**GitHub:-**

GitHub is a web-based platform used for collaborative work and managing project-related data. It allows users to store, share, modify, and delete project data seamlessly.

**Features of GitHub**

* **Open Source**: Encourages open-source project development.
* **Cloud-Based**: It provides cloud storage, enabling access from anywhere.
* **Community Support**: A vast community for help and collaboration.
* **Collaboration Tools**: Tools to manage team workflows.
* **Code Reviews**: Enables peer review of code.
* **Code Hosting and Sharing**: Reliable platform to host and share code.
* **Code Management**: Tools for organizing and managing code repositories.
* **Code Security**: Security features to protect your code.
* **Code Backup**: Ensures code is safely backed up.
* **CI/CD**: Continuous Integration and Continuous Deployment support.
* **Version Control**: Ability to revert to previous versions of a project.
* **Repository**: A storage location for all project files, which tracks changes.
  + **Public**: Accessible to everyone.
  + **Private**: Accessible only to the owner or authorized users.

**If There Was No GitHub**

* **Local Versioning**: Managing versions locally without centralized control.
* **Email and File Sharing**: Relying on email and manual file sharing.
* **Community Isolation**: Lack of a central community hub.
* **Impact on Open Source**: Potential negative impact on open-source development.
* **Efficiency**: Reduced efficiency in collaborative work.

GitHub is a platform where we can store, share, track and modify data

By using GitHub pages we can deploy the project. It is maintained by microsoft

GitHub features **Open Source and Community Support, Collaboration Tools, Code Review, Code Hosting and Sharing, Security Features and Continuous Integration and Deployment (CI/CD)** by using GitHub.

**How to Create a Repository and Upload Files Manually on GitHub**

**Creating a Repository**

1. **Sign In to GitHub**:
   * Go to [GitHub](https://github.com) and sign in to your account.
2. **Create a New Repository**:
   * Click on the **+** icon in the upper right corner of the GitHub dashboard.
   * Select **New repository** from the dropdown menu.
3. **Repository Details**:
   * **Repository Name**: Enter a name for your repository.
   * **Description** (optional): Add a short description of your project.
   * **Public/Private**: Choose whether your repository will be public or private.
   * **Initialize Repository**:
     + Optionally, select to add a README file, .gitignore file, and a license.
4. **Create Repository**:
   * Click the **Create repository** button to finish setting up your repository.

**Uploading Files Manually**

1. **Navigate to Your Repository**:
   * After creating the repository, you will be directed to the repository’s main page.
   * If you are returning to the repository later, navigate to it by clicking on your profile icon, selecting **Your repositories**, and clicking on the repository name.
2. **Add Files**:
   * On the repository main page, click the **Add file** button.
   * Select **Upload files** from the dropdown menu.
3. **Upload Files**:
   * Drag and drop the files or folders from your local machine into the upload area, or click **choose your files** to select files manually.
4. **Commit Changes**:
   * **Commit Message**: Enter a commit message describing the files you are uploading.
   * **Commit Options**:
     + **Commit directly to the main branch**: Choose this if you want the changes to be applied immediately.
     + **Create a new branch for this commit and start a pull request**: Choose this if you want to create a new branch and start a pull request.
   * Click the **Commit changes** button to upload the files to the repository.

**Git (global information tracker):-**

Git is a distributed version control system (VCS) that tracks changes in any set of computer files, usually used for coordinating work among programmers who are collaboratively developing source code during development.

Git can acts as a medium between development environment to storing platform

Git features **Open Source and Community Support, Branching and Merging, Distributed Version Control, Staging Area** by using git we can do pull request and push the code

**Three components of Git**

The three areas of Git refer to the three main components where Git manages and stores data internally.

**Working Directory**: The working directory is a directory on your system where you’ll manipulate files.

**Staging Area**: Staging area acts as a medium between your working directory and repository.

**Repository**: It is referred as git respository where you will track and commit changes

**Git installation**

**Download Git Installer**: Visit the official Git website at <https://git-scm.com/> and navigate to the download page.

**Verify Installation**: After the installation is complete, open a terminal or command prompt and run “**git --version**”

**Verify configuration**

**git config --list**

**Setting Configuration: Set config using below commands**

**git config --global user.email "your@email.com"**

**git config --global user.name "Your GitHub Name"**

**Command lines of Git**

‘’**cd**” It's a command-line used to change the directory

“**ls**” It's a command-line used to list the files

“**cd..**” To move up one level

“**git version**” used to check the version also gives you the whether it is installed or not

‘**’git init**’’ Initializes a new Git repository in the current directory.

“**git status**” displays the status

“**git clone url**” clones a remote repository into your local machine.

“**git add file.name**” used to add changes to the staging area.

“**git commit**” commit stages changes to the repository with a message’

“**git push –u origin branchname**” it pushes local commits to the remote repository.

“**git branch**” it displays the branches.

“**git checkout –b branchname**” used to switches the branches in working directory.

The **git config** command in Git is used to set or get configuration options for Git.

**git config --global user.name "Your Name"**

**git config --global user.email** [your@email.com](mailto:your@email.com)

**config reset**

rm ~/.gitconfig

How to upload files from our working directory to remote repository

**Process-1**

In GitHub

1. create repository in GitHub with your project name
2. add the description of project
3. select the security of file (public or private)
4. Add a read.me file for the documentation of your project
5. After creating the repository, click on the code and copy the https url

In your working directory

1. Select a folder open git bash terminal
2. Enter git init **(optional)**
3. Enter **git clone “url”**………enter the url which is created in GitHub
4. Check the status by using **ls**
5. Change the directory using **cd foldname**
6. Copy the files into **foldname**
7. Check the status using git status
8. Add the files using “**git add filename**”
9. Commit the message using “**git commit –m “meaningfull message**”
10. Push the code using “**git push –u origin main**”

**Process-2**

In GitHub

1. create repository in GitHub with your project name
2. add the description of project
3. select the security of file (public or private)
4. create the repository
5. below command lines is displayed
6. open your working directory

In your working directory

1. open the source code
2. open the terminal
3. paste the below commands one by one

git init

git add README.md

git commit -m "first commit"

git branch -M main

git remote add origin git@GitHub.com:saiteja-yernagula/b-n-.git

git push -u origin main

git is user-friendly software so it can display the help or suggestions , if it still remains unresolved go through below websites for reference.