



Git Basics

Training Details

Training Time	:	1 Day
Capacity	:	12
Prerequisites	:	There are no prerequisites for this course.

About Training

About Training

This course is divided into a number of individual modules, many modules have an associated lab exercise that will help to reinforce the learning experience. The Git basics training course provides a comprehensive understanding of basic Git concepts and looks at the different ways Git can be used and implemented

What You'll Learn

At the end of the Clearvision Git Basics course students will understand basic Git concepts and will be able to use Git to provide configuration management support for software development projects. For a more in-depth look at how Git can be used to support various ways of working, as well looking at maintenance of your Git environment and integrations with other tools, Clearvision recommend this course is followed by the Git Advanced course.

Outline

Module 1: Introduction

Module 2: Git Concepts

- Git Object Types: Blobs/ Trees/ Commits/ Tags/ Git Reference Types/ Object Model Example
- Hash Values (SHA-1)
- The Git Data Model
- Verify Git Installations
- Identify the git repository

- Ask Git for help

Module 3: Getting started

- How to create a new Git repository
- Setting up Your Git Environment
- Basic workflow
- Adding new files
- Changing existing files
- Committing changes to the repository
- The Git Log
- Identify yourself to Git
- Create a Git repository
- How to add new files/folders
- How to commit new files/folders
- Clone an existing repository
- Check the status of your Git repository

Module 4: Working with Git

- Workflow
- The Status command
- Ignoring file types
- View specific changes
- Removing files & directories
- Moving files & directories
- Stashing Changes
- Undoing or fixing errors
- Reset a change
- Revert a change
- Checkout a change
- Use the git log command to help you clarify if you are committing the correct version of a file
- Use the git status command to help you clarify the current state of your repository

Module 5: Branching and Merging

- What is a branch
- Creating a branch
- Switching between branches
- What is a merge
- Fast forward merge
- 3-way merge
- Resolving merge conflicts
- Merge tools
- Removing branches
- Branch management in Git
- Create branches in Git
- Add files to branches

- Checkout branches
- Merge into branches
- Rebase and squash a feature branch

Module 6: Collaboration Basics

- Cloning repositories
- Remotes
- Remote Branches and Tracking Branches
- Fetching, Pulling and Pushing Changes
- Bare and development repositories
- Publishing repositories Lab Exercises
- Clone a repository
- Create or Identify a remote
- Pull changes
- Push changes
- Merging between repositories

Module 7: Collaboration Strategies

- Branching Strategies
- Structural Strategies
- A flexible exercise implementing the strategy you intend to use moving forward

Module 8: Tagging

- What is a tag
- Viewing tags
- Creating tags
- Signed tags
- Tagging later
- Sharing tags
- Create tags in Git
- Lightweight tags
- Annotated tags
- Checkout tags in Git
- How to view tags
- How to checkout tags