Git workshop reference sheet

- *param* = required parameter
- [*param*] = optional parameter
- [flag] = optional flag (no italics)
- [*param…*] = one or more optional parameters

Basic command-line navigation

Command(s)	Parameters	Description	Examples
cd *location*	*location* – a path relative to the current location	Changes the current location of the shell	cd .. # go back one directory cd ~ # go to your user directory cd MyRepo # go to the MyRepo folder
ls [-l] [-a] [*pattern*]	*pattern* – A glob-matching pattern	Prints a list of all folders and files at the current location of the shell	ls # lists all folders, files in current directory ls C* # lists all folders, files starting with "C" ls -la # list format; includes hidden files and folders in listing
touch *name*	*name* – name of file to create	Makes a file with the specified name in the current directory	touch users.txt # makes a file named users.txt
rm *file* rm -r *folder*	*file* – name of file to delete *folder* – name of folder to delete	Deletes the file or folder with the specified name	rm dist/scripts-old.js # removes scripts-old.js rm -r dist # removes dist folder rm -rf . # removes every item in current directory (WARNING: dangerous!!)

Git command reference

- **Note:** all commands below are preceded with "git"

Command	Parameters	Description	Examples
clone *url*	*url* – Git URL of the repo	Makes a local copy of the specified repository	git clone https://github.com/twitter/pants # makes a copy of the Twitter Pants repository
status [-s]	(none)	Prints information about the repository's status	git status git status -s # display status in short form
diff	(none)	Shows differences between working directory and staging area	git diff
add *f1* [*f2 …*]	f1, f2… – pathspecs	Adds the specified paths to the staging area	git add new-index.html # adds the specified file git add test/*.js # adds all .js files within the test folder
rm [--cached] *f1* [*f2 …*]	f1, f2… – pathspecs	Removes the specified paths from the staging area	git rm –cached new-index.html # removes the specified file
commit –m "*message*"	message – summary of changes	Creates a commit containing all specified changes	git commit –m "Added tests" git commit –m "Removed unnecessary helper function"
log	(none)	Shows commit logs	git log
push	(none)	Pushes the changes to the remote repo	git push
pull	(none)	Retrieves and applies the most recent change(s) from the remote repo	git pull

Other commands: **git stash [apply], git reset [--hard], git revert *SHA*, git checkout *SHA-or-branch-name***