

Course Title: GIT Intermediate & Advanced

Course Description:

A powerful and complex tool yet widely used open source Version Control System (VCS) which helps easy to track the changes to files of an application/software. It helps to figure out the changes made by team or another including what has been changed and why. Every developer has the full history of their code repository locally. This makes the initial clone of the repository slower, but subsequent operations such as commit, blame, diff, merge, and log amazingly faster.

Course Duration: 2 Days (16 Hrs.)

Course Objective:

- Understand Git concepts
- Apply various techniques to visualize data using multiple graphs and dashboards
- Implement Git in the organization to monitor operational intelligence
- Troubleshoot various application log issues using SPL (Search Processing Language)
- Implement indexers, forwarders, deployment servers and deployers in Git

Pre Requisites:

- Have basic mathematic knowledge
- Want to learn more about Git

Course Contents:

Course Contents: **Git Intermediate Training**

Day 1:

Navigating the Commit Tree

- Referencing commits
- Exploring tree listings
- Getting more from the commit log
- Viewing commits
- Comparing commits

Branching

- Branching overview
- Viewing and creating branches
- Switching branches
- Creating and switching branches
- Switching branches with uncommitted changes

- Comparing branches
- Renaming branches
- Deleting branches

Merging Branches

- Merging code
- Using fast-forward merge vs. true merge
- Merging conflicts
- Resolving merge conflicts
- Exploring strategies to reduce merge conflicts
- Fast Forward Merge
- Three way Merge
- Rebasing

Stashing Changes

- Saving changes in the stash
- Viewing stashed changes
- Retrieving stashed changes
- Deleting stashed changes

Remotes

- Using local and remote repositories
- Adding a remote repository
- Creating a remote branch
- Cloning a remote repository
- Tracking remote branches
- Pushing changes to a remote repository
- Fetching changes from a remote repository
- Merging in fetched changes
- Checking out remote branches
- Pushing to an updated remote branch
- Deleting a remote branch
- Enabling collaboration
- A collaboration workflow

Tools and Next Steps

- Setting up aliases for common commands
- Using SSH keys for remote login
- Exploring integrated development environments
- Exploring graphical user interfaces
- Understanding Git hosting
- Conclusion

Tagging

- Lightweight tags

- Annotated tags
- Signed Tags
- Create tags in Git
- Create tags in Stash
- Checkout tags in Git
- How to view tags
- How to checkout tags

Course Contents: **Git Advanced Training**

Day 2

Tools and Next Steps

- Setting up aliases for common commands
- Using SSH keys for remote login
- Exploring integrated development environments
- Exploring graphical user interfaces
- Understanding Git hosting
- Conclusion

Tagging

- Lightweight tags
- Annotated tags
- Signed Tags
- Create tags in Git
- Create tags in Stash
- Checkout tags in Git
- How to view tags
- How to checkout tags

Workflow Strategies

- Branching basics
- Forking basics
- Centralized Workflow
- Feature Branch Workflow
- Gitflow Workflow
- Forking Workflow
- Git Troubleshooting

SourceTree: Local Repositories

- What is SourceTree?
- Installing SourceTree
- Creating a Local Repository Through SourceTree
- Editing a Committed File

- Adding a Tag
- Branching
- .gitignore in SourceTree
- Removing Files

SourceTree: Remote Repositories

- Creating a Repository Through Bitbucket
- Cloning the Repository
- Working with the Remote Repository