LONDON METROPOLITAN UNIVERSITY

Islington college
(इस्लिङ्टन कलेज)

Choose a Module

Choose Coursework Percentage **Individual Coursework**

2023-24 Choose an item.

Student Name: ASHISH BUDHA

London Met ID: 22067313

College ID: NP01AI4A220031

Assignment Due Date: 13 May 2024

Assignment Submission Date: Monday, May 13, 2024

Word Count: 1083

I confirm that I understand my coursework needs to be submitted online via MySecondTeacher under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a marks of zero will be awarded.

Contents

Data Understanding

What is the dataset about?

The provided dataset contains different data of employees that plays a part in determining the salary of the employee. These variables are : ‘work_year’ , ‘experience_level’, ‘employment_type’, ‘job_title’, ‘salary’, ‘salary_currency’, ‘salary_in_USD’, ‘employee_residence’, ‘remote_ratio’, ‘company_location’, ‘company_size’. We are to identify correlations between the employee’s salary and other variables.

Table 1 : Data Table

S.No	Column Name	Description	Data Type
1	Work_year	This is a column that shows the year the employee worked in. We may be able to observe patterns in work year and salary, overtime a job may become more in demand or saturated, affecting the salary of the employee.	Integer
2	Experience_level	This is a column that shows the experience of the employee. The different experiences are : SE or Senior / Expert, MI or Medium Level / Intermediate, EN or Entry Level, EX or Executive Level.	String
3	Employment_type	This is a column showing the type of employment the employee is working as. Example it could be full type employment (FT) or a contract employment (CT)	
4	Job_title	This column shows us the job of the employee.	String
5	Salary	The Salary of the employee. This can be in any currency.	Integer
6	Salary Currency	The columns will tell us what the currency of the salary in the 5th column is.	String
7	Salary_in_usd	This column will show the employees salary in USD, if in previous columns	Integer

		it was shown in another currency, it will be converted to USD.	
8	Employee_residence	This column shows the place the employee is currently residing in. People living in low cost of living could be paid less and vice versa.	String
9	Remote_ratio	This column shows the extent to which remote work is embraced by a company. This could affect the salary structure of the employees.	Integer
10	Company_location	This column shows the location of the country. Certain company locations could mean a higher / lower salary.	String
11	Company_size	The size of the company such as : S, M, and L.	String

Data Preparation

Python Program to load data into pandas Framework

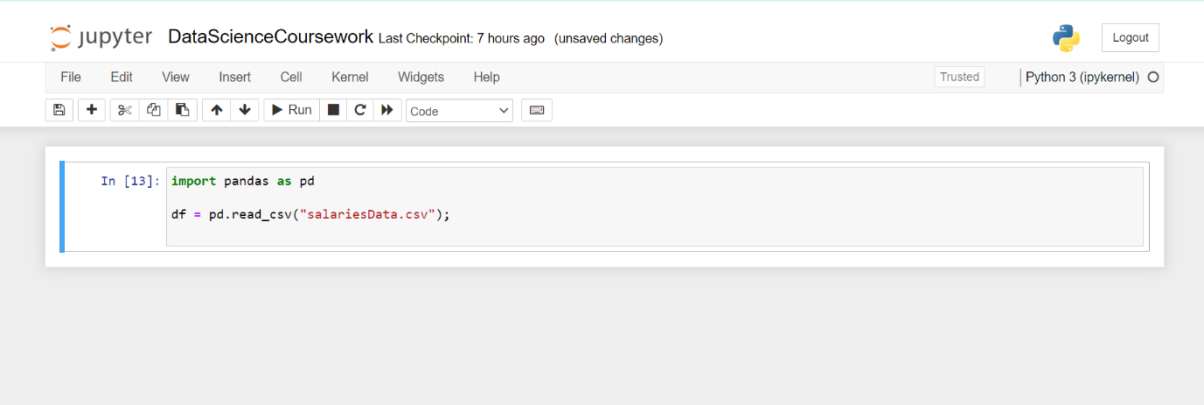


Figure 1: Load Data

Python Program to remove unnecessary columns i.e., salary and salary currency.

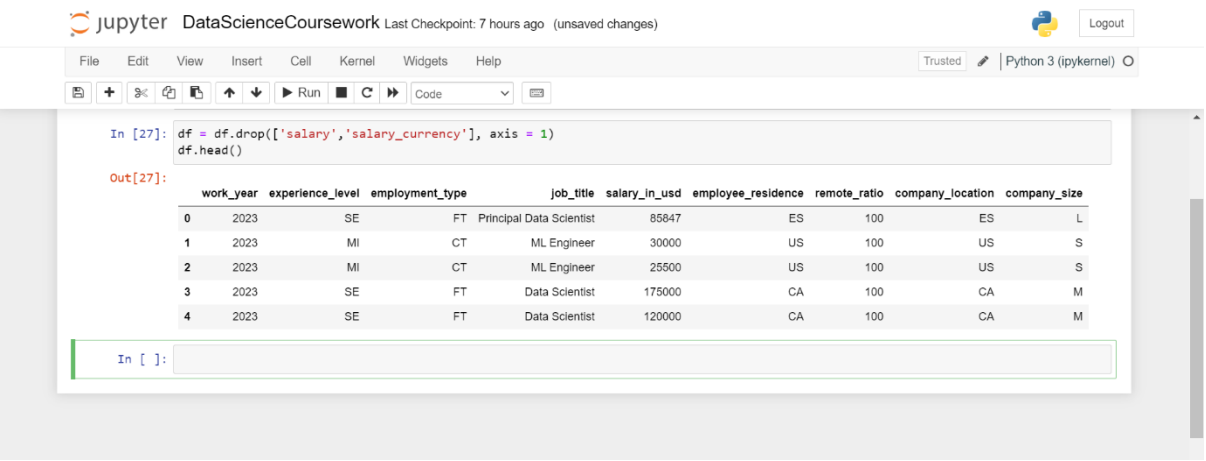


Figure 2: Drop Data

The columns “salary” and “salary_currency” are no longer seen on the DataFrame df.

Write a python program to remove NaN missing values from updated dataframe.

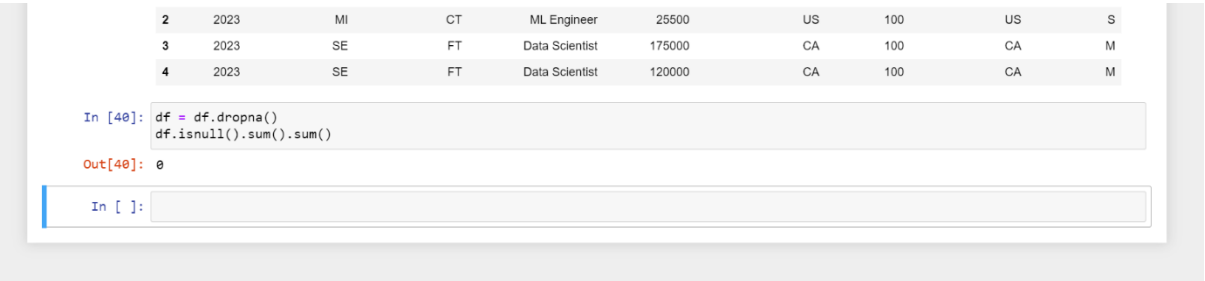


Figure 3: Remove NaN

Method called to drop NaN numbers. Checked to see if there are any NaN present in the new updated dataframe, there is no NaN present.

Python Program to check duplicate values in the dataframe

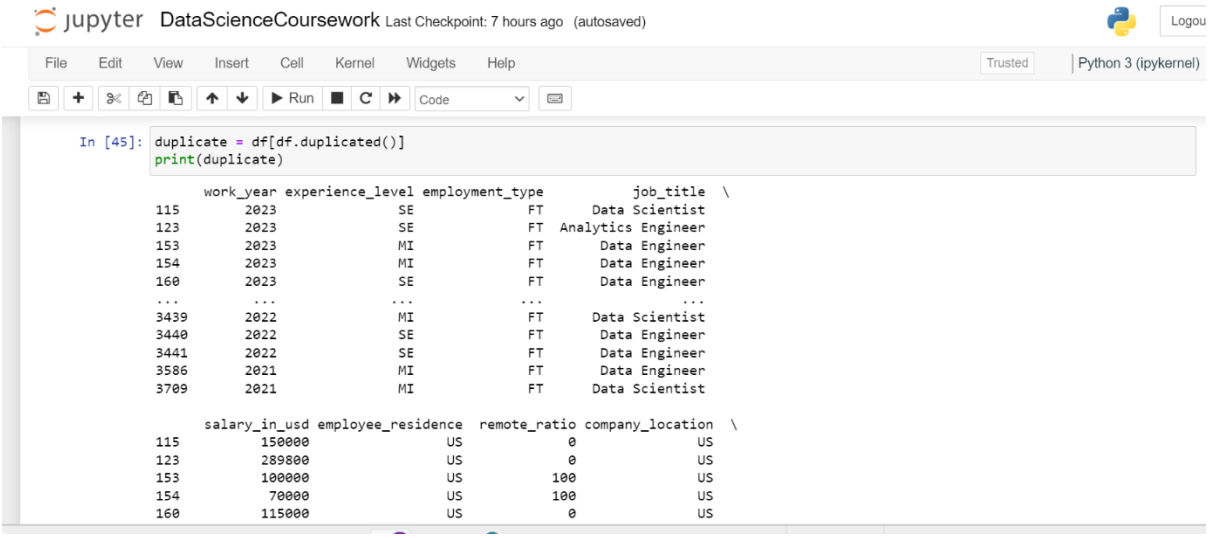


Figure 4:Check Duplicate

The .duplicated() method of the dataframe is used to check for duplicate values. It returns a boolean series, indicating a value of true if the row is a duplicate of a previous row.

This boolean Series is then used as condition for printing the dataframe rows.

Python program to see the unique values from all the columns in the dataframe.

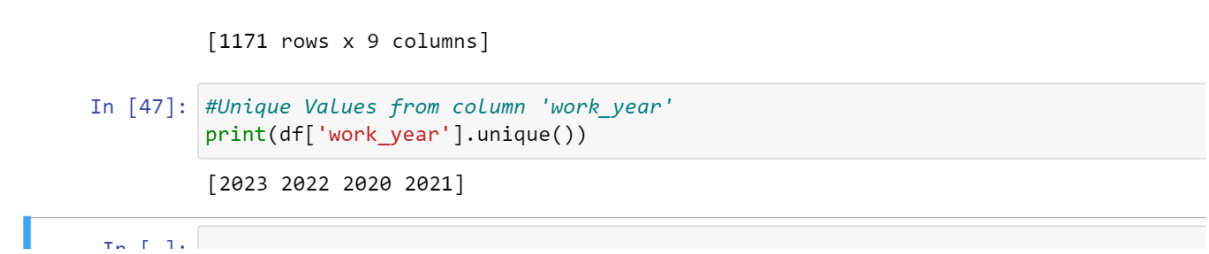


Figure 5: Unique Work Year

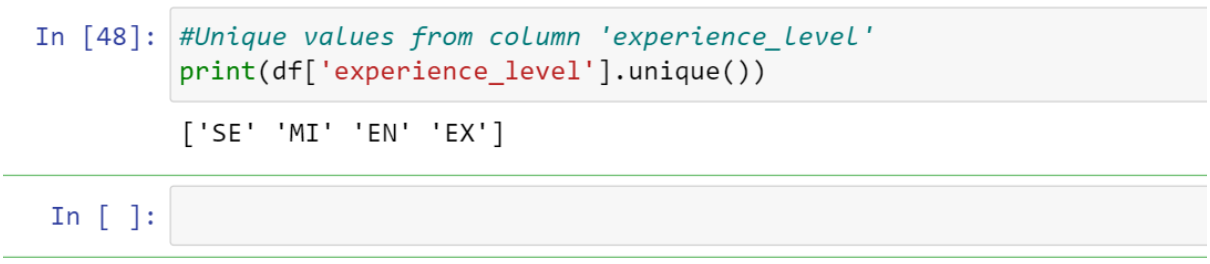


Figure 6: Unique Experience Level

```
In [50]: #Unique values form column 'employment_type'
         print(df['employment_type'].unique())

         ['FT' 'CT' 'FL' 'PT']

In [ ]:
```

Figure 7:Unique Employment Type

```
In [52]: #Unique values form column 'job_title'
         print(df['job_title'].unique())

         ['Principal Data Scientist' 'ML Engineer' 'Data Scientist'
          'Applied Scientist' 'Data Analyst' 'Data Modeler' 'Research Engineer'
          'Analytics Engineer' 'Business Intelligence Engineer'
          'Machine Learning Engineer' 'Data Strategist' 'Data Engineer'
          'Computer Vision Engineer' 'Data Quality Analyst'
          'Compliance Data Analyst' 'Data Architect'
          'Applied Machine Learning Engineer' 'AI Developer' 'Research Scientist'
          'Data Analytics Manager' 'Business Data Analyst' 'Applied Data Scientist'
          'Staff Data Analyst' 'ETL Engineer' 'Data DevOps Engineer' 'Head of Data'
          'Data Science Manager' 'Data Manager' 'Machine Learning Researcher'
          'Big Data Engineer' 'Data Specialist' 'Lead Data Analyst'
          'BI Data Engineer' 'Director of Data Science'
          'Machine Learning Scientist' 'MLOps Engineer' 'AI Scientist'
          'Autonomous Vehicle Technician' 'Applied Machine Learning Scientist'
          'Lead Data Scientist' 'Cloud Database Engineer' 'Financial Data Analyst'
          'Data Infrastructure Engineer' 'Software Data Engineer' 'AI Programmer'
          'Data Operations Engineer' 'BI Developer' 'Data Science Lead'
```

Figure 8: Unique Job Title

```
'Data Infrastructure Engineer' 'Software Data Engineer' 'AI Programmer'
'Data Operations Engineer' 'BI Developer' 'Data Science Lead'
'Deep Learning Researcher' 'BI Analyst' 'Data Science Consultant'
'Data Analytics Specialist' 'Machine Learning Infrastructure Engineer'
'BI Data Analyst' 'Head of Data Science' 'Insight Analyst'
'Deep Learning Engineer' 'Machine Learning Software Engineer'
'Big Data Architect' 'Product Data Analyst'
'Computer Vision Software Engineer' 'Azure Data Engineer'
'Marketing Data Engineer' 'Data Analytics Lead' 'Data Lead'
'Data Science Engineer' 'Machine Learning Research Engineer'
'NLP Engineer' 'Manager Data Management' 'Machine Learning Developer'
'3D Computer Vision Researcher' 'Principal Machine Learning Engineer'
'Data Analytics Engineer' 'Data Analytics Consultant'
'Data Management Specialist' 'Data Science Tech Lead'
'Data Scientist Lead' 'Cloud Data Engineer' 'Data Operations Analyst'
'Marketing Data Analyst' 'Power BI Developer' 'Product Data Scientist'
'Principal Data Architect' 'Machine Learning Manager'
'Lead Machine Learning Engineer' 'ETL Developer' 'Cloud Data Architect'
'Lead Data Engineer' 'Head of Machine Learning' 'Principal Data Analyst'
'Principal Data Engineer' 'Staff Data Scientist' 'Finance Data Analyst']
```

Figure 8.1:Unique Job Title

```
In [55]: #Unique values from column 'salary_in_usd'
         print(df['salary_in_usd'].unique())

         [ 85847  30000  25500 ...  28369 412000  94665]
```

Figure 9:Unique Salary

```
In [57]: #Unique values from column 'employee_residence'
         print(df['employee_residence'].unique())

['ES' 'US' 'CA' 'DE' 'GB' 'NG' 'IN' 'HK' 'PT' 'NL' 'CH' 'CF' 'FR' 'AU'
 'FI' 'UA' 'IE' 'IL' 'GH' 'AT' 'CO' 'SG' 'SE' 'SI' 'MX' 'UZ' 'BR' 'TH'
 'HR' 'PL' 'KW' 'VN' 'CY' 'AR' 'AM' 'BA' 'KE' 'GR' 'MK' 'LV' 'RO' 'PK'
 'IT' 'MA' 'LT' 'BE' 'AS' 'IR' 'HU' 'SK' 'CN' 'CZ' 'CR' 'TR' 'CL' 'PR'
 'DK' 'BO' 'PH' 'DO' 'EG' 'ID' 'AE' 'MY' 'JP' 'EE' 'HN' 'TN' 'RU' 'DZ'
 'IQ' 'BG' 'JE' 'RS' 'NZ' 'MD' 'LU' 'MT']
```

Figure 10: Unique Employee Residence

```
In [58]: #Unique values from column 'remote_ratio'
         print(df['remote_ratio'].unique())

         [100   0  50]

In [ ]:
```

Figure 11: Unique Remote Ratio

```
In [59]: #Unique values from column 'company_location'
         print(df['company_location'].unique())

         ['ES' 'US' 'CA' 'DE' 'GB' 'NG' 'IN' 'HK' 'NL' 'CH' 'CF' 'FR' 'FI' 'UA'
          'IE' 'IL' 'GH' 'CO' 'SG' 'AU' 'SE' 'SI' 'MX' 'BR' 'PT' 'RU' 'TH' 'HR'
          'VN' 'EE' 'AM' 'BA' 'KE' 'GR' 'MK' 'LV' 'RO' 'PK' 'IT' 'MA' 'PL' 'AL'
          'AR' 'LT' 'AS' 'CR' 'IR' 'BS' 'HU' 'AT' 'SK' 'CZ' 'TR' 'PR' 'DK' 'BO'
          'PH' 'BE' 'ID' 'EG' 'AE' 'LU' 'MY' 'HN' 'JP' 'DZ' 'IQ' 'CN' 'NZ' 'CL'
          'MD' 'MT']
```

Figure 12: Unique Company Location

```
In [61]: #Unique values form column 'company_size'
         print(df['company_size'].unique())

         ['L' 'S' 'M']

In [ ]:
```

Figure 13: Unique Company Size

Renaming experience level column

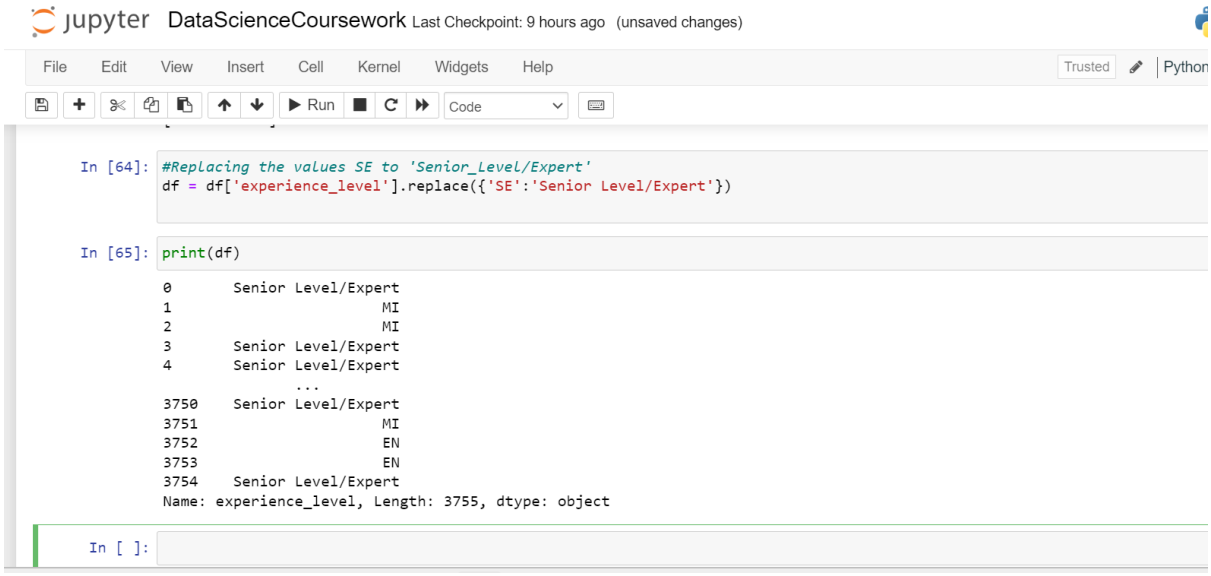


Figure 14: Replacing SE

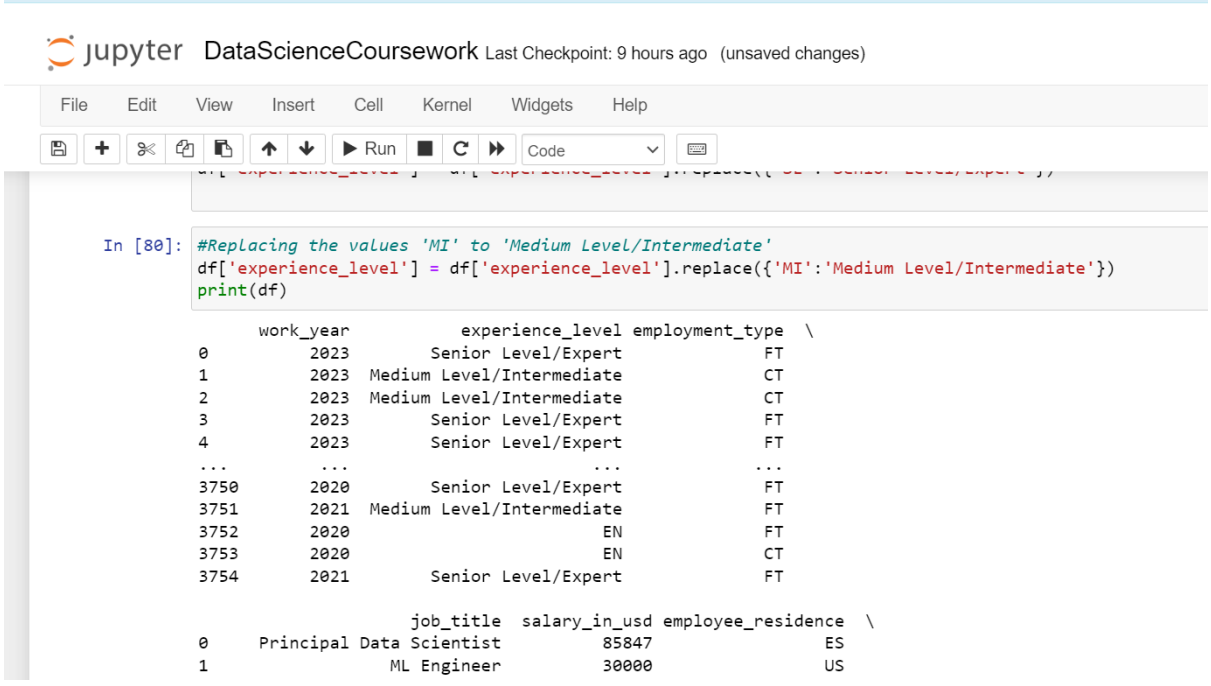


Figure 15: Replacing MI

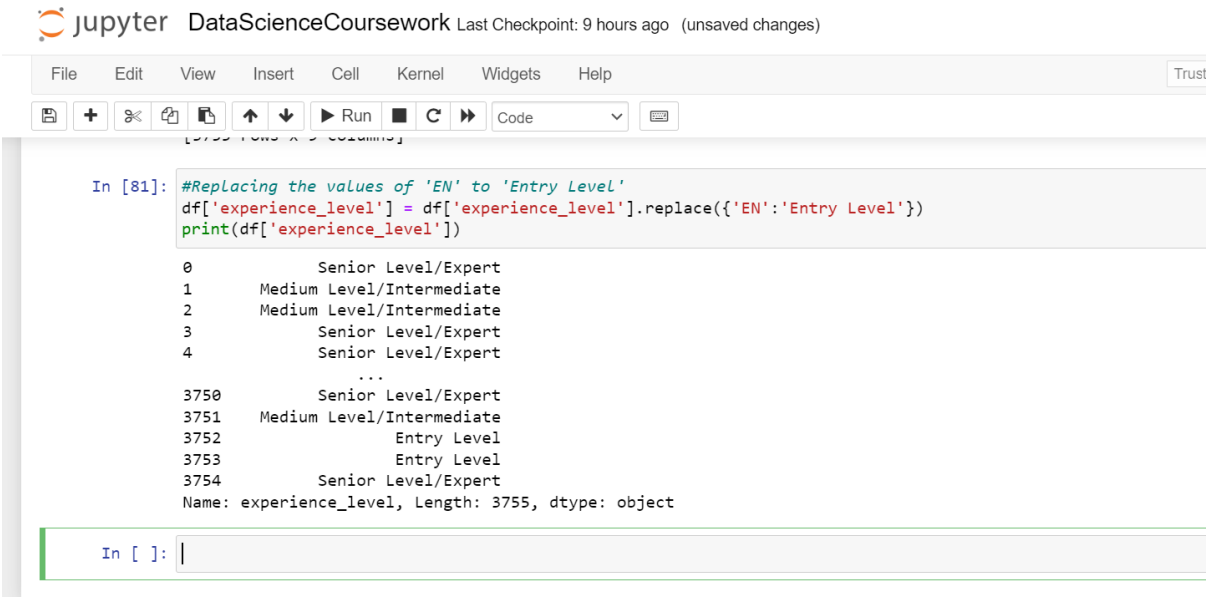


Figure 16: Replacing EN

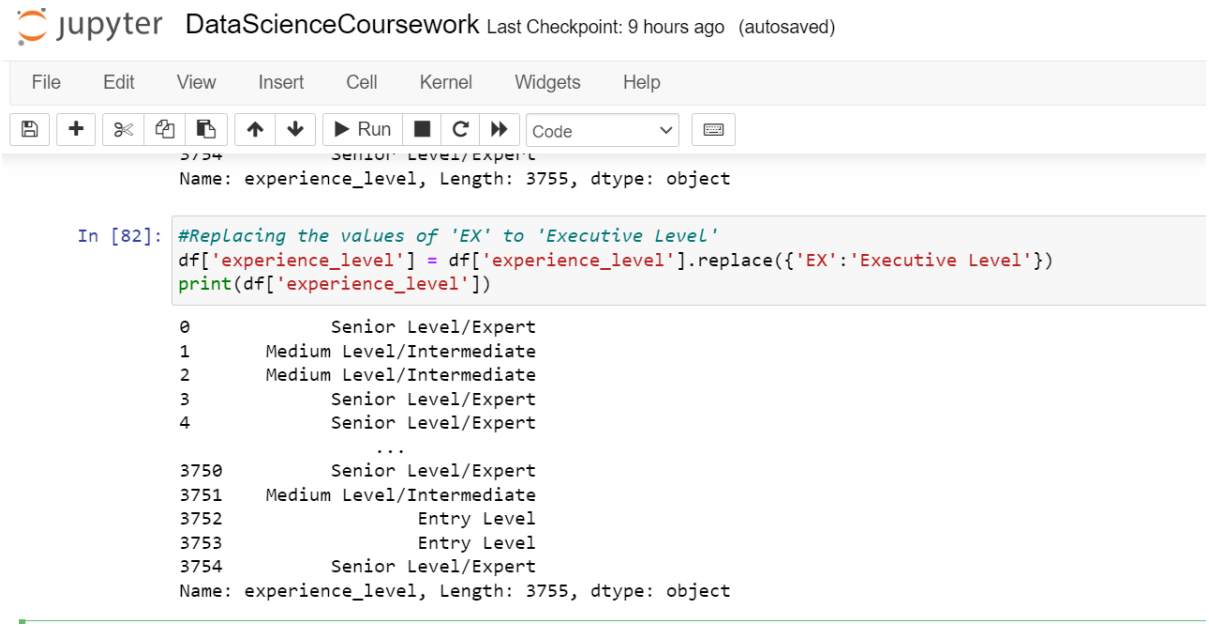


Figure 17: Replacing EX

Data Analysis

Python Program to show summary statistics of sum, mean, standard deviation, skewness, and kurtosis of any chosen variable

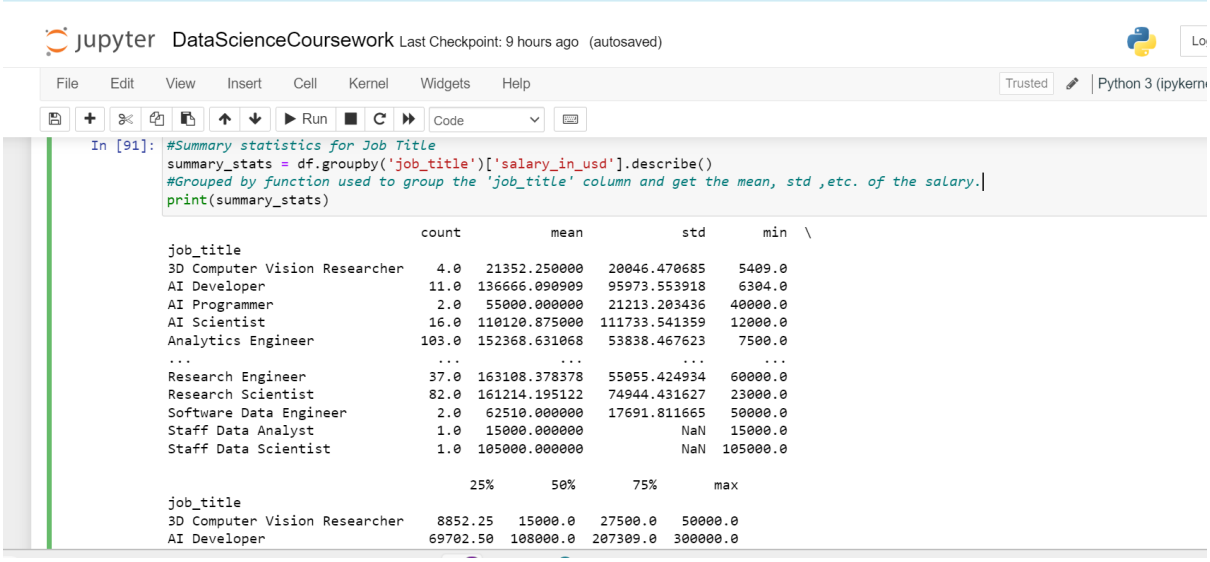


Figure 18: Summary Statistics

Summary Stats are being shown here using the .describe() method of the dataframe object. The dataframe is first grouped the on 'job title' and then we calculated the summary statistics of the 'salary' based on 'job title'

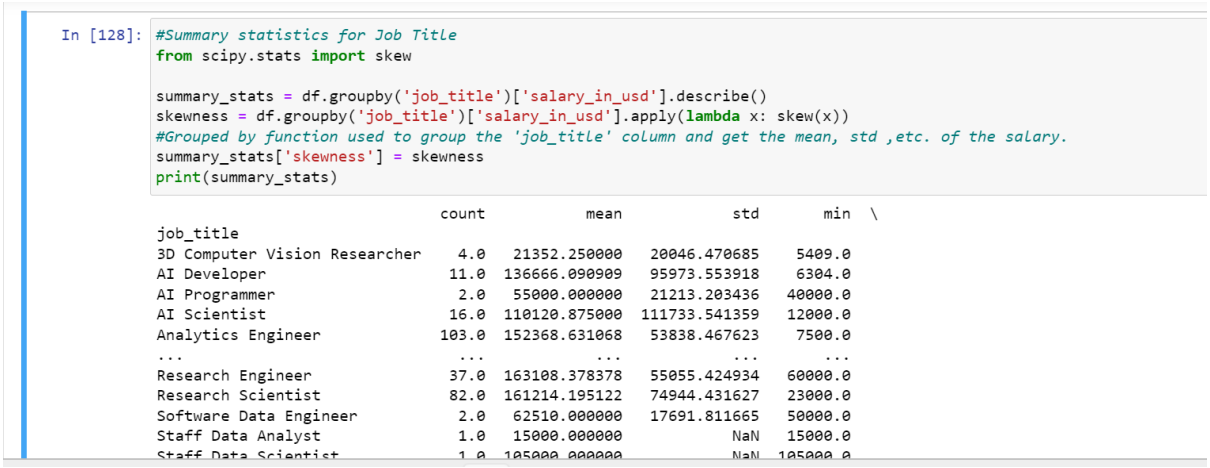
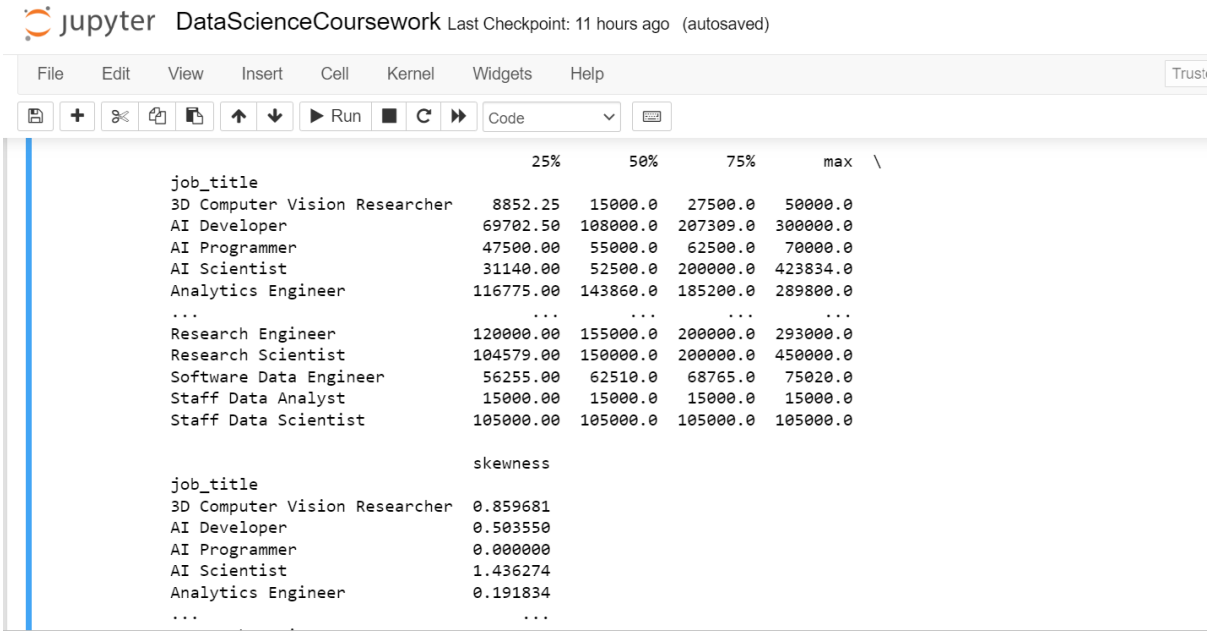


Figure 19: Skewness

Here I have used the scipy libraries' skew() method to get the skew of the data.




```
                                          25%       50%       75%       max  \
job_title
3D Computer Vision Researcher    8852.25   15000.0   27500.0   50000.0
AI Developer                    69702.50  108000.0  207309.0  300000.0
AI Programmer                   47500.00   55000.0   62500.0   70000.0
AI Scientist                    31140.00   52500.0  200000.0  423834.0
Analytics Engineer             116775.00  143860.0  185200.0  289800.0
...                                  ...       ...       ...       ...
Research Engineer              120000.00  155000.0  200000.0  293000.0
Research Scientist             104579.00  150000.0  200000.0  450000.0
Software Data Engineer          56255.00   62510.0   68765.0   75020.0
Staff Data Analyst              15000.00   15000.0   15000.0   15000.0
Staff Data Scientist           105000.00  105000.0  105000.0  105000.0

                                skewness
job_title
3D Computer Vision Researcher   0.859681
AI Developer                    0.503550
AI Programmer                   0.000000
AI Scientist                    1.436274
Analytics Engineer              0.191834
...                                  ...
```

Figure 19.1: Skewness

```
Staff Data Analyst              15000.00   15000.0   15000.0   15000.0
Staff Data Scientist           105000.00  105000.0  105000.0  105000.0

                                skewness  kurtosis
job_title
3D Computer Vision Researcher   0.859681  2.078886
AI Developer                    0.503550 -0.910019
AI Programmer                   0.000000       NaN
AI Scientist                    1.436274  2.906718
Analytics Engineer              0.191834  0.151166
...                                  ...       ...
Research Engineer               0.540204 -0.198917
Research Scientist              0.953457  2.139420
Software Data Engineer          0.000000       NaN
Staff Data Analyst                   NaN       NaN
Staff Data Scientist                 NaN       NaN

[93 rows x 10 columns]
```

Figure 20: Kurtosis

Skewness values for the job title 'Staff Data Analyst' and 'Staff Data Scientist' is NaN as there is only 1 person for each of the job title. Standard Deviation is NaN as well.

Write a python program to calculate and show the correlation of all variables

```
In [130]: correlation = df.corr()
          print(correlation)

                      work_year  salary_in_usd  remote_ratio
work_year              1.00000       0.228290     -0.236430
salary_in_usd          0.22829       1.000000     -0.064171
remote_ratio          -0.23643      -0.064171      1.000000

C:\Users\ashis\AppData\Local\Temp\ipykernel_32764\1693706327.py:1: FutureWarning: The def
```

Figure 21: Correlation

From this output, we can see that work_year and salary_in_usd has a positive correlation of 0.22, which means that an increase in the year would likely mean an increase in salary.

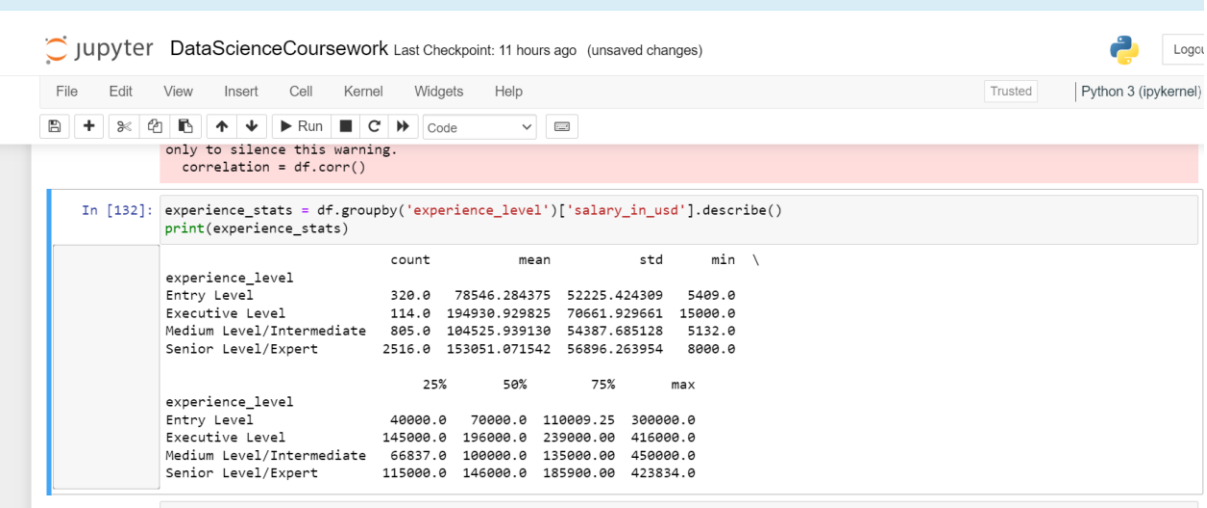


Figure 22: Correlation Non-Numeric

We can observe a correlation between the non-numeric data (experience_level) and the salary variable. Experience level 'Executive' earns the highest, followed by 'Senior Level/Expert, followed by 'Medium Level/Intermediate', followed by 'Entry Level'

Data Exploration

Write a python program to find out the top 15 jobs. Make a bar graph of sales as well.

Figure 23: Top 15 Jobs

Here we have grouped the data by ‘job title’ and then selected the ‘salary_in_usd’ column which we then calculated the mean for.

The plot.figure(figsize = (10,6)) creates a new figure with a size of 10 (width) by 6 (height).

We then used the top15 Series value to plot the bar. With it's x-label being defined using the method .xlabel() and y-label being defined using the .ylabel()

The .xticks() method rotates the label for better readability.

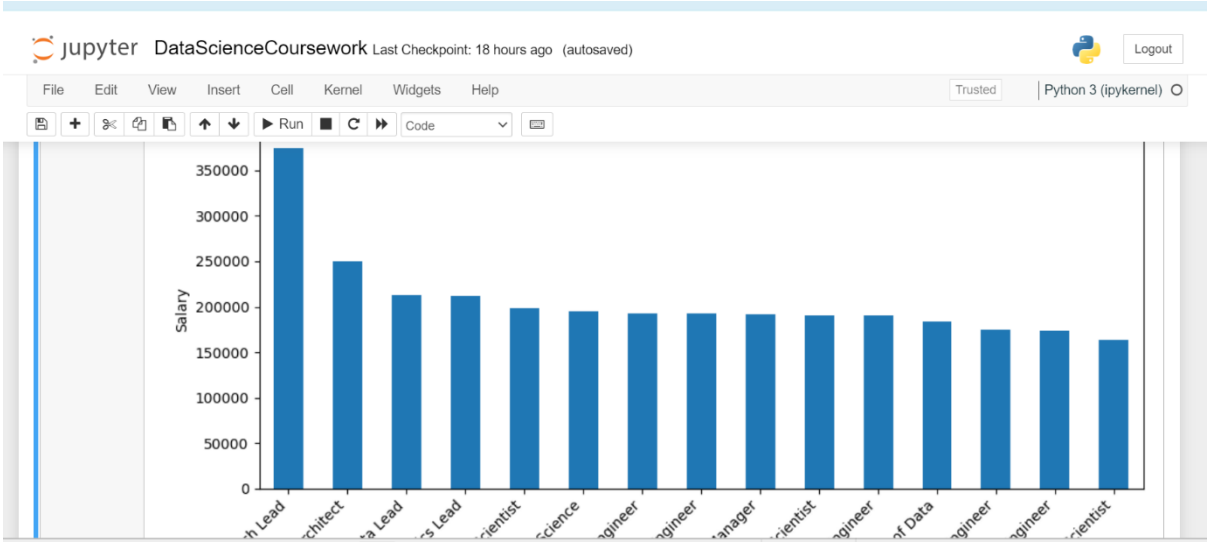


Figure 23.1: Top 15 Jobs

The figure that was shown when executing the code, here we can see top 15 jobs on the basis of the jobs salary mean value.

Which job has the highest salary? Illustrate with bar graph

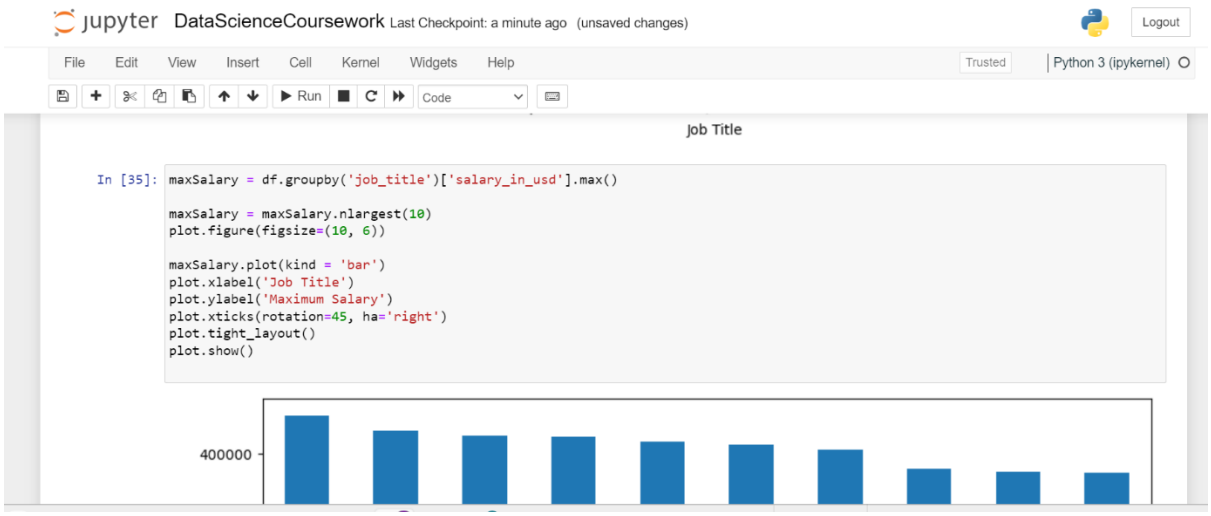


Figure 24: Highest Salary

We grouped the data by ‘job title’ and then selected the ‘salary_in_usd’ column which we then identified the maximum value.

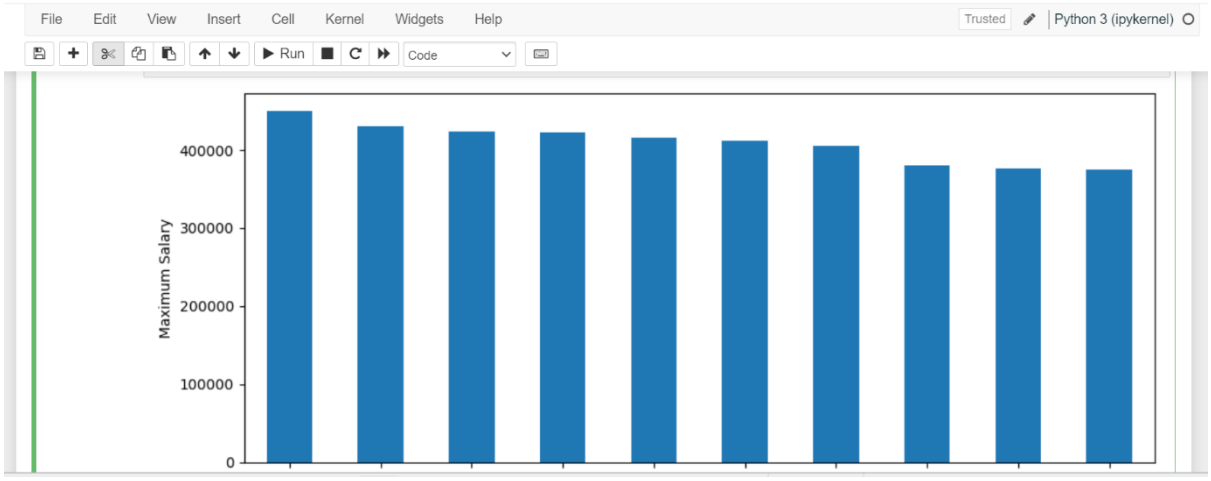


Figure 24.1: Highest Salary

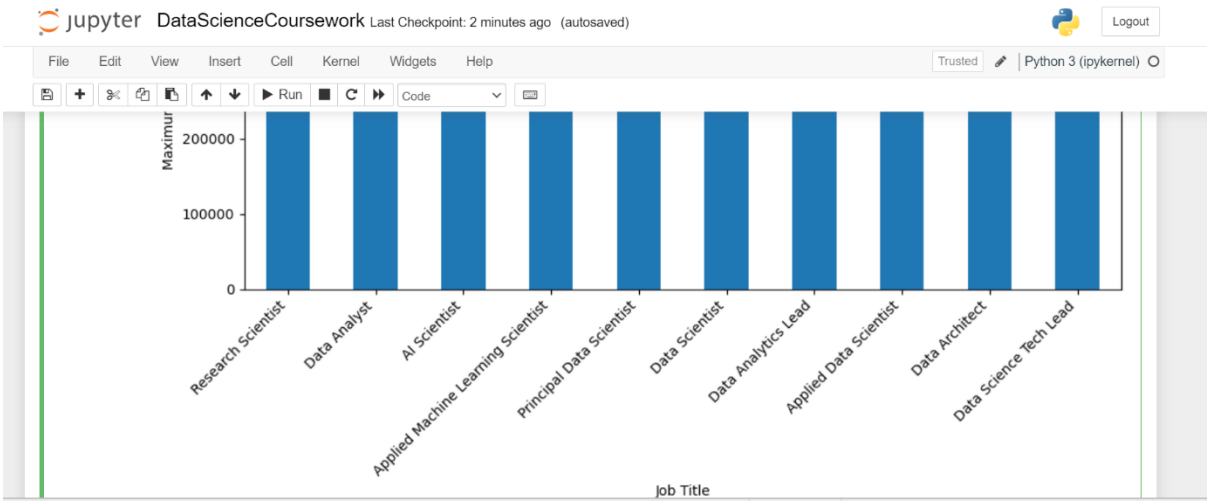


Figure 24.2: Highest Salary

From the bar graph, it shows that the job ‘Research Scientist’ has the highest salaries

Python program to find out salaries based on experience level.

Figure 25: Experience Level Salary

Here the dataframe is grouped by ‘experience_level’ and then it’s ‘salary_in_usd’ column selected, then the mean of the selected column is calculated for each group.

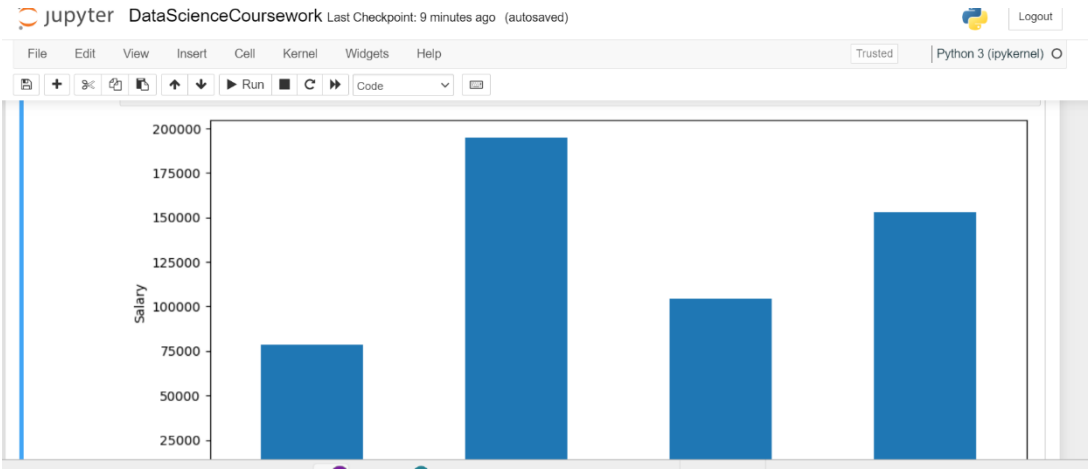


Figure 25.1: Experience Level Salary

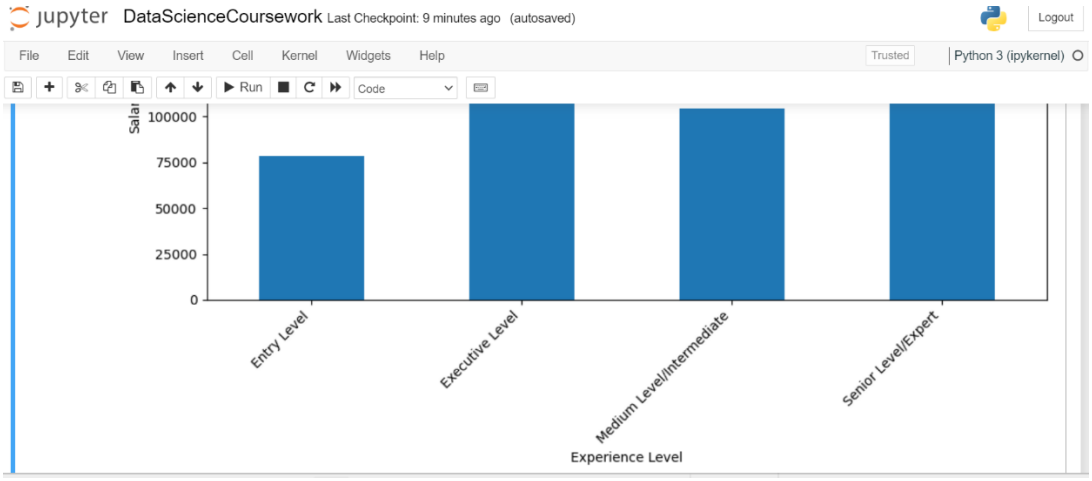


Figure 25.2: Experience Level Salary

This bar chart shows that people with experience ‘Executive Leve’ is paid the highest, followed by ‘Senior Level/Expert’, followed by ‘Medium Level / Intermediate’ , followed by ‘Entry Level’

Python program to show histogram and boxplot of any chosen different variables

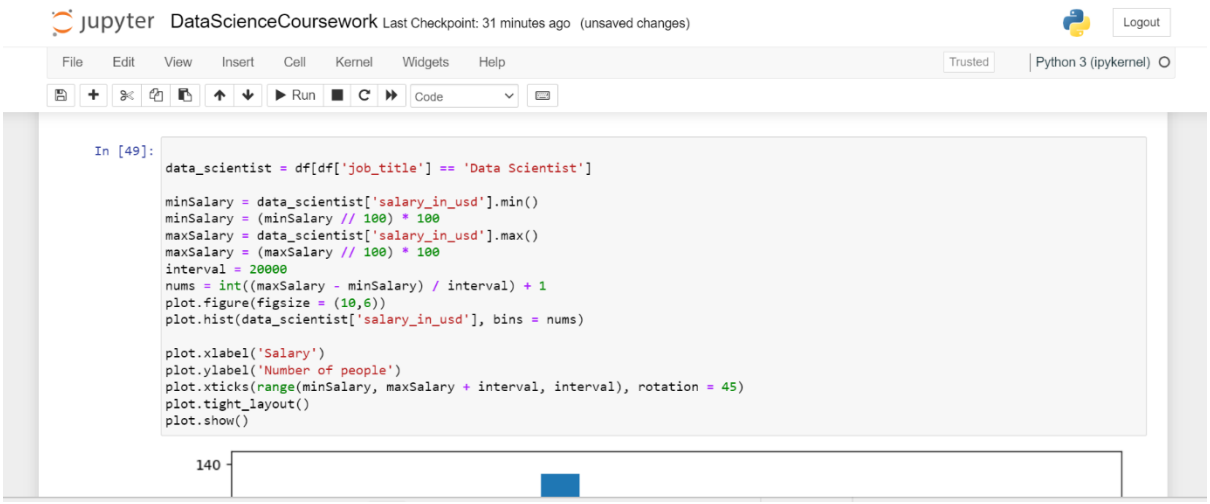


Figure 26: Histogram

Here the statement creates a histogram based on salary as the interval and the height of the histogram representing the number of people in the job 'Data Scientist' having an earning within that specific interval.

The minSalary = (minSalary // 100) * 100 is used to round down the value to the nearest hundred.

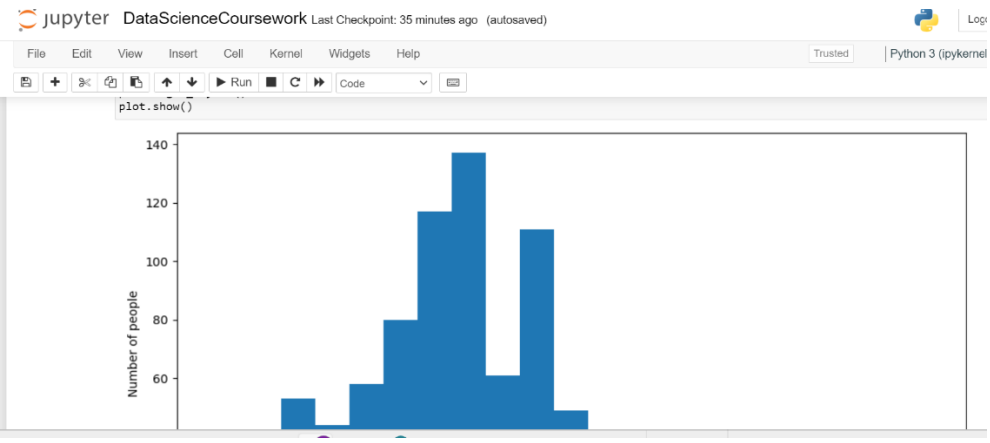


Figure 26.1: Histogram

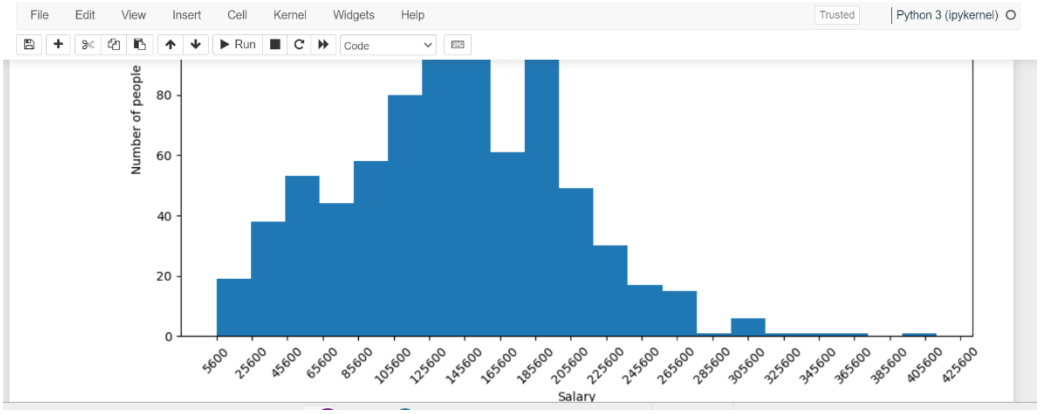


Figure 26.2: Histogram

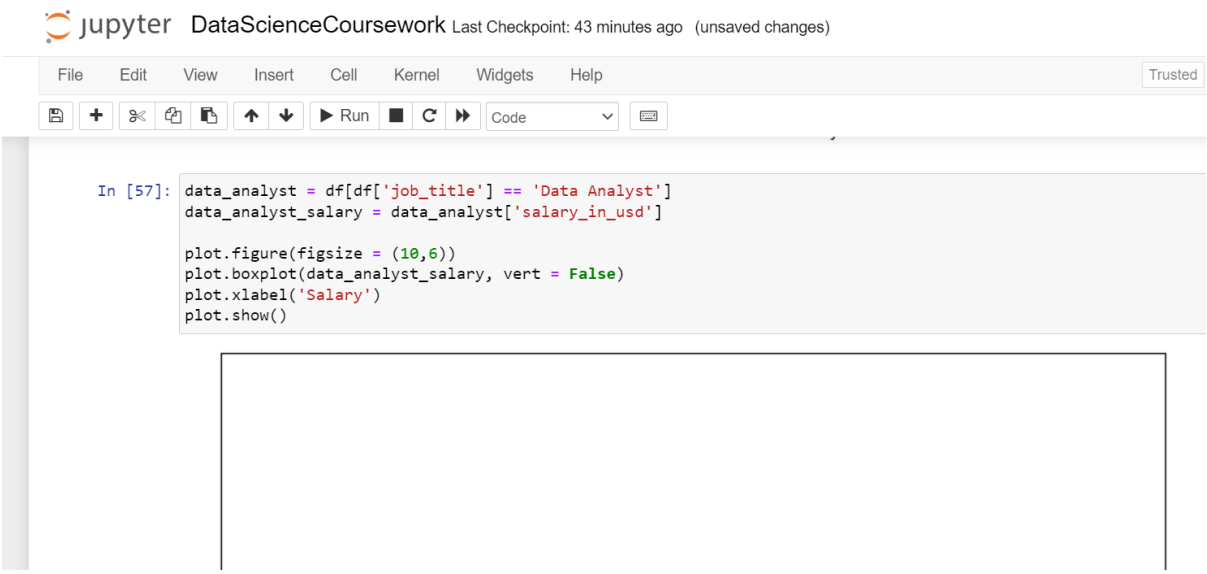


Figure 27: Box Plot

This code creates a box plot of salary of people with the job title ‘Data Analyst’

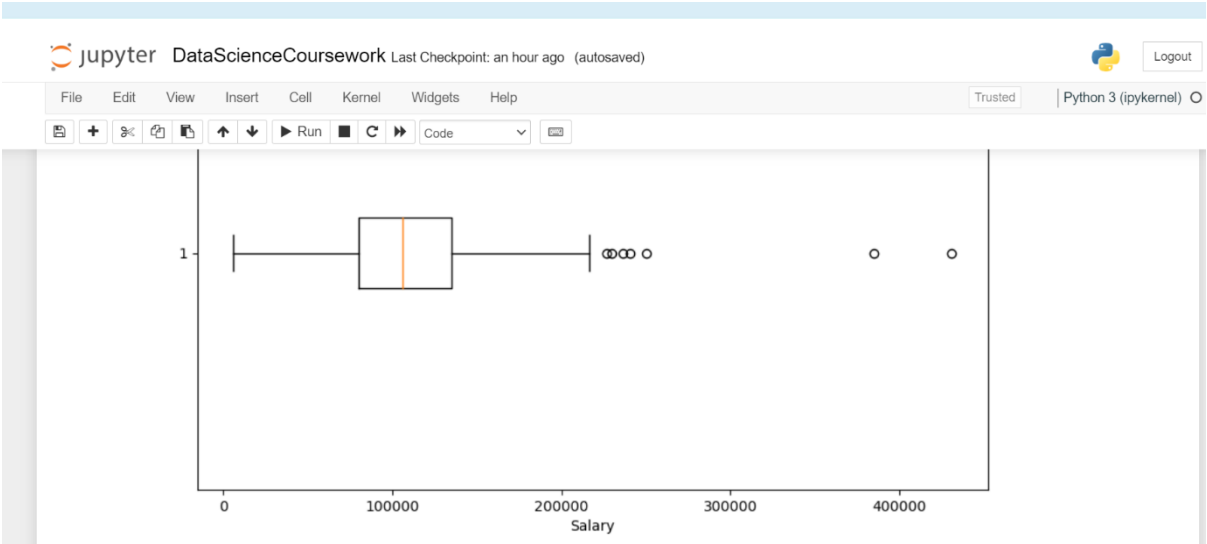


Figure 27.1: Box Plot

We can see the Q1, Q3, median, min, max, and outliers from this boxplot.