

TLG300 : GIT for Developers

Code : TLG300 **Duration :** 1 day

Category : Tooling

Audience :

This course is intended for developers who want to use Git and GitHub for distributed version control.

Prerequisites :

To participate in this course knowledge of and experience with software development in a modern language such as Java, C++, C# or PHP is required.

Realization :

The theory is discussed on the basis of the presentation slides and is interspersed with exercises. Demo projects are used to clarify the concepts. The course material is in English.



GIT for Developers



Contents :

In the course Git for developers participants learn to understand the concepts of distributed version control and to begin with the GitHub suite of tools effectively. Attention is paid to repositories that can be located both local and remote, how versions can be committed and possibly undone. It is further treated what Git branching is and which strategies you can use for branching and merging. The control of Git from the command line is also discussed. Furthermore attention is paid to typical GitHub workflows, to undoing errors and history manipulation in a distributed environment. Finally some of the available tools are treated like Revision Selection, Interactive Staging, Rewriting History and Debugging with bisect.

Module 1 : Git Intro

- What is Git?
- Git and DVCS concepts
- GitHub platform
- Repository
- Repository on the web
- Desktop repositories
- Recording Changes to the Repository
- Viewing the Commit History
- Undoing Things
- Tagging
- Document versioning
- Local and remote repositories
- Distributed version control
- Configuration and customization

Module 2 : Git Branching

- Branching and Merging
- Branch Management
- Branching Workflows
- Remote Branches
- Rebasing
- Branching strategies
- Merge strategies
- Conflict resolution
- Multiple remotes
- Fork maintenance
- Temporary branching solutions

Module 3 : Git on the Server

- The Protocols
- Getting Git on a Server
- Generating SSH Public Key
- Setting up the Server
- Public Access
- Hosted Git (GitHub)
- Working with Remotes
- Remote repository interaction
- Repository integration

Module 4 : Distributed Git

- Distributed Workflows
- Contributing to a Project
- Maintaining a Project
- Synchronization
- Collaboration patterns
- Collaboration workflows
- Project management
- Git Configuration
- Attributes and Hooks
- Local and remote synchronization
- Shortcuts and efficiencies

Module 5 : Git Tools

- Revision Selection
- Interactive Staging
- History reordering
- History editing
- Rewriting History
- Debugging with bisect
- Submodules
- Subtree Merging
- Accidental commits
- Ignore patterns uses
- Aliases beyond-the-basics