Git Cheat Sheet

For more amazing cheat sheets visit inventive.io!

INVENTIVE

Create a Repository

From scratch -- Create a new local repository
```
$ git init {project name}
```

Download from an existing repository
```
$ git clone {your url}
```

Watch your Repository

List new or modified files not yet committed
```
$ git status
```

Show the changes to files not yet staged
```
$ git diff
```

Show the changes to staged files
```
$ git diff --cached
```

Show all staged and unstaged file changes
```
$ git diff HEAD
```

Show the changes between two commit ids
```
$ git diff {commit1} {commit2}
```

List the change datest and authors for a file
```
$ git blame {file}
```

Show the file changes for a commit id and/or file
```
$ git show {commit} : {file}
```

Show full change history
```
$ git log
```

Show change history for file/directory including diffs
```
$ git log -p {file/directory}
```

Working with Branches

List all local branches
```
$ git branch
```

List all branches, local and remote
```
$ git branch -av
```

Switch to a branch, {your-branch}
```
$ git branch {your-branch}
```

Create a new branch called {your-branch}
```
$ git branch {your-branch}
```

Delete the branch called {your-branch}
```
$ git branch -d {your-branch}
```

Merge branch-a into branch-b
```
$ git checkout branch-b
$ get merge branch-a
```

Tag the current commit
```
$ git tag my-tag
```

Synchronize

Get the latest changes from origin (no merge)
```
$ git fetch
```

Fetch the latest changes from origin and merge
```
$ git pull
```

Fetch the latest changes from origin and rebase
```
$ git pull --rebase
```

Push local changes to the origin
```
$ git push
```

Make a Change

Stages the file, ready for commit
```
$ git add {file}
```

Stage all changed files, ready for commit
```
$ git add .
```

Commit all staged files to versioned history
```
$ git commit -m "message"
```

Commit all your tracked files to versioned history
```
$ git commit -am "message"
```

Unstages file, keeping the file changes
```
$ git reset {file}
```

Revert everything to the last commit
```
$ git reset --hard
```

Finally!

When in doubt, use git help
$ git command --help

Or visit https://training.github.com/ for official GitHub training.