SQL QUERY TO COPY, DUPLICATE AND BACKUP TABLE

In relational databases, we often deal with different tables and perform various operations using different database software like MYSQL, Oracle, PostgreSQL, etc.

Sometimes, while performing operations on a table, we might need to create a table backup. We can create a backup of the table by creating its **copy** or **duplicate**. This allows users to maintain the integrity and security of the original database.

We can track changes in data using the backup table when we perform various modification operations. So, in this article, we will learn how to **copy**, **duplicate**, and **backup tables in SQL**. Creating a table copy or duplicate is the same as creating the backup.

Demo Table

We will be using the following table "Student Information" which consists of data as shown below:

Student Information

ID	Age	Student Name	Sex
1	22	Harry	Male
2	23	Vishal	Male
3	20	Snehal	Female
4	25	Ram	Male
5	24	Hina	Female

Create Backup Of A Table

We can create a backup of a table by creating a duplicate or copy of original database.

Syntax

The syntax to create a copy/duplicate/backup of a table in SQL is:

```
CREATE TABLE [backup_table_name] AS SELECT *
FROM [source_table_name];
```

Here:

+ **backup_table_name**: The name of the backup table.

+ **AS**: Aliasing

SQL Copy, Duplicate And Backup Table Examples

Let's look at some examples on how to copy/duplicate table in SQL to create a backup table.

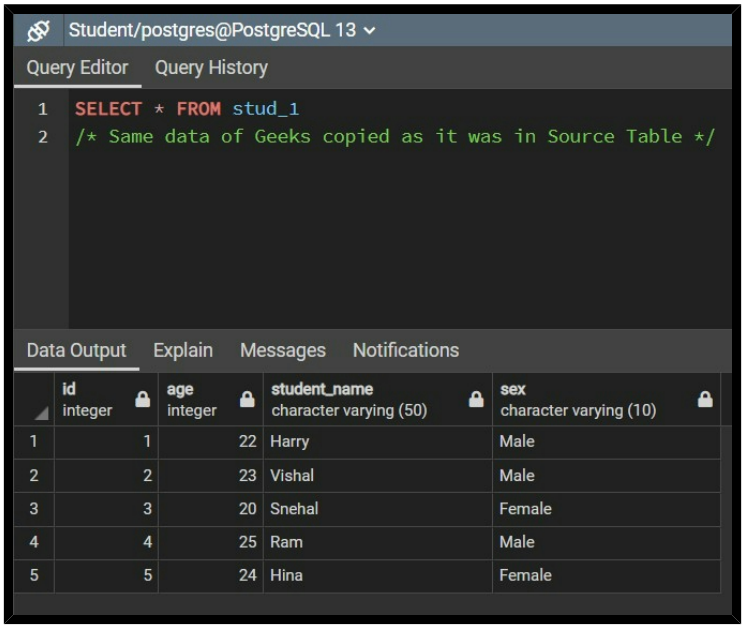
SQL Backup Table with All Columns Data Example

In this example, we will create a backup table"stud_1″ of "student_information" table by creating a copy of "student_information" table and copying its all columns with data.

Query

```
CREATE TABLE stud_1 AS SELECT * FORM student_information;
SELECT * FROM stud_1;
```

Output



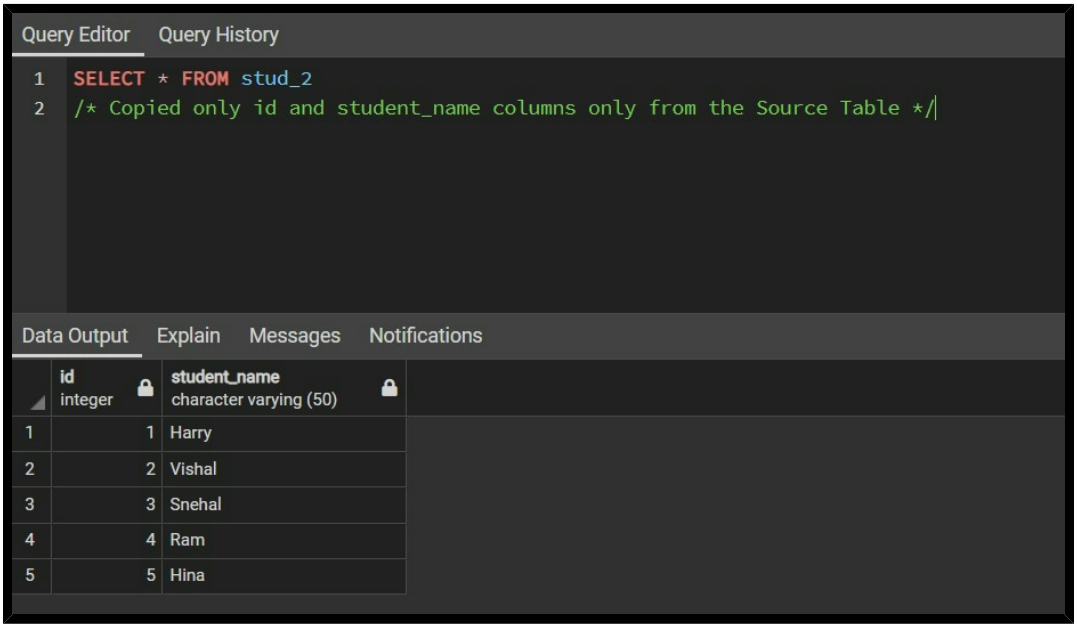
SQL Backup Table With Specific Column Data Example

In this example, we will create a backup table"stud_2″ of "student_information" table by creating a copy of "student_information" table and only copying specific columns with data.

Query

```
CREATE TABLE stud_2 AS
SELECT id, student_name FORM student_information;
SELECT * FROM stud_2;
```

Output



Till now we have seen how to create a clone of the source table. In the above backup table, the data is also copied along with the table. However, we can also create a backup table without copying the data.

SQL Backup Table With No Data Example

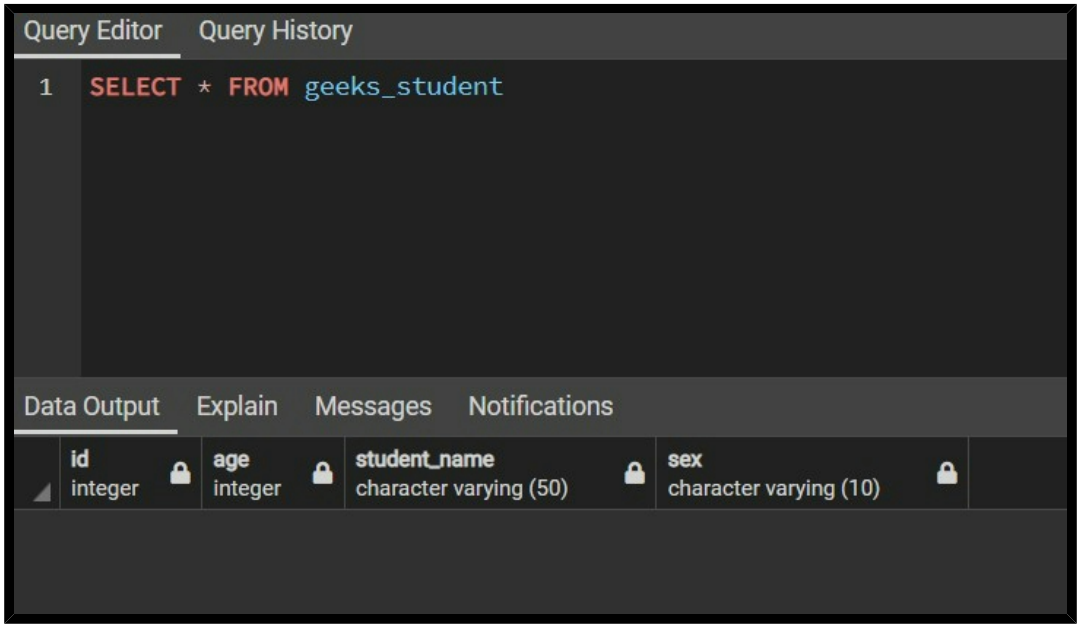
So, to create a table without any data being copied we can use the help of the **WHERE** clause which needs to return a FALSE value. For example, we can use **WHERE 2<2 or WHERE 1=2**.

In this example, we will create a backup table "geeks_student" of "student_information" table by creating a copy of "student_information" table and copying its all columns without data.

Query

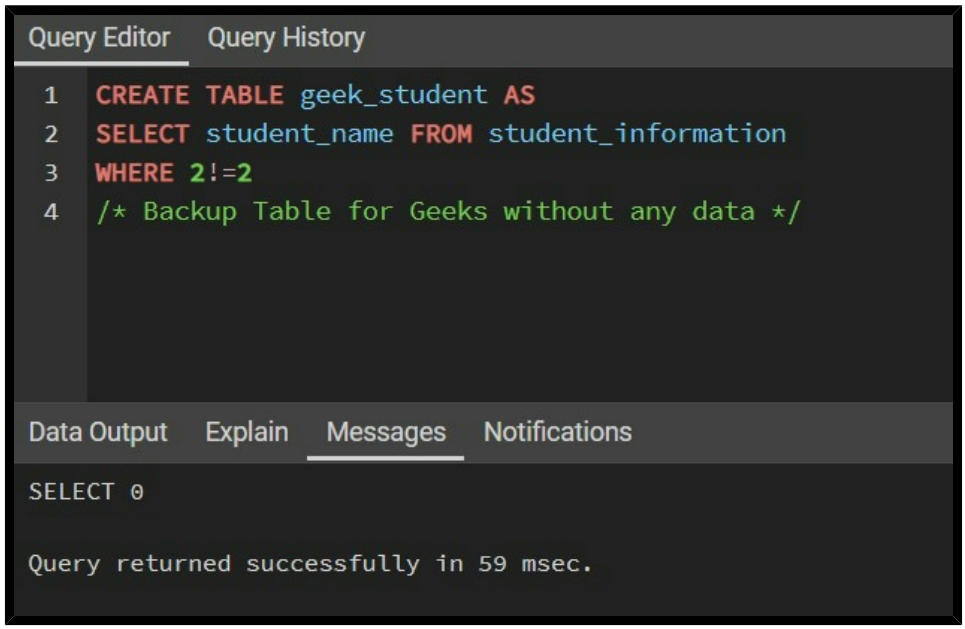
```
CREATE TABLE geeks_student AS
SELECT * FROM student_information;
WHERE 1!=1;
SELECT * FROM stud_2;
```

Output

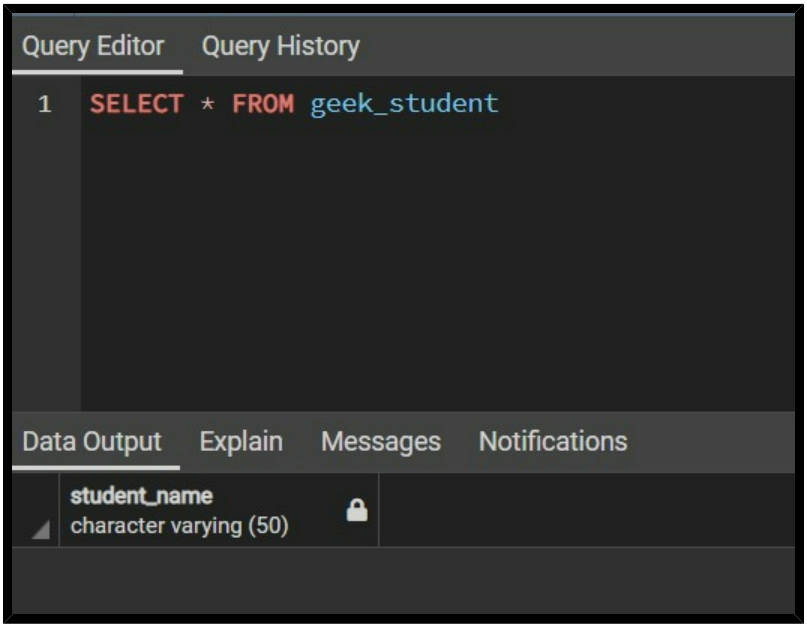


SQL Backup Table With Specific Columns And No Data Example

In this example, we will create a backup table “geek_student” of “student_information” table by creating a copy of “student_information” table and copying specific columns without data.



Output


Query Editor
Query History
1
SELECT * FROM geek_student
Data Output
Explain
Messages
Notifications
student_name
character varying (50)