Gaurav Katiyar

PRACTICAL ASSIGNMENT

Master of Operational Research (II-Sem)-2020

Course-DBMS

SQL PRACTICAL 1.

Creating the Customers Table –

```
CREATE TABLE Customers (
   Cust_id INT,
   Cust_Name VARCHAR(255),
   City VARCHAR(255),
   Country VARCHAR(255)
);
```

Inserting the data-

```
INSERT INTO Customers (Cust_id, Cust_Name, City , Country)
VALUES (1, 'Sanji Vinsmoke', 'Marseille', 'France'),
        (2, 'Scarlett Johansson', 'NewYork', 'USA'),
        (3, 'Kokashi', 'Kanagawa', 'Japan'),
        (4, 'Jose Mourinho', 'Setabal', 'Portugal'),
       (5, 'Sosuke Uchiha', 'China', 'Japan'),
        (6, 'Kimi Raikkonen', 'Espoo', 'Finland'),
        (7, 'Gintoki Sokata', 'Kyoto', 'Japan'),
       (8, 'Sir Alex Ferguson', 'Govan', 'UK'),
       (9, 'Arsene Wenger', 'Starbourg', 'France'),
```

(10, 'Michal Jordan', 'Chicago', 'USA')

;

Cust_id	Cust_Name	City	Country
1	Sanji Vinsmoke	Marseille	France
2	Scarlett Johansson	NewYork	USA
3	Kokashi	Kanagawa	Japan
4	Jose Mourinho	Setabal	Portugal
5	Sosuke Uchiha	China	Japan
6	Kimi Raikkonen	Espoo	Finland
7	Gintoki Sokata	Kyoto	Japan
8	Sir Alex Ferguson	Govan	UK
9	Arsene Wenger	Starbourg	France
10	Michal Jordan	Chicago	USA

Queries:

1. Display the data in ascending order of their customer id.

```
SELECT
   *
FROM
   Customers
ORDER BY Cust_id ASC;
```

Cust_id	Cust_Name	City	Country
1	Sanji Vinsmoke	Marseille	France
2	Scarlett Johansson	NewYork	India
3	Kokashi	Kanagawa	Japan
4	Jose Mourinho	Setabal	Portugal
5	Sosuke Uchiha	China	Japan
6	Kimi Raikkonen	Espoo	Finland
7	Gintoki Sokata	Kyoto	Japan
8	Sir Alex Ferguson	Govan	UK
9	Arsene Wenger	Starbourg	France
10	Michal Jordan	Chicago	India

2. Display the data in ascending order (order by City).

```
SELECT
   *
FROM
   Customers
ORDER BY City ASC;
```

Cust_id	Cust_Name	City	Country
10	Michal Jordan	Chicago	India
5	Sosuke Uchiha	China	Japan
6	Kimi Raikkonen	Espoo	Finland
8	Sir Alex Ferguson	Govan	UK
3	Kokashi	Kanagawa	Japan
7	Gintoki Sokata	Kyoto	Japan
1	Sanji Vinsmoke	Marseille	France
2	Scarlett Johansson	NewYork	India
4	Jose Mourinho	Setabal	Portugal
9	Arsene Wenger	Starbourg	France

3. Count the number of Cities.

```
SELECT

      COUNT(City)

FROM

     Customers

;
```

	COUNT(City)
▶	10

4. Display the name and frequency of every city.

```
SELECT
   City, COUNT(City)
FROM
   Customers
group by City;
```

	City	COUNT(City)
▶	Marseille	1
	NewYork	1
	Kanagawa	1
	Setabal	1
	China	1
	Espoo	1
	Kyoto	1
	Govan	1
	Starbourg	1
	Chicago	1

5. Display the record where county is ‘Japan’.

```
SELECT
   *
FROM
   Customers
WHERE
   Country = 'Japan';
```

	Cust_id	Cust_Name	City	Country
▶	3	Kokashi	Kanagawa	Japan
	5	Sosuke Uchiha	China	Japan
	7	Gintoki Sokata	Kyoto	Japan

6. Update the table by changing the value of Country as ‘India’ where it is ‘USA’.

```
UPDATE Customers
SET
   Country = 'India'
WHERE
   Country = 'USA';
```

	Cust_id	Cust_Name	City	Country
▶	1	Sanji Vinsmoke	Marseille	France
	2	Scarlett Johansson	NewYork	India
	3	Kokashi	Kanagawa	Japan
	4	Jose Mourinho	Setabal	Portugal
	5	Sosuke Uchiha	China	Japan
	6	Kimi Raikkonen	Espoo	Finland
	7	Gintoki Sokata	Kyoto	Japan
	8	Sir Alex Ferguson	Govan	UK
	9	Arsene Wenger	Starbourg	France
	10	Michal Jordan	Chicago	India

SQL Practical 2:

Create the database of 20 students with the following attributes: student id, name, address, class and marks in economics.

Creating the table of Students -

```
CREATE TABLE Students (
    Student_id INT,
    Student_Name VARCHAR(50),
    Address VARCHAR(250),
    Class VARCHAR(10),
    Marks_eco INT
);
```

Inserting the data-

```
INSERT INTO Students (Student_id ,Student_Name ,Address ,Class ,Marks_eco)
VALUES (1, 'Karishma', 'B/1 Lajpat Nagar, Delhi', '12th',75),
    (2, 'Selena', 'B/2 Vijay Nagar, Delhi', '12th',80),
    (3, 'Krishteena', 'B/3 Lajpat Nagar, Delhi', '12th',85),
    (4, 'Salmaan', 'B/4 Kidwai Nagar, Delhi', '12th',86),
    (5, 'Simran', 'B/5 Lajpat Nagar, Delhi', '12th',84),
    (6, 'Akshay', 'B/6 Govind Puri, Delhi', '12th',75),
    (7, 'Shahrukh', 'B/7 Palam, Delhi', '12th',78),
    (8, 'Chrish', 'B/8 Palam, Delhi', '12th',81),
    (9, 'Krish', 'B/9 Satya Niketan, Delhi', '12th',82),
    (10, 'Priyanka', 'B/10 Moti bagh, Delhi', '12th',72),
    (11, 'Taapsee', 'B/11 Palam, Delhi', '12th',86),
```

(12, 'Kanika', 'B/12 Palam, Delhi', '12th',83),

(13, 'Kangna', 'B/13 Nangal Raya, Delhi', '12th',80),

(14, 'Abhishek', 'B/14 Palam, Delhi', '12th',71),

(15, 'Ironman', 'B/15 R.K Puram, Delhi', '12th',84),

(16, 'Batman', 'B/16 Palam, Delhi', '12th',75),

(17, 'Superman', 'B/17 Palam, Delhi', '12th',70),

(18, 'Dweyn', 'B/18 Delhi Cant, Delhi', '12th',80),

(19, 'Ryan', 'B/19 Kashmiri Gate, Delhi', '12th',81),

(20, 'Rock', 'B/20 Satya Niketan, Delhi', '12th',79)

;

Student_id	Student_Name	Address	Class	Marks_eco
1	Karishma	B/1 Lajpat Nagar, Delhi	12th	75
2	Selena	B/2 Vijay Nagar, Delhi	12th	80
3	Krishteena	B/3 Lajpat Nagar, Delhi	12th	85
4	Salmaan	B/4 Kidwai Nagar, Delhi	12th	86
5	Simran	B/5 Lajpat Nagar, Delhi	12th	84
6	Akshay	B/6 Govind Puri, Delhi	12th	75
7	Shahrukh	B/7 Palam, Delhi	12th	78
8	Chrish	B/8 Palam, Delhi	12th	81
9	Krish	B/9 Satya Niketan, Delhi	12th	82
10	Priyanka	B/10 Moti bagh, Delhi	12th	72
11	Taapsee	B/11 Palam, Delhi	12th	86
12	Kanika	B/12 Palam, Delhi	12th	83
13	Kangna	B/13 Nangal Raya, Delhi	12th	80
14	Abhishek	B/14 Palam, Delhi	12th	71
15	Ironman	B/15 R.K Puram, Delhi	12th	84
16	Batman	B/16 Palam, Delhi	12th	75
17	Superman	B/17 Palam, Delhi	12th	70
18	Dweyn	B/18 Delhi Cant, Delhi	12th	80
19	Ryan	B/19 Kashmiri Gate, Delhi	12th	81
20	Rock	B/20 Satya Niketan, Delhi	12th	79

Queries:

1. Display the table. Arrange the data in ascending order of total marks

```
SELECT
   *
FROM
   students
ORDER BY Marks_eco ASC;
```

Student_id	Student_Name	Address	Class	Marks_eco
17	Superman	B/17 Palam, Delhi	12th	70
14	Abhishek	B/14 Palam, Delhi	12th	71
10	Priyanka	B/10 Moti bagh, Delhi	12th	72
1	Karishma	B/1 Lajpat Nagar, Delhi	12th	75
6	Akshay	B/6 Govind Puri, Delhi	12th	75
16	Batman	B/16 Palam, Delhi	12th	75
7	Shahrukh	B/7 Palam, Delhi	12th	78
20	Rock	B/20 Satya Niketan, Delhi	12th	79
2	Selena	B/2 Vijay Nagar, Delhi	12th	80
13	Kangna	B/13 Nangal Raya, Delhi	12th	80
18	Dweyn	B/18 Delhi Cant, Delhi	12th	80
8	Chrish	B/8 Palam, Delhi	12th	81
19	Ryan	B/19 Kashmiri Gate, Delhi	12th	81
9	Krish	B/9 Satya Niketan, Delhi	12th	82
12	Kanika	B/12 Palam, Delhi	12th	83
5	Simran	B/5 Lajpat Nagar, Delhi	12th	84
15	Ironman	B/15 R.K Puram, Delhi	12th	84
3	Krishteena	B/3 Lajpat Nagar, Delhi	12th	85
4	Salmaan	B/4 Kidwai Nagar, Delhi	12th	86
11	Taapsee	B/11 Palam, Delhi	12th	86

2. Display the description of the table.

```
Describe students;
```

	Field	Type	Null	Key	Default	Extra
▶	Student_id	int	YES		NULL	
	Student_Name	varchar(50)	YES		NULL	
	Address	varchar(250)	YES		NULL	
	Class	varchar(10)	YES		NULL	
	Marks_eco	int	YES		NULL	

3. Display the detail of the student having max and min marks in economics.

For MIN marks in economics-

```
SELECT
   *
FROM
   students
WHERE
   Marks_eco = (SELECT
         MIN(Marks_eco)
      FROM
         students);
```

	Student_id	Student_Name	Address	Class	Marks_eco
▶	17	Superman	B/17 Palam, Delhi	12th	70

For MAX marks in economics-

```
SELECT
   *
FROM
```

```
   students
WHERE
   Marks_eco = (SELECT
         MAX(Marks_eco)
      FROM
         students);
```

	Student_id	Student_Name	Address	Class	Marks_eco
▶	4	Salmaan	B/4 Kidwai Nagar, Delhi	12th	86
	11	Taapsee	B/11 Palam, Delhi	12th	86

4. Update the student address as ‘Rajasthan’ where Roll_no . =14

```
UPDATE students
SET
   Address = 'Rajasthan'
WHERE
   Student_id = 14;
```

Student_id	Student_Name	Address	Class	Marks_eco
1	Karishma	B/1 Lajpat Nagar, Delhi	12th	75
2	Selena	B/2 Vijay Nagar, Delhi	12th	80
3	Krishteena	B/3 Lajpat Nagar, Delhi	12th	85
4	Salmaan	B/4 Kidwai Nagar, Delhi	12th	86
5	Simran	B/5 Lajpat Nagar, Delhi	12th	84
6	Akshay	B/6 Govind Puri, Delhi	12th	75
7	Shahrukh	B/7 Palam, Delhi	12th	78
8	Chrish	B/8 Palam, Delhi	12th	81
9	Krish	B/9 Satya Niketan, Delhi	12th	82
10	Priyanka	B/10 Moti bagh, Delhi	12th	72
11	Taapsee	B/11 Palam, Delhi	12th	86
12	Kanika	B/12 Palam, Delhi	12th	83
13	Kangna	B/13 Nangal Raya, Delhi	12th	80
14	Abhishek	Rajasthan	12th	71
15	Ironman	B/15 R.K Puram, Delhi	12th	84
16	Batman	B/16 Palam, Delhi	12th	75
17	Superman	B/17 Palam, Delhi	12th	70
18	Dweyn	B/18 Delhi Cant, Delhi	12th	80
19	Ryan	B/19 Kashmiri Gate, Delhi	12th	81
20	Rock	B/20 Satya Niketan, Delhi	12th	79

5. Update the class of the student to 10 where Roll_no . =14.

```
UPDATE students
SET
   class = '10th'
WHERE
   Student_id = 14;
```

Student_id	Student_Name	Address	Class	Marks_eco
1	Karishma	B/1 Lajpat Nagar, Delhi	12th	75
2	Selena	B/2 Vijay Nagar, Delhi	12th	80
3	Krishteena	B/3 Lajpat Nagar, Delhi	12th	85
4	Salmaan	B/4 Kidwai Nagar, Delhi	12th	86
5	Simran	B/5 Lajpat Nagar, Delhi	12th	84
6	Akshay	B/6 Govind Puri, Delhi	12th	75
7	Shahrukh	B/7 Palam, Delhi	12th	78
8	Chrish	B/8 Palam, Delhi	12th	81
9	Krish	B/9 Satya Niketan, Delhi	12th	82
10	Priyanka	B/10 Moti bagh, Delhi	12th	72
11	Taapsee	B/11 Palam, Delhi	12th	86
12	Kanika	B/12 Palam, Delhi	12th	83
13	Kangna	B/13 Nangal Raya, Delhi	12th	80
14	Abhishek	Rajasthan	10th	71
15	Ironman	B/15 R.K Puram, Delhi	12th	84
16	Batman	B/16 Palam, Delhi	12th	75
17	Superman	B/17 Palam, Delhi	12th	70
18	Dweyn	B/18 Delhi Cant, Delhi	12th	80
19	Ryan	B/19 Kashmiri Gate, Delhi	12th	81
20	Rock	B/20 Satya Niketan, Delhi	12th	79

6. Alter the table by adding two more attributes say Marks_maths and Marks_his.

```
ALTER TABLE students
ADD (Marks_maths INT,
      Marks_his INT);
```

Inserting the values of Marks_maths and Marks_his into the table of students-

```
   UPDATE Students
SET
  Marks_maths = CASE
     WHEN student_id = 1 THEN 81
     WHEN student_id = 2 THEN 82
     WHEN student_id = 3 THEN 84
     WHEN student_id = 4 THEN 75
     WHEN student_id = 5 THEN 79
```

```
    WHEN student_id = 6 THEN 78
    WHEN student_id = 7 THEN 81
    WHEN student_id = 8 THEN 82
    WHEN student_id = 9 THEN 84
    WHEN student_id = 10 THEN 75
    WHEN student_id = 11 THEN 79
    WHEN student_id = 12 THEN 78
    WHEN student_id = 13 THEN 81
    WHEN student_id = 14 THEN 82
    WHEN student_id = 15 THEN 84
    WHEN student_id = 16 THEN 75
    WHEN student_id = 17 THEN 79
    WHEN student_id = 18 THEN 78
    WHEN student_id = 19 THEN 70
    WHEN student_id = 20 THEN 69
  END;

UPDATE Students
SET
  Marks_his = CASE
    WHEN student_id = 1 THEN 78
    WHEN student_id = 2 THEN 71
    WHEN student_id = 3 THEN 82
    WHEN student_id = 4 THEN 75
    WHEN student_id = 5 THEN 79
    WHEN student_id = 6 THEN 78
    WHEN student_id = 7 THEN 81
    WHEN student_id = 8 THEN 67
    WHEN student_id = 9 THEN 68
    WHEN student_id = 10 THEN 75
    WHEN student_id = 11 THEN 79
    WHEN student_id = 12 THEN 78
    WHEN student_id = 13 THEN 81
    WHEN student_id = 14 THEN 82
    WHEN student_id = 15 THEN 61
    WHEN student_id = 16 THEN 75
    WHEN student_id = 17 THEN 79
    WHEN student_id = 18 THEN 78
    WHEN student_id = 19 THEN 70
    WHEN student_id = 20 THEN 69
  END;
```

Student_id	Student_Name	Address	Class	Marks_eco
1	Karishma	B/1 Lajpat Nagar, Delhi	12th	75
2	Selena	B/2 Vijay Nagar, Delhi	12th	80
3	Krishteena	B/3 Lajpat Nagar, Delhi	12th	85
4	Salmaan	B/4 Kidwai Nagar, Delhi	12th	86
5	Simran	B/5 Lajpat Nagar, Delhi	12th	84
6	Akshay	B/6 Govind Puri, Delhi	12th	75
7	Shahrukh	B/7 Palam, Delhi	12th	78
8	Chrish	B/8 Palam, Delhi	12th	81
9	Krish	B/9 Satya Niketan, Delhi	12th	82
10	Priyanka	B/10 Moti bagh, Delhi	12th	72
11	Taapsee	B/11 Palam, Delhi	12th	86
12	Kanika	B/12 Palam, Delhi	12th	83
13	Kangna	B/13 Nangal Raya, Delhi	12th	80
14	Abhishek	Rajasthan	10th	71
15	Ironman	B/15 R.K Puram, Delhi	12th	84
16	Batman	B/16 Palam, Delhi	12th	75
17	Superman	B/17 Palam, Delhi	12th	70
18	Dweyn	B/18 Delhi Cant, Delhi	12th	80
19	Ryan	B/19 Kashmiri Gate, Delhi	12th	81
20	Rock	B/20 Satya Niketan, Delhi	12th	79

7. Display the name and Marks_eco, Marks_his and Marks_maths.

```
SELECT
   student_name, Marks_eco, Marks_his, Marks_maths
FROM
   students;
```

	student_name	Marks_eco	Marks_his	Marks_maths
▶	Karishma	75	78	81
	Selena	80	71	82
	Krishteena	85	82	84
	Salmaan	86	75	75
	Simran	84	79	79
	Akshay	75	78	78
	Shahrukh	78	81	81
	Chrish	81	67	82
	Krish	82	68	84
	Priyanka	72	75	75
	Taapsee	86	79	79
	Kanika	83	78	78
	Kangna	80	81	81
	Abhishek	71	82	82
	Ironman	84	61	84
	Batman	75	75	75
	Superman	70	79	79
	Dweyn	80	78	78
	Ryan	81	70	70
	Rock	79	69	69

8. **Create another attribute containing the total of Marks_eco, Marks_his, Marks_maths and name it as 'Total marks'**.

```
Alter table students

Add Total_marks INT ;
```

Inserting the value of Total marks into students table-

```
UPDATE Students

SET

    Total_marks = CASE
```

```
        WHEN student_id = 1 THEN Marks_eco + Marks_maths + marks_his
        WHEN student_id = 2 THEN Marks_eco + Marks_maths + marks_his
        WHEN student_id = 3 THEN Marks_eco + Marks_maths + marks_his
        WHEN student_id = 4 THEN Marks_eco + Marks_maths + marks_his
        WHEN student_id = 5 THEN Marks_eco + Marks_maths + marks_his
        WHEN student_id = 6 THEN Marks_eco + Marks_maths + marks_his
        WHEN student_id = 7 THEN Marks_eco + Marks_maths + marks_his
        WHEN student_id = 8 THEN Marks_eco + Marks_maths + marks_his
        WHEN student_id = 9 THEN Marks_eco + Marks_maths + marks_his
        WHEN student_id = 10 THEN Marks_eco + Marks_maths + marks_his
        WHEN student_id = 11 THEN Marks_eco + Marks_maths + marks_his
        WHEN student_id = 12 THEN Marks_eco + Marks_maths + marks_his
        WHEN student_id = 13 THEN Marks_eco + Marks_maths + marks_his
        WHEN student_id = 14 THEN Marks_eco + Marks_maths + marks_his
        WHEN student_id = 15 THEN Marks_eco + Marks_maths + marks_his
        WHEN student_id = 16 THEN Marks_eco + Marks_maths + marks_his
        WHEN student_id = 17 THEN Marks_eco + Marks_maths + marks_his
        WHEN student_id = 18 THEN Marks_eco + Marks_maths + marks_his
        WHEN student_id = 19 THEN Marks_eco + Marks_maths + marks_his
        WHEN student_id = 20 THEN Marks_eco + Marks_maths + marks_his
END;
```

Student_id	Student_Name	Address	Class	Marks_eco	Marks_maths	Marks_his	Total_marks
1	Karishma	B/1 Lajpat Nagar, Delhi	12th	75	81	78	234
2	Selena	B/2 Vijay Nagar, Delhi	12th	80	82	71	233
3	Krishteena	B/3 Lajpat Nagar, Delhi	12th	85	84	82	251
4	Salmaan	B/4 Kidwai Nagar, Delhi	12th	86	75	75	236
5	Simran	B/5 Lajpat Nagar, Delhi	12th	84	79	79	242
6	Akshay	B/6 Govind Puri, Delhi	12th	75	78	78	231
7	Shahrukh	B/7 Palam, Delhi	12th	78	81	81	240
8	Chrish	B/8 Palam, Delhi	12th	81	82	67	230
9	Krish	B/9 Satya Niketan, Delhi	12th	82	84	68	234
10	Priyanka	B/10 Moti bagh, Delhi	12th	72	75	75	222
11	Taapsee	B/11 Palam, Delhi	12th	86	79	79	244
12	Kanika	B/12 Palam, Delhi	12th	83	78	78	239
13	Kangna	B/13 Nangal Raya, Delhi	12th	80	81	81	242
14	Abhishek	Rajasthan	10th	71	82	82	235
15	Ironman	B/15 R.K Puram, Delhi	12th	84	84	61	229
16	Batman	B/16 Palam, Delhi	12th	75	75	75	225
17	Superman	B/17 Palam, Delhi	12th	70	79	79	228
18	Dweyn	B/18 Delhi Cant, Delhi	12th	80	78	78	236
19	Ryan	B/19 Kashmiri Gate, Delhi	12th	81	70	70	221
20	Rock	B/20 Satya Niketan, Delhi	12th	79	69	69	217

9. Display the record of the students whose total marks is between 250 and 270.

```
SELECT
   *
FROM
   students
WHERE
   Total_marks BETWEEN 250 AND 270;
```

Student_id	Student_Name	Address	Class	Marks_eco	Marks_maths	Marks_his	Total_marks
3	Krishteena	B/3 Lajpat Nagar, Delhi	12th	85	84	82	251

SQL PRACTICAL-3

Creating the Table of Parts-

```
CREATE TABLE Parts (
    Names1 VARCHAR(30),
    Addresses VARCHAR(255),
    City VARCHAR(30)
);
```

Inserting the data into Parts table-

```
INSERT INTO Parts (Names1, Addresses, City)
VALUES ('Truewheel', '550 Husker', 'New'),
       ('Bike Spect', 'CPT Shrive', 'Los Angels'),
       ('Le Shoppe', 'Hometown', 'KS'),
       ('AAA Bike', '10 Oldtown', 'SL'),
       ('Jacks Bike', '24 Eglin', 'FL')
;
```

	Names1	Addresses	City
▶	Truewheel	550 Husker	New
	Bike Spect	CPT Shrive	Los Angels
	Le Shoppe	Hometown	KS
	AAA Bike	10 Oldtown	SL
	Jacks Bike	24 Eglin	FL

Creating the table of Order-

```
CREATE TABLE Order1(
    Ordered_on_date DATE ,
```

```
    Names1 VARCHAR(30),
    PartNum INT,
     Quantity INT)
;
```

Inserting the data into Order table-

```
INSERT INTO Order1 (Ordered_on_date, Names1, PartNum, Quantity)
VALUES ('1996-05-15', 'Truewheel', 23, 6),
       ('1996-05-19', 'Truewheeel', 76,3),
       ('1996-09-02', 'Truewheel', 10, 2),
       ('1996-03-30', 'Truewheel', 42, 8),
       ('1996-06-30', 'Bike Spect', 54, 10),
       ('1996-05-30', 'Bike Spect', 10, 2),
       ('1996-05-20', 'Bike Spect', 23, 8),
       ('1996-01-17', 'Le Shoppe', 76, 5),
       ('1996-06-01', 'Le Shoppe', 10, 3),
       ('1996-06-01', 'AAA Bike', 10, 1),
       ('1996-06-11', 'Jacks Bike', 76, 14)
;
```

	Ordered_on_date	Names1	PartNum	Quantity
▶	1996-05-15	Truewheel	23	6
	1996-05-19	Truewheeel	76	3
	1996-09-02	Truewheel	10	2
	1996-03-30	Truewheel	42	8
	1996-06-30	Bike Spect	54	10
	1996-05-30	Bike Spect	10	2
	1996-05-20	Bike Spect	23	8
	1996-01-17	Le Shoppe	76	5
	1996-06-01	Le Shoppe	10	3
	1996-06-01	AAA Bike	10	1
	1996-06-11	Jacks Bike	76	14

Creating the table of Price-

```
CREATE TABLE Price1 (
    PartNum INT,
    PartName VARCHAR(30),
    Price FLOAT
);
```

Inserting the data Into price table-

```
INSERT INTO Price1 (PartNum, PartName, Price)
VALUES (54, 'Pedals', 54.25),
       (42, 'Seats', 24.5),
       (46, 'Tyres', 15.25),
       (23, 'Mountain Bike', 350.45),
       (76, 'Road Bike', 530),
       (10, 'Tandem', 1200);
```

	PartNum	PartName	Price
▶	54	Pedals	54.25
	42	Seats	24.5
	46	Tyres	15.25
	23	Mountain Bike	350.45
	76	Road Bike	530
	10	Tandem	1200

1. Display the tables.

```
SELECT
    *
FROM
    Parts;
```

	Names1	Addresses	City
▶	Truewheel	550 Husker	New
	Bike Spect	CPT Shrive	Los Angels
	Le Shoppe	Hometown	KS
	AAA Bike	10 Oldtown	SL
	Jacks Bike	24 Eglin	FL

```
SELECT
    *
FROM
    Order1;
```

	Ordered_on_date	Names1	PartNum	Quantity
▶	1996-05-15	Truewheel	23	6
	1996-05-19	Truewheeel	76	3
	1996-09-02	Truewheel	10	2
	1996-03-30	Truewheel	42	8
	1996-06-30	Bike Spect	54	10
	1996-05-30	Bike Spect	10	2
	1996-05-20	Bike Spect	23	8
	1996-01-17	Le Shoppe	76	5
	1996-06-01	Le Shoppe	10	3
	1996-06-01	AAA Bike	10	1
	1996-06-11	Jacks Bike	76	14

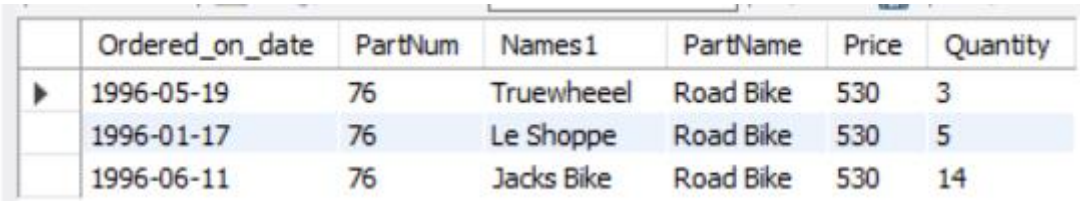
```
SELECT
    *
FROM
    price1;
```

	PartNum	PartName	Price
▶	54	Pedals	54.25
	42	Seats	24.5
	46	Tyres	15.25
	23	Mountain Bike	350.45
	76	Road Bike	530
	10	Tandem	1200

2. Find out the order details of parts having part name like ‘road_____’ .

```
SELECT
    o.Ordered_on_date,
    o.PartNum,
    o.Names1,
```

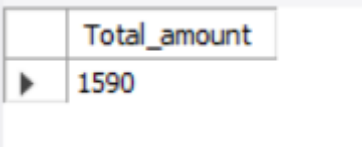
```
    p.PartName,
    p.Price,
    o.Quantity
FROM
    Order1 AS o
INNER JOIN
    price1 AS p ON o.PartNum = p.PartNum
WHERE
    p.PartName LIKE 'road%'
;
```



Ordered_on_date	PartNum	Names1	PartName	Price	Quantity
1996-05-19	76	Truewheeel	Road Bike	530	3
1996-01-17	76	Le Shoppe	Road Bike	530	5
1996-06-11	76	Jacks Bike	Road Bike	530	14

3. Determine the total order amount for part ‘road bike’. Save the result with the name of the attribute ‘total amount’.

```
SELECT
    SUM(p.Price) AS Total_amount
FROM
    Order1 AS o
INNER JOIN
    price1 AS p ON o.PartNum = p.PartNum
WHERE
    p.PartName = 'Road Bike'
;
```



Total_amount
1590

4. Display customer name, order date and total value of those customers whose order values is more than average order value.

```
SELECT
    Names1, ordered_on_date, SUM(Quantity) AS Total_value
FROM
    Order1
GROUP BY ordered_on_date
HAVING SUM(Quantity) > (SELECT
                AVG(Quantity)
            FROM
                order1);
```

Names1	ordered_on_date	Total_value
Truewheel	1996-05-15	6
Truewheel	1996-03-30	8
Bike Spect	1996-06-30	10
Bike Spect	1996-05-20	8
Jacks Bike	1996-06-11	14

5. Alter table ‘Orders’ to add one more column ‘Remark’ and set as paid if the quantity > 6 and unpaid if the quantity < 6.s

```
ALTER TABLE Order1
Add Remark VARCHAR(15);ss
```

Inserting the value of Remark column-

```
UPDATE Order1
SET   Remark =
      CASE
            WHEN Quantity > 6 THEN 'Paid'
```

ELSE 'Unpaid' END ;

Ordered_on_date	Names1	PartNum	Quantity	Remark
1996-05-15	Truewheel	23	6	Unpaid
1996-05-19	Truewheeel	76	3	Unpaid
1996-09-02	Truewheel	10	2	Unpaid
1996-03-30	Truewheel	42	8	Paid
1996-06-30	Bike Spect	54	10	Paid
1996-05-30	Bike Spect	10	2	Unpaid
1996-05-20	Bike Spect	23	8	Paid
1996-01-17	Le Shoppe	76	5	Unpaid
1996-06-01	Le Shoppe	10	3	Unpaid
1996-06-01	AAA Bike	10	1	Unpaid
1996-06-11	Jacks Bike	76	14	Paid