Task 1

I first created a small neural network.

I added 2 dense layers with 128 neurons and output layer with 10 neurons.

As the dataset is multi-labeled, I used SoftMax activation.

I used Adam as optimizer. It starts with a initial learning rate and decreases with epoch to avoid the gradient overshooting problem.

The evaluated metrics are:

	Train	Valid	Test
Accuracy (%)	99.63	97.34	97.58
Loss	.0174	.0949	.0840

I started with a small one as it will give me an insight to how the dataset is performing with a simple model, should I increase parameters, should I further tweak hyper parameters to improve model performance. I could use CNN blocks to further optimize as the MNIST dataset is a image dataset.

Further optimizations may be done with adding more complex blocks.

Task 2

I created a simple SQL database to work with. The database structure is as follows:

The database is "pythonsqlite.db"

I create a table named ‘solution’

The solution table:

ID (int primary key)	Name (text)	status (text)	begin_date (text)	end_date (text)
1	Task 2 begin	In progress	2023-11-08	2023-11-11

I organize the database using ‘sqlite3’ from python. It’s a built-in library and lightweight. I first connected to the database. Next, I create a table structured as above. Then I create the 1st entry and furthermore upate the entry. I retrieved all the data and printed in the console. Finally I deleted the entry thus completing all the CRUD operations.

Task 3

I used Google API to retrieve data from google sheets. The sheet link can be found at:

https://docs.google.com/spreadsheets/d/1HSJarrTevcqeSblr61I_jv1Z14PAAK0yjdSwhFYhyMA/edit?usp=sharing

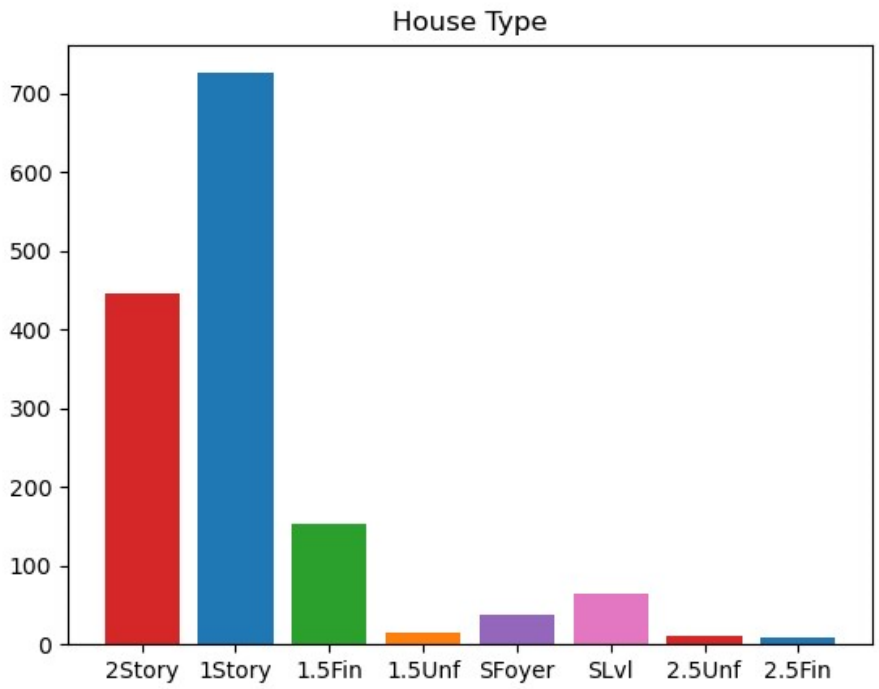
It is the house regression price dataset. I enabled console API in my account, saved the credentials in credentials.json. (I cannot publicly provide the credentials.json or token.json because its from my personal account.)

I used credentials.json to log in and create token.json for future api calls without re logging in. I called the sheets api using SPREADSHEET_ID varable. I stored the results to values variable.

The received values are individual lists such as

[[‘a’], [‘b’], [‘c‘]]

I had to convert that to [‘a’, ‘b’, ‘c’]. I retrieved the ‘HouseStyle’ column and plotted a bar graph. The graph is as follows:



Further preprocessing can be made if I need to train a regression model on this. I can apply scaling, normalization to individual column before training.