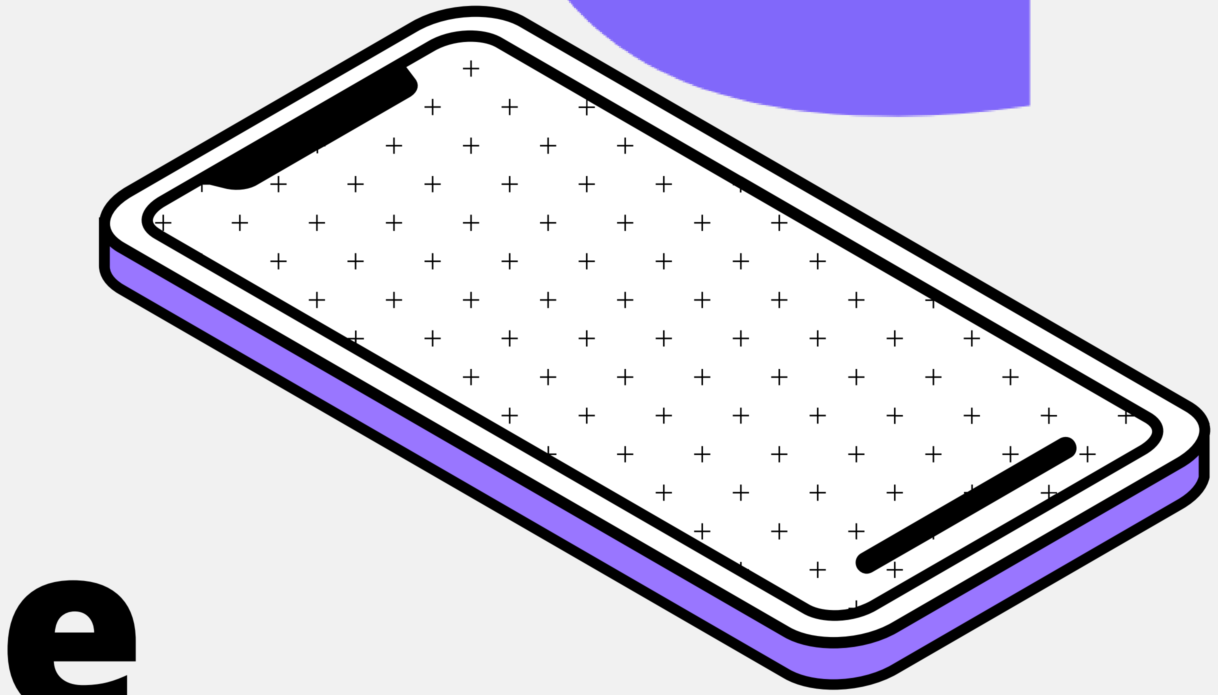
Industrial Training Program

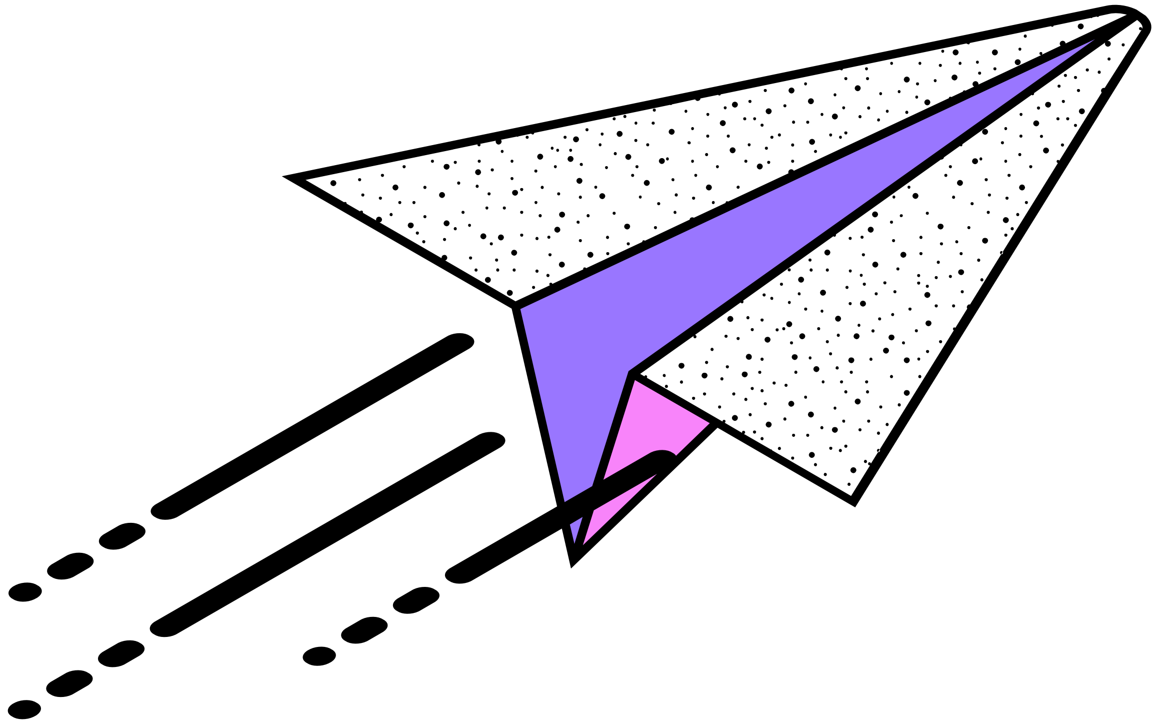
by CWIPEDIA

Instructor: Shivam Hande



Day One

Agenda



Introduction to Training Program

What is Software (SDLC)?

What is git and github?

What is the working directory, staging area, and local repo?

Getting started with git and github

Contributing on GitHub

Introduction to Training Program

LinkedIn post:

I'm excited to announce that I have enrolled in CWIPEDIA Industrial Training Program. I'm super happy to learn and practice here with the team CWIPEDIA.

What is Software (SDLC)?

Process of creating software

- Requirement analysis
- Planning
- Software design
- Software development
- Testing
- Deployment

What is git and github?

Git is a version control system, It keeps track of projects and files as they change over time (Git helps keep track of changes made to a code)

Github:
GitHub is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere.

Commits

A commit is a series of changes that you have made to a file in the repository. A commit has a unique SHA1 hash which is used to keep track of files changed in the past. A series of commits comprises the Git history.

Branches

A branch is a version of the repository. Repo can have one or many branches

Now what is the working directory, staging area, and local repo?

The working directory is where new files are created, old files are deleted, or where changes are made to already existing files.

Once changes are made, they are added to the staging area. The staging area is also sometimes called the index.

Once the changes are complete, the staging area will contain one or more files that need to be committed

Getting started

To get started with Git, go to your terminal and run the following command in your project directory. This initializes a project directory.

```
git init
```

Run the following command to add files for Git to track. This will add these files to the staging area.

```
git add <filename_one>
```

Run the following command to commit your changes to these files.

git commit -m "<add a commit message here>"

We can push our changes through once we’re done.

git status

The git status command displays the state of the working directory and the staging area

git push

Making any more changes in the master branch will require these changes to be committed again.

git checkout

Use to checkout working branches

git clean

Removes untracked files from the working directory.

git fetch

Fetching downloads a branch from another repository

git pull

The Git pull command is used to fetch and merge code changes from the remote repository to the local repository.

git clone

Creates a copy of an existing Git repository.

Difference between git pull and git fetch

Git Fetch is the command that tells the local repository that there are changes available in the remote repository without bringing AND git pull on the other hand does that AND brings (copy) those changes from the remote repository.

Contributing on GitHub

Why contribute to open source?

- builds your resume
- gives you practice with Git and GitHub

Step 1: Sign in to GitHub

Step 2: Fork the project repository

Step 3: Clone your fork

Step 4: Navigate to your local repository

Step 5: Add the project repository as the "upstream" remote

```
git remote add upstream
```

Step 6: Pull the latest changes from upstream into your local repository

Step 7: Create a new branch

Step 8: Make changes in your local repository

Step 9: Commit your changes

Step 10: Push your changes to your fork

Step 11: Begin the pull request and create

Pull requests let you tell others about changes you've pushed to a branch in a repository on GitHub.

Thank You!