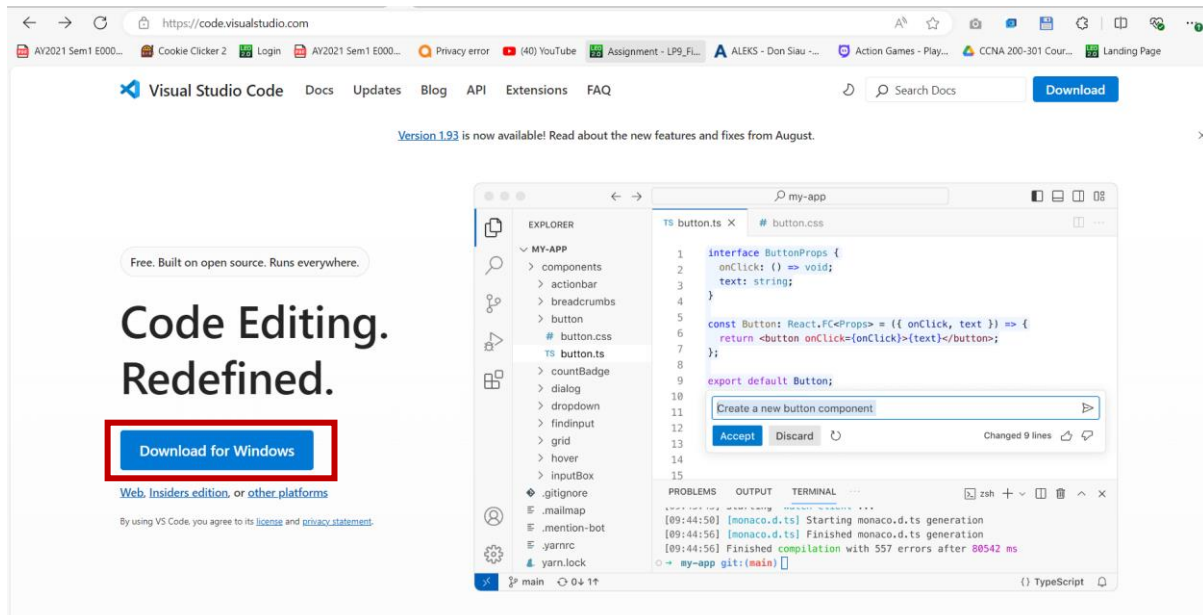


GitHub Link: https://github.com/DonSiau/SQL_Grading.git

Prerequisites:

1)Visual Studio Code (VSC):

https://code.visualstudio.com/?wt.mc_id=vscom_downloads



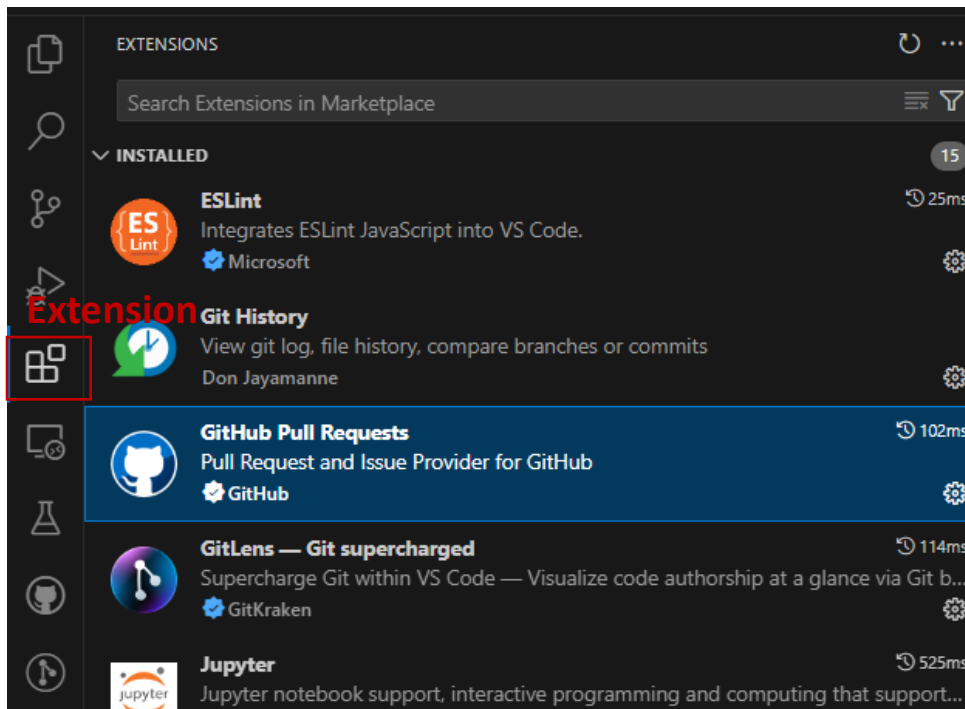
2)Python 3.12.3 [Python Release Python 3.12.3 | Python.org](https://www.python.org/release/python-3.12.3/)

Scroll down on the website to see download files, and download the version highlighted in red

A screenshot of the Python 3.12.3 release page on python.org. The page shows the 'Files' section with a table of download links. The row for 'Windows installer (64-bit)' is highlighted with a red box. The table has columns for Version, Operating System, Description, MD5 Sum, File Size, GPG, Sigstore, and SBOM.

Version	Operating System	Description	MD5 Sum	File Size	GPG	Sigstore	SBOM
Gzipped source tarball	Source release		3c5498a34d5226c9b746b1199f0bf2d9	25.9 MB	SIG	.sigstore	SPDX
XZ compressed source tarball	Source release		8defb33f0c37aa4bdd3a38ba52abde4e	19.7 MB	SIG	.sigstore	SPDX
macOS 64-bit universal2 installer	macOS	for macOS 10.9 and later	6114a3bb9b288f23ab38dbbb959be1bf	43.6 MB	SIG	.sigstore	
Windows installer (64-bit)	Windows	Recommended	c86949710e0471a065db970290819489	25.5 MB	SIG	.sigstore	
Windows installer (32-bit)	Windows		a95c4fbdce0b6a22ca7cfb450f57c616	24.2 MB	SIG	.sigstore	
Windows installer (ARM64)	Windows	Experimental	ef016521b5a147f3ded730801d36a350	24.7 MB	SIG	.sigstore	
Windows embeddable package (64-bit)	Windows		38cce2bf5b4de76db19a31f46a0720de	10.5 MB	SIG	.sigstore	
Windows embeddable package (32-bit)	Windows		65d873c723db66d6746e9872df5a715e	9.4 MB	SIG	.sigstore	
Windows embeddable package (ARM64)	Windows		3229271cac55ff913aad1bb46ebc9931	9.8 MB	SIG	.sigstore	

3)Python and GitHub Pull Requests extensions: Download from VSC extensions on the Extension panel. After installed, click on the extensions and enable them



Python v2024.14.1

Microsoft microsoft.com | 136,489,645 | ★★★★★ (597)

Python language support with extension access points for IntelliSense (Pylance), Debugging (Python Debugger), linting, formatting, refactoring, unit tests, ...

[Disable](#) [Uninstall](#) [Switch to Pre-Release Version](#) ☒ Auto Update



GitHub Pull Requests v0.96.0

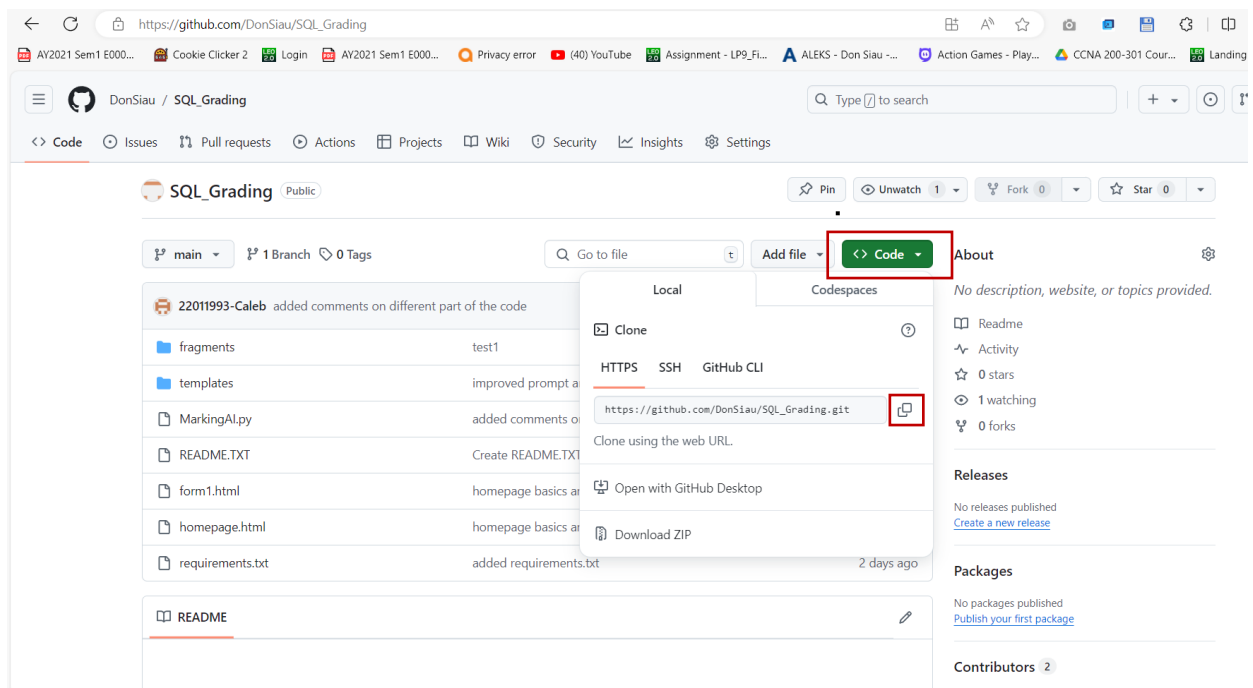
GitHub github.com | 24,352,936 | ★★★★★ (167)

Pull Request and Issue Provider for GitHub

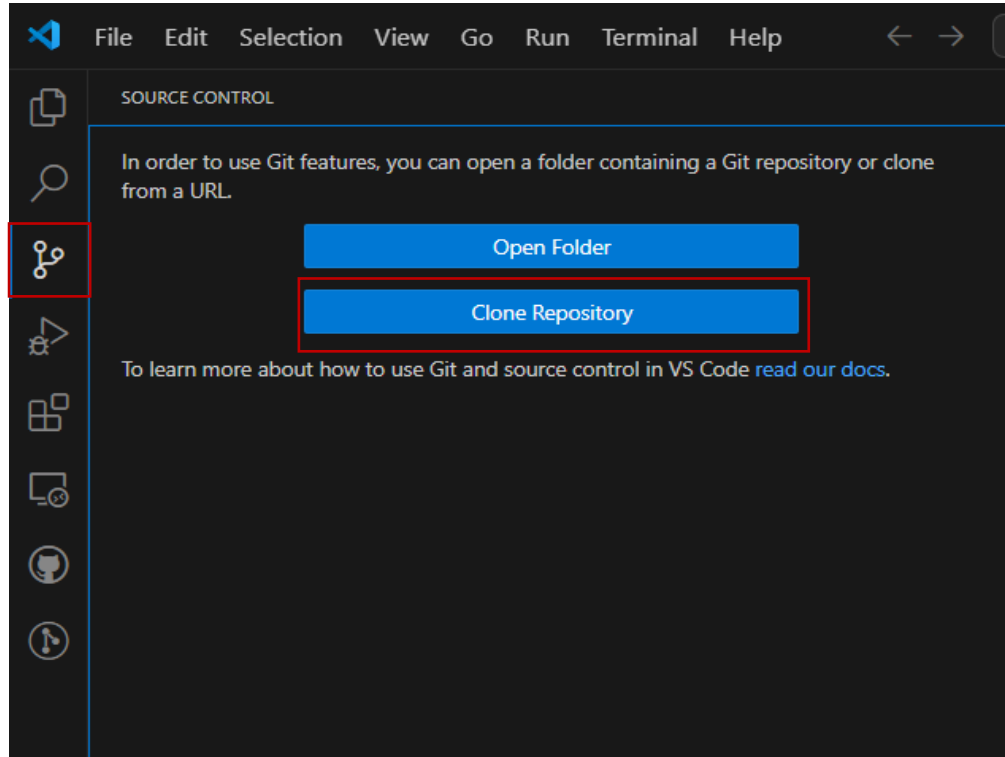
[Disable](#) [Uninstall](#) [Switch to Pre-Release Version](#) ☒ Auto Update

Installation

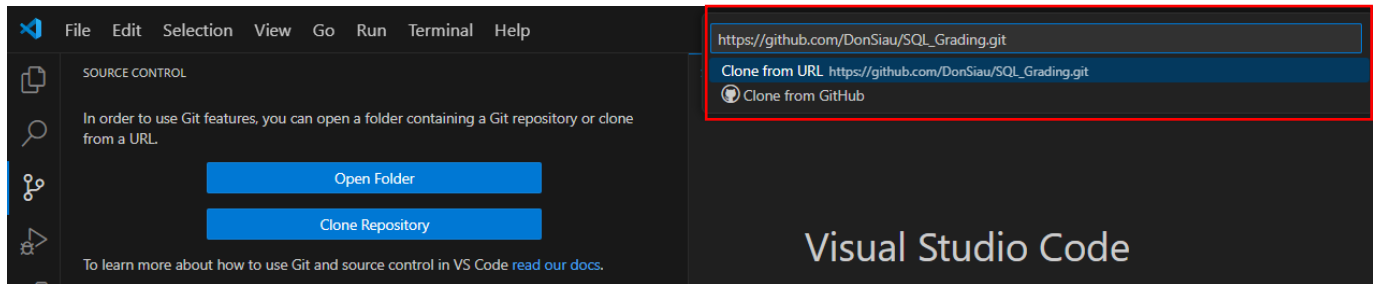
Step 1: Go to the GitHub link. Click on “Code”, then copy the HTTPS link



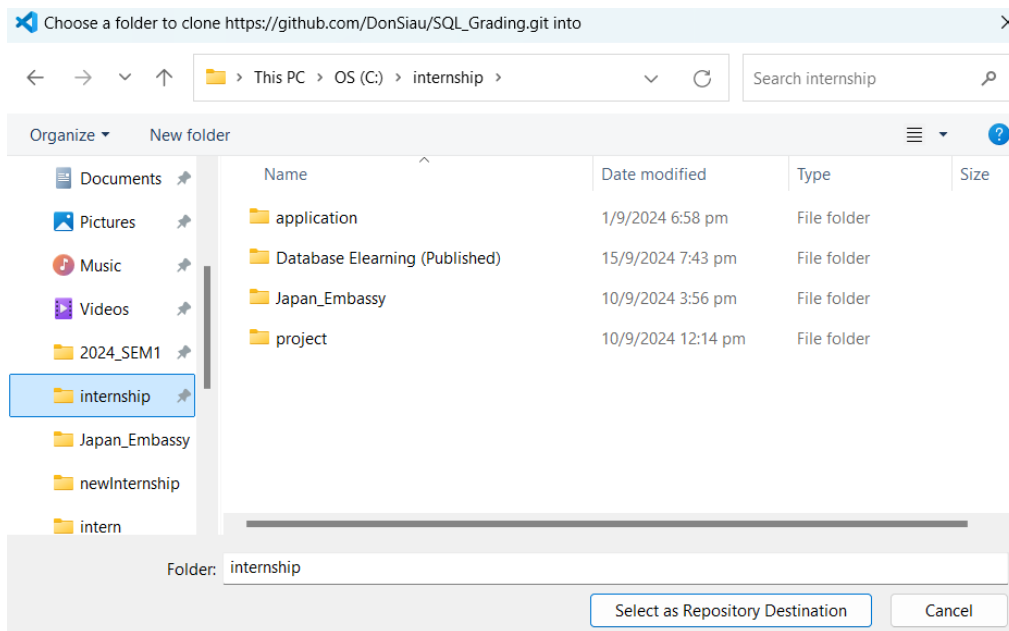
Step 2: on VSC the repository. Click the “Source control” panel and click “Clone Repository”



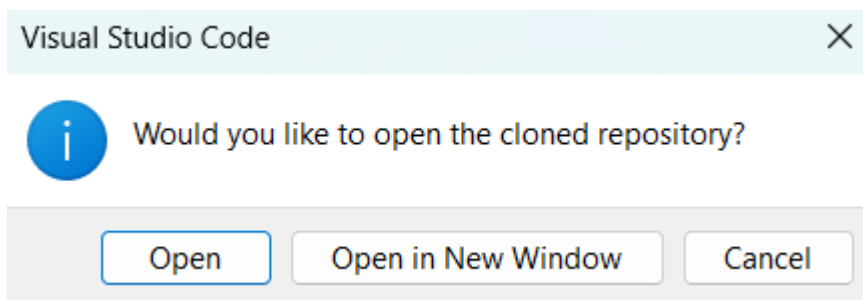
Step 3: Paste the HTTPS link into the search bar. Then press enter



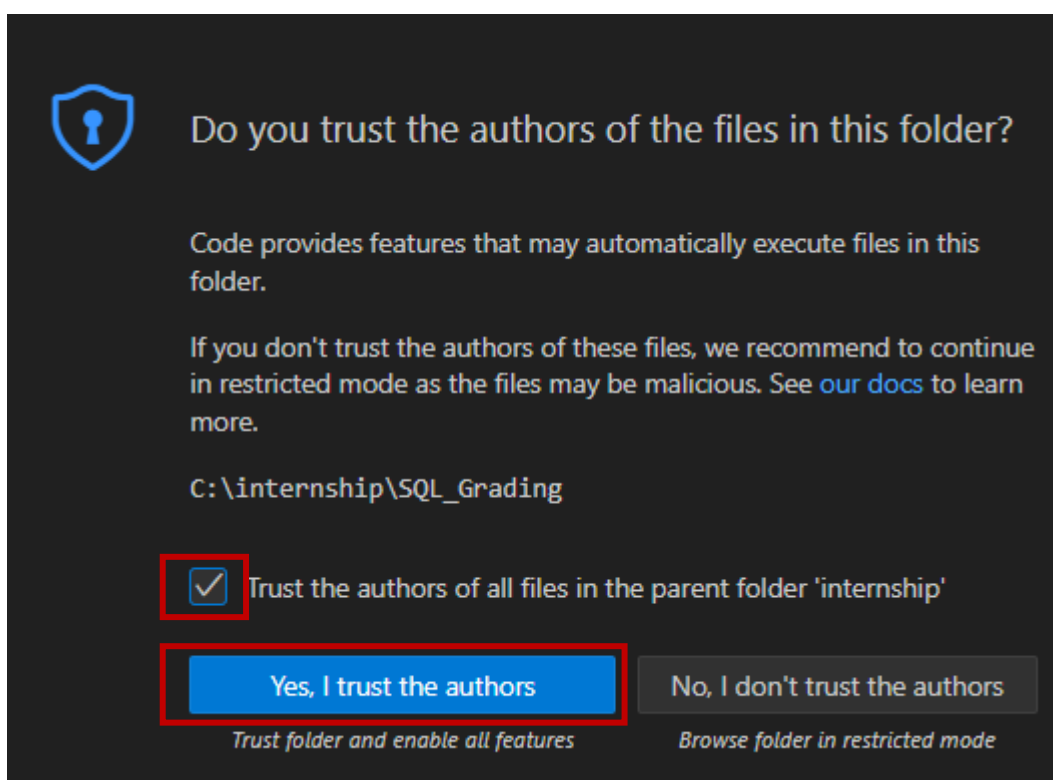
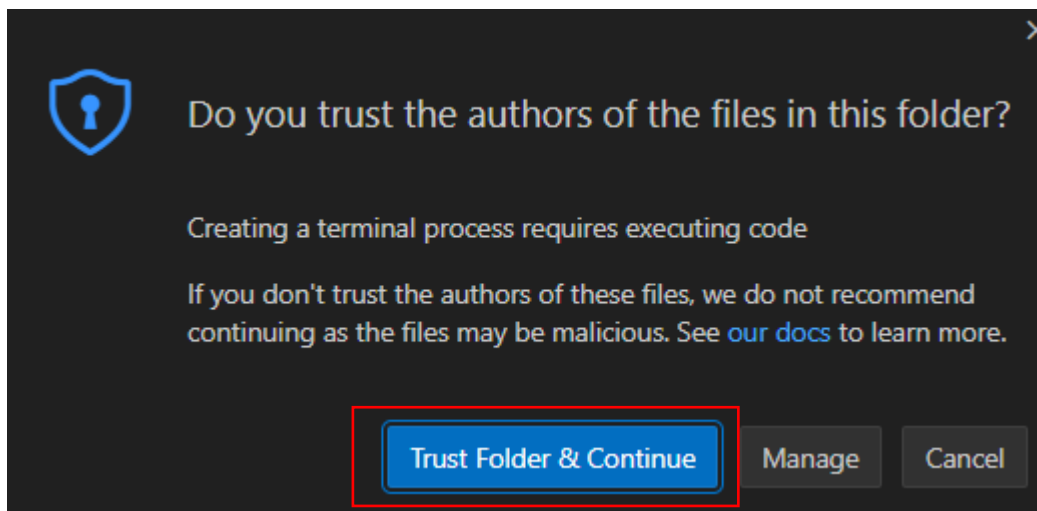
Step 4: Choose a folder to clone the repository to. Then click “Select as Repository Destination”



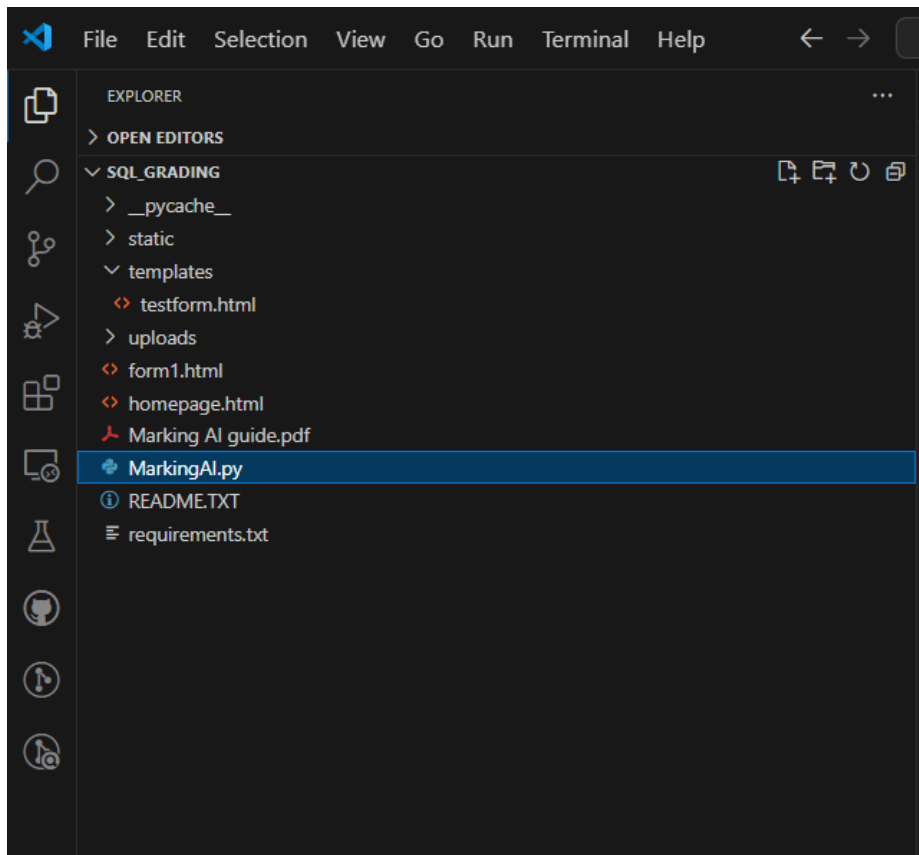
Step 5: On VSC this prompt appears. press open.



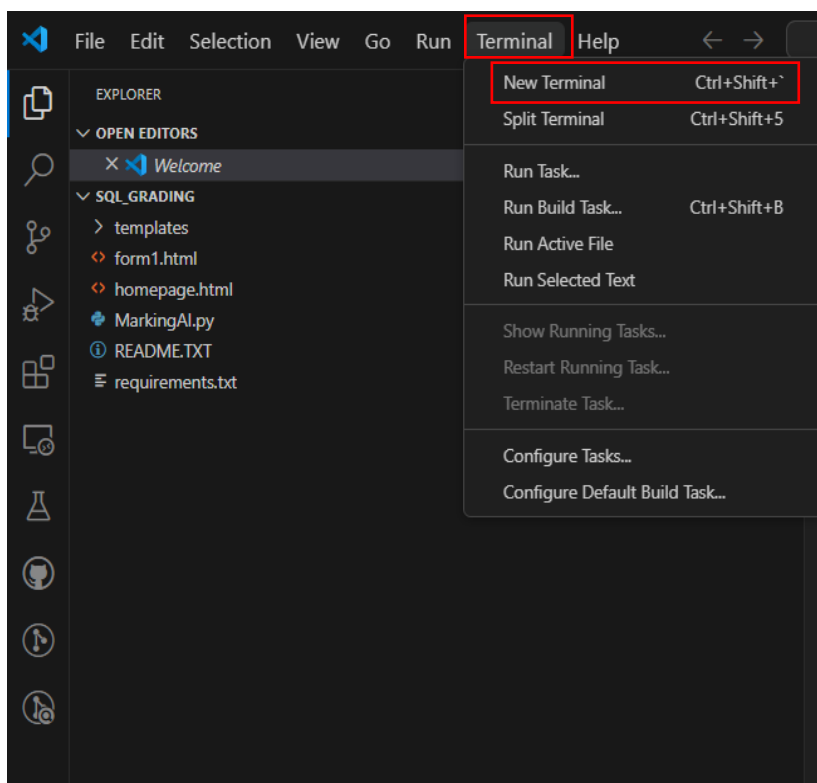
Step 6: On VSC these prompts appear. Click that you trust the authors like the images below



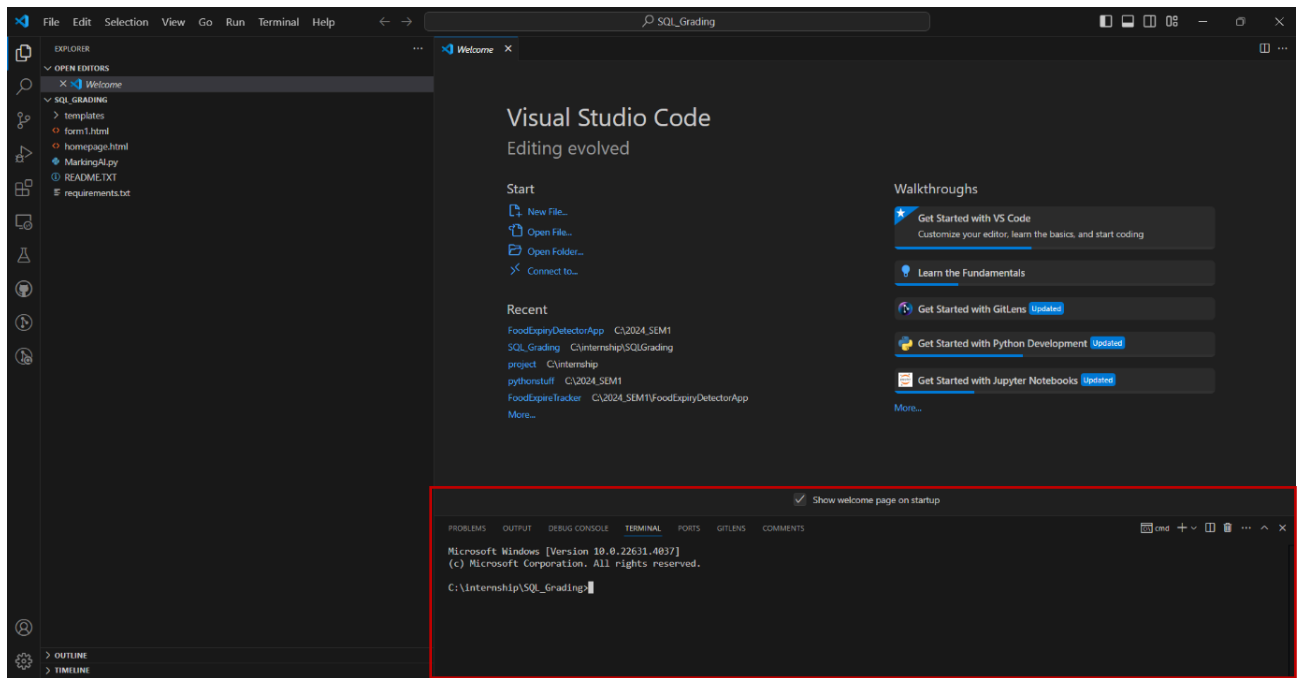
Now on the sidebar the project has been imported



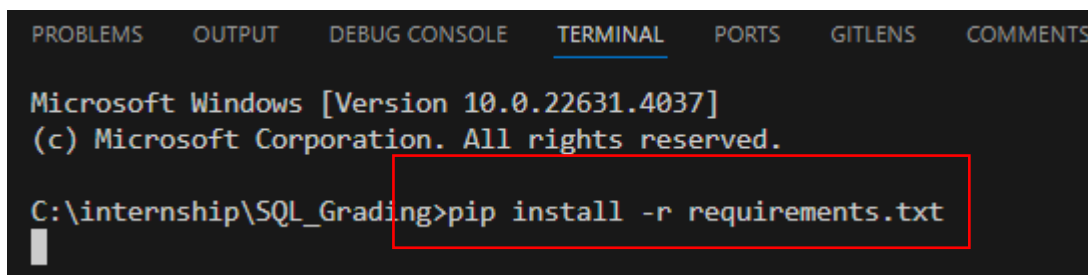
Step 7: click “Terminal” and “New Terminal”



A terminal appears at the bottom of screen



Step 8: Input “pip install -r requirements.txt” and press enter (do it in the SQL_Grading directory)



If it is not working, manually install the requirements on the SQL_Grading directory by running the commands one by one

pip install Flask==3.0.3

pip install pandas==2.2.2

pip install Requests==2.32.3

pip install openpyxl==3.1.5

Below is an example for installing Flask and Pandas manually Do the same for the other requirements if the “pip install -r requirements.txt” command does not work

```
C:\internship\SQL_Grading>pip install Flask==3.0.3
```

```
Requirement already satisfied: Flask==3.0.3 in c:\python312\lib\site-packages (3.0.3)
Requirement already satisfied: Werkzeug>=3.0.0 in c:\python312\lib\site-packages (from Flask==3.0.3) (3.0.3)
Requirement already satisfied: Jinja2>=3.1.2 in c:\python312\lib\site-packages (from Flask==3.0.3) (3.1.4)
Requirement already satisfied: itsdangerous>=2.1.2 in c:\python312\lib\site-packages (from Flask==3.0.3) (2.2.0)
Requirement already satisfied: click>=8.1.3 in c:\python312\lib\site-packages (from Flask==3.0.3) (8.1.7)
Requirement already satisfied: blinker>=1.6.2 in c:\python312\lib\site-packages (from Flask==3.0.3) (1.8.2)
Requirement already satisfied: colorama in c:\python312\lib\site-packages (from click>=8.1.3->Flask==3.0.3) (0.4.6)
Requirement already satisfied: MarkupSafe>=2.0 in c:\python312\lib\site-packages (from Jinja2>=3.1.2->Flask==3.0.3) (2.1.5)
```

```
[notice] A new release of pip is available: 24.0 -> 24.2
```

```
[notice] To update, run: python.exe -m pip install --upgrade pip
```

```
C:\internship\SQL_Grading>
```

```
C:\internship\SQL_Grading>pip install pandas==2.2.2
```

```
Requirement already satisfied: pandas==2.2.2 in c:\users\22001326\appdata\roaming\python\python312\site-packages (2.2.2)
Requirement already satisfied: numpy>=1.26.0 in c:\users\22001326\appdata\roaming\python\python312\site-packages (from pandas==2.2.2) (1.26.4)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\22001326\appdata\roaming\python\python312\site-packages (from pandas==2.2.2) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in c:\users\22001326\appdata\roaming\python\python312\site-packages (from pandas==2.2.2) (2024.1)
Requirement already satisfied: tzdata>=2022.7 in c:\users\22001326\appdata\roaming\python\python312\site-packages (from pandas==2.2.2) (2024.1)
Requirement already satisfied: six>=1.5 in c:\python312\lib\site-packages (from python-dateutil>=2.8.2->pandas==2.2.2) (1.16.0)
```

Now the requirements are installed (note in the image the requirements were already preinstalled, so it would look different on your end)

```
Microsoft Windows [Version 10.0.22631.4037]
(c) Microsoft Corporation. All rights reserved.
```

```
C:\internship\SQL_Grading>pip install -r requirements.txt
```

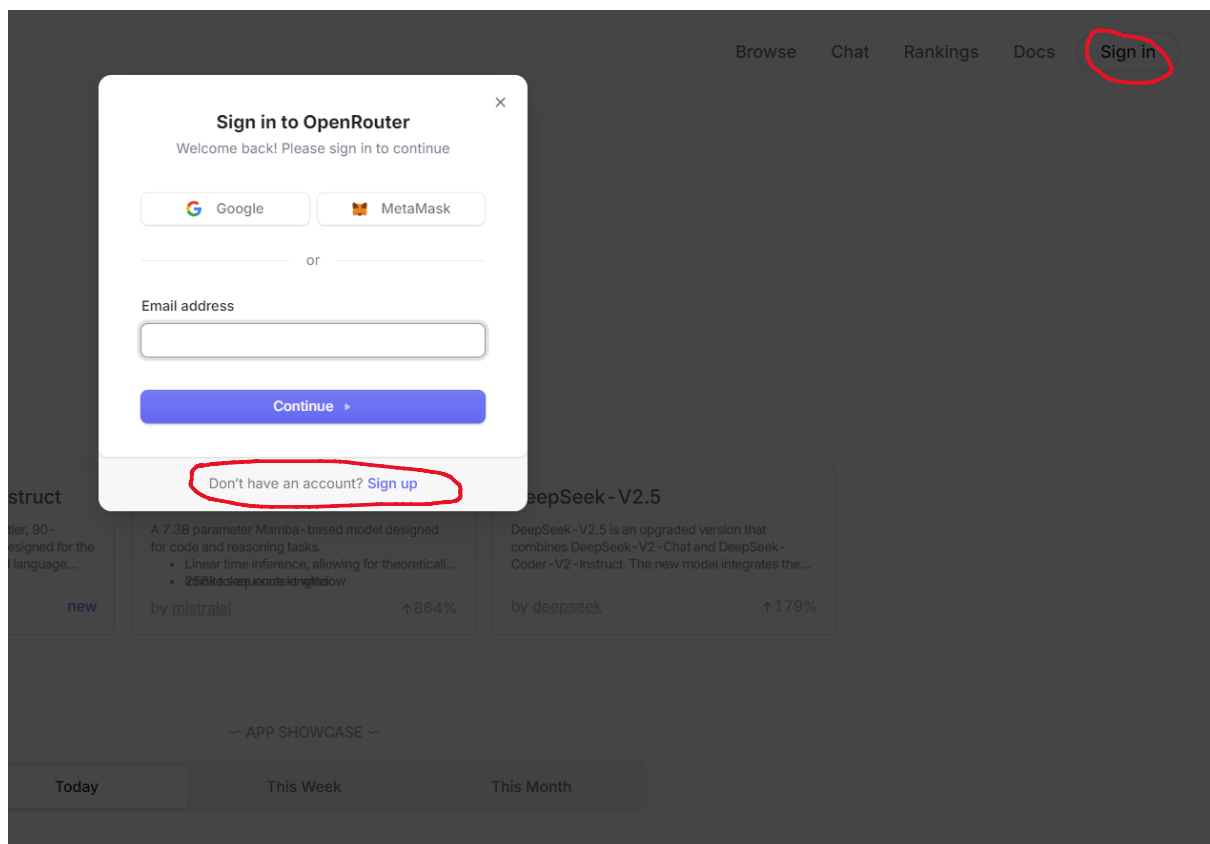
```
Requirement already satisfied: Flask==3.0.3 in c:\python312\lib\site-packages (from -r requirements.txt (line 1)) (3.0.3)
Requirement already satisfied: pandas==2.2.2 in c:\users\22001326\appdata\roaming\python\python312\site-packages (from -r requirements.txt (line 2)) (2.2.2)
Requirement already satisfied: Requests==2.32.3 in c:\users\22001326\appdata\roaming\python\python312\site-packages (from -r requirements.txt (line 3)) (2.32.3)
Requirement already satisfied: Werkzeug>=3.0.0 in c:\python312\lib\site-packages (from Flask==3.0.3->-r requirements.txt (line 1)) (3.0.3)
Requirement already satisfied: Jinja2>=3.1.2 in c:\python312\lib\site-packages (from Flask==3.0.3->-r requirements.txt (line 1)) (3.1.4)
Requirement already satisfied: itsdangerous>=2.1.2 in c:\python312\lib\site-packages (from Flask==3.0.3->-r requirements.txt (line 1)) (2.2.0)
Requirement already satisfied: click>=8.1.3 in c:\python312\lib\site-packages (from Flask==3.0.3->-r requirements.txt (line 1)) (8.1.7)
Requirement already satisfied: blinker>=1.6.2 in c:\python312\lib\site-packages (from Flask==3.0.3->-r requirements.txt (line 1)) (1.8.2)
Requirement already satisfied: numpy>=1.26.0 in c:\users\22001326\appdata\roaming\python\python312\site-packages (from pandas==2.2.2->-r requirements.txt (line 2)) (1.26.4)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\22001326\appdata\roaming\python\python312\site-packages (from pandas==2.2.2->-r requirements.txt (line 2)) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in c:\users\22001326\appdata\roaming\python\python312\site-packages (from pandas==2.2.2->-r requirements.txt (line 2)) (2024.1)
Requirement already satisfied: tzdata>=2022.7 in c:\users\22001326\appdata\roaming\python\python312\site-packages (from pandas==2.2.2->-r requirements.txt (line 2)) (2024.1)
Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\22001326\appdata\roaming\python\python312\site-packages (from Requests==2.32.3->-r requirements.txt (line 3)) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in c:\python312\lib\site-packages (from Requests==2.32.3->-r requirements.txt (line 3)) (3.8)
Requirement already satisfied: urllib3<3,>=1.21.1 in c:\python312\lib\site-packages (from Requests==2.32.3->-r requirements.txt (line 3)) (2.2.1)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\22001326\appdata\roaming\python\python312\site-packages (from Requests==2.32.3->-r requirements.txt (line 3)) (2023.7.22)
Requirement already satisfied: colorama in c:\python312\lib\site-packages (from click>=8.1.3->Flask==3.0.3->-r requirements.txt (line 1)) (0.4.6)
Requirement already satisfied: MarkupSafe>=2.0 in c:\python312\lib\site-packages (from Jinja2>=3.1.2->Flask==3.0.3->-r requirements.txt (line 1)) (2.1.5)
Requirement already satisfied: six>=1.5 in c:\python312\lib\site-packages (from python-dateutil>=2.8.2->pandas==2.2.2->-r requirements.txt (line 2)) (1.16.0)
```

```
[notice] A new release of pip is available: 24.0 -> 24.2
```

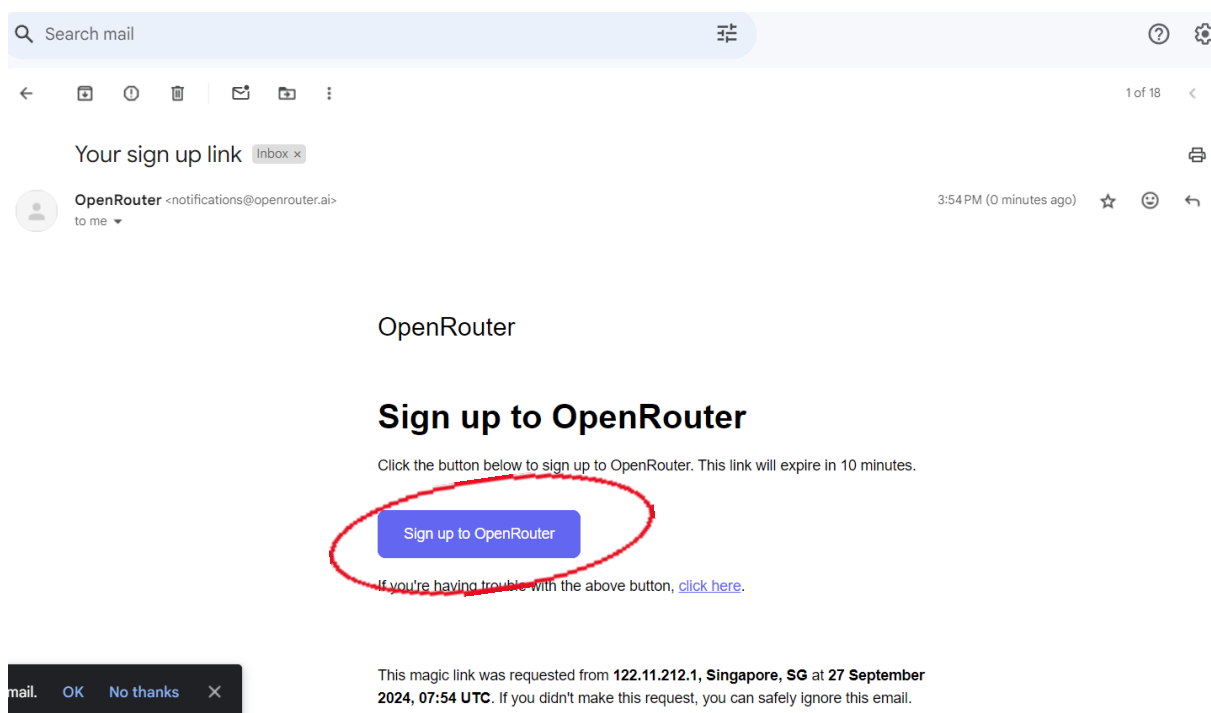
```
[notice] To update, run: python.exe -m pip install --upgrade pip
```

```
C:\internship\SQL_Grading>
```

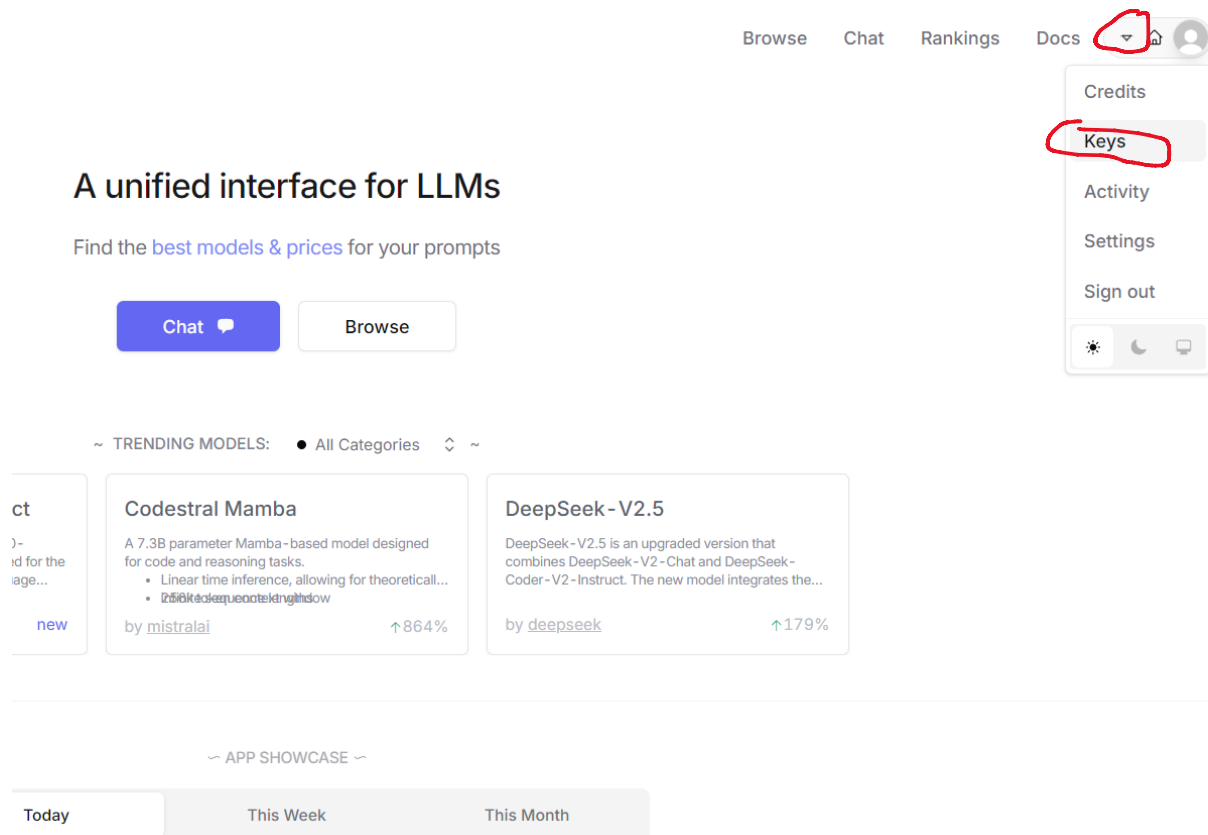

Step 9: the next few steps are for the AI model. Go to this website, which is the service provider for the AI ([OpenRouter](#)). Select Sign In and then signup for the service



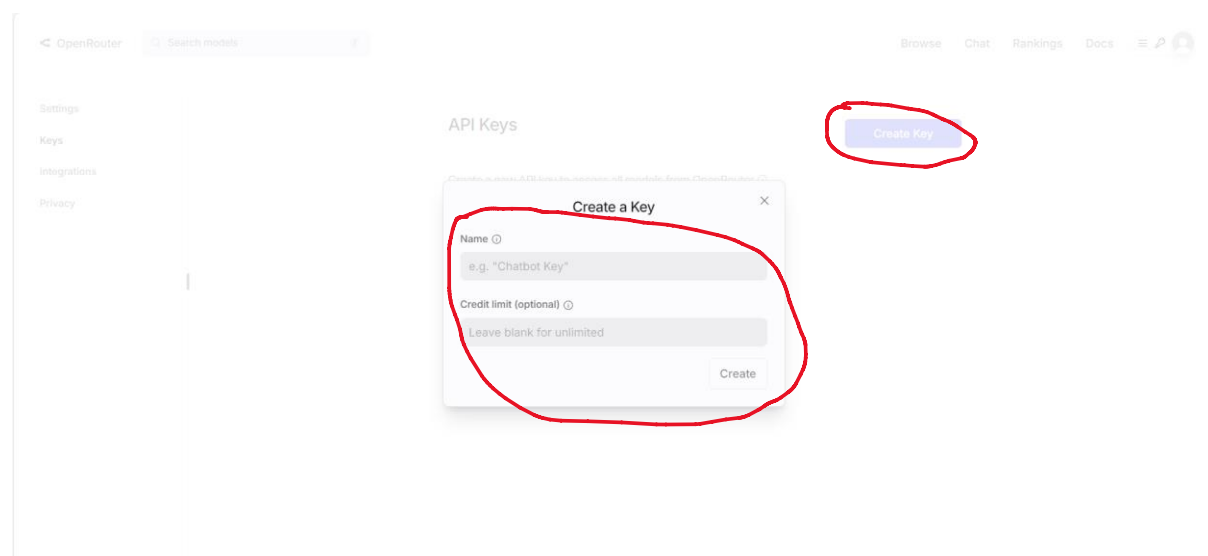
After signing up, email should be sent for you to sign in. click the sign up button, which should sign you in



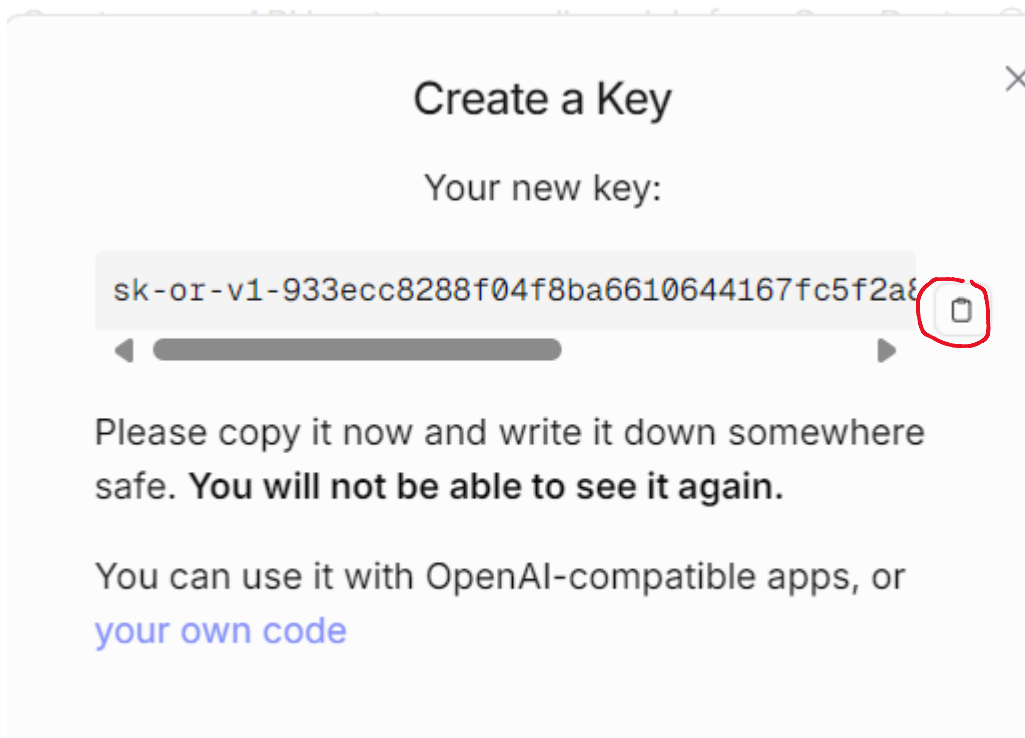
Step 10: now that you are signed in you need to get a api key. Go to the icon in the top right and select keys



Step 11: generate your api key. Click “Create Key”, the blue button highlighted in red, and then name your key before clicking create



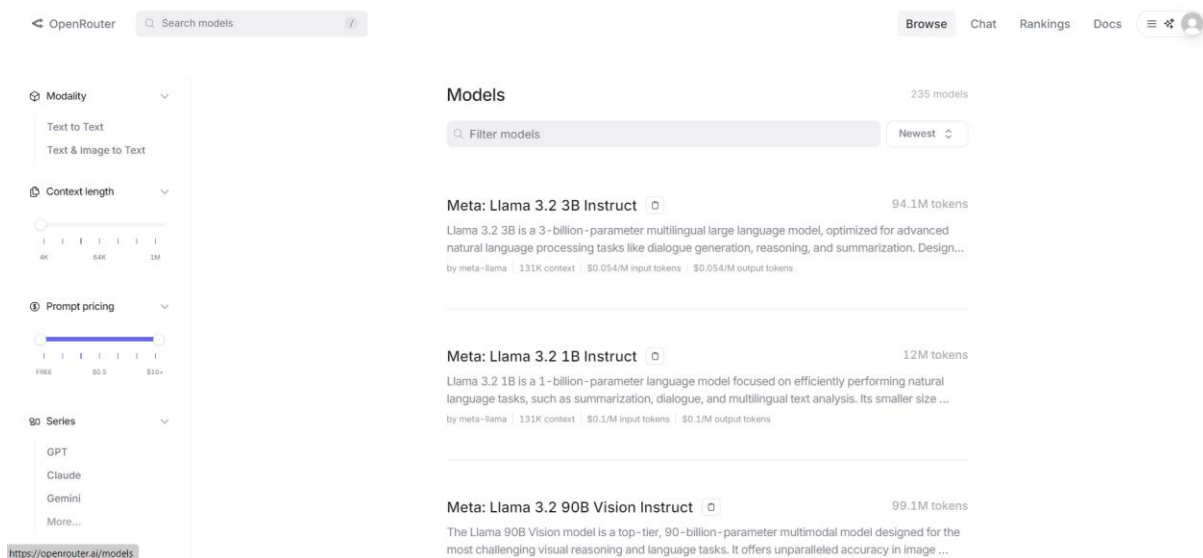
Step 12: copy the Key



Step 13: on the application on line 19 in MarkingAI.py, replace the key with your own key

```
7 from openpyxl.utils.dataframe import dataframe_to_rows
8 from openpyxl.styles import numbers
9
10 app = Flask(__name__)
11 app.secret_key = "secret_key" # Required to use Flask's flash messaging
12
13 # Use absolute path for UPLOAD_FOLDER
14 UPLOAD_FOLDER = os.path.join(os.path.dirname(os.path.abspath(__file__)), 'uploads')
15 app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER
16
17 URL = "https://openrouter.ai/api/v1/chat/completions"
18 HEADERS = {
19     "Authorization": f"Bearer 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", # School should use their own key
```

Step 14: click on browse to choose models. I have some general tips for picking a model



1. must use a "Instruct" model, chat models do not work. Do not use a vision model

2. more parameters are typically better, (405B is better than 70B)

Meta: Llama 3.1 405B Instruct

699M tokens

The highly anticipated 400B class of Llama3 is here! Clocking in at 128k context with impressive eval scores, the Meta AI team continues to push the frontier of open-source LLMs. Meta's latest ...

by meta-llama | 131K context | \$1.79/M input tokens | \$1.79/M output tokens

Meta: Llama 3.1 70B Instruct

6.39B tokens

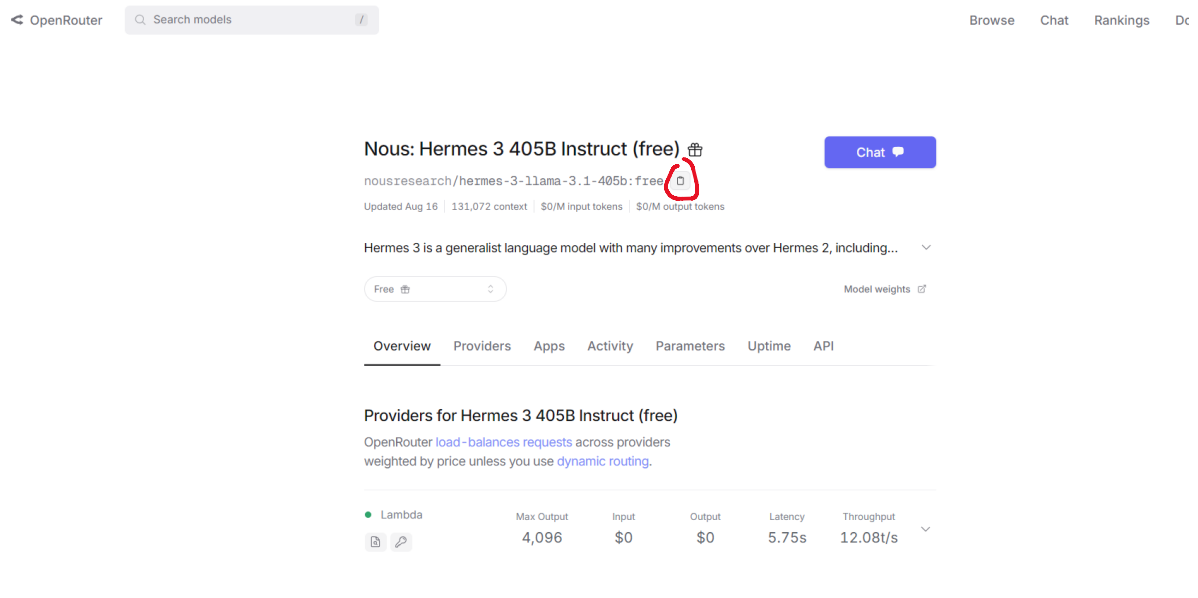
Technology/Web (#6) Programming (#7) Science (#7) Finance (#9) Programming/Scripting (#9)

Meta's latest class of model (Llama 3.1) launched with a variety of sizes & flavors. This 70B instruct-tuned version is optimized for high quality dialogue usecases. It has demonstrated strong...

by meta-llama | 131K context | \$0.3/M input tokens | \$0.3/M output tokens

3. price. Price is shown above. For example \$0.3/M output and input tokens. A word is 3-4 tokens, so basically it is \$0.3 each per 250,000 words of input and output


Step 15: the model we used is [Hermes 3 405B Instruct \(free\) - API, Providers, Stats | OpenRouter](#), however you may use other models such as [Llama 3.1 405B Instruct - API, Providers, Stats | OpenRouter](#) or other ones. Simply copy the name of the model



OpenRouter Search models /



Browse Chat Rankings Dc

Nous: Hermes 3 405B Instruct (free)

nousresearch/hermes-3-llama-3.1-405b:free 

Updated Aug 16 | 131,072 context | \$0/M input tokens | \$0/M output tokens

Hermes 3 is a generalist language model with many improvements over Hermes 2, including...

Free  Model weights 

Overview Providers Apps Activity Parameters Uptime API

Providers for Hermes 3 405B Instruct (free)

OpenRouter load-balances requests across providers weighted by price unless you use dynamic routing.

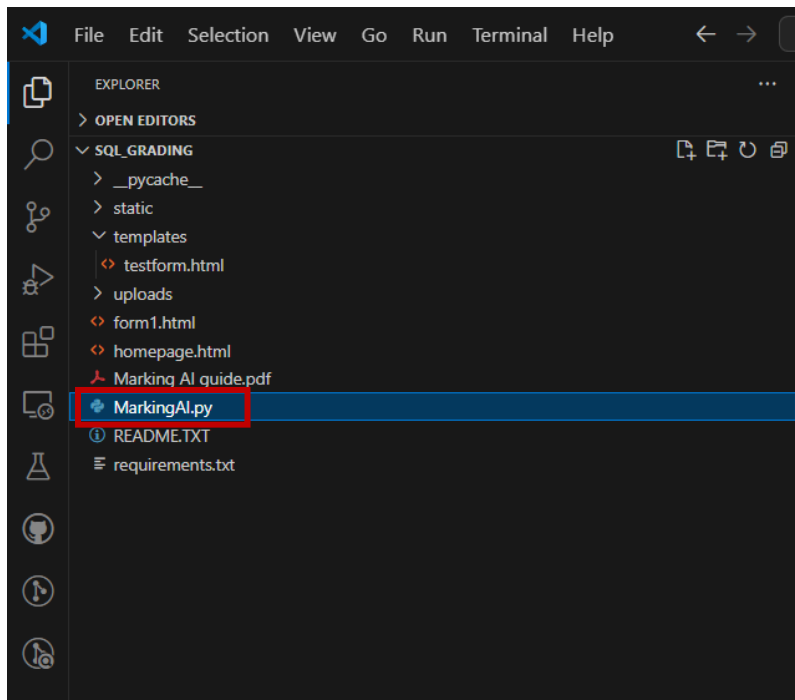
	Max Output	Input	Output	Latency	Throughput
Lambda	4,096	\$0	\$0	5.75s	12.08t/s

And then replace the model in line 134 with what you copied

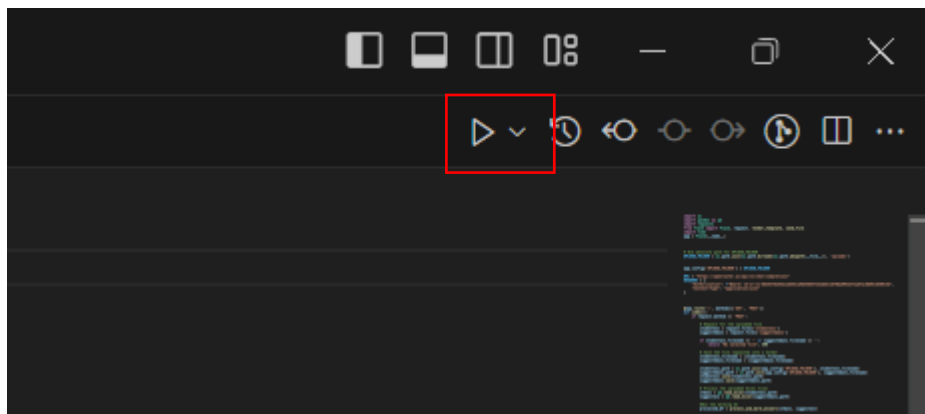
```
132 # Run the AI
133 payload = {
134     "model": "nousresearch/hermes-3-llama-3.1-405b:free", # Make sure to use an instruct model, not a chat model https://
135     "messages": [
136         {"role": "user", "content": prompt}
137     ]
138 }
```

Running the application

Step 9: Select the “MarkingAI.py”



Press the run icon on the top right of the screen

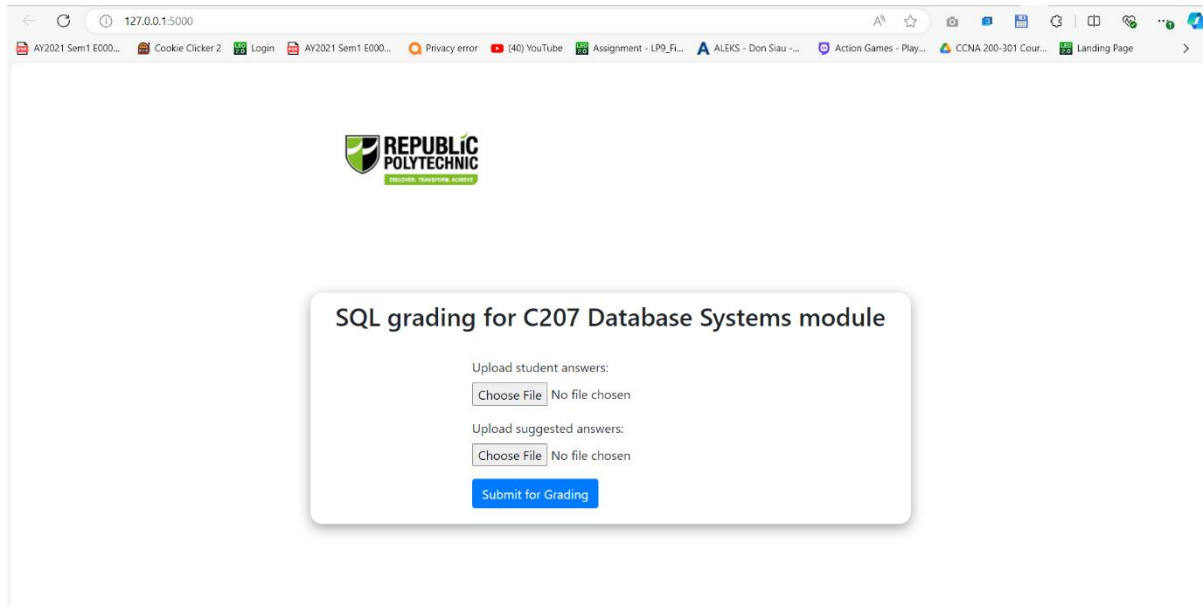


Step 10: On the bottom of the screen in the this should run on the terminal. Hold the ctrl key whilst you click the link highlighted in the red box. (the address of the link might be different on your end, but it should still work)

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS COMMENTS
Microsoft Windows [Version 10.0.22631.4037]
(c) Microsoft Corporation. All rights reserved.

C:\internship\SQL_Grading>C:/Python312/python.exe c:/internship/SQL_Grading/MarkingAI.py
* Serving Flask app 'MarkingAI'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5000
* Running on http://10.175.21.175:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 122-589-219
```

On your browser the application will appear



Step 11: Find the csv file for each problem statement. It contains the student answers. Find the xlsx file which contains the suggested answers. For each lesson, find the files that correspond with one another. For example:

For lesson 1, find:

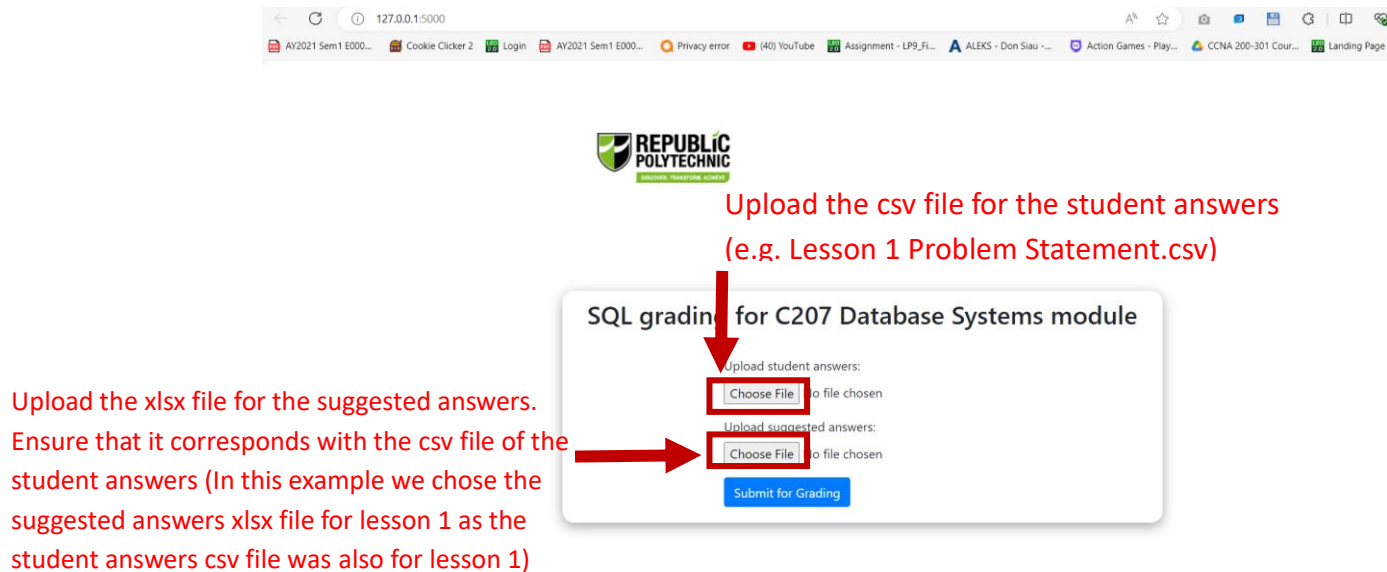
 Lesson 1 problem statement.csv	16/9/2024 11:45 am	Microsoft Excel Co...
 suggestedAnswerL1.xlsx	16/9/2024 11:29 am	Microsoft Excel W...

For lesson 3, find:

 Lesson 3 problem statement.csv	16/9/2024 11:45 am	Microsoft Excel Co...
 suggestedAnswerL3.xlsx	17/9/2024 12:24 pm	Microsoft Excel W...

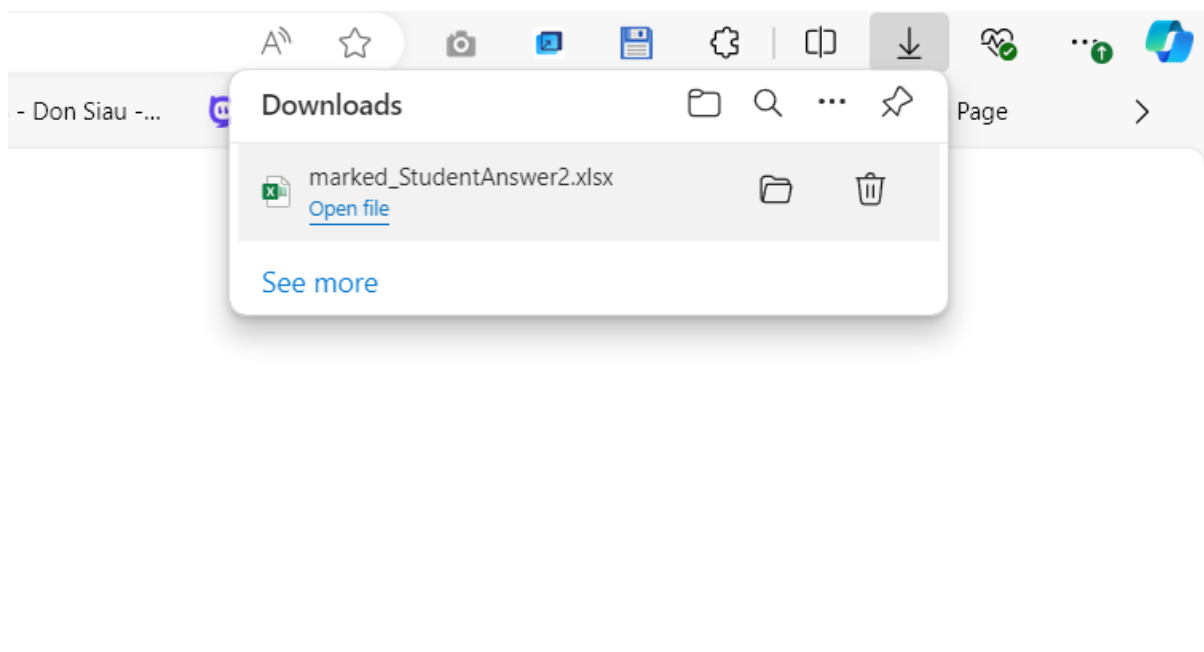
and the same for the other problem statements

Step 12: Now we will submit the quiz results for grading. Click the “Choose File” icons to do so

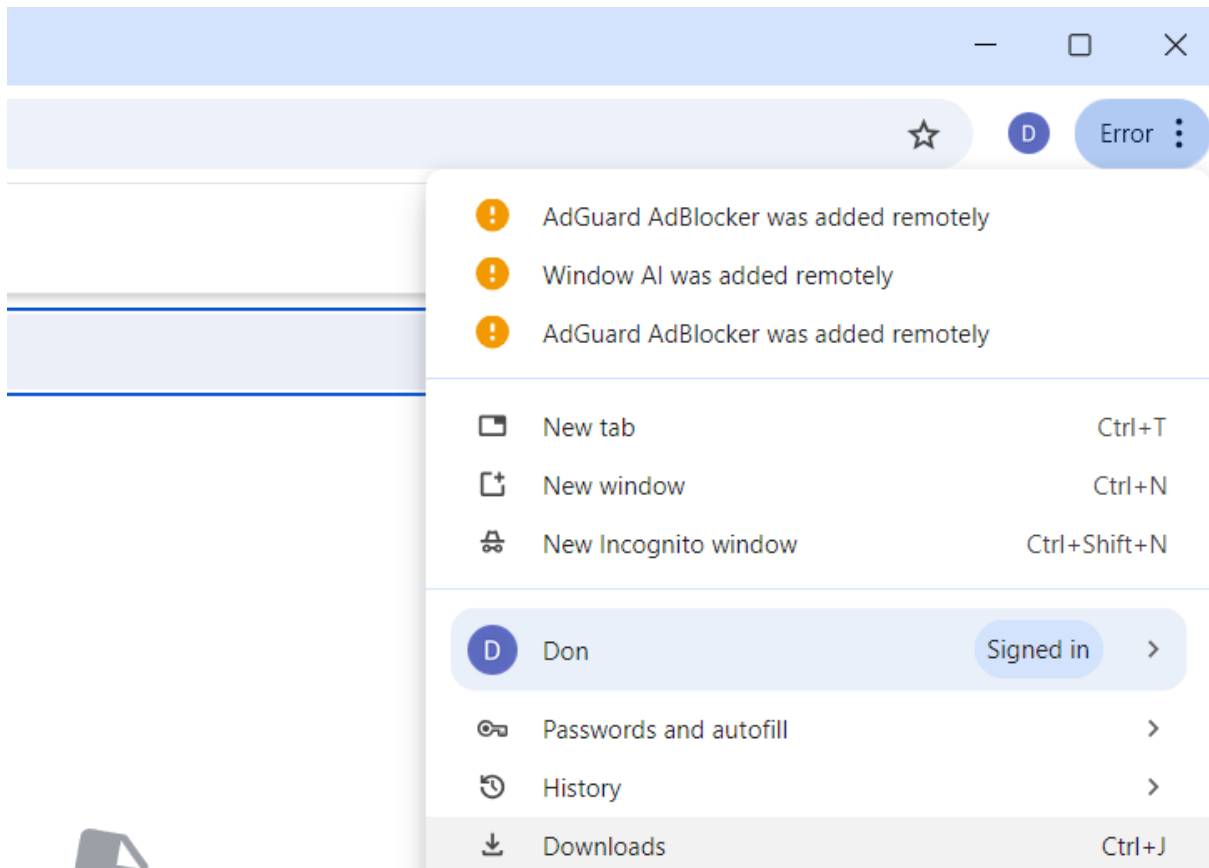


Step 13: Wait for the AI to run and once it is done your browser will download a marked excel version of the student's answers. (May take a while to download).

Edge browser



Chrome browser



Step 14: download it and view it. 2 marks means correct answer 1 mark partially correct 0 marks the answer is blank. The left side is the student Info and their answers, the right side are the marks for each individual question, total marks and percentage

marked_Lesson 1 Problem Statement (6).xlsx No Label • Saved to this PC

File Home Insert Page Layout Formulas Data Review View Automate Help

Clipboard Font Alignment Number Conditional Formatting

A1 Timestamp

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Timestamp	Class	StudentID	Name	Q1)	Q2)	Q3)	marked->	Q1)_Mark	Q2)_Mark	Q3)_Mark	total_mar	Total Score	%
2	2024/09/1	W66C	22001326						0	0	0	0	0.00%	
3	2024/09/1	W66E	22001329		SELECT titl	SELECT pu	SELECT isbn, title, lan		1	1	1	3	50.00%	
4	2024/09/1	W66E	22001332		SELECT titl	SELECT pu	SELECT isbn, title, lan		2	2	2	6	100.00%	
5	2024/09/1	W66C	22001291		SELECT titl	SELECT pu	SELECT isbn, title, lan		2	2	2	6	100.00%	

