

90DaysOfDevOps

Git cheatsheet

Git is a Free and Open Source Software (FOSS) software. It is a Distributed Version Control System(DVCS). It was a project started by Linus Torvalds - also the founder of The Linux Kernel. GitLab and GitHub are two most popular web-hosting platforms for Git repositories of which GitLab is again FOSS while GitHub is a subsidiary of Microsoft and is closed-source, proprietary.

Installation on Linux

Debian-based	<code>sudo apt install git</code>
Arch-based	<code>sudo pacman -S git</code>
Fedora-based	<code>sudo dnf install git</code>

SETUP

- set global username

```
git config --global user.name
```

- set global user.email

```
git config --global user.email
```

- for more

```
man git-config
```

INITIALIZING A REPOSITORY

- Setup version control to an existing directory

DIFFERENCES

- Show difference between commits

```
git diff <commit1> <commit2>
```

- Diff of staged but not committed files

```
git diff --staged
```

BRANCHING

- Create a new branch (Do not switch to it)

```
git branch <newBranchName>
```

- Create a new branch (Switch to it)

```
git init
```

- Get remote repository on local machine

```
git clone <url.git>
```

- *Fork a repository from someone else's remote repository to your remote repository then use git clone to work locally.*

STATUS and TRACKING

- Check status of repo - show **untracked** or **modified** files (edits, adds, deletes, etc)

```
git status
```

- Add files to tracking stage

```
git add <file1> <file2> ..
```

- Commit your staged content with a message for future reference

```
git commit -m "some message"
```

- Stop tracking certain files

```
git rm <file1> <file2> ...
```

```
git checkout -b <newBranchName>
```

- Switch to a branch and checkout in working directory

```
git branch <branchName>
```

- List all the branches

```
git branch -a
```

MERGE BRANCHES

- Merge two branches

```
git merge <branchToMerge>
```

- Print detailed logs

```
git log <branchName>
```

- Print oneliner logs

```
git log --oneline <branchName>
```

- Merge if changes have been pushed

```
git rebase <branchName>
```

REMOTE

- Connect remote repo to local repo for the first time

PUSH/PULL CHANGES WITH REMOTE REPOSITORY

- push changes to a branch

```
git push -u origin <branch>
```

- fetch and pull remote commits from the tracking branch

```
git pull
```

TEMPORARY COMMITS

- Save staged and modified changes before switching to other branch

```
git stash
```

- List stash file changes in LIFO order

```
git stash list
```

- Bring back the changes stashed

```
git stash pop
```

```
git remote add origin <url>
```

- Change the url of an existing remote repo

```
git remote set-url origin <url>
```
