

## ASSIGNMENT - III

### TITLE :

Design at least 10 SQL queries for suitable database application using SQL DML Statements.

**NAME :** Shinde Shubham Dnyandev,      **DIV :** SY-B,      **ROLL NO. :** 23107121.

### QUERIES :

#### 1) CREATE :

a) CREATE DATABASE PR3;

```
USE PR3;
```

```
CREATE TABLE DEPARTMENT(  
D_ID INT PRIMARY KEY,  
D_NAME VARCHAR(50),  
D_CONTACT BIGINT  
);
```

```
DESC DEPARTMENT;
```

The screenshot shows the MySQL Workbench interface with the 'Result Grid' tab selected. A table named 'DEPARTMENT' is displayed with the following schema:

	Field	Type	Null	Key	Default	Extra
▶	D_ID	int	NO	PRI	NULL	
	D_NAME	varchar(50)	YES		NULL	
	D_CONTACT	bigint	YES		NULL	

On the right side of the interface, there is a vertical toolbar with three icons: 'Result Grid' (selected), 'Form Editor', and 'SQL Editor'.

b) CREATE TABLE STUDENT(

```
S_ID INT PRIMARY KEY,  
NAME VARCHAR(50),  
ROLL_NO INT,  
AGE INT,  
D_ID INT,  
FOREIGN KEY (D_ID) REFERENCES DEPARTMENT(D_ID)  
);  
DESC STUDENT;
```

Result Grid | Filter Rows: Export: Wrap Cell Content: □

	Field	Type	Null	Key	Default	Extra
▶	S_ID	int	NO	PRI	NULL	
	NAME	varchar(50)	YES		NULL	
	ROLL_NO	int	YES		NULL	
	AGE	int	YES		NULL	
	D_ID	int	YES	MUL	NULL	

Result Grid Form Editor Field Types

## 2) INSERT :

### a) INSERT INTO DEPARTMENT VALUES

```
(465, 'AI&DS', 9544646642),
(847, 'COMP', 7656452365),
(364, 'IT', 8766647845),
(763, 'E&TC', 9867765894),
(273, 'MECH', 7768956543);
```

```
SELECT * FROM DEPARTMENT;
```

Result Grid | Filter Rows: Edit: Export/Import: Wr

	D_ID	D_NAME	D_CONTACT
▶	273	MECH	7768956543
	364	IT	8766647845
	465	AI&DS	9544646642
	763	E&TC	9867765894
	847	COMP	7656452365
*	HULL	HULL	HULL

Result Grid Form Editor Field Types

### b) INSERT INTO STUDENT VALUES

```
(6587, 'Ayush', 45, 24, 465),
(4938, 'Vinit', 07, 19, 847),
(3498, 'Aditya', 56, 27, 364),
(9476, 'Vivek', 19, 28, 763),
(3947, 'Yash', 60, 31, 273);
```

```
SELECT * FROM STUDENT;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Write: |

	S_ID	NAME	ROLL_NO	AGE	D_ID
▶	3498	Aditya	56	27	364
	3947	Yash	60	31	273
	4938	Vinit	7	34	847
	6587	Ayush	45	24	465
*	9476	Vivek	19	28	763
	NULL	NULL	NULL	NULL	NULL

### 3) UPDATE :

UPDATE STUDENT

SET AGE = AGE + 15

WHERE S\_ID = 4938;

SELECT \* FROM STUDENT;

Result Grid | Filter Rows: | Edit: | Export/Import: | Write: |

	S_ID	NAME	ROLL_NO	AGE	D_ID
▶	3498	Aditya	56	27	364
	3947	Yash	60	31	273
	4938	Vinit	7	34	847
	6587	Ayush	45	24	465
*	9476	Vivek	19	28	763
	NULL	NULL	NULL	NULL	NULL

### 4) DELETE :

DELETE FROM STUDENT

WHERE S\_ID = 9476;

SELECT \* FROM STUDENT;

Result Grid | Filter Rows: | Edit: | Export/Import: | Write: |

	S_ID	NAME	ROLL_NO	AGE	D_ID
▶	3498	Aditya	56	27	364
	3947	Yash	60	31	273
	4938	Vinit	7	34	847
	6587	Ayush	45	24	465
*	NULL	NULL	NULL	NULL	NULL

## 5) UNION :

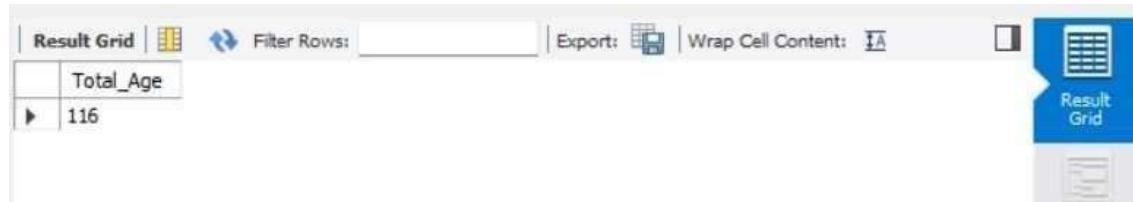
```
SELECT NAME, D_ID  
FROM STUDENT  
WHERE D_ID = 364  
UNION  
SELECT NAME, D_ID  
FROM STUDENT  
WHERE D_ID = 273;
```



	NAME	D_ID
▶	Aditya	364
	Yash	273

## 6) SUM :

```
SELECT SUM(AGE) AS Total_Age FROM STUDENT;
```



	Total_Age
▶	116

## 7) GROUPBY :

```
SELECT S_ID, AVG(AGE) AS Avg_Age  
FROM STUDENT  
GROUP BY S_ID;
```



	S_ID	Avg_Age
▶	3498	27.0000
	3947	31.0000
	4938	34.0000
	6587	24.0000

### 8) MAX & MIN :

```
SELECT MAX(AGE) AS Highest_Age, MIN(AGE) AS Lowest_Age FROM STUDENT;
```

The screenshot shows a database result grid with the following data:

	Highest_Age	Lowest_Age
▶	34	24

### 9) COUNT :

```
SELECT D_ID, COUNT(*) AS Student_Count  
FROM STUDENT  
GROUP BY D_ID;
```

The screenshot shows a database result grid with the following data:

D_ID	Student Count
273	1
364	1
465	1
847	1

### 10) UPDATE (UPPERCASE) :

```
UPDATE STUDENT  
SET NAME = UPPER(NAME)  
WHERE S_ID = 4938;
```

```
SELECT * FROM STUDENT;
```

The screenshot shows a database result grid with the following data:

S_ID	NAME	ROLL_NO	AGE	D_ID
3498	Aditya	56	27	364
3947	Yash	60	31	273
4938	VINIT	7	34	847
6587	Ayush	45	24	465
	HULL	HULL	HULL	HULL

## 11) BITWISE OPERATOR :

```
SELECT * FROM STUDENT  
WHERE NAME = 'Aditya' OR AGE > 20;
```

The screenshot shows a database query results grid titled "Result Grid". The grid has columns: S\_ID, NAME, ROLL\_NO, AGE, and D\_ID. The data is as follows:

	S_ID	NAME	ROLL_NO	AGE	D_ID
▶	3498	Aditya	56	27	364
	3947	Yash	60	31	273
	4938	VINIT	7	34	847
*	6587	Ayush	45	24	465
*	HULL	HULL	HULL	HULL	HULL

## 12) SPECIAL OPERATOR :

a) SELECT \* FROM STUDENT  
WHERE NAME LIKE 'V%';

The screenshot shows a database query results grid titled "Result Grid". The grid has columns: S\_ID, NAME, ROLL\_NO, AGE, and D\_ID. The data is as follows:

	S_ID	NAME	ROLL_NO	AGE	D_ID
▶	4938	VINIT	7	34	847
*	HULL	HULL	HULL	HULL	HULL

b) SELECT \* FROM STUDENT  
WHERE NAME LIKE '%sh';

The screenshot shows a database query results grid titled "Result Grid". The grid has columns: S\_ID, NAME, ROLL\_NO, AGE, and D\_ID. The data is as follows:

	S_ID	NAME	ROLL_NO	AGE	D_ID
▶	3947	Yash	60	31	273
*	6587	Ayush	45	24	465
*	HULL	HULL	HULL	HULL	HULL

### 13) ROUND :

UPDATE STUDENT

```
SET AGE = ROUND(AGE / 20) * 20  
WHERE S_ID = 3498;
```

```
SELECT * FROM STUDENT;
```

The screenshot shows a database result grid with the following data:

S_ID	NAME	ROLL_NO	AGE	D_ID
3498	Aditya	56	40	364
3947	Yash	60	31	273
4938	VINIT	7	34	847
6587	Ayush	45	24	465
*	NULL	NULL	NULL	NULL

### 14) CONCAT :

```
SELECT CONCAT(NAME, ', ', AGE) AS NAME_AND_CONTACT  
FROM STUDENT;
```

The screenshot shows a database result grid with the following data:

NAME_AND_CONTACT
Aditya , 40
Yash , 31
VINIT , 34
Ayush , 24

### 15) IS NULL :

```
SELECT * FROM STUDENT  
WHERE ROLL_NO IS NULL;
```

The screenshot shows a database result grid with the following data:

S_ID	NAME	ROLL_NO	AGE	D_ID
*	NULL	NULL	NULL	NULL