

|           |
|-----------|
| DATE      |
|           |
| EXPT. NO. |
|           |

## Table Operations :-

pioneerpaper.in  
PAGE NO. 1

1) Create a table Employee with EName, Eid, Dept, Address, Salary, DOJ.

⇒ Create table Employee3 (E\_Name varchar(15), E\_id varchar(15), Dept varchar(15), Address varchar(20), salary varchar(10), DOJ Date);

2) Display the schema of the table.

⇒ desc Employee3;

3) Insert four values in the table

⇒ insert into Employee3 values ('Tanu', 'tanu@gmail.com', 'HR', 'Kolkata', '12000', '01-aug-2022');

insert into Employee3 values ('Ram', 'ram@gmail.com', 'Manager', 'Kolkata', '180000', '21-aug-2012');

insert into Employee3 values ('Akash', 'akash@gmail.com', 'CEO', 'Delhi', '200000', '01-aug-2020');

insert into Employee3 values ('Soheb', 'soheb@gmail.com', 'HR', 'USA', '12000', '01-aug-2022');

4) Display the entire table along with its values.

⇒ Select \* from Employee3;

5) Retrieve only the EName and Salary of the Employee

⇒ Select E\_Name, salary from Employee3;

DATE

EXPT. NO.

- 6) Display details of employees whose address = 'Kolkata'.  
⇒ Select \* from Employee3 where Address = 'Kolkata';
- 7) Display details of employees whose ~~address~~ department = 'HR'.  
⇒ Select \* from Employee3 where dept = 'HR';
- 8) Delete all records from the relation.  
⇒ Delete from Employee3 ;
- 9) Delete the relation.  
⇒ drop table Employee3 ;
- 10) Check if relation is deleted or not.  
⇒ desc Employee3 ;



Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:56

Table created.

0.01 seconds

Object Type TABLE Object EMPLOYEE3

| Table     | Column  | Data Type | Length | Precision | Scale | Primary Key | Nullable | Default | Comment |
|-----------|---------|-----------|--------|-----------|-------|-------------|----------|---------|---------|
| EMPLOYEE3 | E_NAME  | Varchar2  | 15     | -         | -     | -           | ✓        | -       | -       |
|           | E_ID    | Varchar2  | 20     | -         | -     | -           | ✓        | -       | -       |
|           | DEPT    | Varchar2  | 15     | -         | -     | -           | ✓        | -       | -       |
|           | ADDRESS | Varchar2  | 20     | -         | -     | -           | ✓        | -       | -       |
|           | SALARY  | Varchar2  | 10     | -         | -     | -           | ✓        | -       | -       |
|           | D_O_J   | Date      | 7      | -         | -     | -           | ✓        | -       | -       |

1 - 6

1 row(s) inserted.

0.01 seconds

| E_NAME | E_ID            | DEPT    | ADDRESS | SALARY | D_O_J     |
|--------|-----------------|---------|---------|--------|-----------|
| Tarun  | dey@gmail.com   | HR      | kolkata | 12000  | 01-AUG-22 |
| Ram    | ram@gmail.com   | Manager | kolkata | 18000  | 21-AUG-12 |
| Akas   | akas@gmail.com  | CEO     | Dilli   | 200000 | 01-AUG-20 |
| Saheb  | saheb@gmail.com | HR      | usa     | 12000  | 01-AUG-22 |

4 rows returned in 0.02 seconds

[CSV Export](#)

| E_NAME | SALARY |
|--------|--------|
| Tarun  | 12000  |
| Ram    | 18000  |
| Akas   | 200000 |
| Saheb  | 12000  |

4 rows returned in 0.01 seconds



Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:57

| E_NAME                          | E_ID          | DEPT    | ADDRESS | SALARY | D_O_J                      |
|---------------------------------|---------------|---------|---------|--------|----------------------------|
| Tarun                           | dey@gmail.com | HR      | kolkata | 12000  | 01-AUG-22                  |
| Ram                             | ram@gmail.com | Manager | kolkata | 18000  | 21-AUG-12                  |
| 2 rows returned in 0.00 seconds |               |         |         |        | <a href="#">CSV Export</a> |

| E_NAME                          | E_ID            | DEPT | ADDRESS | SALARY | D_O_J                      |
|---------------------------------|-----------------|------|---------|--------|----------------------------|
| Tarun                           | dey@gmail.com   | HR   | kolkata | 12000  | 01-AUG-22                  |
| Saheb                           | saheb@gmail.com | HR   | usa     | 12000  | 01-AUG-22                  |
| 2 rows returned in 0.00 seconds |                 |      |         |        | <a href="#">CSV Export</a> |

4 row(s) deleted.

0.00 seconds

Table dropped.

0.03 seconds

Object to be described could not be found.

14/09/23



Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:57

Lab:2

1) Create a table EMPLOYEE with the columns First\_Name, Second\_Name, Address1, Address2, Ph\_No, E\_Id, salary, D\_O\_J.

Create table EMPLOYEE (First\_Name <sup>varchar</sup>(15), Second\_Name <sup>varchar</sup>(15),  
 Address1 <sup>varchar</sup>(20), Address2 <sup>varchar</sup>(20), Ph\_No  
<sup>varchar</sup>(10), E\_Id <sup>varchar</sup>(10), salary <sup>varchar</sup>(10),  
 D\_O\_J Date);

2) Describe the table.

desc EMPLOYEE

3) Insert three values into the table.

insert into EMPLOYEE values ('Tom', 'Dey', 'Kolkata', 'Bankura',  
 '77648512', 'E001', '50000', '01-jan-2015');

insert into EMPLOYEE values ('Soheb', 'Pai', 'mumbai', 'Barkur',  
 '77845512', 'E002', '40000', '01-feb-2017');

insert into EMPLOYEE values ('Raj', 'Sen', 'Kolkata', 'Bankura',  
 '795855', 'E004', '45500', '01-sep-2018');

4) Select all the records from the relation.

select \* from EMPLOYEE;



Shot on vivo Z1x  
 Vivo AI camera

5) Create a MASTER table with same field in the previous relation modifying E\_Id = Id .

Create table MASTER (First\_Name varchar(15), Second\_Name varchar(15), Address1 varchar(20), Address2 varchar(20), Ph\_no varchar(10), Id varchar(10), Doj Date);

6) Delete all the records from the relation . (MASTER)

delete from MASTER ;

7) Delete the row where Id = "E001"

delete from MASTER where Id = 'E001' ;

8) update any field in the MASTER relation .

9) Add Email in the MASTER table relation .

Alter table MASTER ADD Email varchar(20);

DATE

EXPT. NO.

10) Drop address from MASTER table .

Alter table MASTER drop column Address1 ;

Alter table MASTER drop column Address2 ;

11) Rename MASTER to MID

rename MASTER to MID ;

12) Drop MID .

drop table MID ;

PIONEER®



Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:57

Table created.

0.00 seconds

Object Type TABLE Object EMPLOYEE6

| Table     | Column       | Data Type | Length | Precision | Scale | Primary Key | Nullable | Default | Comment |
|-----------|--------------|-----------|--------|-----------|-------|-------------|----------|---------|---------|
| EMPLOYEE6 | FIRST_NAME   | VARCHAR2  | 15     | -         | -     | -           | ✓        | -       | -       |
|           | SECOUND_NAME | VARCHAR2  | 15     | -         | -     | -           | ✓        | -       | -       |
|           | ADDRESS1     | VARCHAR2  | 20     | -         | -     | -           | ✓        | -       | -       |
|           | ADDRESS2     | VARCHAR2  | 20     | -         | -     | -           | ✓        | -       | -       |
|           | PH_NO        | VARCHAR2  | 10     | -         | -     | -           | ✓        | -       | -       |
|           | E_ID         | VARCHAR2  | 10     | -         | -     | -           | ✓        | -       | -       |
|           | SALARY       | VARCHAR2  | 10     | -         | -     | -           | ✓        | -       | -       |
|           | D_O_J        | DATE      | 7      | -         | -     | -           | ✓        | -       | -       |

1 - 8

1 row(s) inserted.

0.01 seconds

| FIRST_NAME | SECOUND_NAME | ADDRESS1 | ADDRESS2 | PH_NO    | E_ID | SALARY | D_O_J     |
|------------|--------------|----------|----------|----------|------|--------|-----------|
| Tanun      | Dey          | Kolkata  | Bankura  | 77648512 | E001 | 50000  | 01-JAN-15 |
| Saheb      | Pal          | Mumbai   | Bankura  | 77845512 | E002 | 40000  | 01-FEB-17 |
| Raj        | Sen          | Kolkata  | Bankura  | 745855   | E004 | 45500  | 01-SEP-18 |

3 rows returned in 0.03 seconds

[CSV Export](#)

Object Type TABLE Object MASTER

| Table  | Column       | Data Type | Length | Precision | Scale | Primary Key | Nullable | Default | Comment |
|--------|--------------|-----------|--------|-----------|-------|-------------|----------|---------|---------|
| MASTER | FIRST_NAME   | VARCHAR2  | 15     | -         | -     | -           | ✓        | -       | -       |
|        | SECOUND_NAME | VARCHAR2  | 15     | -         | -     | -           | ✓        | -       | -       |
|        | ADDRESS1     | VARCHAR2  | 20     | -         | -     | -           | ✓        | -       | -       |
|        | ADDRESS2     | VARCHAR2  | 20     | -         | -     | -           | ✓        | -       | -       |
|        | PH_NO        | VARCHAR2  | 10     | -         | -     | -           | ✓        | -       | -       |
|        | ID           | VARCHAR2  | 10     | -         | -     | -           | ✓        | -       | -       |
|        | SALARY       | VARCHAR2  | 10     | -         | -     | -           | ✓        | -       | -       |
|        | D_O_J        | DATE      | 7      | -         | -     | -           | ✓        | -       | -       |

1 - 8



Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:57

| FIRST_NAME | SECOUND_NAME | ADDRESS1 | ADDRESS2 | PH_NO    | ID   | SALARY | D_O_J     |
|------------|--------------|----------|----------|----------|------|--------|-----------|
| Tarun      | Dey          | Kolkata  | Bankura  | 77648512 | E001 | 50000  | 01-JAN-15 |
| Saheb      | Pal          | Mumbai   | Bankura  | 77845512 | E002 | 40000  | 01-FEB-17 |
| Raj        | Sen          | Kolkata  | Bankura  | 745855   | E004 | 45500  | 01-SEP-18 |

3 rows returned in 0.00 seconds

[CSV Export](#)

3 row(s) deleted.

0.00 seconds

1 row(s) deleted.

0.02 seconds

| FIRST_NAME | SECOUND_NAME | ADDRESS1 | ADDRESS2 | PH_NO    | ID   | SALARY | D_O_J     |
|------------|--------------|----------|----------|----------|------|--------|-----------|
| Raj        | Sen          | Kolkata  | Bankura  | 745855   | E004 | 45500  | 01-SEP-18 |
| Saheb      | Pal          | Mumbai   | Bankura  | 77845512 | E002 | 40000  | 01-FEB-17 |

2 rows returned in 0.00 seconds

[CSV Export](#)

Table altered.

0.02 seconds

Object Type TABLE Object MASTER

| Table  | Column       | Data Type | Length | Precision | Scale | Primary Key | Nullable | Default | Comment |
|--------|--------------|-----------|--------|-----------|-------|-------------|----------|---------|---------|
| MASTER | FIRST_NAME   | VARCHAR2  | 15     | -         | -     | -           | ✓        | -       | -       |
|        | SECOUND_NAME | VARCHAR2  | 15     | -         | -     | -           | ✓        | -       | -       |
|        | ADDRESS1     | VARCHAR2  | 20     | -         | -     | -           | ✓        | -       | -       |
|        | ADDRESS2     | VARCHAR2  | 20     | -         | -     | -           | ✓        | -       | -       |
|        | PH_NO        | VARCHAR2  | 10     | -         | -     | -           | ✓        | -       | -       |
|        | ID           | VARCHAR2  | 10     | -         | -     | -           | ✓        | -       | -       |
|        | SALARY       | VARCHAR2  | 10     | -         | -     | -           | ✓        | -       | -       |
|        | D_O_J        | DATE      | 7      | -         | -     | -           | ✓        | -       | -       |
|        | E_MAIL       | VARCHAR2  | 20     | -         | -     | -           | ✓        | -       | -       |

1 - 9



Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:57

Table dropped.

0.12 seconds

| Object Type TABLE Object MASTER |             |           |        |           |       |             |          |         |         |
|---------------------------------|-------------|-----------|--------|-----------|-------|-------------|----------|---------|---------|
| Table                           | Column      | Data Type | Length | Precision | Scale | Primary Key | Nullable | Default | Comment |
| MASTER                          | FIRST_NAME  | Varchar2  | 15     | -         | -     | -           | ✓        | -       | -       |
|                                 | SECOND_NAME | Varchar2  | 15     | -         | -     | -           | ✓        | -       | -       |
|                                 | PH_NO       | Varchar2  | 10     | -         | -     | -           | ✓        | -       | -       |
|                                 | ID          | Varchar2  | 10     | -         | -     | -           | ✓        | -       | -       |
|                                 | SALARY      | Varchar2  | 10     | -         | -     | -           | ✓        | -       | -       |
|                                 | D_O_J       | Date      | 7      | -         | -     | -           | ✓        | -       | -       |
|                                 | E_MAIL      | Varchar2  | 20     | -         | -     | -           | ✓        | -       | -       |

1-7

Statement processed.

0.01 seconds

| Object Type TABLE Object MID |             |           |        |           |       |             |          |         |         |
|------------------------------|-------------|-----------|--------|-----------|-------|-------------|----------|---------|---------|
| Table                        | Column      | Data Type | Length | Precision | Scale | Primary Key | Nullable | Default | Comment |
| MID                          | FIRST_NAME  | Varchar2  | 15     | -         | -     | -           | ✓        | -       | -       |
|                              | SECOND_NAME | Varchar2  | 15     | -         | -     | -           | ✓        | -       | -       |
|                              | PH_NO       | Varchar2  | 10     | -         | -     | -           | ✓        | -       | -       |
|                              | ID          | Varchar2  | 10     | -         | -     | -           | ✓        | -       | -       |
|                              | SALARY      | Varchar2  | 10     | -         | -     | -           | ✓        | -       | -       |
|                              | D_O_J       | Date      | 7      | -         | -     | -           | ✓        | -       | -       |
|                              | E_MAIL      | Varchar2  | 20     | -         | -     | -           | ✓        | -       | -       |

1-7

Table dropped.

0.10 seconds



Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:57

|           |
|-----------|
| DATE      |
|           |
| EXPT. NO. |
|           |

## LAB: 3

1) Create table STUDENT with the columns Name, Roll, Address, E-mail, marks .

Create table student (Name varchar(15), Roll int, Address varchar(15), E-mail varchar(15), Marks int);

2) Add Primary Key to the column Roll .

Insert into student (Name, Roll, Address, Email, marks)

3) Select Name, Roll , Percentage\_of\_Marks from the relation .

Select \* from student ; Name, Roll, (Marks \* 100.0 / (select max(Marks) from student)) As Percentage from student ;

4) Rename column Marks as Percentage\_of\_Marks .

Alter table student rename marks to Percentage\_of\_Marks ;

5) Select Name, Roll and Address of the student whose Marks is between 50 to 80 .

Select Name, Roll, Address where Percentage\_of\_marks between 50 to 80 ;

Shot on vivo Z1x

Vivo AI camera

DATE

EXPT. NO.

6) Retrieve Name, Email, of the student whose Name starts with "S".

Select Name, Email from student whose Name is like 'S%';

7) Select the tuples where Address is Durgapur or Kolkata.

Select \* from student where Address = 'Durgapur' || 'Kolkata';

8) Select Name of students whose Name is sumit, Ankit, Rajon.

Select Name ~~Address~~ from student where name in ('sumit', 'Ankit', 'Rajon');

9) Select Name of students whose Name is except sumit, Ankit, Rajon.

Select Name, from student where name not in ('sumit', 'Ankit', 'Rajon');

PIONEER®



Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:57

DATE

EXPT. NO.

10) Retrieve the tuple where address are Durgapur, Holdia, Kolkata.

Select \* from student where address in ('Durgapur', 'Holdia', 'Kolkata');

11) Select the details of the student ascending to their marks.

Select \* from student order by percentage\_of\_marks ASC;

12) Select the details of the student descending to their marks.

Select \* from student order by percentage\_of\_marks DESC;

13) Retrieve the student whose Name starts with 'S' and end with 'N'.

Select \* from student where Name like 'S%.N';

14) Drop the relation.

Drop table student;

PIONEER®



Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:58

### Input

```
CREATE TABLE STUDENT (
    Name VARCHAR(255),
    Roll INT,
    Address VARCHAR(255),
    E_mail VARCHAR(255),
    Marks INT
);
```

Run SQL

### Output

SQL query successfully executed. However, the result set is empty.

### Input

```
INSERT INTO STUDENT (Name, Roll, Address, E_mail, Marks)
VALUES
    ('Aryan', 102, 'DLF-2', 'aryan@example.com', 88),
    ('Soham', 103, 'KOLKATA', 'Soham@example.com', 75),
    ('Sumit', 104, 'ASANSOL', 'sumit@example.com', 35),
    ('Amit', 105, 'HALDIA', 'ami@example.com', 65),
    ('Tarun', 106, 'BANKURA', 'tarun@example.com', 48),
    ('Arijeet', 107, 'DUMDUM', 'ari@example.com', 74),
    ('Rahul', 108, 'DURGAPUR', 'rahul@example.com', 48),
    ('Ranjan', 109, 'KOLKATA', 'ranhan@example.com', 45),
    ('Pratyay', 110, 'DURGAPUR', 'pratya@example.com', 66 ),
    ('Shaan', 111, 'KOLKATA', 'shaan@example.com', 85);
SELECT * FROM STUDENT;
```

Run SQL

### Output

| Name    | Roll | Address  | E_mail             | Marks |
|---------|------|----------|--------------------|-------|
| Aryan   | 102  | DLF-2    | aryan@example.com  | 88    |
| Soham   | 103  | KOLKATA  | Soham@example.com  | 75    |
| Sumit   | 104  | ASANSOL  | sumit@example.com  | 35    |
| Amit    | 105  | HALDIA   | ami@example.com    | 65    |
| Tarun   | 106  | BANKURA  | tarun@example.com  | 48    |
| Arijeet | 107  | DUMDUM   | ari@example.com    | 74    |
| Rahul   | 108  | DURGAPUR | rahul@example.com  | 48    |
| Ranjan  | 109  | KOLKATA  | ranhan@example.com | 45    |
| Pratyay | 110  | DURGAPUR | pratya@example.com | 66    |
| Shaan   | 111  | KOLKATA  | shaan@example.com  | 85    |
| Aryan   | 102  | DLF-2    | aryan@example.com  | 88    |
| Soham   | 103  | KOLKATA  | Soham@example.com  | 75    |
| Sumit   | 104  | ASANSOL  | sumit@example.com  | 35    |
| Amit    | 105  | HALDIA   | ami@example.com    | 65    |

### Input

```
SELECT Name, Roll, (Marks * 100.0 / (SELECT MAX(Marks) FROM STUDENT)) AS Percentage
FROM STUDENT;
```

Run SQL

### Output

| Name    | Roll | Percentage        |
|---------|------|-------------------|
| Aryan   | 102  | 100               |
| Soham   | 103  | 86.22222222222222 |
| Sumit   | 104  | 35.77272727272727 |
| Tarun   | 106  | 71.86363636363636 |
| Arijeet | 107  | 64.54545454545454 |

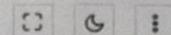


Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:58

**Input**

```
ALTER TABLE STUDENT
RENAME COLUMN Marks TO Percentage_of_marks;
```

**Run SQL****Output**

| Name    | Roll | Address  | E-mail              | Percentage_of_marks |
|---------|------|----------|---------------------|---------------------|
| Aryan   | 102  | DLF-2    | aryan@example.com   | 88                  |
| Soham   | 103  | KOLKATA  | soham@example.com   | 75                  |
| Sumit   | 104  | ASANSOL  | sumit@example.com   | 35                  |
| Amit    | 105  | HALDIA   | ami@example.com     | 65                  |
| Tarun   | 106  | BANKURA  | tarun@example.com   | 48                  |
| Arijeet | 107  | DUMDUM   | ari@example.com     | 74                  |
| Rahul   | 108  | DURGAPUR | rahul@example.com   | 48                  |
| Ranjan  | 109  | KOLKATA  | ranjan@example.com  | 45                  |
| Pratyay | 110  | DURGAPUR | pratyay@example.com | 66                  |
| Shaan   | 111  | KOLKATA  | shaan@example.com   | 55                  |

**Input**

```
SELECT Name, E_mail
FROM STUDENT
WHERE Marks >= 50 AND Marks <= 80;
```

**Output**

| Name    | E_mail              |
|---------|---------------------|
| Soham   | Soham@example.com   |
| Amit    | ami@example.com     |
| Arijeet | ari@example.com     |
| Pratyay | pratyay@example.com |

**Output****Input**

```
SELECT Name, E_mail
FROM STUDENT
WHERE Name LIKE 'S%';
```

| Name  | E_mail            |
|-------|-------------------|
| Soham | Soham@example.com |
| Sumit | sumit@example.com |
| Shaan | shaan@example.com |

**Input**

```
SELECT Name, Address
FROM STUDENT
WHERE Name IN ('Sumit', 'Amit', 'Ranjan');
```

**Output**

| Name  | Address |
|-------|---------|
| Sumit | ASANSOL |
| Amit  | HALDIA  |

Shot on vivo Z1x  
Vivo AI camera

**Input**

```
SELECT Name, Address
FROM STUDENT
WHERE Name NOT IN ('Sumit', 'Amit', 'Ranjan');
```

| Name    | Address  |
|---------|----------|
| Aryan   | DLF-2    |
| Soham   | KOLKATA  |
| Tarun   | BANKURA  |
| Arijeet | DUMDUM   |
| Rahul   | DURGAPUR |
| Pratyay | DURGAPUR |
| Shaan   | KOLKATA  |

2023.11.21 19:58

**Input**

```
SELECT *
FROM STUDENT
WHERE Address IN ('DURGAPUR', 'HALDIA', 'KOLKATA');
```

**Output**

| Name    | Roll | Address  | E_mail             | Marks |
|---------|------|----------|--------------------|-------|
| Soham   | 103  | KOLKATA  | Soham@example.com  | 75    |
| Amit    | 105  | HALDIA   | ami@example.com    | 65    |
| Rahul   | 108  | DURGAPUR | rahul@example.com  | 48    |
| Ranjan  | 109  | KOLKATA  | ranhan@example.com | 45    |
| Pratyay | 110  | DURGAPUR | pratya@example.com | 66    |
| Shaan   | 111  | KOLKATA  | shaan@example.com  | 85    |

**Input**

```
SELECT *
FROM STUDENT
WHERE Address IN ('DURGAPUR', 'KOLKATA');
```

**Output**

| Name    | Roll | Address  | E_mail             | Marks |
|---------|------|----------|--------------------|-------|
| Soham   | 103  | KOLKATA  | Soham@example.com  | 75    |
| Rahul   | 108  | DURGAPUR | rahul@example.com  | 48    |
| Ranjan  | 109  | KOLKATA  | ranhan@example.com | 45    |
| Pratyay | 110  | DURGAPUR | pratya@example.com | 66    |
| Shaan   | 111  | KOLKATA  | shaan@example.com  | 85    |

**Input**

```
SELECT *
FROM STUDENT
ORDER BY Marks ASC;
```

| Name     | Roll | Address  | E_mail             | Marks |
|----------|------|----------|--------------------|-------|
| Sumit    | 104  | ASANSOL  | sumit@example.com  | 35    |
| Ranjan   | 109  | KOLKATA  | ranhan@example.com | 45    |
| Tarun    | 106  | BANKURA  | tarun@example.com  | 48    |
| Rahul    | 108  | DURGAPUR | rahul@example.com  | 48    |
| Amit     | 105  | HALDIA   | ami@example.com    | 65    |
| Pratyay  | 110  | DURGAPUR | pratya@example.com | 66    |
| Arijeeet | 107  | DUMDUM   | ari@example.com    | 74    |
| Soham    | 103  | KOLKATA  | Soham@example.com  | 75    |



Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:58

DATE

EXPT. NO.

pioneerpaper.in  
PAGE NO. 3

**Input**

```
SELECT *  
FROM STUDENT  
ORDER BY Marks DESC;
```

| Name    | Roll | Address  | E_mail             | Marks |
|---------|------|----------|--------------------|-------|
| Aryan   | 102  | DLF-2    | aryan@example.com  | 88    |
| Shaan   | 111  | KOLKATA  | shaan@example.com  | 85    |
| Soham   | 103  | KOLKATA  | Soham@example.com  | 75    |
| Arijeet | 107  | DUMDUM   | ari@example.com    | 74    |
| Pratyay | 110  | DURGAPUR | pratya@example.com | 66    |
| Amit    | 105  | HALDIA   | ami@example.com    | 65    |
| Tarun   | 106  | BANKURA  | tarun@example.com  | 48    |
| Rahul   | 108  | DURGAPUR | rahul@example.com  | 48    |

**Input**

```
SELECT *  
FROM STUDENT  
WHERE Name LIKE 'S%N';
```

| Name  | Roll | Address | E_mail            | Marks |
|-------|------|---------|-------------------|-------|
| Shaan | 111  | KOLKATA | shaan@example.com | 85    |

**Output**

**Input**

```
DROP TABLE STUDENT;
```

Error: no such table: STUDENT

PIONEER®



Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:58

## LAB-4

1) Create table Employee master with the ~~6~~ columns:-

First\_Name, Second\_Name, Address1, Address2, Phone\_No,  
Emp\_ID, salary, Date\_of\_joining .

⇒ Create table Employee\_master (First\_Name varchar(50),  
Second\_Name varchar(50), Address1 varchar(100), Address2  
varchar(100), Phone\_no varchar(15), Emp\_Id INT, salary  
decimal (10,2), Date\_of\_joining date );

2) Insert atleast two rows with update field.

⇒ Insert into Employee\_master values ('john', 'doe', '123 main st',  
'APT 45' '555-1234', 1, 50000.00, '2023-01-01'), ('jane',  
'smith', '456 oak st', 'suite 78', '555-5678', 2, 60000.00,  
'2023-02-15');

3) Create the table employee which contains the  
Emp\_ID field is unique and with other field Name,  
Address1, Address2, Date of joining and phone number.

⇒ Create table Employee (Emp\_ID INT unique, Name varchar(100),  
Address1 varchar(100), Address2 varchar(100), Date\_of\_joining  
date, Phone\_no varchar(15) );

DATE  
EXPT. NO.

4) Insert data into table with the utility of unique field.

⇒ Insert into Employee values(1, 'john doe', '123 Main St', 'Apt 45', '2023-01-01', '555-1234'), (2, 'jane smith', '956 Oak St', 'suite 78', '2023-02-15', '555-5678');

5) Create a table Employee with Emp\_ID as Primary Key and with other field Name, Address1, Address2, Date of joining, Phone Number, Salary.

⇒ Create table Employee (Emp\_ID int Primary Key, Name varchar(100), Address1 varchar(100), Address2 varchar(100), Date\_of\_joining date, Phone\_No varchar(15), salary decimal (10, 2));

6) Create a table Emp\_ID Primary Key and foreign key as Reference Emp\_ID in employee table the other field are basic, DA, and Gross.

⇒ Create table Salary (Emp\_ID int Primary Key, Basic decimal (8, 2), Da decimal (8, 2), Gross decimal (10, 2), Foreign Key (Emp\_ID) References Employee (Emp\_ID));

PIONEER®



Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:58

7) Insert two rows in the table .

⇒ Insert into salary values (1, 40000.00, 10000.00, 50000.00);  
(2, 50000.00, 10000.00, 60000.00);

8) Modify the Name column with new size .

⇒ Alter table Employee modify column Name varchar (255);

9) Delete the table Employee from database .

⇒ Drop table Employee;

10) Delete the table salary from database .

⇒ Drop table salary .

| FIRST_NAME | SECOND_NAME | ADDRESS1    | ADDRESS2 | PHONE_NO | EMP_ID | SALARY   |
|------------|-------------|-------------|----------|----------|--------|----------|
| John       | Doe         | 123 Main St | Apt 45   | 555-1234 | 1      | 50000.00 |
| Jane       | Smith       | 456 Oak St  | Suite 78 | 555-5678 | 2      | 60000.00 |

| EMP_ID | NAME       | ADDRESS1    | ADDRESS2 | DATE_OF_JOINING | PHONE_NO |
|--------|------------|-------------|----------|-----------------|----------|
| 1      | John Doe   | 123 Main St | Apt 45   | 2023-11-16      | 555-1234 |
| 2      | Jane Smith | 456 Oak St  | Suite 78 | 2023-11-17      | 555-5678 |



Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:58

| EMP_ID | NAME       | ADDRESS1    | ADDRESS2 | DATE_OF_JOINING | PHONE_NO |
|--------|------------|-------------|----------|-----------------|----------|
| 1      | John Doe   | 123 Main St | Apt 45   | 2023-11-16      | 555-1234 |
| 2      | Jane Smith | 456 Oak St  | Suite 78 | 2023-11-17      | 555-5678 |

| EMP_ID | BASIC    | DA       | GROSS    |
|--------|----------|----------|----------|
| 1      | 40000.00 | 10000.00 | 50000.00 |
| 2      | 50000.00 | 10000.00 | 60000.00 |



Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:58

DATE

EXPT. NO.

## LAB-5

1) Create table student\_Details with the fields:-

Roll\_No, Name, Address, Email & Phone\_No.

=> Create table student\_Details ( Roll\_NO int Primary Key, Name varchar(100), Address varchar(200), Email varchar(100), phone\_No varchar(15) );

2) Create table subject\_Details with the fields:-

Roll\_No, Sub\_code, Sub\_Name, sub\_Teacher, & Marks.

=> Create table ~~student~~ subject\_Details ( Roll\_NO int, sub\_code varchar(10), sub\_Name varchar(100), sub\_Teacher varchar(100), Marks int, primary key(Roll\_No, sub\_code), foreign key (Roll\_No) references student\_Details (Roll\_No));

3) Insert data in both table.

Insert into student\_Details values (1, 'John Doe', '123 Main St', 'john@example.com', '555-1234'), (2, 'Jane Smith', '456 Oak St', 'jane@example.com', '555-5678');

Insert into subject\_Details values (1, 'BCA 401', 'math', 'Prof. Johnson', 80), (1, 'BCA 402', 'computer science', 'Prof. Smith', 75), (2, 'BCA 401', 'math', 'Prof. Johnson', 85), (2, 'BCA 402', 'computer science', 'Prof. Smith', 70);



Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:58

DATE  
XPT. NO.

4) Select average marks of all students for the subject code 'BCA 401'.

⇒ select avg (marks) as average\_marks from subject\_details where sub\_code = 'BCA 401';

5) Select the roll\_no of the student who has got minimum marks in 'DBMS'.

⇒ select roll\_no from subject\_details where sub\_name = 'DBMS' ordered by marks limit 1;

6) Retrieve the Name and Roll\_no of the students who has failed.

⇒ select Name, Roll\_no from student\_Details where Roll\_no in (select Roll\_no from subject\_Details where marks < 40);

7) Retrieve the student Name and Roll\_no who has got maximum Marks in all subjects.

⇒ select Name, Roll\_no, from student\_Details where Roll\_no IN (select Roll\_no from subject\_Details ordered by marks desc limit 1);



Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:59

8) Retrieve the sub\_Name & Teacher Name whose Marks is

Best -

⇒ Select sub\_Name, sub\_Teacher from subject\_Details order by marks desc limit 1;

9) Retrieve the Roll\_NO, Name, Email & Address of the topper student .

⇒ select sub\_Name, sub\_F s.Roll\_No, s.Name, s.Email, s.Address from Student\_Details s join (select Roll\_No from subject\_Details order by marks desc limit 1 ) T on s.Roll\_No = T.Roll\_No;

10) Retrieve the Roll\_NO, Name, Address of the student who failed in exam .

⇒ Select Roll\_No, Name, Address from Student\_Details where Roll\_No in (select Roll\_No from subject\_Details where marks < 40 );

| ROLL_NO | AVERAGE_MARKS |
|---------|---------------|
| 1       | 85            |
| 2       | 75            |

| ROLL_NO | NAME       |
|---------|------------|
| 2       | Jane Smith |

| ROLL_NO | NAME     |
|---------|----------|
| 1       | John Doe |



Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:59

| SUB_NAME                   | SUB_TEACHER   |
|----------------------------|---------------|
| Database Management System | Prof. Johnson |

| ROLL_NO | NAME     | EMAIL            | ADDRESS     |
|---------|----------|------------------|-------------|
| 1       | John Doe | john@example.com | 123 Main St |

| ROLL_NO | NAME       | ADDRESS    |
|---------|------------|------------|
| 2       | Jane Smith | 456 Oak St |



Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:59

DATE

EXPT. NO.

## LAB - 6

pioneerpaper.in  
PAGE NO. 16

- 1) Create table orderdetails with the fields:- order\_no, Product\_no, Quantity\_order & Quantity\_Dispose.  
⇒ Create table orderDetails ( order\_no int Primary Key, Product\_no varchar(10), quantity\_order int, quantity\_Dispose int, foreign key (Product\_no) references Product (product\_no) );
- 2) Create table salesorden with the fields:- order\_no, Client\_no, Order\_date, The 'order\_no' reference to the order\_no, of the table 'orderdetails'.  
⇒ Create table salesorden (order\_no int Primary Key, Client\_no int, Order\_date Date, foreign key (order\_no) references orderdetails (order\_no));
- 3) Create table client with the fields:- client\_no, Name, Balance, Due, the 'client\_no' reference to the 'client\_no' of the table 'salesorden'.  
⇒ Create table client (client\_no int Primary Key, Name varchar(100), Balance decimal (10,2), Due decimal (10,2), foreign key (client\_no) references salesorden (client\_no));

PIONEER®



Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:59

DATE

EXPT. NO.

4) Create table ~~client~~ with the fields:- Product with the fields :- Product\_No , Description , the 'Product\_No' reference to 'product\_no' of the table 'SalesOrder'.  
'OrderDetails'.

⇒ Create table Product (product\_no varchar(10) primary key, Description varchar(100));

5) Insert atleast two records in each table .

⇒ Insert into Product values ('P001', 'Product 1 Description'),  
( 'P002', 'Product 2 Description');

Insert into OrderDetails, values (1, 'P001', 10, 5); (2, 'P002', 8, 8);

Insert into SalesOrder values (1, 101, '2023-01-15');  
(2, 102, '2023-01-20');

6) Retrieve the product\_no & the total quantity order & for each product from the table 'OrderDetails'.

⇒ Select product\_no, sum (quantity\_order) as Total\_Quantity\_order from OrderDetails group by product\_no;

PIONEER®



Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:59

DATE  
EXPT. NO.

- 7) Retrieve the Product\_No & the total quantity ordered for product 'P001' & 'P004' from table 'ordenDetails'.
- ⇒ Select Product\_No, sum(Quantity\_Ordered) as Total\_Quantity\_Ordered from ordenDetails where Product\_No in ('P001', 'P002') group by Product\_No;
- 8) Retrieve all orden placed by a client named 'Jack' from Client table.
- ⇒ Select o.Order\_No, o.Client\_No, o.Order\_Date from SalesOrder o join Client c on o.Client\_No = c.Client\_No where c.Name = 'Jack';
- 9) Find out all Product that are not be sold from the Product table based on the product actually sold as shown in the table 'ordenDetails'.
- ⇒ Select p.Product\_No, p.Description from Product p left join ordenDetails od on p.Product\_No = od.Product\_No where od.Product\_No is null;
- 10) Retrieve the details of the client who has place his orden at the last of this month.
- ⇒ Select c.Client\_No, c.Name, c.Balance, c.Due, from Client c join SalesOrder so no c.Client\_No = so.Client\_No where so.Order\_Date = (select max(Order\_Date) from SalesOrder where Month(Order\_Date) = Month(Current\_date))
- Shot on vivo Z1x  
Vivo AI camera

| CLIENT_NO | NAME | BALANCE | DUE    |
|-----------|------|---------|--------|
| 201       | John | 1000.00 | 500.00 |
| 202       | Jack | 1500.00 | 200.00 |

| ORDER_NO | CLIENT_NO | ORDER_DATE |
|----------|-----------|------------|
| 1        | 201       | 2023-11-16 |
| 2        | 202       | 2023-11-17 |

| ORDER_NO | PRODUCT_NO | QUANTITY_ORDER | QUANTITY_DISPENSE |
|----------|------------|----------------|-------------------|
| 1        | 101        | 5              | 5                 |
| 2        | 102        | 3              | 2                 |

| CLIENT_NO | NAME | BALANCE | DUE    |
|-----------|------|---------|--------|
| 202       | Jack | 1500.00 | 200.00 |



Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:59

| PRODUCT_NO | DESCRIPTION |
|------------|-------------|
|------------|-------------|

|     |          |
|-----|----------|
| 102 | ProductB |
|-----|----------|

| ORDER_NO | CLIENT_NO | ORDER_DATE |
|----------|-----------|------------|
|----------|-----------|------------|

|   |     |            |
|---|-----|------------|
| 2 | 202 | 2023-11-17 |
|---|-----|------------|

| PRODUCT_NO | TOTAL_QUANTITY_ORDERED |
|------------|------------------------|
|------------|------------------------|

|     |   |
|-----|---|
| 101 | 5 |
|-----|---|

| PRODUCT_NO | TOTAL_QUANTITY_ORDER |
|------------|----------------------|
|------------|----------------------|

|     |   |
|-----|---|
| 101 | 5 |
| 102 | 3 |

| PRODUCT_NO | DESCRIPTION |
|------------|-------------|
|------------|-------------|

|     |          |
|-----|----------|
| 101 | ProductA |
| 102 | ProductB |



Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:59

|           |  |
|-----------|--|
| DATE      |  |
| EXPT. NO. |  |

## LAB-7

- a) Retrieve the list of names and the cities of all the clients.  
 $\Rightarrow$  select client\_name, city from client\_master;
  
  
  
  
  
- b) List the various products available.  
 $\Rightarrow$  select product\_number, product\_name from product\_master;
  
  
  
  
  
- c) Find the names of all clients having 'a' as the second letter in their names.  
 $\Rightarrow$  select client\_name from client\_master where substring(client\_name, 2, 1) = 'a';
  
  
  
  
  
- d) List all the clients who are located in TEZPUR.  
 $\Rightarrow$  select \* from product\_master where selling\_price > 2000 and selling\_price <= 5000;  
 $\Rightarrow$  select \* from client\_master where city = 'TEZPUR';

PIONEER®



Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:59

e) Find the products whose selling price is greater than 2000 and less than or equal to 5000.

⇒ select \* from product\_master where selling\_price > 2000 and selling\_price <= 5000;

f) Add a new column NEW\_PRICE into the product\_master table.

⇒ Alter table product\_master add column new\_price int;

g) Rename the column Product\_rate of sales\_order\_Details to new\_product\_rate.

⇒ Alter table sales\_order\_Details rename column Product\_rate to new\_product\_rate;

h) List the products in sorted order of their description.

⇒ select \* from product\_master Order by Product\_description;

DATE

EXPT. NO.

i) Display the order number and date on which the clients placed their order.

⇒ select order\_number, order\_date from sales\_order;

j) Delete all the records having delivery date before 25th August, 2008.

⇒ delete from sales\_order where delivery\_date < '2008-08-25';

k) Change the delivery date of order number ON01008 to 16-08-08.

⇒ update sales\_order set delivery\_date = '2008-08-16' where order\_number = 'ON01008';

l) Change the bal\_due to client\_no CN01003 to 1200.

⇒ update client\_master set bal\_due = 1200 where client\_number = 'CN01003';

PIONEER®



Shot on vivo Z1x  
Vivo AI camera

2023.11.21 19:59

m) Find the Product with description as 'HDD1034' and 'DVDRW' .

⇒ Select \* from product\_master where product\_description in ('HDD1034', 'DVDRW');

n) List the names, city and state of the clients not in the state of 'ASSAM' .

⇒ Select client\_name, city, state from client\_master where state <> 'ASSAM' ;

o) List # of all orders that were canceled in the month of March.

⇒ Select \* from sales\_order where orden\_Status = 'canceled' and Extract(month from orden\_date) = 3;

=