1. **Setup GitHub repository of the ml/dl project for other developers to collaborate simultaneously.**
2. **Create folder of the project in local system & create virtual environment and activate** *virtualenv venv  
   venv\Scripts\activate*
3. **Clone github repository & sync with github to commit all our code using below codes in command line:***git init 🡪 initialize an empty repo on git*

*git add README.md 🡪 create a description file in local system & add in git repo*

*git commit -m "first commit" 🡪 commit the README.md file*

*git branch -M main 🡪 change branch to main before pushing what you’ve commited*

*git remote add origin* [*https://github.com/iamprashantjain/ml-ops\_laptop-price-predictor.git*](https://github.com/iamprashantjain/ml-ops_laptop-price-predictor.git) *--> add the address of the repo where its needed to be pushed*

*git push -u origin main 🡪 push the changes to git*

1. **Create python** *.gitignore* **file on github web interface to ignore some files which need not to be pushed to github repo like venv, .ipynb files etc**
2. **Create** *setup.py* **to build machine learning model as package which can be installed, distributed, used & also deployed &****create** *requirements.txt*  **file**
3. **Create a new folder “src” & create a new file inside it** *“\_\_init\_\_.py”* **to find this folder as a package.. so whenever find\_packages() function is called from setup.y file, it will look through all packages inside src folder**
4. **Add “-e .” flag in requirements.txt file to install the package directly from the current directory**
5. **Now run “pip install -r requirements.txt” & it will create a package file “***mlops\_project.egg-info***” inside src folder**
6. **All the new folders & code development will be done inside that folder**
7. **Now – “git add .”, check status “git status”, “git commit -m ‘updated setup file & created src folder’” & lastly push it to github repo “git push -u origin main”**
8. **Create a new folder “components” inside src folder & new file inside components “\_\_init\_\_.py” this will be all the modules of the project.. like data ingestion (reading data), data transformation (like applying OHE etc), model\_trainer (to train, evaluate model, r2 score etc)**
9. **Create another folder inside src folder “pipeline” & a file inside pipeline folder “train\_pipeline.py” which will be used to call all the modules in components folder, predict\_pipeline.py for making predictions & also “\_\_init\_\_.py” to import this folder**
10. **Since entire project implementation will be inside src folder.. so create 3 files:  
    logger.py – for logging  
    exception.py – for exception handling, write custom exceptions  
    utils.py – for basic helper utilities**
11. **Now push to github repo**