

Digital Career Institute

Python Course - Database - Basic Usage



Views

- In SQL, a **view** is a statement that has been given a name.
- It works like a function. It can be executed later.
- Only **SELECT** statements are used in Views.
- Every time a view is called/executed, the underlying statement is executed.

Define a View

```
CREATE VIEW <name> AS <statement>;
```

```
CREATE VIEW friend_messages AS  
SELECT friends.name, message.text  
FROM friends, message  
WHERE friends.id = message.friend_id;
```

Use a View

```
personal=# SELECT * FROM friend_messages;
      name      |      text
-----+-----
 Lisa Klepp     | How are you doing?
 Maria Schmidt  | Will you come to the party?
 Karen O'Mailey | Have you seen my wallet?
(3 rows)
```

The view returns a temporary table. This table can be used to perform additional queries.

```
personal=# SELECT * FROM friend_messages WHERE text LIKE 'H%';
      name      |      text
-----+-----
 Lisa Klepp     | How are you doing?
 Karen O'Mailey | Have you seen my wallet?
(2 rows)
```

Rename and Remove a View

```
ALTER VIEW [IF EXISTS] friend_messages  
RENAME TO full_name_messages;
```

```
DROP VIEW [IF EXISTS] full_name_messages;
```

Change a View

DLI

```
CREATE OR REPLACE VIEW <name> AS <statement>;
```

```
CREATE OR REPLACE VIEW friend_messages AS  
SELECT friends.name, friends.age, message.text  
FROM friends, message  
WHERE friends.id = message.friend_id;
```

Updatable Views

```
INSERT INTO teenage_friends(name, age)  
VALUES ('Amina', 30);
```

INSERT, **UPDATE** and **DELETE** can be used on a view, only if the view is defined with one single table and the columns modified are present in the view.

```
CREATE OR REPLACE VIEW teenage_friends AS  
SELECT friends.name, friends.age  
FROM friends  
WHERE friends.age BETWEEN 13 AND 19;
```

The new record will be added to the **friends** table. The values inserted do not need to match the query's conditions.

Updatable Views

```
INSERT INTO teenage_friends(name, age)
VALUES ('Amina', 30);
```

The values inserted do not need to match the conditions in the view.

```
CREATE OR REPLACE VIEW teenage_friends AS
SELECT friends.name, friends.age
FROM friends
WHERE friends.age BETWEEN 13 AND 19;
```

Updatable Views

Adding **WITH CHECK OPTION** will require the inserted values to match the conditions in the query defined in the view.

```
CREATE OR REPLACE VIEW teenage_friends AS
SELECT friends.name, friends.age
FROM friends
WHERE friends.age BETWEEN 13 AND 19
WITH CHECK OPTION;
```

```
personal=# INSERT INTO teenage_friends(name, age) VALUES('Amina', 30);
ERROR:  new row violates WITH CHECK OPTION for view "teenage_friends"
DETAIL:  Failing row contains (null, null, null, 30, null, null, 7, Amina).
```

Materialized Views

- A materialized view is a view that has been made persistent by storing its results in a temporary table.
- Subsequent calls to the view, will not process the underlying query, but will return the previously stored data.
- The query will not be executed unless the materialized view is refreshed (re-evaluated).

Define a Materialized View

```
CREATE MATERIALIZED VIEW friend_messages AS  
SELECT friends.name, message.text  
FROM friends, message  
WHERE friends.id = message.friend_id;
```

The usage of a materialized view is the same as a standard view.

Refresh a Materialized View

```
REFRESH MATERIALIZED VIEW friend_messages;
```

Refreshing the materialized view will execute again the query and store the results.

We learned ...

- That a query can be given a name.
- That named queries are called views and can be reused many times.
- That calling a view executes the underlying query.
- That there are special views, who store the results of the query and do not get executed again every time.
- That these views are called materialized views and can be refreshed when required.
- That materialized views can be used to cache complex queries and improve the user experience.

A large group of people, mostly young adults, are posing for a group photo in a room with a projector screen in the background. They are arranged in several rows, with some people sitting on the floor in the front. Many are making peace signs or other celebratory gestures. The text "THANK YOU" is overlaid in large white letters in the center of the image.

THANK YOU

Contact Details
DCI Digital Career Institute gGmbH