2. Set up a GitHub account & Create a GitHub repository

Introduction:

GitHub is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere. GitHub essentials like repositories, branches, commits, and pull requests.

What is GitHub?

GitHub is a collaboration platform built on top of a distributed version control system called Git.

Figure 1. GitHub's beloved Octocat logo.

In addition to being a place to host and share your Git projects, GitHub provides a number of features to help you and your team collaborate more effectively. These features include:

- Issues
- Pull Requests
- Organizations and Teams

- Create and use a repository
- Start and manage a new branch
- Make changes to a file and push them to GitHub as commits
- Open and merge a pull request

To need a GitHub account and Internet access. You don't need to know how to code, use the command line, or install Git (the version control software that GitHub is built on).

Step 1: Set up a GitHub account

You will need a GitHub account to create a GitHub repository where the revision will be stored.

If you already have a GitHub account, skip ahead to Step 2: Create a GitHub repository

1. Go to https://github.com/join.
2. Type a user name, your email address, and a password.
3. Choose Sign up for GitHub, and then follow the instructions

1. Creating an account

To sign up for an account on GitHub.com, navigate to https://github.com/ and follow the prompts.

To keep your GitHub account secure you should use a strong and unique password. For more information, see "Creating a strong password."

2. Choosing your GitHub product

You can choose GitHub Free or GitHub Pro to get access to different features for your personal account. You can upgrade at any time if you are unsure at first which product you want.
For more information on all of GitHub's plans, see "GitHub's products.

3. Verifying your email address

To ensure you can use all the features in your GitHub plan, verify your email address after signing up for a new account. For more information, see "Verifying your email address."

4. Configuring two-factor authentication

Two-factor authentication, or 2FA, is an extra layer of security used when logging into websites or apps. We strongly urge you to configure 2FA for the safety of your account.

5. Viewing your GitHub profile and contribution graph

Your GitHub profile tells people the story of your work through the repositories and gists you've pinned, the organization memberships you've chosen to publicize, the contributions you've made, and the projects you've created.

Figure 2. Key GitHub Features.

Rather than force you into a "one size fits all" ecosystem, GitHub strives to be the place that brings all of your favorite tools together. You may even find some new, indispensable tools like continuous integration and continuous deployment to help you and your team build software better, together.

Exploring a GitHub Repository

A repository is the most basic element of GitHub. It is easiest to imagine as a project's folder. However, unlike an ordinary folder on your laptop, a GitHub repository offers simple yet

powerful tools for collaborating with others. A repository contains all of the project files (including documentation), and stores each file's revision history. Whether you are just curious or you are a major contributor, knowing your way around a repository is essential!

4. GitHub Repositories.

Step 2: Create a GitHub repository

You will need a GitHub repository to store the revision.If you already have a GitHub repository, be sure to substitute its name for CodeDeployGitHubDemo throughout this tutorial, and then skip ahead to Step 3: Upload a sample application to your GitHubrepository

1. On the GitHub home page, do one of the following:

- In Your repositories, choose New repository.
- On the navigation bar, choose Create new (+), and then choose New repository.

2. In the Create a new repository page, do the following:

- In the Repository name box, enter CodeDeployGitHubDemo.
- Select Public.

Note:

Selecting the default Public option means that anyone can see this repository. You can select the Private option to limit who can see and commit to the repository.

- Clear the Initialize this repository with a README check box. You will create a README.md file manually in the next step instead.
- Choose Create repository.

3. Follow the instructions for your local machine type to use the command line to create the repository

On local Windows machines:

1. From a command prompt running as an administrator, run the following commands, one at a time:

mkdir c:\temp\CodeDeployGitHubDemo

cd c:\temp\CodeDeployGitHubDemo

notepad README.md