

NOT SO LONG AGO.  
IN A GALAXY CLOSE BY...

HEY Bengt!

WHAT'S UP?

I ACCIDENTILY  
DELETED ANOTHER  
PAGE OF MY  
dagbok

NOT THIS STUPID  
'SUN BATTLE' THING  
AGAIN...

IT'S NOT STUPID!  
- YOUR STUPID!

OH WELL...  
YOU HAD IT ALL  
UNDER VERSION  
CONTROL RIGHT?

VERSION CON-WHAT?

UGH...

# Crush course in Version Control

---

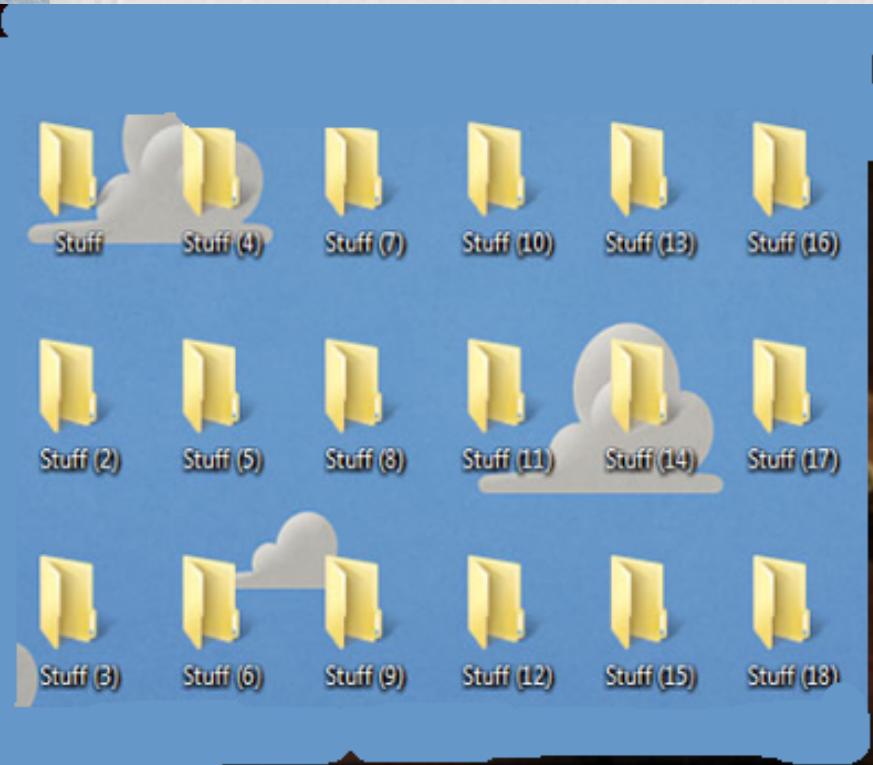
*Roderick Karlemstrand  
feat. Bengt Molin*



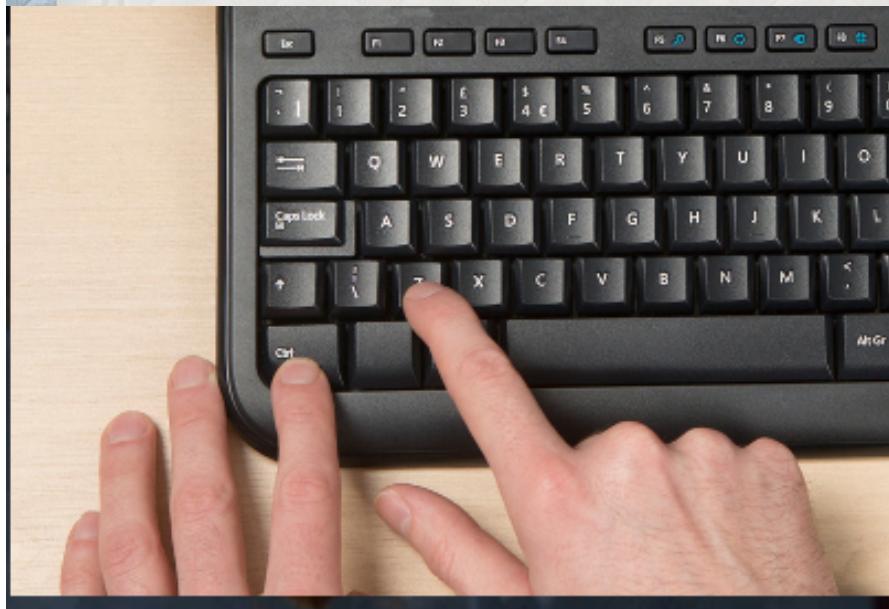
Google Drive



I prefer the real Version Control



No, the real Version Control





ORACLE®

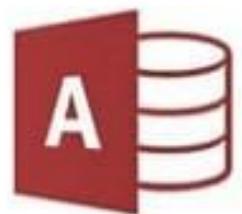
PostgreSQL



MySQL

SYBASE

IBM DB2.



Microsoft®  
Access



Excel

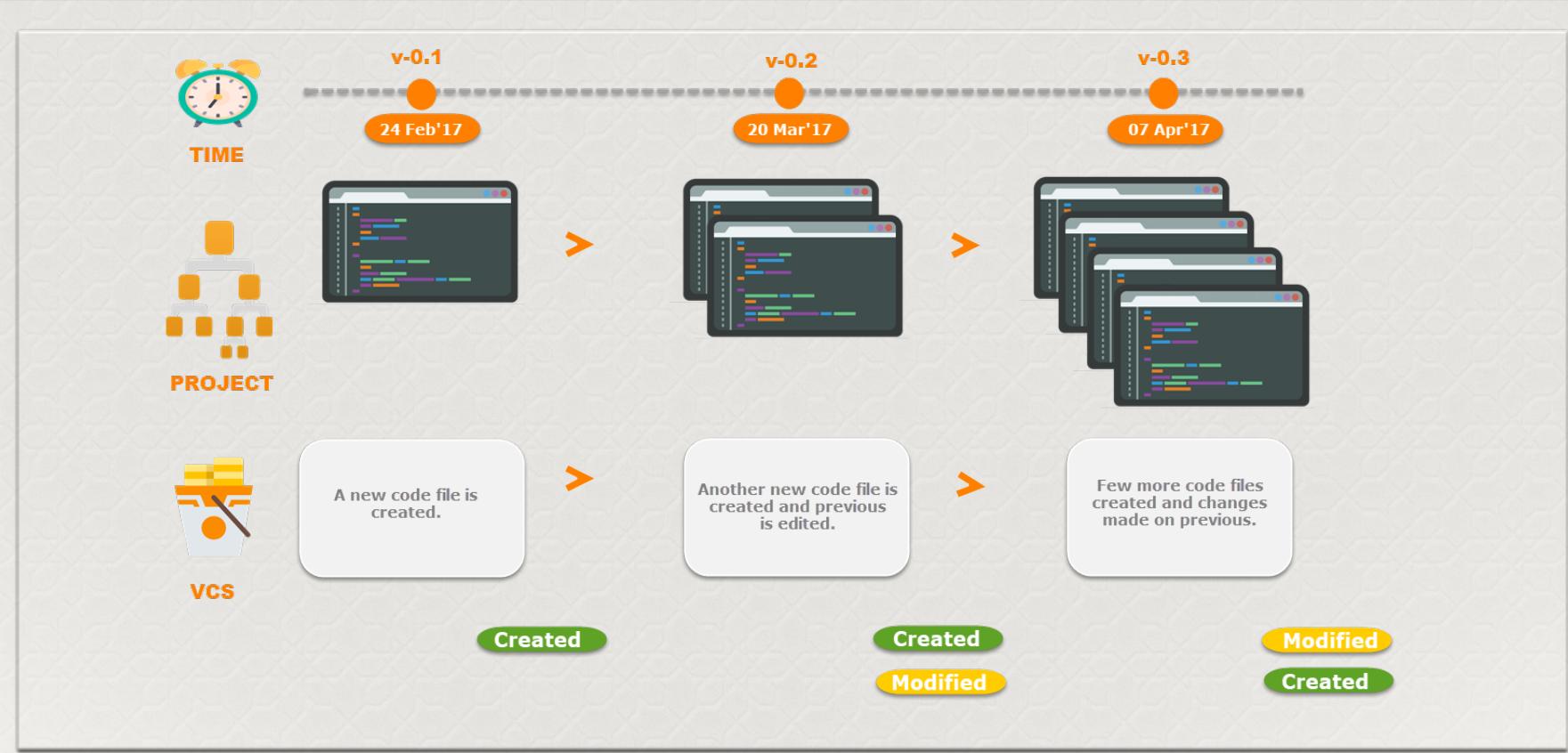


# *AGENDA*

- ◆ *Version Control*
- ◆ *Git Basics*
- ◆ *Github*
- ◆ *Git workflow*
- ◆ *Intermediate Git*
- ◆ *Advanced Git (skipped)*

# Version Control

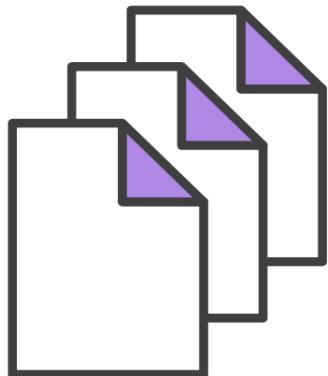
*Tracks changes in tracked files  
Keeps an eye on untracked files*



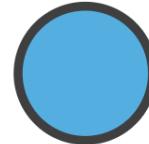
# Git Basics

*Like cooking in a super narrow kitchen*

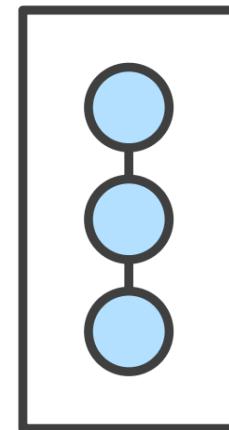
The main components of a Git repository



Working  
Directory



Staged  
Snapshot



Commit  
History

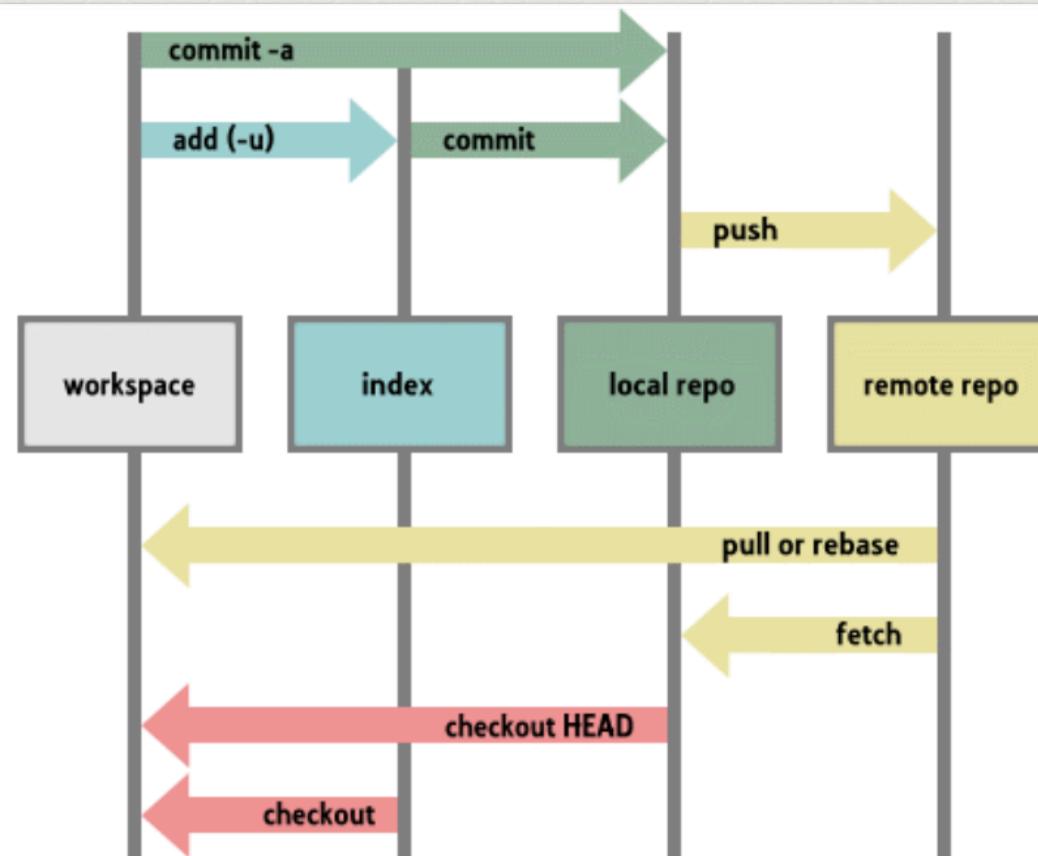
# Git Basics

*HEAD*

*Staging area*

*Local git-repo*

*Remote/Origin git-repo*



# Git Basics

*\*git config <username, email>*

*git clone <repo>*

*git checkout -b <new-branch>*

*git add <file>*

*\*git fetch origin*

*git pull origin <branch>*

*git commit -m <message>*

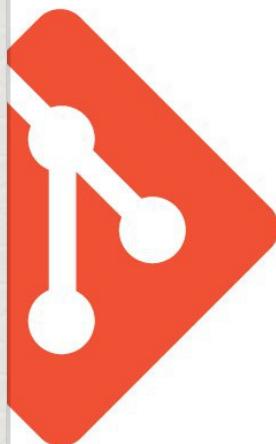
*git push origin <branch>*

*git checkout <file>*

*git checkout <branch>*

*git reset <commit>*

*git reset <file>*



git

*Demo*

# Github

*Remote repo*

*Organise projects and  
members*

*Code difference*

*Pull request*

*Discussion*

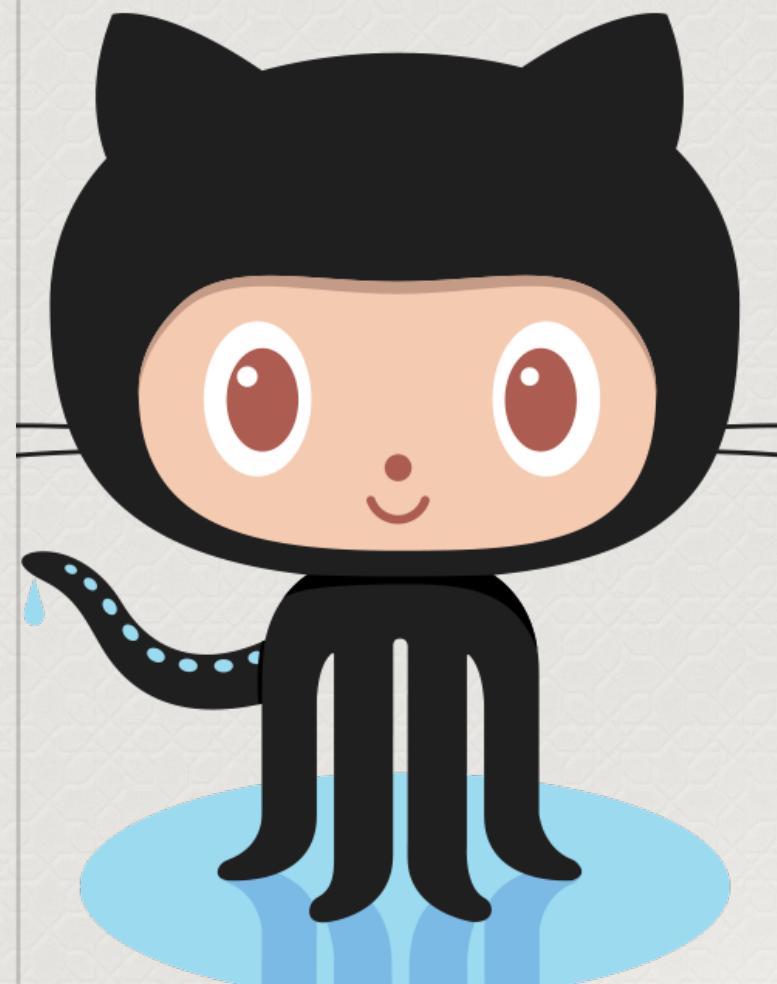
*Better history chart*

*Track bugs*

*Automated deployment*

*Test coverage*

.....



*Demo*

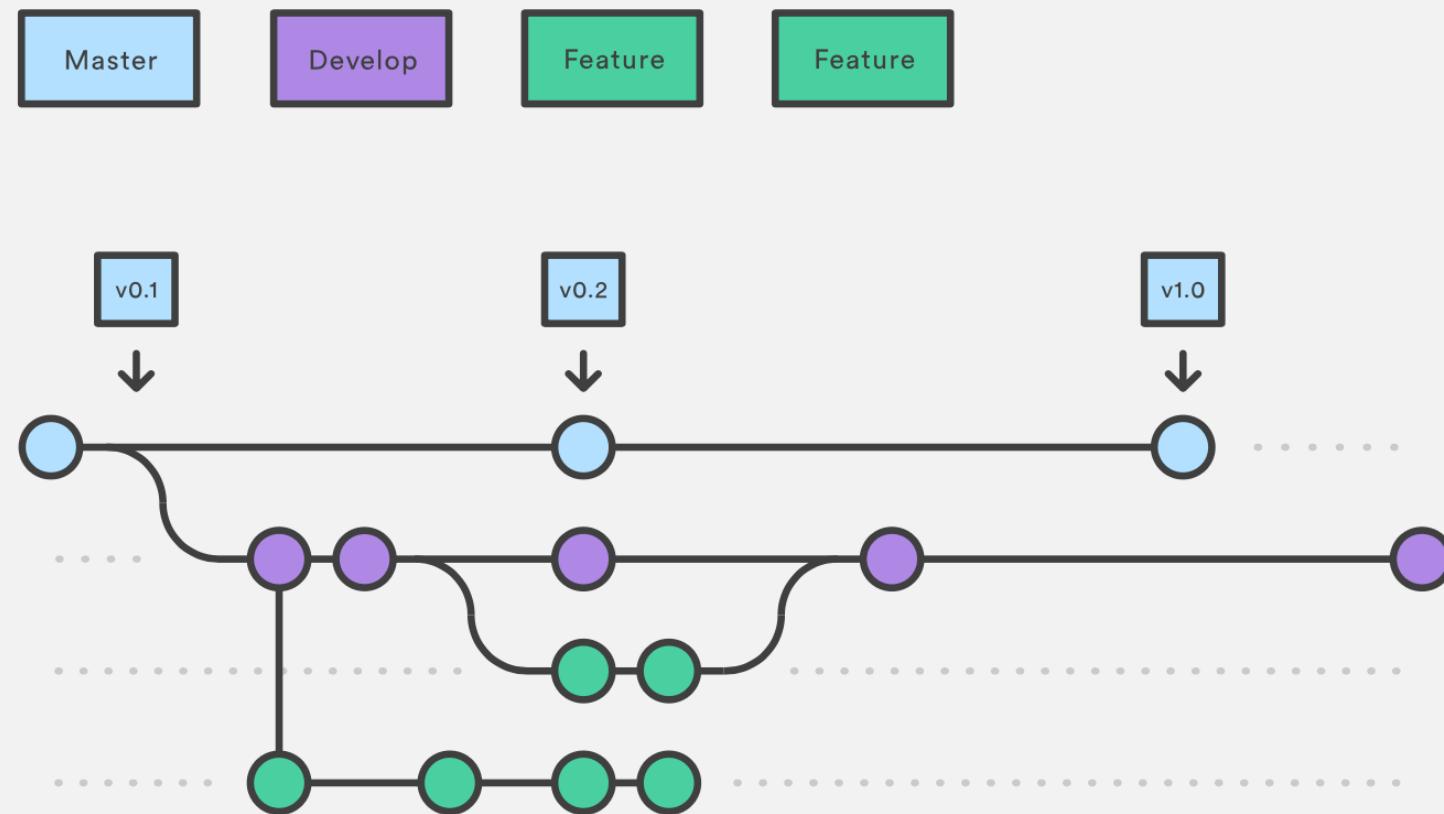
# Git workflow

*0. git status*

*1. git pull origin <branch>*

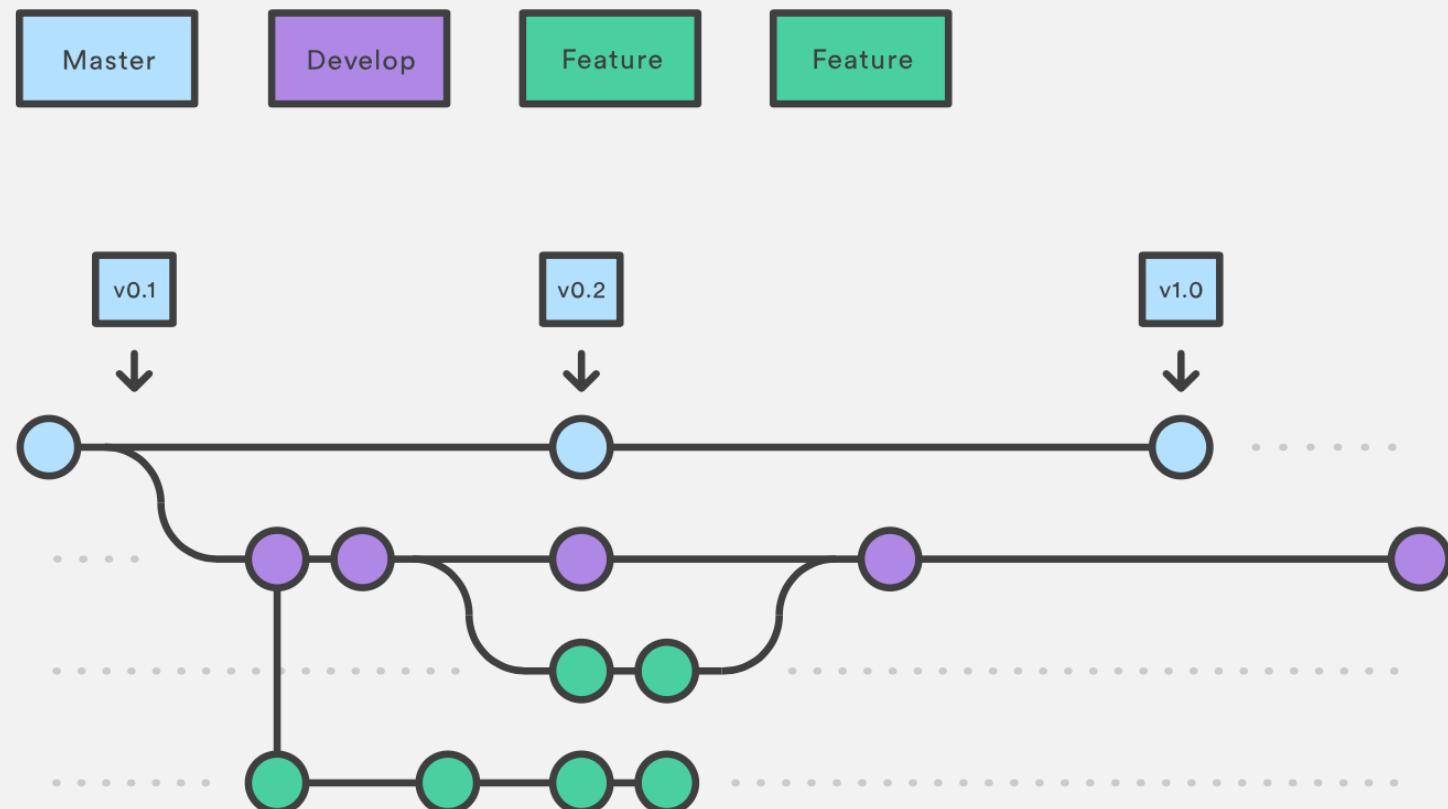
*2. git checkout -b <new branch>*

*3. DO YOUR WORK HERE*



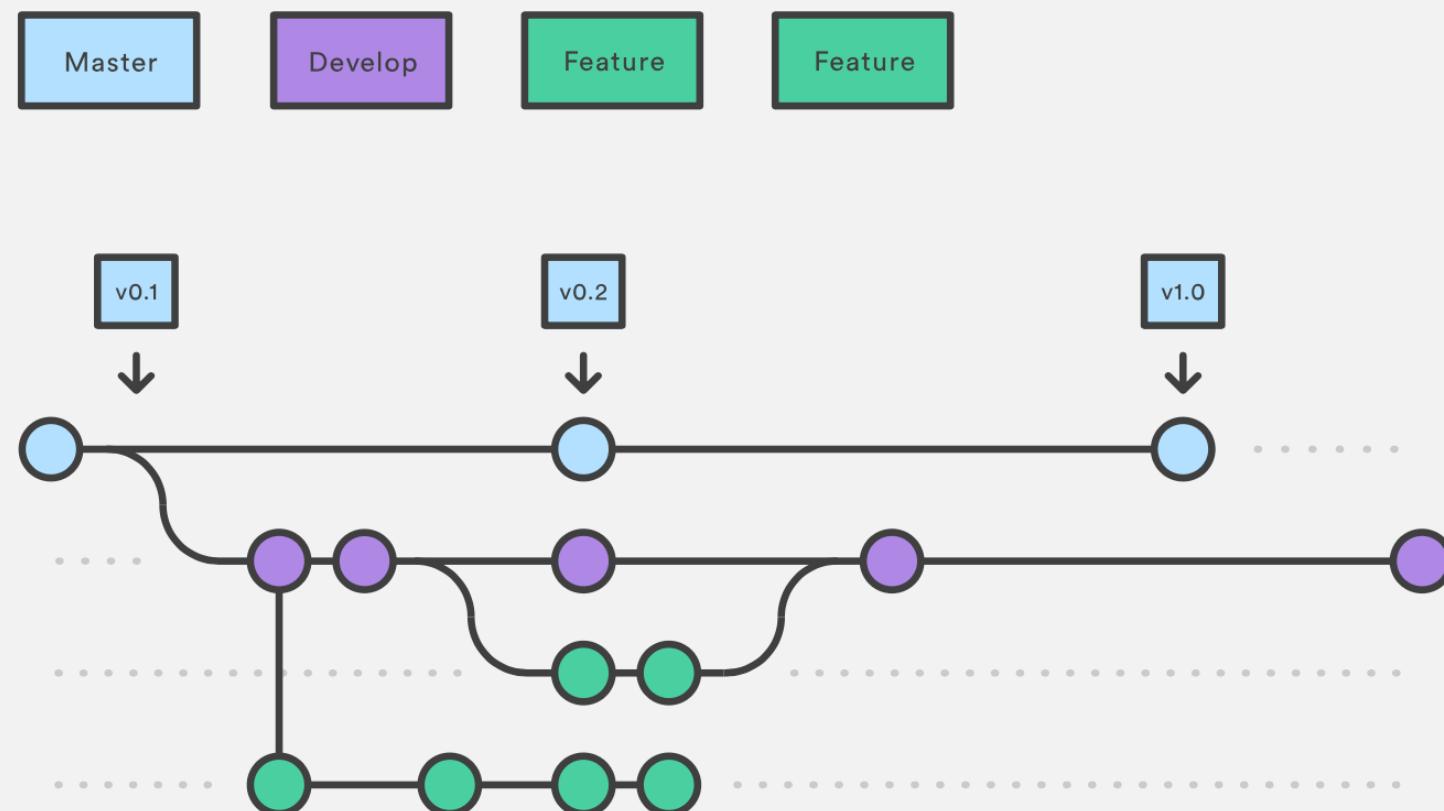
# Git workflow

4. *git add <file>*  
5. *git status*



# Git workflow

6. *git fetch origin*
7. *pull origin <branch>*
- SOLVE CONFLICTS*
8. *git commit -m <message>*
9. *git push origin <branch>*



# Git workflow



- 10. Open PR on Github*
- 11. Discuss about code changes*
- 12. Review approved*
- 13. Merge into master*
- 14. git branch -D <merged branch>*
- 15. git reset --hard origin master*
- 16. git pull origin master*

*Demo*

# Intermediate Git

*git checkout <file>*

*git checkout <branch>*

*git reset <commit>*

*git reset <file>*

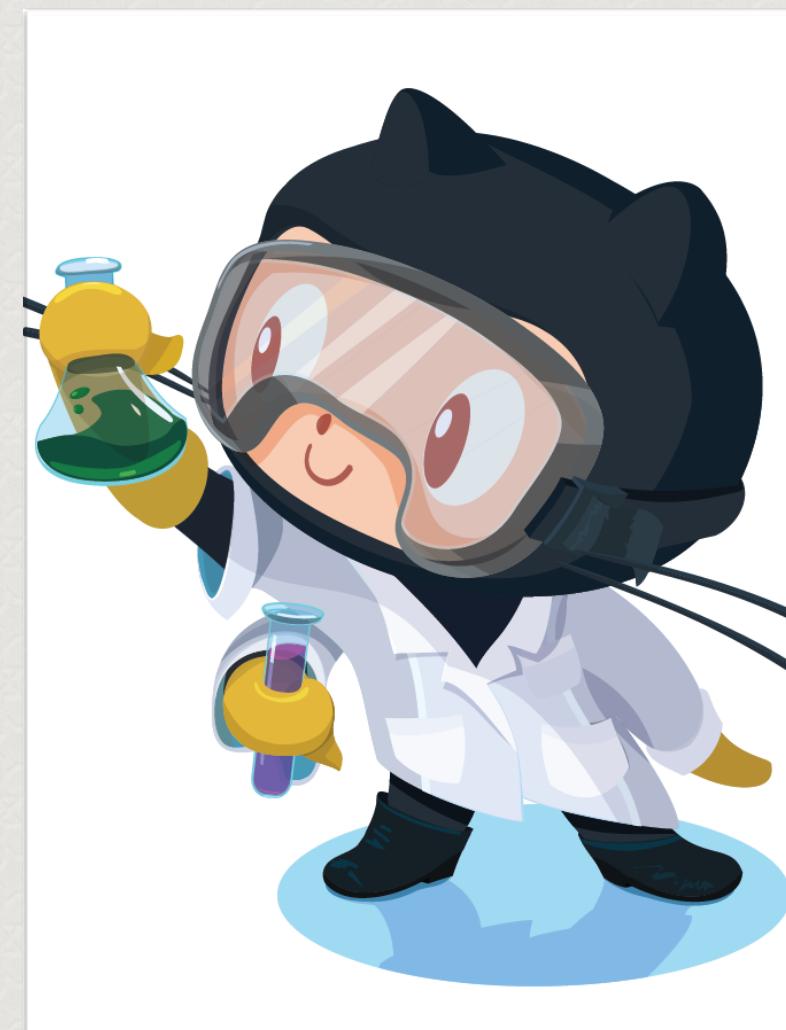
*git rebase <branch>*

*pick*

*edit*

*Squash*

*git blame*



*Demo*

# Advanced Git



1. *git rebase --onto master <br1> <br2>*

2. *git log --graph --abbrev-commit --decorate --format=format:'%C(bold blue)%h%C(reset) - %C(bold green)(%ar)%C(reset) %C(white)%s%C(reset) %C(dim white)- %an%C(reset)%C(bold yellow)%od%C(reset)' --all*

...