1. **Upload React Project to GitHub**

* **Create a new React app using npx create-react-app myapp**Initialize a git repo and push to GitHub
* Use .gitignore to exclude node\_modules
* Create multiple branches: feature/navbar, feature/form
* Practice merge and pull requests (can use GitHub GUI)

**Create a new React app using npx create-react-app myapp**

**Install Node.js**

* First, you need to have **Node.js** installed on your system.
* Download it from https://nodejs.org.
* Installing Node.js will also install **npm (Node Package Manager)** automatically.

**Step 1**: Create React App

Make sure Node.js and npm are installed on your system.

Open the command line terminal.

**Run the command**:

npx create-react-app myapp

This will create a new React project folder named "myapp" with all necessary files and dependencies.

**Navigate into the project folder**:

cd myapp

**Start the React development server to verify the app runs**:

npm start

Initialize a git repo and push to GitHub

**Step 2**: Initialize Git Repository

**In the same project folder (myapp), initialize a local Git repository:**

git init

**Add all project files to the Git staging area**:

git add .

**Commit the files with a message**:

git commit -m "Initial commit - create React app"

**Step 3:**Create GitHub Repository

Go to GitHub website and create a new repository (do not initialize with README or other files).

Copy the repository URL (HTTPS or SSH).

**Step 4**: Link Local Repo to GitHub and Push

**Add the GitHub repository as a remote origin**:

git remote add origin https://github.com/yourusername/your-repo-name.git

**Push the local commits to GitHub**:

git push -u origin master

Use .gitignore to exclude node\_modules

* Create or open a .gitignore file in the root directory of your project.
* Add the line node\_modules/ to the .gitignore file. This tells Git to ignore the entire node\_modules directory and its contents.
* Save and close the .gitignore file.

**If node\_modules was previously added to Git tracking, remove it from the repository's index by running:**

git rm -r --cached node\_modules

**Commit the changes to record the update**:

git commit -m "Exclude node\_modulesusing .gitignore"

**Push the commit to the remote repository**:

git push origin main

Create multiple branches: feature/navbar, feature/form

* Open terminal in your project folder.

Switch to the main branch and update it:

git checkout main

git pull origin main

Create and switch to the first branch feature/navbar:

git checkout -b feature/navbar

Push the first branch to remote (optional):

git push -u origin feature/navbar

Switch back to the main branch:

git checkout main

Create and switch to the second branch feature/form:

git checkout -b feature/form

Push the second branch to remote (optional):

git push -u origin feature/form

Practice merge and pull requests (can use GitHub GUI)

**Create Repository (GitHub GUI)**

Go to GitHub → New Repository → give name → Create.

Directly Add File in main Branch

In the repo, click Add file → Create new file → name it file1.txt → write something → Commit to main.

**Create a New Branch in GitHub GUI**

In the repo → top left branch dropdown → type feature-branch → click Create branch.

Add Another File in New Branch

Switch to feature-branch.

Click Add file → Create new file → file2.txt → commit changes.

**Open Pull Request**

Go to Pull requests tab.

Click New pull request.

Select base: main ← compare: feature-branch.

Click Create pull request → give title & description.

**Merge Pull Request**

Review changes → click Merge pull request.

Confirm by clicking Confirm merge.

(Optional) Delete feature-branch.

**2.creating a pull request on GitHub after cloning a repository**

**Step 1: Clone the GitHub Repository**

* Open your terminal or Git GUI.
* Run the command: git clone <repository-url> to create a local copy of the repo.

**Step 2: Create a Branch Locally**

* Navigate to the cloned repo folder: cd <repo-folder>
* Create and switch to a new branch: git checkout -b my-feature-branch

**Step 3: Make Changes and Commit**

* Make your intended code or file changes in this branch.
* Stage changes: git add .
* Commit changes: git commit -m "Add feature or fix description"

**Step 4: Push Branch to GitHub**

* Push the branch: git push origin my-feature-branch

**Step 5: Open a Pull Request via GitHub GUI**

* Go to the GitHub repo page.
* Switch to the "Pull requests" tab.
* Click New Pull Request.
* Select the base branch (often main or master) and your feature branch.
* Provide a title and description for your pull request.
* Click Create Pull Request.

**Step 6: Review the Pull Request**

* Review the changes shown in GitHub.
* Add comments if needed for discussion or requested changes.

**Step 7: Merge the Pull Request**

* If all checks pass and review is complete, click Merge pull request.
* Choose merge type (merge commit, squash, or rebase).
* Confirm the merge.
* Optionally delete the branch after merging.

**Step 8: Update Your Local Repository**

* On your local machine, switch to the base branch: git checkout main
* Pull the latest changes: git pull