1. Command to get in some folder: cd foldername
2. Command to get out some folder: cd..   
    to get multiple times out: cd../..
3. To check if git is configured: git config –global user.name (then give your name)  
   To check if git is configured: git config –global user.email (then give your email)  
   If not then set their values: git config –global user.name “Your Name”

: git config –global user.email “Your email@example”

1. To check remote set up of git: git remote -v  
   If nothing is listed then: git remote add origin <https://github.com/yourusername/your-repo.git>
2. If you make a new file & want to add it to repository then first check the status of the whole folder, it will talk about the files that have changed: git status  
    git add filename.withextension

git commit -m “comment”

git push origin main   
 Refresh the repository and the files are added.

1. For the deletion of a file: git status  
    : git add -A  
    : git commit -m “comment”  
    : git push origin main
2. Npm (node package manager)- installs and then executes
3. Npx (does not install, but executes)- Node package execution
4. Make a react application through vite: npm create vite@latest

Name\_of\_project  
 ..

1. If by doing dir, node folder is not present then: npm i (it will install all the important files)
2. Spa (single page application) -> index.html
3. React himself makes a DOM (virtual DOM in memory) and does only the changes when the real DOM (browser DOM) gets changes and changes only that element.
4. <> </> this is called fragment, it is returned in the jsx as an empty kind of tag in the place of div, because only one entity can be returned in the jsx.
5. Whenever you are making a component take a capitalised name in vite or in react both, Good Practice to make file capitalised name.
6. createRoot-> create DOM just like that of browser, and changes only the changed part. But your browser changes the whole DOM that is called Page reload