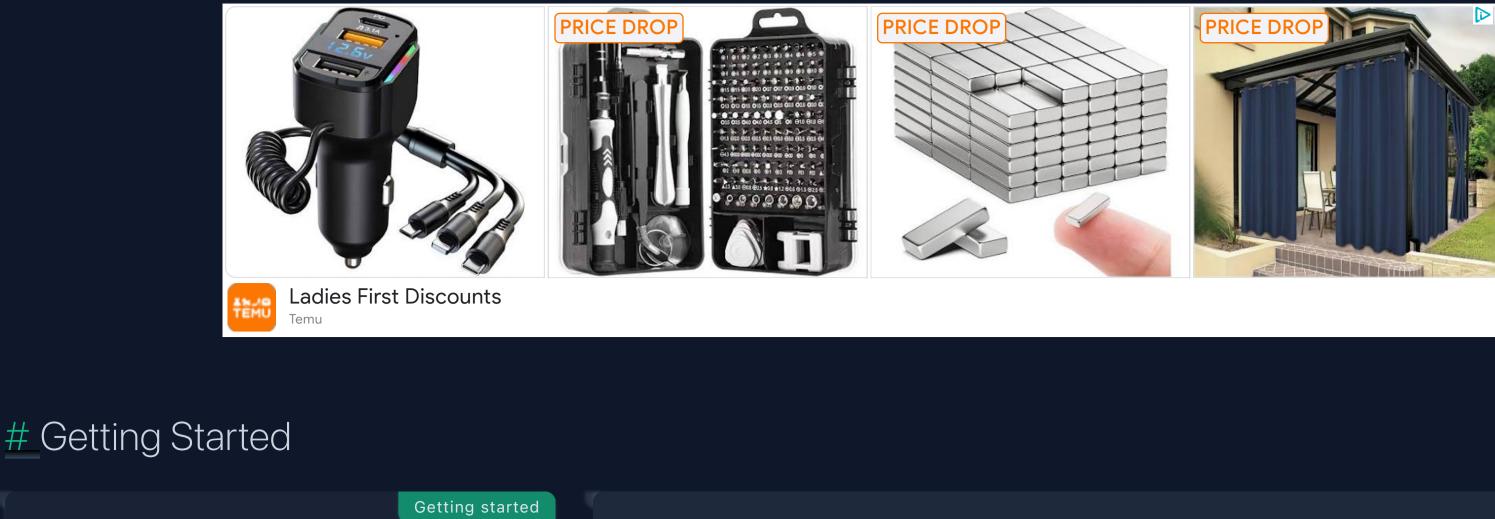


The PostgreSQL cheat sheet provides you with the common PostgreSQL commands and statements.

PostgreSQL cheatsheet

PRICE DROP PRICE DROP



Example

psql -d mydb

Option

psql commands

Connecting to database

Description

sudo -u postgres psql

PostgreSQL Working

Switch and connect

	–U	psql -U john mydb	Connecting as a specific user
List all databases	−h −p	psql -h localhost -p 5432 mydb	Connecting to a host/port
postgres=# \l	–U −h −p −d	psql -U admin -h 192.168.1.5 -p 2506 -d mydb	Connect remote PostgreSQL
Connect to the database named postgres	-W	psql -W mydb	Force password
postgres=# \c postgres	-с	psql -c '\c postgres' -c '\dt'	Execute a SQL query or command
Discourset	-Н	psql -c "\l+" -H postgres > database.html	Generate HTML report
Disconnect		psql -l	List all databases
<pre>postgres=# \q postgres=# \!</pre>	-f	psql mydb -f file.sql	Execute commands from a file
	-V	psql -V	Print the psql version
\h Help on syntax of SQL commands			
\h Help on syntax of SQL commands			
\h DELETE			
\? List of PostgreSQL command			
Run in PostgreSQL console			

Recon	Databases	Tables
Show version	List databases	List tables, in current db
SHOW SERVER_VERSION;	\1	\dt
Show system status	Connect to database	<pre>SELECT table_schema, table_name FROM information_schema.tables ORDER BY</pre>
\conninfo	\c <database_name></database_name>	table_schema,table_name;
Show environmental variables	Show current database	List tables, globally
SHOW ALL;	<pre>SELECT current_database();</pre>	\dt *.*.
List users	Create database	<pre>SELECT * FROM pg_catalog.pg_tables</pre>
<pre>SELECT rolname FROM pg_roles;</pre>	<pre>CREATE DATABASE <database_name> WITH OWNER <username>;</username></database_name></pre>	List table schema
Show current user		<pre>\d <table_name> \d+ <table_name></table_name></table_name></pre>
SELECT current_user;	Drop database	<pre>SELECT column_name, data_type, character_maximum_length</pre>
Show current user's permissions	<pre>DROP DATABASE IF EXISTS <database_name>;</database_name></pre>	FROM INFORMATION_SCHEMA.COLUMNS WHERE table_name = ' <table_name>';</table_name>
\du	Rename database	Create table
Show current database	ALTER DATABASE <old_name> RENAME TO <new_name>;</new_name></old_name>	CREATE TABLE <table_name>(</table_name>
<pre>SELECT current_database();</pre>		<pre><column_name> <column_type>,</column_type></column_name></pre>
Show all tables in database		
\dt		Create table, with an auto-incrementing primary key
List functions		<pre>CREATE TABLE <table_name> (</table_name></pre>
\df <schema></schema>		Delete table
		<pre>DROP TABLE IF EXISTS <table_name> CASCADE;</table_name></pre>
Permissions	Columns	Data
Become the postgres user, if you have permission errors	Add column	Select all data
sudo su – postgres psql	ALTER TABLE <table_name> IF EXISTS ADD <column_name> <data_type></data_type></column_name></table_name>	<pre>SELECT * FROM <table_name>;</table_name></pre>
Grant all permissions on database	[<constraints>];</constraints>	Read one row of data
GRANT ALL PRIVILEGES ON DATABASE	Update column	<pre>SELECT * FROM <table_name> LIMIT 1;</table_name></pre>
<pre><db_name> T0 <user_name>;</user_name></db_name></pre>	ALTER TABLE <table_name> IF EXISTS ALTER <column_name> TYPE <data_type></data_type></column_name></table_name>	Search for data
Grant connection permissions on database	[<constraints>];</constraints>	SELECT * FROM <table_name> WHERE</table_name>

Delete column

KEY;

key

DROP <column_name>;

ALTER TABLE <table_name>

INSERT INTO <table_name>

<column2_name>)

List schemas

SELECT schema_name FROM

pg_catalog.pg_namespace;

SELECT nspname FROM

Create schema

information_schema.schemata;

\dn

Tables

Describe table

Describe table with details

List tables from all schemas

List tables for a schema

List tables from current schema

VALUES (DEFAULT, <value1>);

VALUES (<value1>, <value2>);

ALTER TABLE <table_name> IF EXISTS

Update column to be an auto-incrementing primary key

ADD COLUMN <column_name> SERIAL PRIMARY

Insert into a table, with an auto-incrementing primary

INSERT INTO <table_name> (<column1_name>,

GRANT CONNECT ON DATABASE <db_name> TO <user_name>; Grant permissions on schema GRANT USAGE ON SCHEMA public TO <user_name>; Grant permissions to functions GRANT EXECUTE ON ALL FUNCTIONS IN SCHEMA public TO <user_name>; Grant permissions to select, update, insert, delete, on a all tables TABLES IN SCHEMA public TO <user_name>; Grant permissions, on a table <table_name> TO <user_name>; Grant permissions, to select, on a table GRANT SELECT ON ALL TABLES IN SCHEMA public TO <user_name>; Users List roles

SELECT rolname FROM pg_roles;

CREATE USER <user_name> WITH PASSWORD

DROP USER IF EXISTS <user_name>;

ALTER ROLE <user_name> WITH PASSWORD

PostgreSQL Commands

Create user

Drop user

Alter user password

CREATE SCHEMA IF NOT EXISTS <schema_name>; Drop schema DROP SCHEMA IF EXISTS <schema_name> CASCADE; Edit the query buffer (or file) Edit function definition Show the contents Reset (clear) the query buffer Display history or save it to file Show current time with time zone SELECT current_time; Make dates using integers SELECT make_date(2021,03,25); Informational List all databases List schemas List indexes List roles List sequences

<column_name> = <value>;

<value_1>, <value_2>);

UPDATE <table_name>

WHERE <column_1> = <value>;

DELETE FROM <table_name>;

DELETE FROM <table_name>

Show current date YYYY-MM-DD

Calculate <u>age</u> between two dates

SELECT age(timestamp, timestamp);

SELECT current_date;

WHERE <column_name> = <value>;

Dates

INSERT INTO <table_name> VALUES(

SET <column_1> = <value_1>, <column_2> =

Insert data

Update data

<value_2>

Delete all data

Delete specific data

Schema

Query buffer

\dp 	List table access privileges	\w FILE	Write query buffer to file	\df[antw][S+] 	List functions
\det[+]	List foreign tables			\deu[+]	List user mappings
	Connection		Formatting	\dv[S+]	List views
\c [DBNAME]	Connect to new database	т.	oggle between unaligned and	\dl	List large objects
)	Show or set client	\a 	aligned	\dT[S+]	List data types
\encoding [ENCODING]	encoding	\C [STRING] S	Set table title, or unset if none	\da[S] 	List aggregates
\password [USER]	Change the password	\f [STRING]	Show or set field separator for unaligned	\db[+]	List tablespaces
\conninfo	Display information	\H	Toggle HTML output mode	\dc[S+]	List conversions
		\t [on off]	Show only rows	\dC[+]	List casts
		S	et or unset HTML tag	\ddp 	List default privileges
		\T [STRING]	attributes	\dd[S]	Show object descriptions
		\x [on off]	Toggle expanded output	\dD[S+]	List domains
				\des[+]	List foreign servers
	Input/Output Import/export table	\prompt [TEXT] NAME	Variables Set variable	\dew[+]	List foreign-data wrappers
\copy	See also: copy		Set variable (or list all if no	\dF[+]	List text search configurations
\echo [STRING]	Print string	\set [NAME [VALUE]]	parameters)	\dFd[+]	List text search dictionaries
\i FILE	Execute file	\unset NAME	Delete variable	\dFp[+]	List text search parsers
\o [FILE]	Export all results to file			\dFt[+]	List text search templates
\qecho [STRING]	String to output stream			\dL[S+]	List procedural languages
				\do[S]	List operators
\cd [DIR]	Change the directory		Large Objects	\d0[S+]	List collations
\timing [on off]	Toggle timing	• \lo_export LOBOID F		\drds	List per-database role settings
\! [COMMAND]	Execute in shell	• \lo_import FILE [CO	MMENT]	\dx[+]	List extensions
\! ls -1	List all in shell	• \lo_list			
\: 13 -I	List all III stiell	• \lo_unlink LOBOID		S: show system objects,	+: additional detail
#_Miscellaneous	Backup		Restore		Remote access
Use pg_dumpall to backup a	all databases	Restore a database with pso	ql	Get location of postgres	ql.conf
\$ pg_dumpall -U postg	res > all.sql	\$ psql -U user mydb	< mydb_backup.sql	\$ psql -U postgres	-c 'SHOW config_file'
Use pg_dump to backup a da	atabase	Restore a database with pg	_restore	Append to postgresql.co	onf
<pre>\$ pg_dump -d mydb -f</pre>	mydb_backup.sql	\$ pg_restore -d mydb	mydb_backup.sql -c	listen_addresses =	·*·
–a Dump only the data, r	not the schema	–U Specify a database ι	user	Append to pg_hba.conf postgresql.conf)	(Same location as
-s Dump only the schem	na, no data	–c Drop database befor	re recreating		
–c Drop database before	e recreating	–C Create database bef	ore restoring	host all all 0. host all all ::	0.0.0/0 md5 /0 md5

-e Exit if an error has encountered

sql(default))

-F Format (c: custom, d: directory, t: tar, p: plain text

Use pg_restore -? to get the full list of options

Restart PostgreSQL server

sudo systemctl restart postgresql

QuickRef.ME

Share quick reference and cheat sheet for

o in

Mailchimp

DX

Get suggestions for improving your content, targeting, and marketing

increase revenue.

automations to help you

developers.

中文版 #Notes

OPEN

-C Create database before restoring

Export table into CSV file

Related Cheatsheet

MySQL Cheatsheet

Quick Reference

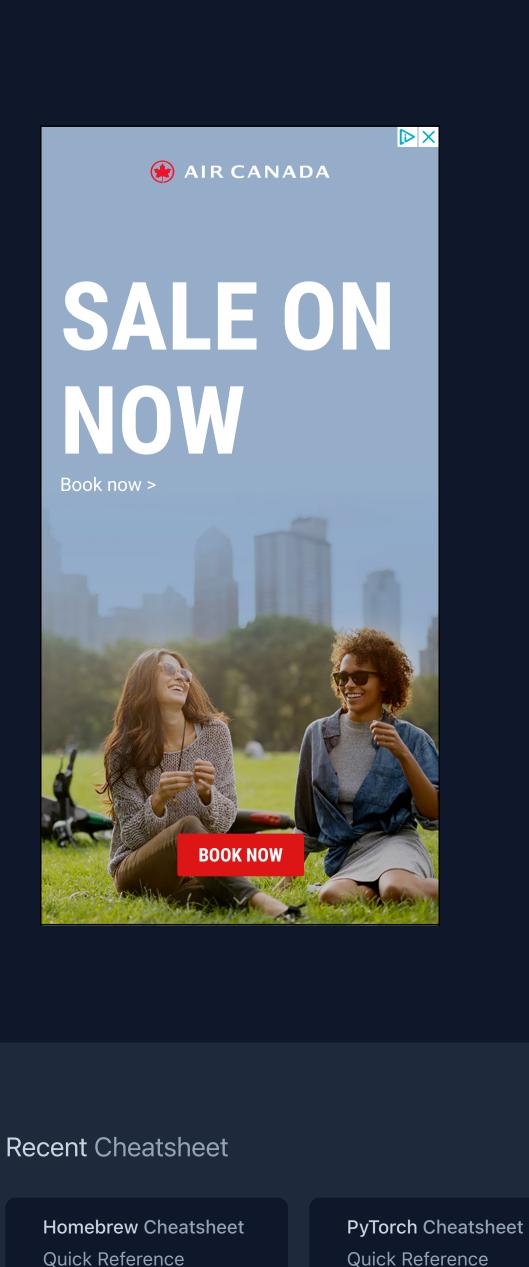
-t Dump the named table(s) only

-F Format (c: custom, d: directory, t: tar)

Use pg_dump -? to get the full list of options

<pre>\copy table TO '<path>' CSV \copy table(col1,col1) TO '<path>' CSV \copy (SELECT) TO '<path>' CSV</path></path></path></pre>	
Import CSV file into table	
<pre>\copy table FROM '<path>' CSV \copy table(col1,col1) FROM '<path>' CSV</path></path></pre>	
See also: <u>Copy</u>	
#_Also see	
Posgres-cheatsheet (gist.github.com)	
	AIR CANADA
	SALE OI
	NOW
	Book now >

Import/Export CSV



MongoDB Cheatsheet

Quick Reference