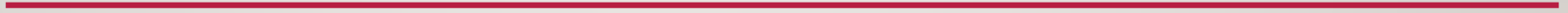


# VIEW



# MYSQL VIEW

---

- A view is a database object that has no values. Its contents are based on the base table. It contains rows and columns similar to the real table. In MySQL, the View is a **virtual table** created by a query by joining one or more tables. It is operated similarly to the base table but does not contain any data of its own. The View and table have one main difference that the views are definitions built on top of other tables (or views). If any changes occur in the underlying table, the same changes reflected in the View also.

# BEST PRACTICES FOR USING VIEWS

---

- 1. Design Views for Specific Use Cases:** Create views that address specific business needs or user requirements.
- 2. Maintain View Consistency:** Update views when the underlying tables change to ensure the view's data remains accurate and up-to-date.
- 3. Monitor View Performance:** Regularly review the performance of views, especially those used in critical business processes.
- 4. Secure Views Appropriately:** Ensure that views are only accessible to authorized users and that they do not expose sensitive data.

# BENEFITS OF DATABASE VIEWS

---

- 1. Data Abstraction:** Views can simplify complex data structures and present a more user-friendly interface to the data.
- 2. Security:** Views can be used to control access to sensitive data by allowing users to see only the data they are authorized to access.
- 3. Performance:** Views can be used to pre-compute and cache frequently used queries, improving overall query performance.

# MYSQL ALLOWS US TO CREATE A VIEW IN MAINLY TWO WAYS:

---

1. MySQL Command line client
2. MySQL Workbench



# SYNTAX

---

1. **CREATE** [OR **REPLACE**] **VIEW** view\_name **AS**
2. **SELECT** columns
3. **FROM** tables
4. [**WHERE** conditions];

MySQL 8.0 Command Line Client

```
mysql> SHOW TABLES;
```

```
+-----+
| Tables_in_testdb |
+-----+
| contact           |
| courses           |
| customer          |
+-----+
```

```
3 rows in set (0.13 sec)
```

```
mysql> CREATE VIEW trainer AS
-> SELECT course_name, trainer
-> FROM courses;
```

```
Query OK, 0 rows affected (0.33 sec)
```

---

DATABASE VIEWS CAN PERFORM A VARIETY OF USEFUL FUNCTIONS. HERE ARE SOME OF THE KEY THINGS THAT VIEWS CAN DO:



# DATA ABSTRACTION

---

- Views can simplify complex data structures by presenting a more user-friendly and intuitive representation of the underlying data.
- This allows users to work with data at a higher level without needing to understand the lower-level table structures.

# CONTROLLED DATA ACCESS

---

- Views can be used to restrict access to sensitive data by exposing only the necessary columns and rows.
- This helps enforce data security and privacy policies within the database.

# QUERY SIMPLIFICATION

---

- Views can encapsulate complex SQL queries, making them easier to understand, maintain, and reuse.
- This can improve the readability and maintainability of database applications.

# PERFORMANCE OPTIMIZATION

---

- Views can be used to pre-compute and cache the results of frequently executed queries.
- This can improve the overall performance of the database by reducing the need to re-execute complex queries.

# DATA CONSOLIDATION

---

- Views can combine data from multiple tables, providing a unified view of the data.
- This can be useful for reporting, analytics, and decision-making purposes.



# LOGICAL DATA INDEPENDENCE:

---

- Changes to the underlying table structures can be hidden behind the view, protecting applications from being affected by these changes.
- This helps maintain the logical data independence of the application from the physical data storage.

# DATA TRANSFORMATION:

---

- Views can be used to transform or calculate data, such as performing calculations, applying business rules, or transforming data formats.
- This can be useful for providing a more meaningful and actionable representation of the data.

- 
- <https://www.javatpoint.com/mysql-view>