| **Feature** | **Temporary Table** | **View** | **Materialized View** | **Common Table Expression (CTE)** |
| --- | --- | --- | --- | --- |
| **Definition** | A table that exists temporarily for the duration of a session or transaction. | A virtual table based on a SELECT query. | A precomputed result set stored physically in the database. | A temporary result set defined within a query. |
| **Persistence** | Exists only during the session or transaction. | Persistent until dropped, not stored physically. | Persisted physically in the database. | Exists only for the duration of the query. |
| **Storage** | Stored in memory (or temp disk) and disappears after session or transaction ends. | No physical storage, it's a stored query. | Physically stored and can be refreshed periodically. | No storage; evaluated at runtime. |
| **Performance** | Can improve performance for repeated complex queries, as it is stored temporarily. | Slow performance for large queries (needs recomputation every time). | Improves performance by storing results physically, avoiding recomputation. | Used to simplify complex queries, but evaluated every time. |
| **Usage** | Ideal for storing intermediate results during a session or for temporary processing. | Good for abstracting complex queries or reusable logic. | Good for frequently used aggregated data or complex calculations that don't change often. | Used to structure queries and make them more readable or reusable within a single query. |
| **Data Modification** | Can insert, update, or delete data (like a regular table). | Cannot modify data directly; only queryable. | Can be refreshed or rebuilt, but direct modification is not allowed. | Cannot modify data; used for query structuring. |
| **Scope** | Limited to the session or transaction. | Available to all sessions (once created). | Available to all sessions (after creation and refresh). | Available only within the query in which it is defined. |
| **Refresh** | No refresh required (disappears automatically). | Always reflects the current state of the underlying data. | Needs to be refreshed manually or on a schedule to reflect changes. | No refresh; re-evaluated every time it's referenced. |
| **Syntax** | CREATE TEMPORARY TABLE | CREATE VIEW | CREATE MATERIALIZED VIEW | WITH CTE AS (SELECT ...) |
| **Example** | CREATE TEMPORARY TABLE temp\_table AS SELECT \* FROM employees; | CREATE VIEW emp\_view AS SELECT \* FROM employees; | CREATE MATERIALIZED VIEW emp\_mv AS SELECT \* FROM employees; | WITH emp\_cte AS (SELECT \* FROM employees) SELECT \* FROM emp\_cte; |

**Key Differences:**

* **Temporary Tables** are useful for temporary storage and modification of data within a session.
* **Views** are virtual tables based on a SELECT query and do not store data physically.
* **Materialized Views** store the result of a query physically, improving performance for frequently accessed data, but need to be refreshed to stay updated.
* **CTEs** are used to simplify complex queries and act as temporary result sets within a single query, with no storage or persistence.