

# Cheat sheet: Git (#1)

## Setting up user data (once per account)

```
git config --global user.name YOUR_NAME
git config --global user.email YOUR_EMAIL
```

## Setting up defaults (once per account)

```
git config --global core.editor YOUR_EDITOR
git config --global diff.tool YOUR_DIFFTOOL
git config --global init.defaultBranch main
```

## Listing git configuration

```
git config --list
```

```
Git config file      ~/.gitconfig
Git ignore file      PROJECT_DIR/.gitignore
```

## Initializing repo (once per project)

```
git init
git -m main          (if not main already)
```

## Getting project status

```
git status
```

## Adding files to stage (from working dir.)

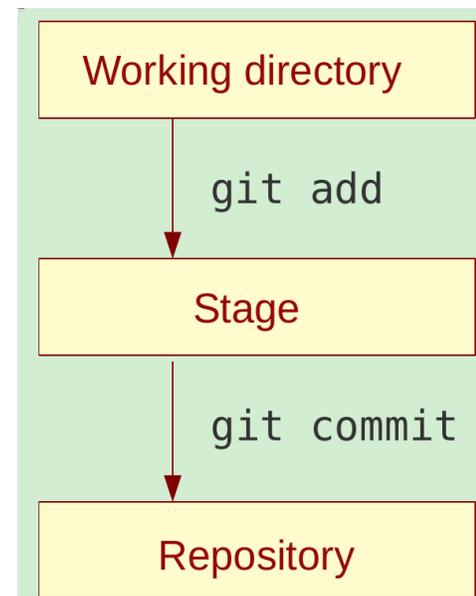
```
git add FILENAME
```

## Commit files to repository (from stage)

```
git commit          Editor opens for log msg.
git commit -m "YOUR_LOG_MESSAGE"
```

## View commit history

```
git log
git log --oneline
```



# Cheat sheet: Git (#2)

## Renaming files / directories

```
git mv OLD NEW
```

## Deleting files / directories

```
git rm FILE_OR_DIR
```

## Investigating differences

```
git diff  
git diff FILE_OR_DIR  
git diff COMMIT_1 COMMIT_2 -- FILE_OR_DIR  
git difftool ...
```

## Discarding changes in ...

```
workdir  git restore FILE_OR_DIR  
stage    git restore --staged FILE_OR_DIR
```

## Switching to an earlier commit (detached head)

```
git switch --detach COMMIT
```

## Switching to most recent commit (attached head)

```
git switch BRANCH
```

## Tagging a version (annotated tag)

```
git tag -a TAG
```

## Defining aliases for git commands

```
git config --global alias.ALIAS COMMAND
```

## Tracking remote repositories

```
git clone REPOSITORY_URL  
git pull
```