DevOps Class 2 Out-Line

Introduction to Version Control Systems

- 0:00 - 0:15: Recap of previous meetup

- 0:15 - 0:45: Basics of version control

```
- What is version control?
- Benefits of version control systems (VCS)
```

What is Version Control?

Definition: Version control, also known as source control, is a system that records changes to a file or set of files over time so that specific versions can be recalled later. It allows you to track your code history, revert back to previous stages, and collaborate on code with others.

How It Works: Version control systems start with a base version of a document and then record subsequent changes as a series of patches or updates. This allows the version control system to reconstruct older versions by applying the patches to the base version.

Benefits of Version Control Systems (VCS)

1. **Collaboration:**
   - **Multiple Developers:** allows multiple people to simultaneously work on a single project. Each developer can independently make changes and later merge those changes into a shared repository.
   - **Branching and Merging:** Developers can create branches, which are parallel versions of a project, to work on updates or new features without disturbing the main project
2. **Track Changes:**
   - **Commit History:** Every change made to files in a repository is tracked along with who made the change, why they made it
   - **Audit Trails:** VCS provides a complete history of the changes, which is useful for debugging issues, understanding decision-making, and determining the origin of code segments.

3. **Backup and Restore:**
    - Files are stored which means you can revert to a previous version at any time. This acts as a full backup for your project.
    - If something breaks due to a recent change, you can fix it by going back to a functional previous version
4. **Integration with DevOps:**
    - VCS are integral to CI/CD pipelines, which automate steps in software delivery processes. They trigger builds, tests, and deployments based on changes detected in the version control system.
5. **Documentation:**
    - By using the commit history and other metadata stored in VCS, the entire project's evolution is documented

- 0:45 - 1:30: Introduction to Git

Git and Git Hub*

Basic Commands exercise

- Git Bash and a GitHub repo needed!
- Create GitHub Repo
- Create a Folder and "Git Init"
- Git clone the remote repo
- Create a Text File,
- git status
- Git add . (text.txt) ,"Git Commit -m 'created a new text file'"
- git log
- create a new file, Python.Py file, commit again, see the new added file, switch to an older hash and check again ls.
- git branch, how to create a branch, how to checkout, how to merge
- show an example of a merge error and how you fix it
- git remote repo url
- git pull
- git push + up stream link
- exercise:
  clone the remote repo to a local repo, create a text file and add content
  create a new branch, add a new text file and add the some content but change one word
  merge the new branch to main branch and solve the conflict

pull the main branch from the remote repo, push the new content to the main branch on the remote repo.

https://git-scm.com/book/en/v2/Getting-Started-A-Short-History-of-Git