■ Complete Git & GitHub Course

■ Goal:
By the end, you will:
- Use Git (the tool) for version control.
- Use GitHub (the platform) for collaboration, portfolio, and projects.
- Apply GitHub in Data Science workflows (Jupyter, datasets, ML projects).

■ Module 1: Introduction (Day 1–2)
1. What is Git? What is GitHub?
- Git = tool for tracking changes in code.
- GitHub = online platform for storing + sharing Git projects.
2. Setup
- Install Git on your PC.
- Configure username & email (git config).
- Learn GitHub interface (repositories, profile, settings).

■ Exercise: Create your first repository on GitHub.

■ Module 2: Git Basics (Day 3–5)
1. Repository (repo) → What it is and why it matters.
2. Basic Commands:
- git init → create repo
- git status → check status
- git add → stage changes
- git commit -m "message" → save changes
- git log → see history
3. .gitignore → hide unnecessary files (like datasets).

■ Exercise: Create a repo, write a small Python file, commit changes.

■ Module 3: Working with GitHub (Day 6–8)
1. Push & Pull
- git remote add origin
- git push (send code to GitHub)
- git pull (download latest code).
2. Branches
- git branch
- git checkout -b new-feature
- git merge.
3. Collaboration
- Forking, cloning, pull requests, issues.

■ Exercise: Create two branches, make changes, merge them on GitHub.

■ Module 4: Intermediate Git (Week 2)

1. Undo & Fixing Mistakes
- git reset
- git revert
- git stash.
2. Merge Conflicts → how to solve them.
3. Tags & Releases → marking project versions.

■ Exercise: Break your code, then restore an older version using Git.

■ Module 5: GitHub for Data Science (Week 3)
1. Using Jupyter Notebooks with GitHub
- Upload .ipynb notebooks.
- Share with teammates.
2. Managing Datasets
- Use .gitignore for big CSV/Excel files.
- Link external data via README.
3. Documentation
- Write a good README.md with project details.
- Use Markdown for formatting.

■ Exercise: Upload a small Data Science project (Python + Notebook + README).

■ Module 6: Advanced GitHub (Week 4)
1. GitHub Actions (Automation)
- Auto test your code.
- Auto run ML training when data updates.
2. Collaboration at Scale
- Project boards (like Trello).
- Wiki pages.
3. Open Source Contribution
- How to contribute to other people's projects.

■ Exercise: Contribute to a beginner-friendly open source repo.

■ Module 7: Portfolio & Career (Final Week)
1. Build your GitHub Portfolio
- Pin your best projects.
- Organize repos professionally.
2. Showcase Data Science Work
- Data cleaning project.
- Visualization dashboard.
- ML model repo.
3. Resume Integration
- Add GitHub profile link to CV.

■ Learning Method:
- Daily Practice (30–60 mins) → Git is learned by doing.
- Use command line Git, not only GUI → more powerful.

- Build small projects (start with Python scripts, then ML notebooks).