Name : Prashant Sharma

Class: 20BCS_KRG_MM_G1

UID : 20BCS9390

Question_1

CODE –

```sql
create table Orders(
order_id number,
customer_id number,
order_date date,
item_id varchar2(255),
quantity   number,
constraint pk primary key(order_id,item_id)
-- primary key (emp_id) -- Primary Key = Must be every emp_id unique and not
null
);


insert into Orders values (1,1,'01-JAN-2023','1',10);
insert into Orders values (2,1,'08-JAN-2023','2',10);
insert into Orders values (3,2,'02-JAN-2023','1',5);
insert into Orders values (4,3,'03-JAN-2023','3',5);
insert into Orders values (5,4,'04-JAN-2023','4',1);
insert into Orders values (6,4,'05-JAN-2023','5',5);
insert into Orders values (7,5,'05-JAN-2023','1',10);
insert into Orders values (8,5,'14-JAN-2023','4',5);
insert into Orders values (9,5,'21-JAN-2023','3',5);
select * from Orders;

create table Items(
item_id varchar2(255) primary key,
item_name varchar2(255),
item_category varchar2(255)
-- primary key (emp_id) -- Primary Key = Must be every emp_id unique and not
null
);


insert into Items values ('1','LC Alg. Book','Book');
insert into Items values ('2','LC DB. Book','Book');
insert into Items values ('3','LC Smartphone','Phone');
insert into Items values ('4','LC Phone 2020','Phone');
insert into Items values ('5','LC SmartGlass','Glasses');
insert into Items values ('6','LC T-Shirt XL','T-Shirt');
```

```
SELECT a.item_category AS CATEGORY,
        sum(CASE
                WHEN to_char(b.order_date, 'd') = '2' THEN b.quantity
                ELSE 0
            END) AS MONDAY,
        sum(CASE
                WHEN to_char(b.order_date, 'd') = '3' THEN b.quantity
                ELSE 0
            END) AS TUESDAY,
        sum(CASE
                WHEN to_char(b.order_date, 'd') = '4' THEN b.quantity
                ELSE 0
            END) AS WEDNESDAY,
        sum(CASE
                WHEN to_char(b.order_date, 'd') = '5'  THEN b.quantity
                ELSE 0
            END) AS THURSDAY,
        sum(CASE
                WHEN to_char(b.order_date, 'd') = '6'  THEN b.quantity
                ELSE 0
            END) AS FRIDAY,
        sum(CASE
                WHEN to_char(b.order_date, 'd') = '7'  THEN b.quantity
                ELSE 0
            END) AS SATURDAY,
        sum(CASE
                WHEN to_char(b.order_date, 'd') = '1'  THEN b.quantity
                ELSE 0
            END) AS SUNDAY
FROM items a
LEFT JOIN orders b ON a.item_id = b.item_id
GROUP BY a.item_category
ORDER BY a.item_category;
```

Output-

ORDER_ID	CUSTOMER_ID	ORDER_DATE	ITEM_ID	QUANTITY
1	1	01-JAN-23	1	10
2	1	08-JAN-23	2	10
3	2	02-JAN-23	1	5
4	3	03-JAN-23	3	5
5	4	04-JAN-23	4	1
6	4	05-JAN-23	5	5
7	5	05-JAN-23	1	10
8	5	14-JAN-23	4	5
9	5	21-JAN-23	3	5

Download CSV

CATEGORY	MONDAY	TUESDAY	WEDNESDAY	THURSDAY	FRIDAY	SATURDAY	SUNDAY
Book	5	0	0	10	0	0	20
Glasses	0	0	0	5	0	0	0
Phone	0	5	1	0	0	10	0
T-Shirt	0	0	0	0	0	0	0

Download CSV

4 rows selected.

Question_2

CODE

```
create table student(
name varchar2(255) not null,
continent varchar2(255)
);

insert into student values ('Prashant','America');
insert into student values ('Paras','Europe');
insert into student values ('Shubhankar','Asia');
insert into student values ('Kunal','America');
```

```
SELECT
  a.name AS America
  ,b.name AS Asia
  ,c.name AS Europe
FROM
(SELECT ROW_NUMBER() OVER (ORDER BY name) AS ID, name FROM student WHERE
continent = 'America') a
FULL JOIN
(SELECT ROW_NUMBER() OVER (ORDER BY name) AS ID, name FROM student WHERE
continent = 'Asia') b
ON a.ID = b.ID
FULL JOIN
(SELECT ROW_NUMBER() OVER (ORDER BY name) AS ID, name FROM student WHERE
continent = 'Europe') c
ON c.ID = b.ID
OR c.ID = a.ID;
```

Output-

AMERICA	ASIA	EUROPE
Kunal	Shubhankar	Paras
Prashant	-	-

Download CSV

2 rows selected.

Question_3

```
CODE – CREATE TABLE Student(student_id number(10), student_name varchar(20));


INSERT INTO Student VALUES(1,'Shivam');
INSERT INTO Student VALUES(2,'Prashant');
INSERT INTO Student VALUES(3,'Shubhankar');
INSERT INTO Student VALUES(4,'Kunal');
INSERT INTO Student VALUES(5,'Rohit');

CREATE TABLE Exam(exam_id number(10), student_id number(10), score
number(10));

INSERT INTO Exam VALUES(10,1,70);
INSERT INTO Exam VALUES(10,2,80);
```

```sql
INSERT INTO Exam VALUES(10,3,90);
INSERT INTO Exam VALUES(20,1,80);
INSERT INTO Exam VALUES(30,1,70);
INSERT INTO Exam VALUES(30,3,80);
INSERT INTO Exam VALUES(30,4,90);
INSERT INTO Exam VALUES(40,1,60);
INSERT INTO Exam VALUES(40,2,70);
INSERT INTO Exam VALUES(40,4,80);

select distinct Student.*
from Student inner join Exam
on Student.student_id = Exam.student_id
where student.student_id not in
    (select e1.student_id
    from Exam as e1 inner join
        (select exam_id, min(score) as min_score, max(score) as max_score
        from Exam
        group by exam_id) as e2
    on e1.exam_id = e2.exam_id
    where e1.score = e2.min_score or e1.score = e2.max_score)
order by student_id
```

Output-

```
2|Prashant


[Execution complete with exit code 0]
```