

دوره آموزش Git

دوره Git و GitHub از مبانی تا کار تیمی حرفه‌ای؛ آموزش عملی `commit`، شاخه‌ها، ادغام، رفع `conflict` و مدیریت مخزن برای پروژه‌های واقعی.

مروری بر دوره

این دوره بر اساس سرفصل ارائه شده، شما را از مبانی Git تا کار تیمی حرفه‌ای با GitHub پیش می‌برد:

- معرفی Git و شیوه دیدن تاریخچه، نصب و پیکربندی ترمینال و ابزارهای GUI مثل GitKraken
- ایجاد و مدیریت مخزن و چرخه `commit` با دستورات پایه مانند `git status/git add/git commit/git log`
- مبانی GitHub کلون، تنظیم `push`، SSH، و مدیریت `remote`
- اصول کامیت‌نویسی و `gitignore`
- کار دقیق با شاخه‌ها و `HEAD` و `switch/checkout`
- ادغام‌ها و حل تعارض و مقایسه تغییرات با `git diff`
- نگهداشت موقت با `stash`
- بازگردانی/لغو با `revert` و `restore`، `reset`
- و در نهایت `pull` و `fetch` و شاخه‌های راه‌دور

تمرکز دوره بر تمرین‌های عملی و درک مفاهیم است تا بتوانید در پروژه‌های واقعی با اطمینان از کنترل نسخه بهره بگیرید.

سرفصل‌ها

- ۱ - Introducing Git
- What is Git?
- Visualizing Git



- Git History
- Git Users
- Git vs. Github

۲ - Git Installation

- Installing Terminal vs. GUI
- Configuring Git Name and Email
- GitKraken
- Terminal Crash Course (Navigation, Creation of Files and Directories, Deletion)

۳ - Git Basics

- What is a Git Repo?
- git init
- git status
- .git Folder
- Committing Workflow
- Staging Changes
- git add
- git commit
- git log

۴ - Github Basics

- What Does it Do?
- Why Do We Need it?
- Cloning Github Repo
- Cloning Non-github Repo
- SSH Config for Github



- Creating First Github Repo
- git push
- git remote
- Github Workflows
- Main and Master Branches

۵ - Committing

- Navigating the Docs
- Atomic Commits
- Commit Messages
- git log Closer Look
- GUI Commit (GitKraken and Pycharm/Intellij)
- git amend
- .gitignore

۶ - Branches in Details

- Introducing Branches
- The Master Branch vs. Main
- HEAD
- git branch
- Creating and Switching Branches
- git checkout vs. git switch
- Switching Branches with Unstaged Changes
- Deleting and Renaming Branches
- HEAD in Details

۷ - Merging



- Introducing Merge
- Fast Forward Merge
- Visualizing Merges
- Generating Merge
- Conflicts
- Resolving Merge Conflicts

۸ - Git Diff

- What is Git Diff?
- Reading Diffs
- Viewing Unstaged Changes
- Viewing working Directory Changes
- Viewing Staged Changes
- Diffing Specific Files
- Comparing Changes Across Branches
- Comparing Changes Across Commits
- Visualizing Diffs with GUI

۹ - Stashing

- Why Do We Need it?
- git stash
- Save and Prop
- git stash apply
- Working With Multiple Stashes
- Dropping and Cleaning the Stashes

۱۰ - Undoing Changes



- Checking Out Old Commits
- Re-Attaching Detached HEAD
- Relative Commit Refrencing
- Discarding Changes With git checkout
- git restore Un-modification !
- git restore Unstaging Changes
- git reset and Undoing Commits
- git revert to Reverting Commits

۱۱ - Fetching and Pulling

- Remote Tracking Branches
- Checking Out Remote Tracking Branches
- Working with Remote Branches
- git fetch
- git pull
- git pull and Merge Conflicts
- Alternate Syntax for git pull

مخاطبان دوره

- برنامه‌نویسان مبتدی تا میان‌رده که می‌خواهند جریان کاری استاندارد مبتنی بر Git ایجاد کنند (شاخه‌بندی، ادغام، حل تعارض).
- دانشجویان و علاقه‌مندان به توسعه نرم‌افزار و علوم داده که نیاز به ثبت منظم تغییرات و کار تیمی دارند)
- اعضای تیم‌های چندنفره (Frontend/Backend/QA/DevOps) که با مخازن راه‌دور، pull/fetch و مرور تغییرات (diff) کار می‌کنند.
- لیدها و مدیران پروژه‌ای که می‌خواهند تاریخچه تغییرات و فرآیندهای بازگردانی/ریویو را بهتر درک و مدیریت کنند.



پیش نیاز ها

- آشنایی مقدماتی با کار با رایانه و یک ویرایشگر کد.
- امکان نصب Git روی سیستم عامل (Windows/macOS/Linux)
- داشتن/ایجاد حساب GitHub برای بخش‌های مربوط به مخازن راه‌دور
- آشنایی با خط فرمان مفید است، اما به دلیل «مرور فشرده ترمینال و دستورات مرتبط» در ابتدای دوره الزامی نیست.

