

Command Line Snippets

Bits and pieces to copy/paste

- [psql](#)
- [robocopy](#)
- [excel](#)
- [azure pipelines](#)
- [git](#)

psql

I always end up looking up the same few things every time I need to interact with psql:

==== Quit ====

```
\q
```

==== Databases ====

== List ==

```
\l
```

== Switch ==

```
\c dbname
```

== Rename ==

```
ALTER DATABASE old RENAME TO new;
```

==== Tables ====

== List ==

```
\dt
```

== Show details ==

```
\d tablename
```

==== Create db with user ====

```
create database mydb;  
create user myuser with encrypted password 'mypass';  
grant all privileges on database mydb to myuser;
```

==== Change user pw ====

```
ALTER USER user_name WITH encrypted password 'new_password';
```

(remove ~/.psql_history after)

==== Drop a user after reassigning privs ====

```
REASSIGN OWNED BY <olduser> TO <newuser>
```

or

```
DROP OWNED BY <olduser>
```

also to be more thorough:

```
REVOKE ALL PRIVILEGES ON ALL TABLES IN SCHEMA public FROM username;  
REVOKE ALL PRIVILEGES ON ALL SEQUENCES IN SCHEMA public FROM username;  
REVOKE ALL PRIVILEGES ON ALL FUNCTIONS IN SCHEMA public FROM username;  
REVOKE ALL PRIVILEGES ON DATABASE database FROM username;  
DROP USER username;
```

<https://stackoverflow.com/questions/3023583/how-to-quickly-drop-a-user-with-existing-privileges>

==== Dump a database ====

```
pg_dump -C -h (ip) -p (port) -U username dbname > dump.sql
```

(omit -C if you want to create the db yourself)

==== Restore a dump ====

```
psql dbname < dump.sql
```

robocopy

copy sourcepath to destpath recursively (/e), log to x:\logs.log.txt (/log), allow resuming large files (/z), and use 16 threads (/mt)

```
robocopy [sourcepath] [destpath] /tee /log:x:\logs\log.txt /z /e /mt:16
```

excel

split by delimiter and take just one part

```
=TAKE(DROP(TEXTSPLIT([@FullName],"\"), 0, 6), 1, 1)
```

- Split the target into a table where each column is a path segment
- Drop the first 6 columns
- Take a single (top leftmost) cell from that table

azure pipelines

```
{{tag>Azure_DevOps}}
```

Snippets for quickly getting a tree of some path:

Look at artifact staging

```
...
```

```
- powershell: |
```

```
  Get-ChildItem -Path $(Build.ArtifactStagingDirectory) -Recurse | Select-Object -ExpandProperty  
  FullName
```

```
  displayName: "Diagnostic: Tree of artifact staging dir."
```

```
  condition: succeededOrFailed()
```

```
...
```

Or everything

```
...
```

```
- powershell: |
```

```
  Get-ChildItem -Path $(Pipeline.Workspace) -Recurse | Select-Object -ExpandProperty FullName
```

```
  displayName: "Diagnostic: Tree of pipeline workspace"
```

```
  condition: succeededOrFailed()
```

```
...
```

Remember to add `condition: succeededOrFailed()` to the step that publishes the artifact, too.

git

find all commits touching a file

today I had some changes to a file sitting on an old branch from a couple months ago, and I wasn't sure *which* branch. [stackoverflow user BigMiner shared this](#), which worked great:

“ I have been looking at this closely and all these answers don't seem to really show me all the commits across all the branches.

Here is what I have come up with by messing around with the [gitk](#) edit view options. This shows me **all the commits for a file** regardless of branch, local, reflog, and remote.

```
gitk --all --first-parent --remotes --reflog --author-date-order -- filename  
Code language: PHP (php)
```

It also works with `git log`:

```
git log --all --first-parent --remotes --reflog --author-date-order --  
filename
```