

[HOME](#)[STORED PROCEDURES](#)[TRIGGERS](#)[VIEWS](#)[INTERFACES](#) ▼[FUNCTIONS](#) ▼

Import CSV File Into PostgreSQL Table

?

Summary: in this tutorial, we will show various ways to import a CSV file into a PostgreSQL table.

We will create a new table named `persons` with the following columns:

`id`: the person id

`first_name`: the first name of the person

`last_name`: the last name of the person

`dob` date of birth of the person

`email`: the email address of the person

We execute the following [CREATE TABLE statement](#) to create the `persons` table:

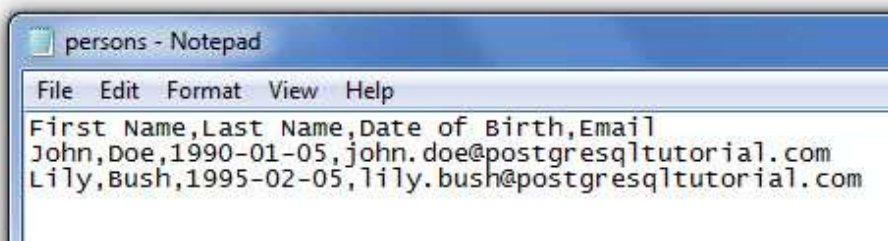
```
1 CREATE TABLE persons
2 (
3   id serial NOT NULL,
4   first_name character varying(50),
5   last_name character varying(50),
6   dob date,
7   email character varying(255),
```

[HOME](#)[STORED PROCEDURES](#)[TRIGGERS](#)[VIEWS](#)[INTERFACES](#) ▾[FUNCTIONS](#) ▾

```
last_name varchar(50)
dob date
email varchar(255)
```

We also prepare a CSV data file by using Microsoft Excel. You can use any applications that available in your PC to do so.

	A	B	C	D
1	First Name	Last Name	Date of Birth	Email
2	John	Doe	1990-01-05	john.doe@postgresqltutorial.com
3	Lily	Bush	1995-02-05	lily.bush@postgresqltutorial.com

 persons

```
persons - Notepad
File Edit Format View Help
First Name,Last Name,Date of Birth,Email
John,Doe,1990-01-05,john.doe@postgresqltutorial.com
Lily,Bush,1995-02-05,lily.bush@postgresqltutorial.com
```

The path of the CSV file is as follows: `C:\tmp\persons.csv`

Import CSV file into table using COPY statement

To import this CSV file into the `persons` table, you use `COPY` statement as follows:

```
1 COPY persons(first_name,last_name,dob,email)
2 FROM 'C:\tmp\persons.csv' DELIMITER ',' CSV HEADER;
```

PostgreSQL gives back the following message:

[HOME](#)[STORED PROCEDURES](#)[TRIGGERS](#)[VIEWS](#)[INTERFACES](#) ▾[FUNCTIONS](#) ▾

Let's check the `persons` table.

```
1 SELECT * FROM persons;
```

id	first_name	last_name	dob	email
1	John	Doe	1990-01-05	john.doe@postgresqltutorial.com
2	Lily	Bush	1995-02-05	lily.bush@postgresqltutorial.com

It works.

Let's dive into the COPY statement in more detail.

```
COPY persons(first_name,last_name,dob,email)
FROM 'C:\tmp\persons.csv' DELIMITER ',' CSV HEADER;
```

First, you specify the table with column names after the `COPY` keyword. The order of the columns must be the same as in the CSV file. In case the CSV file contains all columns of the table, you don't have to specify them explicitly, for example:

```
1 COPY sample_table
2 FROM 'C:\tmp\sample_data.csv' DELIMITER ',' CSV HEADER;
```

Second, you put the CSV file path after the `FROM` keyword. Because CSV file format is used, you need to specify `DELIMITER` as well as `CSV` keywords.

Third, `HEADER` keyword indicates that the CSV file contains a header line with column names. When importing data, PostgreSQL ignores the first line, which is the header line of the file.

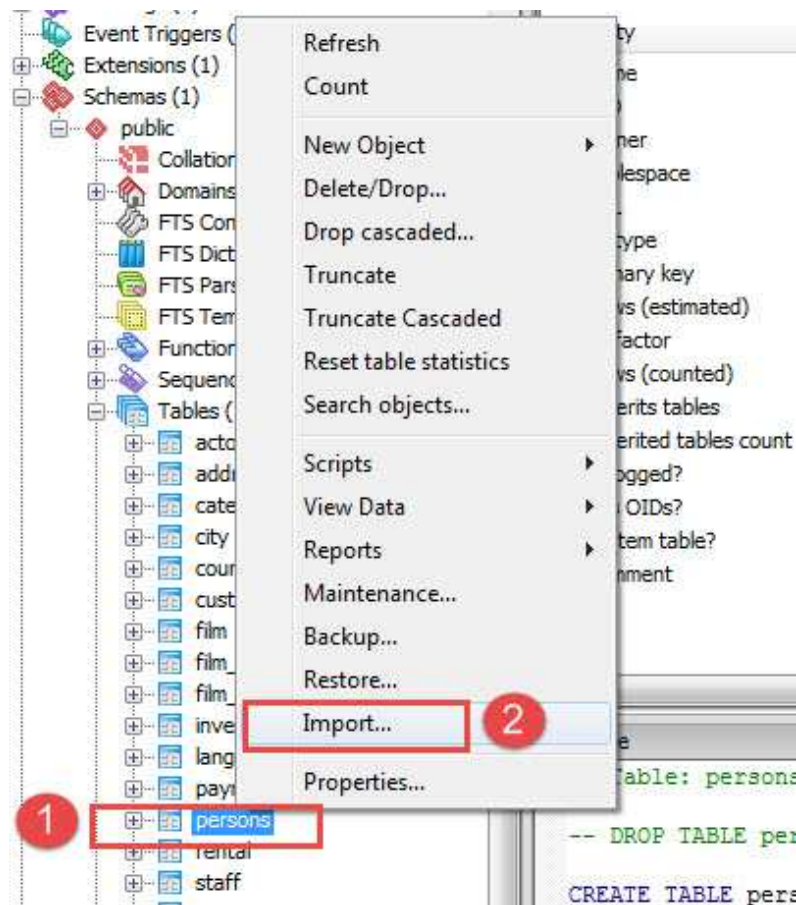
Notice that the file must be read directly by the PostgreSQL server, not by the client application.

Therefore, it must be accessible to the PostgreSQL server machine. Also you can execute COPY statement

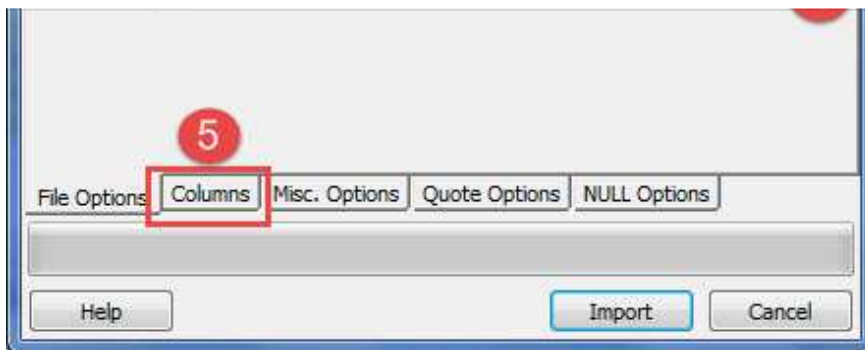
Let's remove all data of the `persons` table so that we can re-import data and see the effect.

```
1 TRUNCATE TABLE persons;
```

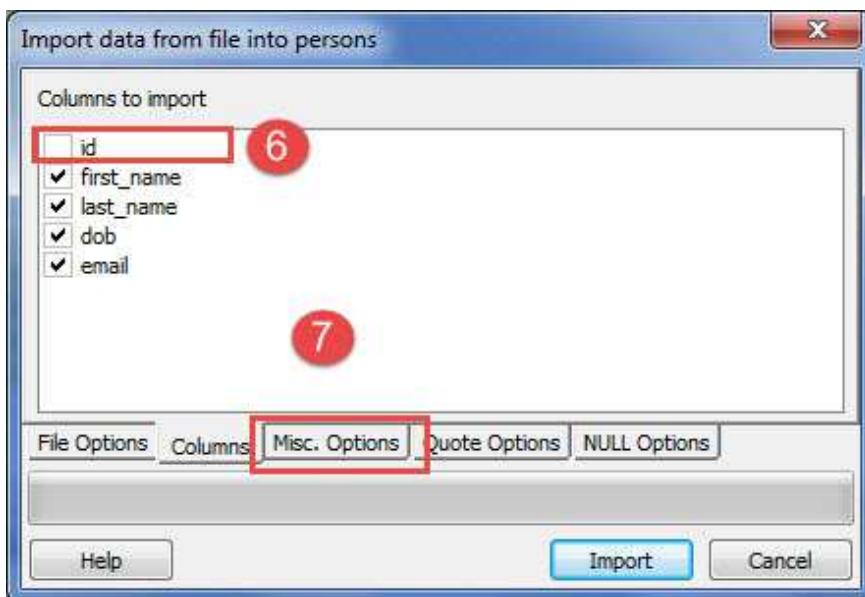
First, right click on the persons table, choose the import... menu item as follows:



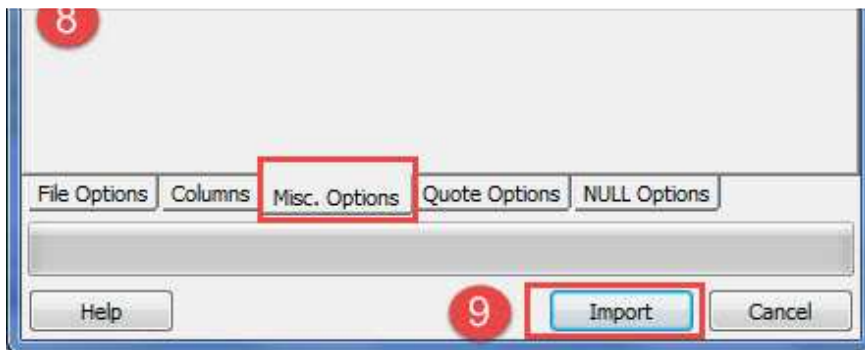
Then, browse the the CSV file, choose format as CSV and click the Columns tab.

[HOME](#)[STORED PROCEDURES](#)[TRIGGERS](#)[VIEWS](#)[INTERFACES ▾](#)[FUNCTIONS ▾](#)

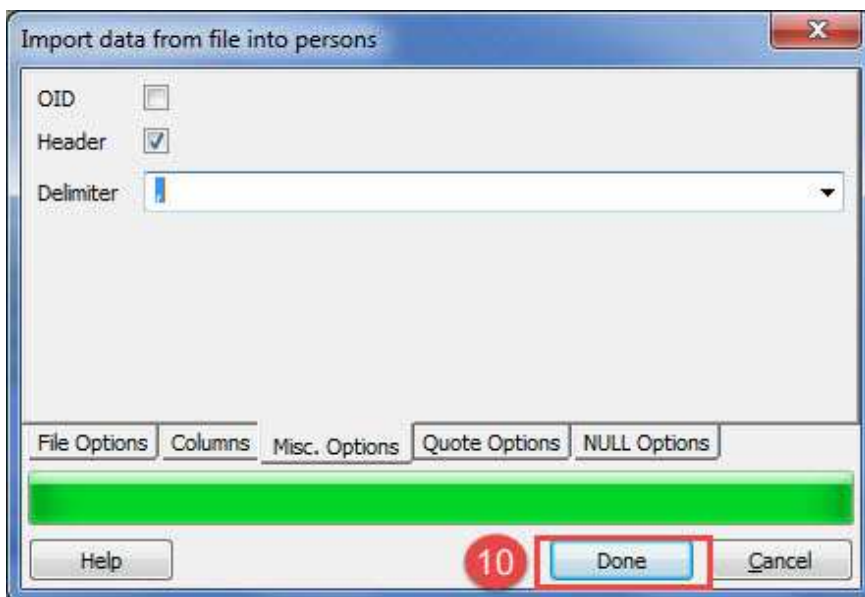
Next, uncheck the id column because we don't import data into this column and click the Misc. Options



After that, check the Header checkbox because our CSV file has a header, choose comma (,) as the delimiter, and click Import button.

[HOME](#)[STORED PROCEDURES](#)[TRIGGERS](#)[VIEWS](#)[INTERFACES](#) ▾[FUNCTIONS](#) ▾

Finally, when the import process completes, click the Done button.



Let's check the `persons` table again to see if the data is really imported.

id	first_name	last_name	dob	email
3	John	Doe	1990-01-05	john.doe@postgresqltutorial.com
4	Lily	Bush	1995-02-05	lily.bush@postgresqltutorial.com

Yes, it is.