

PostgreSQL

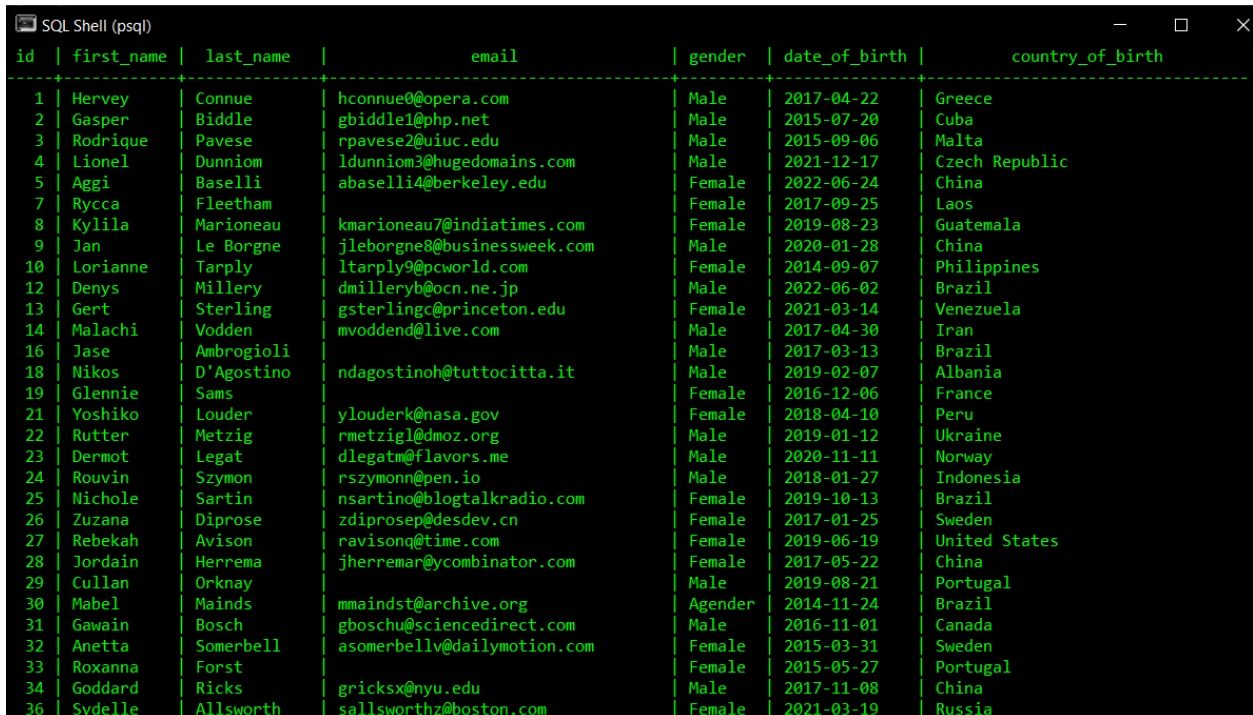
Basic psql commands:

\c db_name => to connect to database;

\d => description of database/table;

\i 'Path' => import data from .sql file;

!\ CIs => clear screen;



The screenshot shows a PostgreSQL SQL Shell window titled "SQL Shell (psql)". It displays a table with 7 columns: id, first_name, last_name, email, gender, date_of_birth, and country_of_birth. The table contains 36 rows of data, each with a unique ID and various personal details.

id	first_name	last_name	email	gender	date_of_birth	country_of_birth
1	Hervey	Connue	hconnue0@opera.com	Male	2017-04-22	Greece
2	Gasper	Biddle	gbiddle1@php.net	Male	2015-07-20	Cuba
3	Rodrique	Pavese	rpavese2@uiuc.edu	Male	2015-09-06	Malta
4	Lionel	Dunniom	ldunniom3@hugedomains.com	Male	2021-12-17	Czech Republic
5	Aggi	Baselli	abaselli4@berkeley.edu	Female	2022-06-24	China
7	Rycca	Fleetham		Female	2017-09-25	Laos
8	Kylila	Marionneau	kmarioneau7@indiatimes.com	Female	2019-08-23	Guatemala
9	Jan	Le Borgne	jleborgne8@businessweek.com	Male	2020-01-28	China
10	Lorianne	Tarply	ltarply9@pcworld.com	Female	2014-09-07	Philippines
12	Denys	Millery	dmilleryb@ocn.ne.jp	Male	2022-06-02	Brazil
13	Gert	Sterling	gsterningc@princeton.edu	Female	2021-03-14	Venezuela
14	Malachi	Vodden	mvoddend@live.com	Male	2017-04-30	Iran
16	Jase	Ambrogio		Male	2017-03-13	Brazil
18	Nikos	D'Agostino	ndagostino@tuttocitta.it	Male	2019-02-07	Albania
19	Glennie	Sams		Female	2016-12-06	France
21	Yoshiko	Louder	ylouderk@nasa.gov	Female	2018-04-10	Peru
22	Rutter	Metzig	rmetzigl@dmoz.org	Male	2019-01-12	Ukraine
23	Dermot	Legat	dlegatm@flavors.me	Male	2020-11-11	Norway
24	Rouvin	Szymon	rszymonn@pen.io	Male	2018-01-27	Indonesia
25	Nichole	Sartin	nsartino@blogtalkradio.com	Female	2019-10-13	Brazil
26	Zuzana	Diprose	zdiprosep@desdev.cn	Female	2017-01-25	Sweden
27	Rebekah	Avison	ravisonq@time.com	Female	2019-06-19	United States
28	Jordain	Herrema	jherremar@ycombinator.com	Female	2017-05-22	China
29	Cullan	Orknay		Male	2019-08-21	Portugal
30	Mabel	Mainds	mmaindst@archive.org	Agender	2014-11-24	Brazil
31	Gawain	Bosch	gboschu@sciencedirect.com	Male	2016-11-01	Canada
32	Anetta	Somerbell	asomerbelly@dailyomotion.com	Female	2015-03-31	Sweden
33	Roxanna	Forst		Female	2015-05-27	Portugal
34	Goddard	Ricks	gricksx@nyu.edu	Male	2017-11-08	China
36	Sydelle	Allsworth	sallsworthz@boston.com	Female	2021-03-19	Russia

Important commands:

All CAPS are postgresql commands and smalls are user-defined variables

- 1) Create database
=> CREATE DATABASE db_name;
- 2) Create a table inside our db
CREATE TABLE person (
 id BIGSERIAL NOT NULL PRIMARY KEY,
 first_name VARCHAR(50) NOT NULL,
 last_name VARCHAR(50) NOT NULL,
 gender VARCHAR(7) NOT NULL,
 email VARCHAR(150),
 date_of_birth Date NOT NULL,
 country_of_birth VARCHAR(50) NOT NULL);

BIGSERIAL => large auto incrementing integer

PRIMARY KEY => is always unique and use to build relation with other tables

NOT NULL => variables with NOT NULL must always be present

VARCHAR => declaration of data types which holds both numbers and letters

3) Insert records into tables

```
INSERT INTO person (id, first_name, last_name, email, gender, date_of_birth,
country_of_birth) VALUES( 'Hervey', 'Connue', 'hconnue0@opera.com', 'Male',
'2017-04-22', 'Greece');
```

INSERT INTO => is a keyword to insert data inside of a tables

4) Order the columns

```
SELECT * FROM person ORDER BY first_name;
```

```
SELECT * FROM person ORDER BY first_name DESC;
```

Note: Default order is in ascending order or ASC can be explicitly defined.

ORDER BY => orders the specified columns either in ascending or descending order

DESC => orders the specified columns in descending order

5) Where clause and AND

```
SELECT * FROM person WHERE gender = 'Male';
```

```
SELECT * FROM person WHERE gender = 'Female' AND country_of_origin = 'China';
```

WHERE => returns the data which contains the specified value

AND => returns the data only when both the value are present

6) Offset, fetch and limit

```
SELECT * FROM person OFFSET 5;
```

```
SELECT * FROM person OFFSET 5 FETCH FIRST 5 ROWS ONLY;
```

```
SELECT * FROM person LIMIT 5;
```

OFFSET => returns data after 5th row

FETCH FIRST 5 ROWS ONLY => returns only the first 5 values

LIMIT => limits rows of data to be returned

7) IN

```
SELECT * FROM person WHERE country_of_birth IN ('Brazil', 'China', 'France')
```

IN => returns the data from the columns which matches the values.

8) Like and iLike

```
SELECT * FROM person WHERE email LIKE '%.com';
```

```
SELECT * FROM person WHERE email ILIKE 'p%';
```

LIKE => It is used for pattern matching it is case sensitive.

% => means anything followed by .com in the above example.

ILIKE => It is also used for pattern matching but is not case sensitive.

9) Group By

```
SELECT country_of_birth, COUNT(*) FROM person GROUP BY country_of_birth
```

COUNT(*) => It is a function which counts the country_of_birth

GROUP BY => group together those rows in a table that have identical data

10) Alter

```
ALTER TABLE person DROP CONSTRAINT person_pkey;
```

```
ALTER TABLE person ADD PRIMARY KEY (id);
```

ALTER => It is used to alter tables in database

DROP CONSTRAINT => It is used to remove constraint from the tables

ADD PRIMARY KEY=> It is used to add a primary key to tables and it can take arrays of columns.

11) Delete

```
DELETE FROM person WHERE id ='2';
```

```
DELETE FROM person WHERE gender ='Male' AND country_of_birth='Nepal';
```

DELETE=> is used to delete data from the database

12) Update

```
UPDATE person SET first_name='Omar', last_name='Montana' WHERE id ='12';
```

UPDATE=> is used to update the tables in the database

SET => it takes an array of values to be updated