DBMS MODEL PRACTICAL

A)

```
mysql> CREATE TABLE book (
    ->     book_name VARCHAR(100),
    ->     author VARCHAR(100),
    ->     price DECIMAL(10, 2),
    ->     quantity INT
    -> );
Query OK, 0 rows affected (0.32 sec)
```

b)

```
mysql> CREATE TABLE customer (
    ->     Cust_id NUMBER PRIMARY KEY,
    ->     Cust_name VARCHAR(100),
    ->     Addr VARCHAR(200),
    ->     ph_no VARCHAR(15),
    ->     pan_no VARCHAR(10)
    -> );
```

```
mysql> INSERT INTO book (book_name, author, price, quantity) VALUES ('Book A', 'Author A', 150.00, 10);
Query OK, 1 row affected (0.04 sec)

mysql> INSERT INTO book (book_name, author, price, quantity) VALUES ('Book B', 'Author B', 200.00, 5);
Query OK, 1 row affected (0.04 sec)

mysql> INSERT INTO book (book_name, author, price, quantity) VALUES ('Book C', 'Author C', 225.00, 8);
Query OK, 1 row affected (0.09 sec)

mysql> INSERT INTO book (book_name, author, price, quantity) VALUES ('Book D', 'Author K', 175.00, 12);
Query OK, 1 row affected (0.03 sec)

mysql> INSERT INTO book (book_name, author, price, quantity) VALUES ('Book E', 'Author K', 300.00, 20);
Query OK, 1 row affected (0.03 sec)

mysql> select * from book;
+-----------+----------+--------+----------+
| book_name | author   | price  | quantity |
+-----------+----------+--------+----------+
| Book A    | Author A | 150.00 |       10 |
| Book B    | Author B | 200.00 |        5 |
| Book C    | Author C | 225.00 |        8 |
| Book D    | Author K | 175.00 |       12 |
| Book E    | Author K | 300.00 |       20 |
+-----------+----------+--------+----------+
5 rows in set (0.00 sec)
```

1)

```
mysql> SELECT author
    -> FROM book
    -> WHERE price = 200.00;
+----------+
| author   |
+----------+
| Author B |
+----------+
```

2)

```
mysql> SELECT price
    -> FROM book
    -> WHERE price BETWEEN 175.00 AND 250.00;
+--------+
| price  |
+--------+
| 200.00 |
| 225.00 |
| 175.00 |
+--------+
```

3)

```
mysql> SELECT *
    -> FROM book
    -> WHERE author LIKE 'K%';
Empty set (0.00 sec)
```

4)

```
DECLARE
    num1 INT := 10; -- You can change the values as needed
    num2 INT := 20;
    num3 INT := 15;
    largest INT;
BEGIN
    IF (num1 >= num2) AND (num1 >= num3) THEN
        largest := num1;
    ELSIF (num2 >= num1) AND (num2 >= num3) THEN
        largest := num2;
    ELSE
        largest := num3;
    END IF;
    DBMS_OUTPUT.PUT_LINE('The largest number is: ' || largest);
END;
/
```

Output:

```
The largest number is: 20
```

5)

```
CREATE OR REPLACE FUNCTION factorial(n INT) RETURN INT IS
    result INT := 1;
BEGIN
    FOR i IN 1..n LOOP
        result := result * i;
    END LOOP;
    RETURN result;
END;
/

DECLARE
    num INT := 5; -- You can change the value as needed
    fact INT;
BEGIN
    fact := factorial(num);
    DBMS_OUTPUT.PUT_LINE('The factorial of ' || num || ' is: ' || fact);
END;
/
```

Output:

```
The factorial of 5 is: 120
```