GIT & GITLAB (WITH GITHUB) — MODERN DEVELOPER WORKFLOW (TTDV7553)

Course Code: 100877

Git and GitLab/GitHub tools are the twin pillars of the developers' Continuous Integration and Continuous Delivery toolbox. By using these tools effectively, developers and DevOps engineers can ensure much higher quality code and better results.

This course provides a hands-on, modern introduction to Git, with a strong focus on GitLab workflows while remaining fully compatible with GitHub. Participants learn best practices for collaboration, branching, rebasing, code review, and CI/CD.

What You'll Learn

Join an engaging hands-on learning environment, where you'll learn:

- Confident use of Git with GitLab or GitHub
- Modern workflows
- Safe collaboration
- CI/CD pipelines

Who Needs to Attend

Developers, DevOps engineers, technical teams

Prerequisites

Basic command-line familiarity

GIT & GITLAB (WITH GITHUB) — MODERN DEVELOPER WORKFLOW (TTDV7553)

Course Code: 100877

VIRTUAL CLASSROOM LIVE	$2,295 USD	3 Day

Virtual Classroom Live Outline

1. Git & Platform Basics

- Core Concepts
  - What version control is and why it matters
  - Git architecture: working tree, index, repository
  - Local vs. remote repositories
  - Distributed version control fundamentals
- Git Installation & Setup
  - Installing Git (Linux, macOS, Windows)
  - Initial configuration (user.name, user.email)
  - Line endings, editors, and defaults
- GitLab & GitHub Overview
  - Navigating GitLab UI
  - Repository structure and permissions
  - Comparing GitLab and GitHub concepts
- Hands-on
  - Initialize a repository
  - Create your first commit
  - Push to GitLab or GitHub
  - Explore repository UI

2. GitLab Flow & Team Workflows

- Workflow Models
  - GitFlow vs. GitLab Flow
  - Trunk-based development
  - When to use feature branches
- Environment Strategy
  - Feature, staging, and production branches
  - Release branches vs. tags

        - Environment promotion patterns
    - Collaboration
        - Issues, merge requests (MRs), and discussions
        - Linking commits to issues
        - Approvals and review policies
    - Optional Platform Features
        - GitLab: Protected branches, approval rules
        - GitHub: Branch protection, required reviews

3. Branching Strategy

- Branching Fundamentals
    - Creating and switching branches
    - Short-lived vs. long-lived branches
    - Naming conventions (feature/, bugfix/, release/)
- Visualization & Cleanup
    - Visualizing branch graphs
    - Deleting merged branches
    - Keeping repositories tidy
- Releases
    - Annotated vs. lightweight tags
    - Semantic versioning
    - Release notes

4. Configuring Git Like a Pro

- Configuration
    - Global vs. local .gitconfig
    - Editor, diff, and merge tools
    - Line ending normalization
- Productivity Boosters
    - Git aliases
    - Shell integrations
- Ignoring Files
    - .gitignore patterns
    - Project vs. global ignores
- Authentication & Security
    - HTTPS vs. SSH
    - Managing SSH keys
    - Credential helpers

5. Rebasing (Safely and Effectively)

- Concepts
    - Rebase vs. merge
    - When rebasing is appropriate
    - Rewriting history safely
- Practical Rebasing
    - Rebasing local branches
    - Interactive rebase (reword, squash, fixup)

        - Cleaning up commit history
    - Conflict Handling
        - Resolving rebase conflicts
        - Aborting and continuing rebases
        - Common mistakes and recovery

6. Merging & Code Reviews

- Merge Types
    - Fast-forward merges
    - No-fast-forward merges
    - Merge commits explained
- Platform-Based Merges
    - Merge requests (GitLab)
    - Pull requests (GitHub)
    - Code review best practices
- Tooling
    - Diff views
    - Inline comments
    - Review checklists

7. Resolving Merge Conflicts

- Understanding Conflicts
    - Why conflicts occur
    - Common conflict scenarios
- Resolution Techniques
    - Manual resolution in editors
    - Using git status and git diff
    - Marking conflicts as resolved
- Best Practices
    - Small, focused commits
    - Frequent pulls/rebases
    - Testing after resolution

8. Working with Remote Repositories

- Remote Basics
    - origin, upstream, and forks
    - Cloning vs. forking
    - Fetch vs. pull
- Collaboration Models
    - Shared repository model
    - Fork-and-merge model
- Tracking & Syncing
    - Upstream branches
    - Keeping forks up to date

9. Exploring & Managing History

- History Inspection

    - • git log, git show, git diff
    - • Graph and pretty formats
- Accountability Tools
    - • git blame
    - • Annotate views in GitLab/GitHub
- Undoing Changes
    - Amend commits
    - Revert vs. reset
    - Recovering lost commits

10. Improving Your Daily Git Workflow

- Everyday Power Tools
    - git stash
    - Interactive staging (git add -p)
    - Reviewing diffs before commit
- Commit Quality
    - Writing meaningful commit messages
    - Atomic commits
    - Conventional commits (optional)
- Automation
    - Aliases for repetitive tasks
    - Git hooks (pre-commit basics)

11. CI/CD with GitLab (and GitHub)

- CI/CD Fundamentals
    - What CI/CD solves
    - Pipelines, stages, and jobs
- GitLab CI/CD
    - GitLab Runner overview
    - .gitlab-ci.yml structure
    - Variables and secrets
    - Artifacts and caching
- Pipelines in Practice
    - Build, test, deploy stages
    - Visualizing and debugging pipelines
    - Optimizing execution time
- Optional GitHub Features
    - GitHub Actions
    - Workflow YAML syntax
    - Marketplace actions

Optional Advanced Topics

- Monorepos vs. multirepos
- Submodules vs. subtrees
- Signed commits and tags
- Security scanning (SAST, dependency scanning)
- Release automation

- GitOps concepts

Course Deliverables

- Hands-on labs
- Sample repositories (GitHub & GitLab)
- CI/CD pipeline examples
- Best-practice checklists
- Real-world workflow patterns

Jun 8 - 10, 2026 | 10:00 AM - 6:00 PM EDT

GIT & GITLAB (WITH GITHUB) — MODERN DEVELOPER WORKFLOW (TTDV7553)

Course Code: 100877

PRIVATE GROUP TRAINING	2 Day

Visit us at www.globalknowledge.com or call us at 1-866-716-6688.

Date created: 4/24/2026 3:52:40 AM