```
insert int department values
  (1, 'CSE' 'Hyderabad'),
  (2, 'ECE', 'mumbai'),
  (3, 'mech', 'Delhi');

insert into student values
(101, upper ('rahul'), 20, 1, 'hyderabad'),
insert into student values
(102, 'Anjali', 22, 2, mumbai');
insert into student values;
(103, 'kiran 19, 1, 'pune'),
insert into student values,
(104, 'mohith', 23, 3, 'Delhi),
insert into student values
(105, 'sara', 21, 1, hyderabad)
select * from students,
```

	student ID	NAME	AGE	DeptID	city	Join Date
1	101	Rahul	20	1	hyderabad	2025-8-26
2	102	Anjali	22	2	mumbai	2025-8-26
3	103	kiran	19	1	pune	2025-8-26
4	104	mohith	23	3	delhi	2025-08-26
5	105	sarakhali	21	1	hyderabad	2025-08-26

```
select * from department;
```

	DeptID	Department	location
1	1	CSE	HYd
2	2	EEE	mumbai
3	3	MECh	Delhi

```
SElect Name, AGE,
From   student
where  Age b/w  19 & 22;
```

	Name	AGE
1	Rahul	20
2	AnJali	22
3	kiran	19
4	srikanth	21

```
select Name, Dept ID,
 From  student
where  Dept ID  IN(1,3)
order  BY Dept ID Desc;
```

Name	Dept ID
Rahul	3
AnJali	1
kiran	1
srikanth	1

```
update  student1
set  AGE = AGE+1
where  DEPT ID = 1 And AGE <21;
```

S.NO	Stu-ID	name	Age	DePt ID	city	Joindat
1	101	Rahul	21	1	Hyd	25-8-26
2	102	Anjali	22	2	mumbai	25-8-26
3	103	kiran	20	1	pune	25-8-26
4	104	mohith	23	3	Delhi	25-8-26
5	105	srikanth	21	1	hyd	25-8-26

```
select DiStinct city
  From student1;
```

S.NO	CITY
1	Delhi
2	hyd
3	mumbai
4	pune

```
select DePt ID, count (*) AS-total-students
  From student1
Group by DePt ID;
```

S.NO	DePt-ID	Total students
1	1	3
2	2	1
3	3	1

```
select DePt ID, count(*) AS total-students
From student1
Group BY DePt ID
Having count(*) >=2;
```

S.NO	DePt-ID	Total-students
1	1	3

VELTECH	
EX No.	3.1
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	-
TOTAL (20)	15
SIGN WITH DATE	[illegible] 25/8/25

Result: The implementation of the clauses operators & functions in the query DDL & DML commands.

TASK (3.2) · AGGREGATE Functions:

Aim: To study & implement aggregate functions (count (), sum (), Avg (), min();
max()) on a sample database.

AGGREGATE FUNCTIONS:

Theyre mostly used with grouped by the rows.

→ count()
→ sum()
→ AVG ()
→ min()
→ max()

```
create table student 2 (
        Roll No int primary key,
        Name varchar (50);
        AGE int,
        Dept ID int,
        marks INT);

insert into student 2 values
(1, "ArJun", 20, 101, 85);
(2, "sneha", 21, 101, 90);
(3, "Ravi", 19, 102, 95);
(4, "priya", 22, 102, 95);
(5, "kiran", 20, 101, 60);
[illegible] 102, 88);
```

```
select* From student 2;
```

	Roll-No	name	Age	Dept ID	marks
1	1	ArJun	20	101	85
2	2	sneha	21	101	90
3	3	Ravi	19	102	70
4	4	priya	22	102	95
5	5	kiran	20	101	80
6	6	Anita	23	103	88

```
select DePt ID , AVG (marks) AS AVG_marks
From student 2
grouped by DePt ID;
```

	Dept Id	TOP mark
1	101	90
2	102	95
3	103	88

```
select DePt ID min (marks) AS least,
mark From student 2
Group by Dept ID;
```

	Dept ID	least mark
1	101	60
2	102	70
3	103	88

```
select DePt ID, AVg (marks) AS AVg-marks
From student2.
grouped by DePt ID;
```

output:

SNO	DePt-ID	AVg-marks
1	101	78
2	102	82
3	103	88

```
select DePt ID, count(*) AS stu-count
From student2,
Group by DePt ID;
```

output:

SNO	DePt-ID	stu-count
1	101	3
2	102	2
3	103	1

VEL TECH	
EX No.	32
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	4
TOTAL (20)	19
SIGN WITH DATE	[illegible] 2/09/25

Result:

implementation of all aggregate functions has been performed successfully on a table.

18/8/25

TASK - 3.1

using clauses, operators and functions in queries:

Aim: To implement of DML commands using clauses, operators and functions in queries.

clauses:

→ where order by, Group by, Having, Distinct.

operators:

→ equal (=)

→ Between

→ AND

→ OR

→ IN

```
create table Department1 (
        DEPT ID  INT primary key,
        Dept Name  varchar(50) unique NotNull;
        location  varchar (50)  Not Null) ;

create table student1 (
        student ID  int  primary key,
        Name - varchar (50)  Not Null,
        Age  int  check (AGE>=18),
        Dept Id  int  Foreign key references
                  Department1 (DEPTID)
        city  varchar(50)  Default 'unknown'.
        Joindate  Datime 'Default GET Date()
```