

GIT

16 Hours

Objectives

The main objective of this course is to learn git, its internals and how git works behind the scenes. On one hand Git is the most popular SCM tool but on the other hand git is the most complicated one of them all so users need to get familiar with git and its abilities and internals.

Expectations and goals

This course is for anyone who has been using any kind of VCS (Version control system) before. In this course the user will get to know git, its internals, how to use efficiently on a daily basis with focus on what are branches and what is the recommended way to use them.

Class Pre-Requisites

To participate in this class, all participants MUST have the following tools, installed on their personal computers (depending on the choice of operating system):

Type	Software	Download URL
IDE	Eclipse or Other	
Git CLI	Git CLI	https://git-scm.com
Git UI	SourceTree or GitKraken	https://www.sourcetreeapp.com/ or https://www.gitkraken.com/

Each participant must have their own personal computer and a mandatory github account (<https://github.com>). The preferred operating system for the class is Linux, however, participants may use Windows or MacOS as well.



Secondary goals

User will learn:

- How to configure git (configuration, hooks, aliases, gitconfig)
- What are the key features of git (DVCS, 3-states)
- What are branches and how to use them efficiently
- How does merges work (ff, no-ff, rebase)
- Git flow and why it is recommended as the daily workflow
- Hooks, how to create hooks and why/when to use them
- How to improve their productivity and code quality (pull request)

Sessions

The course is divided into small sessions.

Each session includes hands on based upon the time frame of the course

- Introduction
- Branches
- Gitflow
- Other stuff (Tools, hooks, aliases, stash and more).

Module 1 – Intro

- History - The history of git SCSS, RCS
- Key Features - DVCS, 3-states, Storage, Heuristics
- 3- states:
 - ~ Source Modification
 - ~ Source Staging
 - ~ Source Commit



Module 2 – Basic Setup with github and usage

- Starting an empty local git repository
- Making your first commit to a remote repository
- Basic Configuration – Aliases, .gitconfig, .gitkeep, .gitignore
- Pushing your source code to a remote repository
- Commands covered: init, add, commit, push, alias

Module 3 – Working with remote public repositories

- Understanding source code forking and the pull-request working paradigm
- Cloning and Forking remote repositories
- Commands covered: clone, rm, status, log
- Understanding commits and who do I blame for breaking my code?

Module 4 – Working with Branches

- Branches – What are branches
- Commands – Branches command
 - ~ Flow
 - ~ Checkout
 - ~ Fetch
 - ~ branch
- Git Flow – What is the Git Flow process – and why is it important?
 - ~ Master Branch
 - ~ Develop Branch
 - ~ Feature Branch
 - ~ Hot-Fix Branch

Module 5 - Pull Requests and merging external code submissions

- What is Fast-Forward and no Fast-Forward statuses
- Creating a pull request
- Merging a pull request
- Using github to review a pull request and selective merging

Module 6 - Advanced topics

- User interfaces to git: Sourcetree and GitKraken
- Source Code merge tools: BeyondCompare and Meld
- Managing releases with Tags
- Using git Hooks to make life easier
- Introduction to CI/CD – The basic theory of CI/CD and why containers are
- important for your software lifecycle

