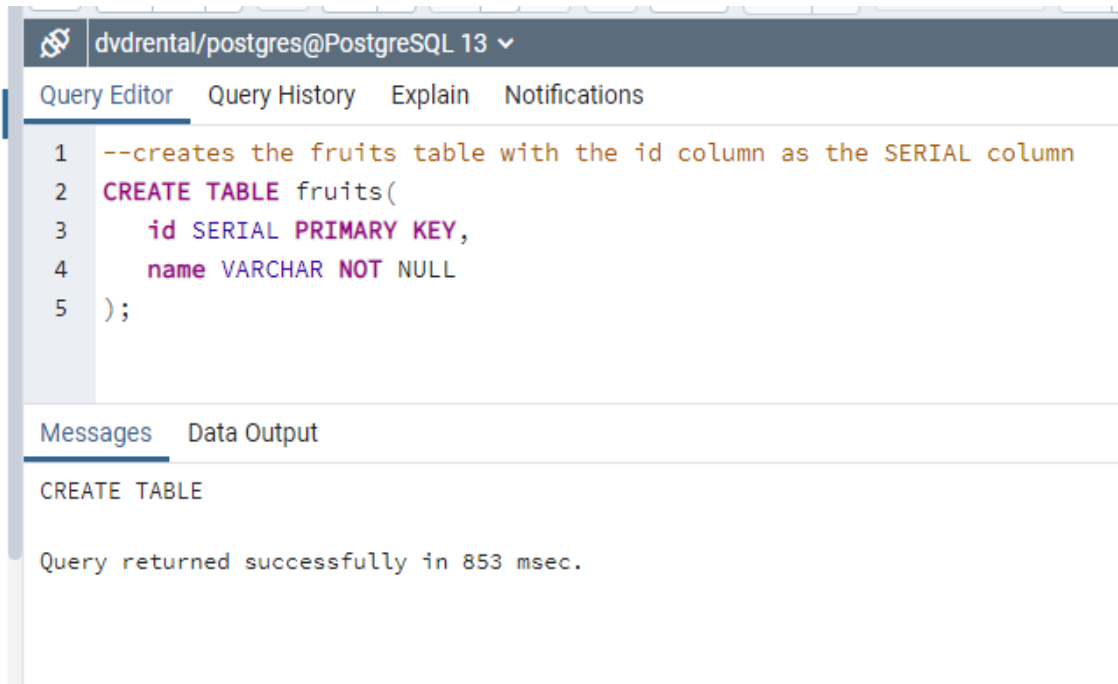


POSTGRESQL AUTO INCREMENT

Using PostgreSQL SERIAL to Create Auto-increment Column



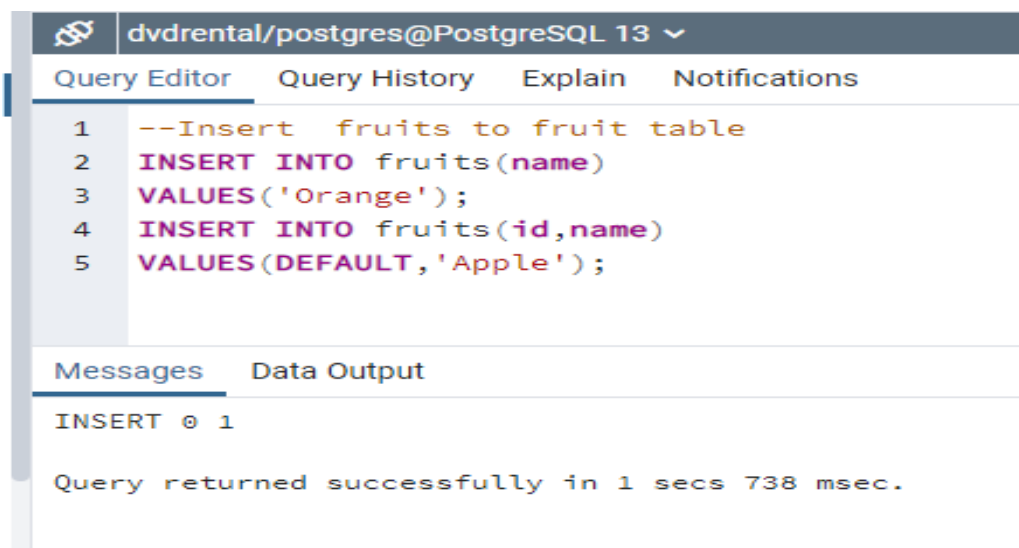
The screenshot shows the PostgreSQL Query Editor interface. The title bar indicates the connection is 'dvdrental/postgres@PostgreSQL 13'. The 'Query Editor' tab is active. The SQL query is as follows:

```
1  --creates the fruits table with the id column as the SERIAL column
2  CREATE TABLE fruits(
3      id SERIAL PRIMARY KEY,
4      name VARCHAR NOT NULL
5  );
```

Below the query editor, the 'Messages' tab is active, showing the output of the query:

```
CREATE TABLE

Query returned successfully in 853 msec.
```



The screenshot shows the PostgreSQL Query Editor interface. The title bar indicates the connection is 'dvdrental/postgres@PostgreSQL 13'. The 'Query Editor' tab is active. The SQL query is as follows:

```
1  --Insert fruits to fruit table
2  INSERT INTO fruits(name)
3  VALUES('Orange');
4  INSERT INTO fruits(id,name)
5  VALUES(DEFAULT,'Apple');
```

Below the query editor, the 'Messages' tab is active, showing the output of the query:

```
INSERT 0 1

Query returned successfully in 1 secs 738 msec.
```

dvdrental/postgres@PostgreSQL 13

Query Editor Query History Explain No

```
1 --View Table
2 SELECT * FROM fruits;
```

Messages Data Output

	id [PK] integer	name character varying
1	1	Orange
2	2	Apple

dvdrental/postgres@PostgreSQL 13

Query Editor Query History Explain Notifications

```
1 -- pass a sequence name to the currval() function to get the recent value generated by the sequence.
2 --For example this query of the recent value generated by the fruits_id_seq object
3 SELECT currval(pg_get_serial_sequence('fruits', 'id'));
```

Messages Data Output

	currval bigint
1	2

dvdrental/postgres@PostgreSQL 13

Query Editor Query History Explain Notifications

```
1 -- insert a new row into the fruits table and returns the value generated for the id column.
2
3 INSERT INTO fruits(name)
4 VALUES('Banana')
5 RETURNING id;
6
7 -- if two concurrent database connections attempt to get the next value from
8 --a sequence, each client will get a different value.
```

Messages Data Output

	id [PK] integer
1	3