

Git workflows

Start a **feature**, work on it, **finish** it.

*Git flow merges **feature** back into **develop***

Work on **develop**, start a **release**, **finish** it

*Git flow merges **release** into **master** and **develop***

Start a **hotfix**, work on it, **finish** it.

*Git flow merges **hotfix** into **master** and **develop***

Initialise a new repo

Normal

```
$ mkgitrepo -n name
```

Puppet

```
$ mkgitrepo -p -n name
```

Normal will create your remote repo in /opt /git/name

Puppet will create your remote repo in /opt /git/ puppet /modules/name

Clone an existing repository

Normal

```
$ git clone git@pv:iaa.sfu.jno.v02:/opt/git/name.git
```

Puppet

```
$ git clone git@pv:iaa.sfu.jno.v02:/trunked/files. If you create new files you need to use git add *
```

Locations for cloning repositories or modules

Branch checking

```
$ git branch -a
```

```
* develop
```

```
master
```

```
remote s/o rig in/ develop
```

```
remote s/o rig in/ master
```

Shows the current remote a local braches

Switching Branches

```
git checkout branchname
```

Switches to a new branch. In git flow you will want to be working on the develop branch or features branch.

Checking git transactions

Command Line

```
$ git log
```

Terminal Interface

```
$ tig
```

Status Checking

```
$ git status
```

Shows the status of the current directory

Committing

Adding new files

```
$ git add *
```

Committing changes

```
$ git commit -a -m "Change description"
```

```
$ git commit -a -m "Change description"
```

git commit -a will only commit already

tracked files. If you create new files you

need to use git add *

Git flow Features branch

Creating Features

```
$ git flow feature start "FeatureName"
```

Publish feature to remote

```
$ git flow feature publish "FeatureName"
```

Get a published feature

```
$ git flow feature pull origin "FeatureName"
```

Finishing a Feature

```
$ git flow feature finish "FeatureName"
```

Features are useful for large repositories with multiple files, or repositories where multiple people will be working at the same time.

Git flow Hotfix branch

Creating a Hotfix

```
$ git flow Hotfix start "Hotfix"
```

Publish Hotfix to remote

```
$ git flow Hotfix publish "Hotfix"
```

Get a published hotfix

```
$ git flow hotfix pull origin "Hotfix"
```

Finishing a Hotfix

```
$ git flow hotfix finish "Hotfix"
```

The hotfix branch is used to create fixes to the master branch, and then merge them back into the develop branch.

Git flow Release branch

Creating a Release

```
$ git flow Release start "Release"
```

Publish Release to remote

```
$ git flow Release publish "Release"
```

Get a published release

```
$ git flow release pull origin "Release"
```

```
$ git flow release pull origin "Release"
```

Finishing a Release

```
$ git flow release finish "Release"
```

The release branch is used to merge the *develop* branch into *master*

Pushing changes

```
$ git push --all
```

Pushes your changes to the remote repository

```
$ git push --all
```

```
$ git push --all
```

```
$ git push --all
```

```
$ git push --all
```

```
$ git push --all
```

```
$ git push --all
```

```
$ git push --all
```

```
$ git push --all
```

```
$ git push --all
```

```
$ git push --all
```

```
$ git push --all
```

```
$ git push --all
```

```
$ git push --all
```

```
$ git push --all
```

```
$ git push --all
```

```
$ git push --all
```

```
$ git push --all
```

```
$ git push --all
```

```
$ git push --all
```

```
$ git push --all
```

```
$ git push --all
```

```
$ git push --all
```

```
$ git push --all
```



By glisignoli

cheatography.com/glisignoli/

Not published yet.

Last updated 11th May, 2016.

Page 1 of 1.

Sponsored by CrosswordCheats.com

Learn to solve cryptic crosswords!

<http://crosswordcheats.com>