Cheatography

Random Short Cuts Cheat Sheet

by monkeymynd via cheatography.com/55007/cs/15867/

Chrome

Ctrl + W	Close tab
Alt + P	Show/Hide Ruler

Visual Code

Format Document	Shift + Alt + F
Bookmarks: Toggle	Ctrl + Alt + K
Bookmarks: Next	Ctrl + Alt + L
Bookmarks: Previous	Ctrl + Alt + J
Go To Symbol	Ctrl + Shift + O [:] to group

Git

See non-staged (non-added) changes to existing files

```
git diff
```

See staged, non-commited changes

```
git diff --cached
```

See differences between local changes and master

```
git diff origin/master
```

See differences between two commits

```
git diff COMMIT1_ID COMMIT2_ID
```

See the files changed between two commits

```
git diff --name-only COMMIT1_ID
COMMIT2_ID
```

See the files changed in a specific commit

```
git diff-tree --no-commit-id --name-only -r
COMMIT_ID
or
git show --pretty="format:" --name-only
COMMIT_ID
```

See diff before push

```
git diff --cached origin/master
```

Git (cont)

See details (log message, text diff) of a commit

```
git show COMMIT_ID
```

Revert to the moment before one commit

```
# reset the index to the desired tree
git reset 56e05fced
# move the branch pointer back to the
previous HEAD
git reset --soft HEAD@{1}
git commit -m "Revert to 56e05fced"
# Update working copy to reflect the new
commit
git reset --hard
```

Undo last commit, preserving local changes

```
git reset --soft HEAD~1
```

Undo last commit, without preserving local changes

```
git reset --hard HEAD~1
```

Undo last commit, preserving local changes in index

```
git reset --mixed HEAD~
or
git reset HEAD~1
```

Undo non-pushed commits

```
git reset origin/master
```

Reset to remote state

```
git fetch origin
git reset --hard origin/master
```

Make some changes, create a patch

```
git diff > patch-issue-1.patch
```

Add a file and create a patch

```
git diff --staged > patch-issue-2.patch
```

Add a file, make some changes, and create a patch

```
git add newfile
git diff HEAD > patch-issue-2.patch
```

Make a patch for a commit

```
git format-patch COMMIT_ID
```

Make patches for the last two commits

```
git format-patch HEAD~2
```

Git (cont)

Make patches for all non-pushed commits

```
git format-patch origin/master
```

Apply a patch

```
git apply -v patch-name.patch
```

Apply a patch created using format-patch

```
git am patch1.patch
```

Show differences ignoring new and deleted files

```
git diff --diff-filter=M
```

By **monkeymynd**

cheatography.com/monkeymynd/

Not published yet.
Last updated 1st June, 2018.