For this project, I worked on predicting salaries using multiple linear regression. In the beginning, I explored a few different datasets such as student performance and car price prediction. However, I found those datasets either too messy or not well-structured for my level of understanding. Eventually, I chose a salary dataset from Kaggle because it seemed clean, easy to understand, and had around 1000 records, which I felt was a manageable size to experiment with. The main reason I picked this dataset was because I thought I could use numerical features like age and experience to make decent predictions without getting too deep into complex features.

The dataset contains features such as Education, Gender, Location, Job Title, Age, Experience, and Salary. At first, I focused mainly on using just the numerical features like Age and Experience because I believed these would be enough for the model to learn from. However, after testing, I realized the model's performance wasn't very good using only those features. That's when I understood the importance of including categorical data like Education and Job Title, and I learned how to process them using one-hot encoding.

For the modeling, I used multiple linear regression. After adding those categorical features and applying one-hot encoding through a ColumnTransformer, the model performance improved significantly. I also used StandardScaler to normalize the numeric columns. Once everything was set up, the model achieved a good R^2 score of about 0.88 on the training data and 0.87 on the test set. The mean absolute error (MAE) was around 7,700–8,100, which I found acceptable considering the mean salary value in the dataset was about 105,833 which means my model was getting better .

Throughout the project, I faced many challenges. Choosing the right dataset was confusing at first because I tried many different ones, and some were too messy or complicated. Preprocessing was also difficult because I didn't initially know how to handle categorical features properly. I also tried other regression models just by using numerical data but they didn't improve results . Watching tutorials and carefully building the pipeline step by step helped me understand the importance of preprocessing. I also realized that staying calm and patient is very important, especially when things don’t work as expected.

Next time, I’ll try to plan better and focus more on the preprocessing step earlier. What I really learned from this project is that patience and data understanding are more important than jumping to results. Preprocessing isn’t just a step to check off, it’s essential. I learned that being anxious and switching between ideas too quickly can lead to mistakes and frustration. Instead, if I stick to one dataset, follow each step logically, and truly try to understand what the data is telling me, the process becomes smoother and the results better.