

**BansilalRamnathAgarwalCharitableTrust's**  
**VishwakarmaInstituteofTechnology, Pune-37**

*(AnautonomousInstituteofSavitribaiPhulePuneUniversity)*



**Department of Multidisciplinary**  
**Engineering**

<b>Division</b>	<b>AI &amp; DS – A</b>
<b>Batch</b>	<b>B2</b>
<b>GR-no</b>	<b>12310204</b>
<b>Rollno</b>	<b>26</b>
<b>Name</b>	<b>Jineshwari Anil Bagul</b>

**Title:** Execute DDL statements which demonstrate the use of views. Try to update the base table using its corresponding view. Also consider restrictions on updatable views and perform view creation from multiple tables

**Description:** A view is defined as database object that allows us to create a virtual table in the database whose contents are defined by a query or taken from one or more tables.

**Virtual Table:** A view does not store data itself but provides a way to access data from one or more base tables.

**Abstraction:** Views can simplify data access by abstracting complex queries and presenting the data in a more convenient format.

**Security:** Views can be used to restrict access to specific columns or rows in a table, thereby enhancing security

### Screenshots/Output:

**1) Simple View :** A simple view in SQL is a virtual table based on a SELECT query.

**Syntax :** CREATE VIEW view\_name AS SELECT column1,column2,....  
FROM table\_name;

```
mysql> CREATE VIEW v1 AS SELECT ngo_name,ngo_phone FROM ngo_details;
Query OK, 0 rows affected (0.04 sec)

mysql> SELECT * FROM v1;
+-----+-----+
| ngo_name          | ngo_phone |
+-----+-----+
| Smile Foundation  | 0123456789 |
| Aasha NGO         | 9876543210 |
| Hope Trust        | 1234567890 |
| Sankalp Foundation | 9871234560 |
+-----+-----+
4 rows in set (0.00 sec)
```

**2)Update :** An UPDATE statement in SQL modifies existing records in a table.

**Syntax :** Update view\_name SET column\_name = value WHERE condition;

```
mysql> UPDATE v5 SET item_no = 40 WHERE user_id = "N004";
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

```
mysql> SELECT * FROM v5;
```

user_id	item_no
N001	50
N002	30
N003	50
N004	40

```
4 rows in set (0.00 sec)
```

**3)Simple view with order by clause :** The SQL ORDER BY clause is used to sort the data in ascending or descending order, based on one or more columns.

**Syntax :** **CREATE VIEW** view\_name **AS SELECT** column1,column2,...  
**FROM** table\_name **ORDER BY** column\_name;

```
mysql> CREATE VIEW v2 AS SELECT ngo_id, ngo_name FROM ngo_details ORDER BY ngo_name;
Query OK, 0 rows affected (0.02 sec)
```

```
mysql> SELECT * FROM v2;
```

ngo_id	ngo_name
N002	Aasha NGO
N003	Hope Trust
N005	Sankalp Foundation
N001	Smile Foundation

```
4 rows in set (0.01 sec)
```

**4)Multitable view:**A multitable view in SQL is a virtual table created from a SELECT query that combines data from multiple tables using joins.

**Syntax :** **CREATE VIEW** view\_name **AS SELECT** column1, column2, ...  
**FROM** table, table2 **WHERE** condition;

```
mysql> CREATE VIEW v4 AS
-> SELECT ngo_details.ngo_id, ngo_details.ngo_name, donation_item.item_name, donation_item.item_no
-> FROM ngo_details, donation_item
-> WHERE ngo_details.ngo_id = donation_item.user_id;
Query OK, 0 rows affected (0.01 sec)

mysql> SELECT * FROM v4
-> ;
+-----+-----+-----+-----+
| ngo_id | ngo_name      | item_name | item_no |
+-----+-----+-----+-----+
| N001   | Smile Foundation | clothes   | 50      |
| N002   | Aasha NGO       | toys     | 30      |
| N003   | Hope Trust      | Medicine  | 50      |
| N005   | Sankalp Foundation | Books    | 90      |
+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

**5)Create or Replace :** `CREATE OR REPLACE VIEW` updates an existing view or creates a new one if it doesn't exist, allowing you to change the view's definition without dropping it.

**Syntax :** **CREATE OR REPLACE VIEW** view\_name **AS SELECT** column1, column2, ... **FROM** table\_name **WHERE** condition;

```
mysql> CREATE OR REPLACE VIEW v4 AS
-> SELECT ngo_details.ngo_id, donation_item.item_name, donation_item.item_no
-> FROM ngo_details, donation_item
-> WHERE ngo_details.ngo_id = donation_item.user_id;
Query OK, 0 rows affected (0.02 sec)

mysql> SELECT * FROM v4;
+-----+-----+-----+
| ngo_id | item_name | item_no |
+-----+-----+-----+
| N001   | clothes   | 50      |
| N002   | toys     | 30      |
| N003   | Medicine  | 50      |
| N005   | Books     | 90      |
+-----+-----+-----+
4 rows in set (0.00 sec)
```

**6)Aggregative functions :** in a view allow you to perform calculations like SUM, AVG, COUNT, MIN, and MAX on a set of data within the view.

**Syntax :** **CREATE VIEW** view\_name **AS SELECT** AGGREGATIVE FUNCTIONS(column\_name) **FROM** view\_name;

**Aggregative functions SUM:**

```
mysql> CREATE VIEW sum AS SELECT SUM(item_no) FROM v4;  
Query OK, 0 rows affected (0.02 sec)
```

```
mysql> SELECT * FROM sum;  
+-----+  
| SUM(item_no) |  
+-----+  
|          220 |  
+-----+  
1 row in set (0.02 sec)
```

### Aggregative functions AVG:

```
mysql> CREATE VIEW avg AS SELECT AVG(item_no) FROM v4;  
Query OK, 0 rows affected (0.02 sec)
```

```
mysql> SELECT * FROM avg;  
+-----+  
| AVG(item_no) |  
+-----+  
|          55 |  
+-----+  
1 row in set (0.00 sec)
```

### Aggregative functions COUNT:

```
mysql> CREATE VIEW count AS SELECT COUNT(item_no) FROM v4;  
Query OK, 0 rows affected (0.02 sec)
```

```
mysql> SELECT * FROM count;  
+-----+  
| COUNT(item_no) |  
+-----+  
|          4 |  
+-----+  
1 row in set (0.00 sec)
```

### Aggregative functions MIN:

```
mysql> CREATE VIEW min AS SELECT MIN(item_no) FROM v4;
Query OK, 0 rows affected (0.02 sec)

mysql> SELECT * FROM min;
+-----+
| MIN(item_no) |
+-----+
| 30           |
+-----+
1 row in set (0.01 sec)
```

### Aggregative functions MAX:

```
mysql> CREATE VIEW max AS SELECT MAX(item_no) FROM v4;
Query OK, 0 rows affected (0.02 sec)

mysql> SELECT * FROM max;
+-----+
| MAX(item_no) |
+-----+
| 90           |
+-----+
1 row in set (0.00 sec)
```

**7)Insert view :** You can insert data into an updatable view using the INSERT INTO statement, similar to inserting into a table.

**Syntax :** INSERT INTO view\_name WHERE condition;

```
mysql> INSERT INTO v5 (user_id,item_no) VALUES ("N004","30");
Query OK, 1 row affected (0.01 sec)
```

```
mysql> SELECT * FROM v5;
```

user_id	item_no
N001	50
N002	30
N003	50
N005	90
N004	30

```
5 rows in set (0.00 sec)
```

**8)Deletion from view :** You can delete data from an updatable view using the `DELETE` statement, just like deleting from a table.

**Syntax : DELETE FROM view\_name WHERE condition;**

```
mysql> DELETE FROM v5 WHERE user_id = "N005";
Query OK, 1 row affected (0.01 sec)
```

```
mysql> SELECT * FROM v5;
```

user_id	item_no
N001	50
N002	30
N003	50
N004	30

```
4 rows in set (0.00 sec)
```

**Drop view :** You can delete a view using the DROP VIEW statement.

**Syntax : DROP view\_name;**

```
mysql> DROP VIEW v5;
Query OK, 0 rows affected (0.02 sec)
```