.00 | NULL  |
+----+-------------+---------+------------+------+---------------+------+---------+------+------+----------+-------+
1 row in set, 1 warning (0.00 sec)
```

mysql> SELECT * FROM STUDENT;

```
+------------+-------------+-----------+----------------+---------------+-------+---------+--------+-------------+-----------+-------------+----------+-------+--------+------------+----------+
| FIRST_NAME | MIDDLE_NAME | LAST_NAME | STUDENT_ENRLNO | DATE_OF_BIRTH | CLASS | SECTION | GENDER | YEAR_OF_JOIN | ADMISSION_NO | ADDRESS1     | ADDRESS2 | CITY  | STATE  | RESPHONE   | PIN_CODE |
+------------+-------------+-----------+----------------+---------------+-------+---------+--------+-------------+-----------+-------------+----------+-------+--------+------------+----------+
| Alice      | Marie       | Smith     |              1 | 2008-05-15    |     6 | A       | Female |         2000 |      1001 | 123 Main St  | Apt 101  | City1 | State1 | 1234567890 | 12345    |
| Bob        | Alexander   | Johnson   |              2 | 2007-08-20    |     7 | B       | Male   |         2001 |      1002 | 456 Elm St   | Apt 202  | City2 | State2 | 2345678901 | 23456    |
| Anna       | Grace       | Williams  |              3 | 2009-01-10    |     6 | A       | Female |         2000 |      1003 | 789 Oak St   | Apt 303  | City3 | State3 | 3456789012 | 34567    |
| Alex       | James       | Brown     |              4 | 2008-11-28    |     6 | B       | Male   |         2000 |      1004 | 012 Pine St  | Apt 404  | City4 | State4 | 4567890123 | 45678    |
| Ava        | Rose        | Miller    |              5 | 2009-03-05    |     5 | A       | Female |         2002 |      1005 | 345 Cedar St | Apt 505  | City5 | State5 | 5678901234 | 56789    |
+------------+-------------+-----------+----------------+---------------+-------+---------+--------+-------------+-----------+-------------+----------+-------+--------+------------+----------+
5 rows in set (0.00 sec)
```

(a) Display all the list of students who are in class - 6, section - A.

mysql> SELECT * FROM STUDENT
-> WHERE CLASS = 6 AND SECTION = 'A';

```
+------------+-------------+-----------+----------------+---------------+-------+---------+--------+-------------+-----------+-------------+----------+-------+--------+------------+----------+
| FIRST_NAME | MIDDLE_NAME | LAST_NAME | STUDENT_ENRLNO | DATE_OF_BIRTH | CLASS | SECTION | GENDER | YEAR_OF_JOIN | ADMISSION_NO | ADDRESS1    | ADDRESS2 | CITY  | STATE  | RESPHONE   | PIN_CODE |
+------------+-------------+-----------+----------------+---------------+-------+---------+--------+-------------+-----------+-------------+----------+-------+--------+------------+----------+
```

```
| Alice      | Marie       | Smith     |             1 | 2008-05-15    |     6 | A       | Female |         2000 |
1001 | 123 Main St | Apt 101  | City1 | State1 | 1234567890 | 12345    |
| Anna       | Grace       | Williams  |             3 | 2009-01-10    |     6 | A       | Female |         2000 |
1003 | 789 Oak St  | Apt 303  | City3 | State3 | 3456789012 | 34567    |
+------------+-------------+-----------+---------------+---------------+-------+---------+--------+--------------+-----
---------+-------------+----------+-------+--------+------------+----------+
2 rows in set (0.01 sec)
```

(b) To display all the students list whose first name starts with 'A'.

```
mysql> SELECT * FROM STUDENT
    -> WHERE FIRST_NAME LIKE 'A%';
+------------+-------------+-----------+---------------+---------------+-------+---------+--------+--------------+-----
---------+-------------+----------+-------+--------+------------+----------+
| FIRST_NAME | MIDDLE_NAME | LAST_NAME | STUDENT_ENRLNO | DATE_OF_BIRTH |
CLASS | SECTION | GENDER | YEAR_OF_JOIN | ADMISSION_NO | ADDRESS1     |
ADDRESS2 | CITY  | STATE  | RESPHONE   | PIN_CODE |
+------------+-------------+-----------+---------------+---------------+-------+---------+--------+--------------+-----
---------+-------------+----------+-------+--------+------------+----------+
| Alice      | Marie       | Smith     |             1 | 2008-05-15    |     6 | A       | Female |         2000 |
1001 | 123 Main St  | Apt 101  | City1 | State1 | 1234567890 | 12345    |
| Anna       | Grace       | Williams  |             3 | 2009-01-10    |     6 | A       | Female |         2000 |
1003 | 789 Oak St   | Apt 303  | City3 | State3 | 3456789012 | 34567    |
| Alex       | James       | Brown     |             4 | 2008-11-28    |     6 | B       | Male   |         2000 |
1004 | 012 Pine St  | Apt 404  | City4 | State4 | 4567890123 | 45678    |
| Ava        | Rose        | Miller    |             5 | 2009-03-05    |     5 | A       | Female |         2002 |
1005 | 345 Cedar St | Apt 505  | City5 | State5 | 5678901234 | 56789    |
+------------+-------------+-----------+---------------+---------------+-------+---------+--------+--------------+-----
---------+-------------+----------+-------+--------+------------+----------+
4 rows in set (0.00 sec)
```

(c) To display all the students list who are girls.

```
mysql> SELECT * FROM STUDENT
    -> WHERE GENDER ='FEMALE'
    -> ;
+------------+-------------+-----------+---------------+---------------+-------+---------+--------+--------------+-----
---------+-------------+----------+-------+--------+------------+----------+
| FIRST_NAME | MIDDLE_NAME | LAST_NAME | STUDENT_ENRLNO | DATE_OF_BIRTH |
CLASS | SECTION | GENDER | YEAR_OF_JOIN | ADMISSION_NO | ADDRESS1     |
ADDRESS2 | CITY  | STATE  | RESPHONE   | PIN_CODE |
+------------+-------------+-----------+---------------+---------------+-------+---------+--------+--------------+-----
---------+-------------+----------+-------+--------+------------+----------+
```

Alice	Marie	Smith	1	2008-05-15	6	A	Female	2000	1001	123 Main St	Apt 101	City1	State1	1234567890	12345
Anna	Grace	Williams	3	2009-01-10	6	A	Female	2000	1003	789 Oak St	Apt 303	City3	State3	3456789012	34567
Ava	Rose	Miller	5	2009-03-05	5	A	Female	2002	1005	345 Cedar St	Apt 505	City5	State5	5678901234	56789

3 rows in set (0.00 sec)

(d) To display all the students whose YEAR-OF-JOIN is 2000.

```
mysql> SELECT * FROM STUDENT
   -> WHERE YEAR_OF_JOIN='2000';
```

FIRST_NAME	MIDDLE_NAME	LAST_NAME	STUDENT_ENRLNO	DATE_OF_BIRTH	CLASS	SECTION	GENDER	YEAR_OF_JOIN	ADMISSION_NO	ADDRESS1	ADDRESS2	CITY	STATE	RESPHONE	PIN_CODE
Alice	Marie	Smith	1	2008-05-15	6	A	Female	2000	1001	123 Main St	Apt 101	City1	State1	1234567890	12345
Anna	Grace	Williams	3	2009-01-10	6	A	Female	2000	1003	789 Oak St	Apt 303	City3	State3	3456789012	34567
Alex	James	Brown	4	2008-11-28	6	B	Male	2000	1004	012 Pine St	Apt 404	City4	State4	4567890123	45678

3 rows in set (0.00 sec)

(e) Sort the records of students with respect to their ADMISSION_NO, in ascending order.

```
mysql> SELECT * FROM STUDENT
   -> ORDER BY ADMISSION_NO ASC;
```

FIRST_NAME	MIDDLE_NAME	LAST_NAME	STUDENT_ENRLNO	DATE_OF_BIRTH	CLASS	SECTION	GENDER	YEAR_OF_JOIN	ADMISSION_NO	ADDRESS1	ADDRESS2	CITY	STATE	RESPHONE	PIN_CODE
Alice	Marie	Smith	1	2008-05-15	6	A	Female	2000	1001	123 Main St	Apt 101	City1	State1	1234567890	12345

Bob	Alexander	Johnson	2	2007-08-20	7	B	Male	2001	1002	456 Elm St	Apt 202	City2	State2	2345678901	23456
Anna	Grace	Williams	3	2009-01-10	6	A	Female	2000	1003	789 Oak St	Apt 303	City3	State3	3456789012	34567
Alex	James	Brown	4	2008-11-28	6	B	Male	2000	1004	012 Pine St	Apt 404	City4	State4	4567890123	45678
Ava	Rose	Miller	5	2009-03-05	5	A	Female	2002	1005	345 Cedar St	Apt 505	City5	State5	5678901234	56789
+------------+-------------+-----------+---------------+---------------+-------+---------+--------+--------------+-----------+--------------+----------+-------+--------+------------+----------+
5 rows in set (0.00 sec)

PRACTICAL NO : 28

Create the following table CATALOG with the following fields : (BOOK ID, BOOK TITLE, AUTHOR,
AUTHOR_ID, PUBLISHER_ID, CATEGORY_ID, YEAR, ISBN, PRICE)
(a) To display all the books of the CATEGORY_ID : COMPUTERS;.
(b) List all the books whose PRICE is greater than or equal to 1000/-.
(c) List all the books whose AUTHOR is 'Tata McGraw-Hill;.
(d) List all the books whose YEAR of publication is 2013.
(e) List all the BOOK_TITLEs whose AUTHOR_ID is 123;.

```
mysql> create table catalog(book_id int, book_title varchar(50), author varchar(50), author_id int
, publisher_id varchar(10),category_id varchar(10), year int,isbn varchar(10), price
decimal(10,2));
Query OK, 0 rows affected (0.02 sec)

mysql> describe table catalog;
+----+-------------+---------+------------+------+---------------+------+---------+------+------+----------+-------+
| id | select_type | table   | partitions | type | possible_keys | key  | key_len | ref  | rows | filtered | Extra |
+----+-------------+---------+------------+------+---------------+------+---------+------+------+----------+-------+
|  1 | SIMPLE      | catalog | NULL       | ALL  | NULL          | NULL | NULL    | NULL |    4 |   100.00 | NULL  |
+----+-------------+---------+------------+------+---------------+------+---------+------+------+----------+-------+
1 row in set, 1 warning (0.00 sec)


mysql> insert into catalog(book_id , book_title , author , author_id  , publisher_id ,category_id ,
year ,isbn , price)values
    -> (1,'dsa','ram',101,'ch1234','computer', 2012,'1234567890',123.45);
Query OK, 1 row affected (0.01 sec)

mysql> insert into catalog(book_id , book_title , author , author_id  , publisher_id ,category_id ,
year ,isbn , price)values
    -> (2,'oop','sham',102,'ch1235','computer', 2013,'5134677890',156.45);
Query OK, 1 row affected (0.01 sec)

mysql> insert into catalog(book_id , book_title , author , author_id  , publisher_id ,category_id ,
year ,isbn , price)values
    -> (3,'java','tata mc graw hill',103,'ch1236','civil', 2014,'9190703890',56.45);
Query OK, 1 row affected (0.00 sec)

mysql> insert into catalog(book_id , book_title , author , author_id  , publisher_id ,category_id ,
year ,isbn , price)values
```

```
   -> (4,'python','parth',104,'ch1237','mechanical', 2013,'5640703890',586.45);
Query OK, 1 row affected (0.01 sec)

mysql> select * from catalog;
+---------+------------+------------------+-----------+--------------+-------------+------+------------+--------+
| book_id | book_title | author           | author_id | publisher_id | category_id | year | isbn       | price  |
+---------+------------+------------------+-----------+--------------+-------------+------+------------+--------+
|       1 | dsa        | ram              |       101 | ch1234       | computer    | 2012 | 1234567890 | 123.45 |
|       2 | oop        | sham             |       102 | ch1235       | computer    | 2013 | 5134677890 | 156.45 |
|       3 | java       | tata mc graw hill |       103 | ch1236       | civil       | 2014 | 9190703890 |  56.45 |
|       4 | python     | parth            |       104 | ch1237       | mechanical  | 2013 | 5640703890 | 586.45 |
+---------+------------+------------------+-----------+--------------+-------------+------+------------+--------+
4 rows in set (0.00 sec)
```

(a) To display all the books of the CATEGORY_ID : COMPUTERS;.

```
mysql> select * from catalog
   -> where category_id ='computer';
+---------+------------+--------+-----------+--------------+-------------+------+------------+--------+
| book_id | book_title | author | author_id | publisher_id | category_id | year | isbn       | price  |
+---------+------------+--------+-----------+--------------+-------------+------+------------+--------+
|       1 | dsa        | ram    |       101 | ch1234       | computer    | 2012 | 1234567890 | 123.45 |
|       2 | oop        | sham   |       102 | ch1235       | computer    | 2013 | 5134677890 | 156.45 |
+---------+------------+--------+-----------+--------------+-------------+------+------------+--------+
2 rows in set (0.00 sec)
```

(b) List all the books whose PRICE is greater than or equal to 400/-.

```
mysql> select * from catalog
   -> where price >= 400;
+---------+------------+--------+-----------+--------------+-------------+------+------------+--------+
| book_id | book_title | author | author_id | publisher_id | category_id | year | isbn       | price  |
+---------+------------+--------+-----------+--------------+-------------+------+------------+--------+
|       4 | python     | parth  |       104 | ch1237       | mechanical  | 2013 | 5640703890 | 586.45 |
+---------+------------+--------+-----------+--------------+-------------+------+------------+--------+
1 row in set (0.00 sec)
```

(c) List all the books whose PUBLISHER_ID is ‘Tata Mc Graw Hill;.

mysql> select * from catalog
-> where author ='tata mc graw hill';

```
+---------+------------+------------------+-----------+-------------+-------------+------+------------+-------+
| book_id | book_title | author           | author_id | publisher_id | category_id | year | isbn       |
price |
+---------+------------+------------------+-----------+-------------+-------------+------+------------+-------+
|       3 | java       | tata mc graw hill |       103 | ch1236       | civil       | 2014 | 9190703890 | 56.45 |
+---------+------------+------------------+-----------+-------------+-------------+------+------------+-------+
1 row in set (0.00 sec)
```

(d) List all the books whose YEAR of publication is 2013.

mysql> select * from catalog
-> where year = 2013;

```
+---------+------------+--------+-----------+-------------+-------------+------+------------+--------+
| book_id | book_title | author | author_id | publisher_id | category_id | year | isbn       | price  |
+---------+------------+--------+-----------+-------------+-------------+------+------------+--------+
|       2 | oop        | sham   |       102 | ch1235       | computer    | 2013 | 5134677890 | 156.45 |
|       4 | python     | parth  |       104 | ch1237       | mechanical  | 2013 | 5640703890 | 586.45 |
+---------+------------+--------+-----------+-------------+-------------+------+------------+--------+
2 rows in set (0.00 sec)
```

(e) List all the BOOK_TITLEs whose AUTHOR_ID is 102;.

mysql> select * from catalog
-> where author_id =102;

```
+---------+------------+--------+-----------+-------------+-------------+------+------------+--------+
| book_id | book_title | author | author_id | publisher_id | category_id | year | isbn       | price  |
+---------+------------+--------+-----------+-------------+-------------+------+------------+--------+
|       2 | oop        | sham   |       102 | ch1235       | computer    | 2013 | 5134677890 | 156.45 |
+---------+------------+--------+-----------+-------------+-------------+------+------------+--------+
1 row in set (0.00 sec)
```