# diff and patch Cheat Sheet

### **diff**

diff is used to find differences between two files. On its own, it’s a bit hard to use; instead, use it with diff -u to find lines which differ in two files:

### **diff -u**

diff -u is used to compare two files, line by line, and have the differing lines compared side-by-side in the same output. See below:

~$ cat menu1.txt

Menu1:

Apples

Bananas

Oranges

Pears

~$ cat menu2.txt

Menu:

Apples

Bananas

Grapes

Strawberries

~$ diff -u menu1.txt menu2.txt

--- menu1.txt   2019-12-16 18:46:13.794879924 +0900

+++ menu2.txt   2019-12-16 18:46:42.090995670 +0900

@@ -1,6 +1,6 @@

-Menu1:

+Menu:

Apples

 Bananas

-Oranges

-Pears

+Grapes

+Strawberries

### **Patch**

Patch is useful for applying file differences. See the below example, which compares two files. The comparison is saved as a .diff file, which is then patched to the original file!

~$ cat hello\_world.txt

Hello World

~$ cat hello\_world\_long.txt

Hello World

It's a wonderful day!

~$ diff -u hello\_world.txt hello\_world\_long.txt

--- hello\_world.txt     2019-12-16 19:24:12.556102821 +0900

+++ hello\_world\_long.txt        2019-12-16 19:24:38.944207773 +0900

@@ -1 +1,3 @@

Hello World

+

+It's a wonderful day!

~$ diff -u hello\_world.txt hello\_world\_long.txt > hello\_world.diff

~$ patch < hello\_world.diff

patching file hello\_world.txt

~$ cat hello\_world.txt

Hello World

It's a wonderful day!

There are some other interesting patch and diff commands such as patch -p1, diff -r !

Check them out in the following references:

* [http://man7.org/linux/man-pages/man1/diff.1.html](http://man7.org/linux/man-pages/man1/diff.1.html" \o "" \t "https://www.coursera.org/learn/introduction-git-github/supplement/cKIs3/_blank)
* [http://man7.org/linux/man-pages/man1/patch.1.html](http://man7.org/linux/man-pages/man1/patch.1.html" \o "" \t "https://www.coursera.org/learn/introduction-git-github/supplement/cKIs3/_blank)

| **Command** | **Explanation & Link** |
| --- | --- |
| git commit -a | [Stages files automatically](https://git-scm.com/docs/git-commit" \l "Documentation/git-commit.txt---all" \o "" \t "https://www.coursera.org/learn/introduction-git-github/supplement/39ZMi/_blank) |
| git log -p | [Produces patch text](https://git-scm.com/docs/git-log" \l "_generating_patch_text_with_p" \o "" \t "https://www.coursera.org/learn/introduction-git-github/supplement/39ZMi/_blank) |
| git show | [Shows various objects](https://git-scm.com/docs/git-show" \o "" \t "https://www.coursera.org/learn/introduction-git-github/supplement/39ZMi/_blank) |
| git diff | [Is similar to the Linux `diff` command, and can show the differences in various commits](https://git-scm.com/docs/git-diff" \o "" \t "https://www.coursera.org/learn/introduction-git-github/supplement/39ZMi/_blank) |
| git diff --staged | [An alias to --cached, this will show all staged files compared to the named commit](https://git-scm.com/docs/git-diff" \o "" \t "https://www.coursera.org/learn/introduction-git-github/supplement/39ZMi/_blank) |
| git add -p | [Allows a user to interactively review patches to add to the current commit](https://git-scm.com/docs/git-add" \o "" \t "https://www.coursera.org/learn/introduction-git-github/supplement/39ZMi/_blank) |
| git mv | [Similar to the Linux `mv` command, this moves a file](https://git-scm.com/docs/git-mv" \o "" \t "https://www.coursera.org/learn/introduction-git-github/supplement/39ZMi/_blank) |
| git rm | [Similar to the Linux `rm` command, this deletes, or removes a file](https://git-scm.com/docs/git-rm" \o "" \t "https://www.coursera.org/learn/introduction-git-github/supplement/39ZMi/_blank) |

There are many useful git cheatsheets online as well. Please take some time to research and study a few, such as [this one](https://github.github.com/training-kit/downloads/github-git-cheat-sheet.pdf" \o "" \t "https://www.coursera.org/learn/introduction-git-github/supplement/39ZMi/_blank).

### **.gitignore files**

.gitignore files are used to tell the git tool to intentionally ignore some files in a given Git repository. For example, this can be useful for configuration files or metadata files that a user may not want to check into the master branch. Check out more at: [https://git-scm.com/docs/gitignore](https://git-scm.com/docs/gitignore" \o "" \t "https://www.coursera.org/learn/introduction-git-github/supplement/39ZMi/_blank).

A few common examples of file patterns to exclude can be found [here](https://gist.github.com/octocat/9257657" \o "" \t "https://www.coursera.org/learn/introduction-git-github/supplement/39ZMi/_blank).