How to add a `requirements.txt` file. This is the standard way to tell other developers (or your future self) exactly which libraries are needed to run your code.

Method 1: The Automatic Way (Recommended)

If you have installed all the libraries in your virtual environment (`venv`) and everything is working, you can ask Python to generate this file for you automatically.

1. Open your VS Code terminal.
2. Ensure your virtual environment is active `(venv)`.
3. Run this command:

pip freeze > requirements.txt

This creates a file named `requirements.txt` in your main folder. It will look something like `pandas==2.0.3`, `numpy==1.24.3`, etc., listing every single library and its specific version.

Method 2: The Manual Way (Cleaner)

Sometimes `pip freeze` includes too much "noise" (dependencies of dependencies). A professional `requirements.txt` often lists only the top-level libraries you actually imported.

Create a new file named `requirements.txt` in your **root folder** (outside of `src`, `data`, etc.) and paste this content:

```
Plaintext
pandas
numpy
scikit-learn
joblib
matplotlib
streamlit
```

(Note: You can specify versions like `pandas>=2.0.0` if you want to be strict, but for a personal project, just the names are fine.)

How to push your project to GitHub. This assumes you already have Git installed on your computer.

Step 1: Create a Repository on GitHub

1. Log in to GitHub.com.
2. Click the **+** icon in the top-right corner and select **New repository**.
3. **Repository name:** `student_grade_predictor` (or whatever you named your folder).
4. **Privacy:** Choose **Public** (so you can show it to recruiters) or **Private**.
5. **Initialize this repository:** Leave all checkboxes (Add README, .gitignore) **UNCHECKED**. We already made these files locally.
6. Click **Create repository**.
7. **Copy the URL:** You will see a link that looks like `https://github.com/YOUR_USERNAME/student_grade_predictor.git`.
8. Copy it.

Step 2: Prepare Your Local Folder

Open your VS Code terminal (ensure you are inside your project folder) and run these commands one by one:

1) **Initialize Git:** This tells Git to start tracking this folder.

   git init

2) **Add a `.gitignore` file (Crucial):** You do **NOT** want to upload your virtual environment (`venv/`) or huge dataset files to GitHub. Create a file named `.gitignore` (no name, just extension) and paste this inside:

   venv/
   __pycache__/
   **.DS_Store*

3) **Stage your files:** This prepares all your files (except the ignored ones) for upload.

   git add .

4) **Commit your changes:** This saves a "snapshot" of your code.
   Bash
   git commit -m "Initial commit - Student Grade Predictor"

Step 3: Connect and Push

Now we link your computer to the GitHub page you just created.

A. **Link the remote repository:** (Paste the URL you copied in Step 1)

git remote add origin https://github.com/YOUR_USERNAME/student_grade_predictor.git

B. **Rename branch to main (Standard practice):**

git branch -M main

C. **Push the code:** This actually uploads the files.

git push -u origin main

Step 4: Verify

Refresh your GitHub page. You should now see all your files (`src/`, `main.py`, `README.md`) listed there.