Importing Libraries

```
import pandas as pd
import numpy as np

import matplotlib.pyplot as plt
import seaborn as sns
from plotly import *
from plotly.express import px, graph_objects, graph_objects, country_name_to_country_alpha3
```

Data's Basic Overview

```
df = pd.read_csv("ds_salaries.csv", index_col=0)
```

```
df
```

	work_year	experience_level	employment_type	job_title	salary	salary_currency	salary_in_usd	employee_residence	remote_ratio	company_location	company_size
0	2020	MI	FT	Data Scientist	70000	EUR	79833	DE	0	DE	L
1	2020	SE	FT	Machine Learning Scientist	260000	USD	260000	JP	0	JP	S
2	2020	SE	FT	Big Data Engineer	85000	GBP	109944	GB	50	GB	M
3	2020	MI	FT	Product Data Analyst	20000	USD	20000	HN	0	HN	S
4	2020	SE	FT	Machine Learning Engineer	150000	USD	150000	US	50	US	L
...	...	...	...	...	...	...	...	...	...	...	...
606	2022	MI	FT	Data Engineer	154000	USD	154000	US	100	US	M
607	2022	MI	FT	Data Engineer	126000	USD	126000	US	100	US	M
608	2022	SE	FT	Cloud Data Engineer	148000	USD	148000	US	100	US	L
609	2022	SE	FT	Data Analyst	90000	USD	90000	US	100	US	M
610	2022	SE	FT	AI Scientist	200000	USD	200000	IN	100	US	L

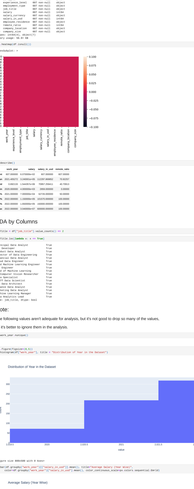
610 rows × 11 columns

```
df.info()
```

```
<class'pandas.core.frame.dataframe'>
int64index: 610 entries, 0 to 610
data columns (total 11 columns):
 #   column              non-null count  dtype
---  ------              --------------  -----
 0   work_year           610 non-null    int64
 1   experience_level    610 non-null    object
 2   employment_type     610 non-null    object
 3   job_title           610 non-null    object
 4   salary              610 non-null    int64
 5   salary_currency     610 non-null    object
 6   salary_in_usd       610 non-null    int64
 7   employee_residence  610 non-null    object
 8   remote_ratio        610 non-null    int64
 9   company_location    610 non-null    object
 10  company_size        610 non-null    object
dtypes: int64(5), object(6)
memory usage: 56.9 kb
```

```
df.corr().sort_values(by='salary_in_usd', ascending=False)
```

```
sns.heatmap(df.corr())
```




```
df.describe()
```

	work_year	salary	salary_in_usd	remote_ratio
count	607.000000	5.770000e+02	607.000000	607.000000
mean	2021.405272	324000.062660	85086.585831	70.574384
std	0.692133	1544357.205288	28369.801689	40.702393
min	2020.000000	5132.000000	2859.000000	0.000000
25%	2021.000000	54000.000000	60757.000000	50.000000
50%	2021.000000	135000.000000	79039.000000	100.000000
75%	2022.000000	207000.000000	99118.000000	100.000000
75%	2022.000000	6000000.000000	450000.000000	100.000000
max	2022.000000	6000000.000000	450000.000000	100.000000

EDA by Columns

```
[df[i].value_counts() for i in df.keys()]
```

```
[df[i].isnull().sum() for i in df.keys()]
```

```
Principal Data Analyst                  True
Data Analyst                            True
Product Data Analyst                    True
Principal Machine Learning Engineer     True
Applied Machine Learning Scientist      True
Lead Machine Learning Engineer          True
Lead Data Scientist                     True
Lead Data Engineer                      True
Head of Machine Learning                True
Head of Data Science                    True
Data Warehouse Analyst                  True
Data Quality Analyst                    True
Big Data Architect                      True
Business Data Analyst                   True
Business Data Analyst                   True
Business Data Analyst                   True
Data Analytics Lead                     True
Data Analytics Lead                     True
```

Note:

The following values aren't adequate for analysis, but it's not good to drop so many of the values, so it's better to ignore them in the analysis.

```
df.work_year.nunique()
```

```
4
```

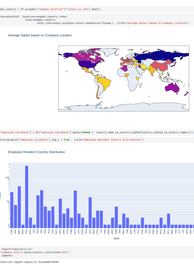
```
px.figure(figure=df.\
px.histogram(df.work_year, title='Distribution of Year in the Dataset')
)
```

```
px.figure(figure=df.\
px.scatter(df.work_year,df.salary_in_usd,df.remote_ratio,color='average Salary Year wise',
color_continuous_scale=px.colors.sequential.Jet)
)
```

```
df.groupby('work_year')[['salary_in_usd']].mean(), 2021.00000 * 100
# Average salary increased by 2021 and 2022 during pandemic
```

```
work_year
2020    76465.000000
2021    85086.585831
2022    108037.275862
Name: salary_in_usd, dtype: float64
```

```
px.figure(figure=df.\
px.bar(df.job_title,df.salary_in_usd,color='Percentage Salary by Job Title',
color_continuous_scale=px.colors.sequential.Jet,barmode='group')
)
```




```
px.figure(figure=df.\
px.bar(df.job_title,df.salary_in_usd,color='Maximum Salary by Job Title',
)
)
```

```
px.figure(figure=df.\
px.bar(df.job_title,df.salary_in_usd,color='Minimum Salary by Job Title',
)
)
```



```
df.groupby('work_year')['job_title'].value_counts()
```

```
work_year  job_title
2020       Data Scientist                                 19
           Data Analyst                                   11
           Machine Learning Scientist                      3
           Big Data Engineer                               3
           Product Data Analyst                            2
           Machine Learning Engineer                       2
           Data Science Consultant                         2
           BI Developer                                    2
2021       Data Scientist                                 42
           Data Engineer                                  32
           Data Analyst                                   16
           Machine Learning Engineer                      12
           Analytics Engineer                              4
           BI Developer                                    4
           Applied Machine Learning Scientist              3
           Applied Scientist                               3
           Research Scientist                              3
           Machine Learning Scientist                      3
Name: job_title, dtype: int64
```

```
px.figure(figure=df.\
px.bar(df.experience_level,df.salary_in_usd,color='Experience of the Individuals',color_discrete_sequence=px.colors.sequential.Jet)
)
```



```
px.bar(df.experience_level,df.salary,color='average Salary as per Experience',
color_continuous_scale=px.colors.sequential.Jet,barmode='group',color_continuous_scale=px.colors.sequential.Plasma)
```



```
px.figure(figure=df.\
px.histogram(df.\'employment_type\', color=df.\'employment_type\', title='Employment Type Distribution')
)
```



```
px.bar(df.employment_type,df.salary_in_usd,color='average Salary based on Employment Type',
color_continuous_scale=px.colors.sequential.Jet,title='average Salary based on Employment Type ')
```



```
df['company_location'].unique()
```

```
array(['DE', 'JP', 'GB', 'HN', 'US', 'HU', 'NL', 'MX', 'ES', 'PK', 'IT',
       'RU', 'IN', 'PR', 'PL', 'AE', 'NG', 'BR', 'PT', 'CN', 'GR', 'AT',
       'FR', 'CA', 'CL', 'MY', 'AR', 'DK', 'IR', 'BE', 'AS', 'CO', 'MA',
       'TR', 'HR', 'RO', 'SI', 'SG', 'VN', 'EE', 'AU', 'BG', 'IQ', 'CH',
       'KE', 'LV', 'CZ', 'UA', 'DZ', 'MK', 'EG', 'MT', 'IL', 'NZ', 'FI',
       'TN', 'PH', 'AM', 'JE', 'HK', 'RS', 'BA', 'PE', 'IE', 'LU', 'UZ',
       'TW', 'PA', 'AL', 'KW', 'SA', 'MD', 'JM', 'NA', 'OM', 'VE', 'SK',
       'BO', 'HT', 'DJ', 'BY', 'SV', 'GH', 'AD', 'LT', 'KH', 'GT', 'MU',
       'KZ', 'CY', 'SE', 'NI', 'TH', 'HV', 'PS', 'BD', 'UG', 'MO', 'TZ',
       'ZA', 'NO', 'AZ', 'MR', 'SD', 'DO', 'MG', 'BS', 'CF', 'ZW', 'CD',
       'KI', 'SM', 'MW', 'MZ', 'AO', 'PG', 'SN', 'SL', 'LR', 'GM', 'BJ',
       'CI', 'TL', 'XK', 'SR', 'ME', 'LK', 'BM', 'AF', 'GA', 'IS', 'LB',
       'MN', 'AX', 'TM', 'KG', 'ME', 'MC', 'MT', 'MD', 'GI', 'GE', 'AE',
       'BF', 'BZ', 'BT', 'BW', 'BI', 'AG', 'VG', 'TG', 'TT', 'TR', 'VI',
       'SC', 'SB', 'RW', 'PY', 'NP', 'MV', 'LA', 'KY', 'HN', 'GU', 'GQ',
       'GD', 'FJ', 'ER', 'CU', 'AI', 'AQ', 'KN', 'MP', 'MH', 'IM', 'EU',
       'ET', 'CG', 'BQ', 'BN', 'AW', 'AI', 'AD', 'AE', 'AE', 'AE', 'AE',
       'AE', 'AE', 'AE', 'AE', 'AE', 'AE', 'AE
```