Git Cheatsheet

This cheatsheet covers all the commands from the workshop and a couple extra commands too!

Setup & Initialization

Initialize a local `git` repository in the current directory

```
git init
```

Change configuration settings

```
git config
```

`--global`	Change config. settings globally
`user.name [name]`	Set username
`user.email [email]`	Set email ID

Staging & Snapshot

Work with the staging area and snapshots

```
git add
```

`[filename]`	Add a file to the staging area
`.`	Can be used in place of filename to add all files in the current directory
`-A | -all`	Used in place of the above params to add all files, including subdirectories, to the staging area

Change configuration settings

```
git commit
```

`-m`	Add commit message

Branch Workflow

Manage branches in a repository

```
git branch
```

	Show all branches in the local repository
`-D [branch-name]`	Force delete a branch with name `branch-name`
`branch-name`	Create a new branch with the given name

Combine changes from a branch with current branch

```
git merge
```

`branch-name`	Branch name to fetch the changes from

Created by: Avaneesh Shetye and Faizan Feroz
(I.I.T. Bombay)
Last updated: 17/Jun/2023
Access the latest updated PDF:
https://www.ee.iitb.ac.in/%7Ebelur/foss/git/cheatsheet.pdf

Switch between branches

```
git checkout
```

`branch-name`	Branch name to switch to

Repository Monitoring

Provides information about the repository

```
git status
```

Display commit history

```
git log
```

Remote Repository

Add a remote repository

```
git remote add
```

`remote-name`	Name of the remote repository
`remote-url`	URL of the remote repository

Download a repository to the folder

```
git clone
```

`repo-url`	The repository URL to clone

Retrieve latest changes from a remote repository but NOT download

```
git fetch
```

`repo-url`	The repository URL to fetch (not required if remote branch has already been added)

Combination of `fetch` and `merge`

```
git pull
```

`branch`	The branch name to pull from
`repo-url`	The repository URL to download from

Send local commits to a remote repository

```
git push
```

`-u`	Set the upstream branch as the one provided below
`remote-name`	The remote repository to upload to
`rem-branch-name`	The remote branch to upload to

Instructions

1. Initialize a local repository

   a. Open the terminal in the folder you want to create the local repository in

   b. Type `git init`

   c. Set global configuration variables by:

      - `git config --global user.name ’username’`
      - `git config --global user.email ’user@email.com’`

2. Add the files to your local repository

   a. Type `git add -A` to add all files to the staging area.

   b. Type `git commit -m ’enter commit message here’` to save the files to your local repository. Make sure to enter a message that is helpful and describes what changes you have made.

3. Generate SSH keys for secure remote repository access

   a. Type the command `ssh-keygen -t rsa -b 4096 -C ’user@email.com’` This command generates an SSH key that uses the RSA encryption, creates a key of size 4096 bytes, and the email is added to the end of the key.

   b. Type `eval $(ssh-agent -s)` for Linux or `eval "$(ssh-agent -s)"` on Windows.

   c. Type `ssh-add ~/.ssh/id_rsa` to add the SSH keys to the above agent. **(Please type this command on your own instead of copy-paste, as it contains special characters)**

   d. Type `cat ~/.ssh/id_rsa.pub` to print your public key. Copy the printed text. **(Please type this command on your own instead of copy-paste, as it contains special characters)**

   e. Open your browser, and open your SSH keys in GitHub settings.

   f. Click on New SSH key, and give it a unique title. Paste the text into the Key field and click Add SSH key.

4. Pushing code to a remote repository

   a. Navigate to a GitHub repository that you own or are a contributor to.

   b. Click on the green button labelled Code, click on SSH and copy the refspec (URL).

   c. Type `git remote add <remote-name> <URL>` in the git bash, replacing `<remote-name>` with a name you want for the remote repository, and `<URL>` with the URL copied in the previous step.

   d. Type `git push -u <remote-name> <branch-name>`, replacing `remote-name` with the name you gave above, and the branch name you want to push your code to.

   e. **NOTE:** In case the push fails, use `git status` and `git branch` to ensure there aren’t any uncommitted changes, and the branch you are working in, in your local repository, is the same as the branch in the remote repository.

Updated PDF can be found here: https://www.ee.iitb.ac.in/%7Ebelur/foss/git/cheatsheet.pdf