Background: "This dataset contains job postings from Glassdoor.com from 2017 with the following features It can be used to analyze the current trends based on job positions, company size, etc."

The goal of this project is to clean the dataset to make it useful for analysis and perform Exploratory Data Analysis (EDA) on the cleaned dataset.

Link: https://www.kaggle.com/datasets/thedevastator/jobs-dataset-from-glassdoor

In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

In [2]:
```
#Loading dataset
data = pd.read_csv('eda_data.csv', usecols=['Job Title', 'Salary Estimate', 'Job De
                                            'Location', 'Size', 'Founded','Type of o
                                            'Sector', 'Revenue','hourly','same_state

#Only certain columns where chosen due to two reasons:
#1. Some columns were deemed not important for the purpose of
#   Exploratory Data Analysis(EDA), such as the 'Unnamed: 0' columns, which, is jus
#2. Some columns were omitted to highlight data cleaning abilities.
```

In [3]:
```
data.head()
```

Out[3]:

	Job Title	Salary Estimate	Job Description	Rating	Company Name	Location	Size	Foun
0	Data Scientist	$53K−91K (Glassdoor est.)	Data Scientist\nLocation: Albuquerque, NM\nEdu...	3.8	Tecolote Research\n3.8	Albuquerque, NM	501 to 1000 employees	1
1	Healthcare Data Scientist	$63K−112K (Glassdoor est.)	What You Will Do:\n\nI. General Summary\n\nThe...	3.4	University of Maryland Medical System\n3.4	Linthicum, MD	10000+ employees	1
2	Data Scientist	$80K−90K (Glassdoor est.)	KnowBe4, Inc. is a high growth information sec...	4.8	KnowBe4\n4.8	Clearwater, FL	501 to 1000 employees	2
3	Data Scientist	$56K−97K (Glassdoor est.)	*Organization and Job ID**\nJob ID: 310709\n\n...	3.8	PNNL\n3.8	Richland, WA	1001 to 5000 employees	1
4	Data Scientist	$86K−143K (Glassdoor est.)	Data Scientist\nAffinity Solutions / Marketing...	2.9	Affinity Solutions\n2.9	New York, NY	51 to 200 employees	1

In [4]:
```
# Checking the amount of columns
```

Cleaning the Data

In [5]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 742 entries, 0 to 741
Data columns (total 15 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   Job Title          742 non-null    object 
 1   Salary Estimate    742 non-null    object 
 2   Job Description    742 non-null    object 
 3   Rating             742 non-null    float64
 4   Company Name       742 non-null    object 
 5   Location           742 non-null    object 
 6   Size               742 non-null    object 
 7   Founded            742 non-null    int64  
 8   Type of ownership  742 non-null    object 
 9   Industry           742 non-null    object 
 10  Sector             742 non-null    object 
 11  Revenue            742 non-null    object 
 12  hourly             742 non-null    int64  
 13  same_state         742 non-null    int64  
 14  age                742 non-null    int64  
dtypes: float64(1), int64(4), object(10)
memory usage: 87.1+ KB
```

There are several columns from the dataset that need to be cleaned before Exploratory Data Analysis (EDA).

Cleaning the Location Data

In [6]:
```
#The Location Data will be separated into two columns, a state and a city column

#First, each state and city will be stored into their respective list variables.
state = []
city = []

for i in range(len(data["Location"])):
    if "Santa Fe Springs" in data['Location'][i]:
        city.append(data['Location'].str.split(',')[i][0].strip())
        state.append(data['Location'].str.split(',')[i][2].strip())
    else:
        city.append(data['Location'].str.split(',')[i][0].strip())
        state.append(data['Location'].str.split(',')[i][1].strip())
```

In [7]:
```
state[0:10]
```

Out[7]:
```
['NM', 'MD', 'FL', 'WA', 'NY', 'TX', 'MD', 'CA', 'NY', 'NY']
```

In [8]:
```
city[0:10]
```

Out[8]:
```
['Albuquerque',
 'Linthicum',
 'Clearwater',
 'Richland',
 'New York',
 'Dallas',
 'Baltimore',
 'San Jose',
 'Rochester',
 'New York']
```

There appears to be some whitespace in the state list, so we will strip any leading whitespaces with the .strip() method.

The same will also be done to the "city" list in case there are any leading whitespaces.

In [9]:
```
for i in range(len(state)):
    city[i] = city[i].strip()
    state[i] = state[i].strip()
```

Now that the location column has been separated, they will be added back to the main dataframe.

In [10]:
```
df_state = pd.DataFrame(state, columns=['State'])
df_city = pd.DataFrame(city, columns=['City'])

data = pd.concat([data, df_state, df_city],axis=1)
```

In [11]:
```
data.head()
```

Out[11]:

	Job Title	Salary Estimate	Job Description	Rating	Company Name	Location	Size	Foun
0	Data Scientist	$53K−91K (Glassdoor est.)	Data Scientist\nLocation: Albuquerque, NM\nEdu...	3.8	Tecolote Research\n3.8	Albuquerque, NM	501 to 1000 employees	1
1	Healthcare Data Scientist	$63K−112K (Glassdoor est.)	What You Will Do:\n\nI. General Summary\n\nThe...	3.4	University of Maryland Medical System\n3.4	Linthicum, MD	10000+ employees	1
2	Data Scientist	$80K−90K (Glassdoor est.)	KnowBe4, Inc. is a high growth information sec...	4.8	KnowBe4\n4.8	Clearwater, FL	501 to 1000 employees	2
3	Data Scientist	$56K−97K (Glassdoor est.)	*Organization and Job ID**\nJob ID: 310709\n\n...	3.8	PNNL\n3.8	Richland, WA	1001 to 5000 employees	1
4	Data Scientist	$86K−143K (Glassdoor est.)	Data Scientist\nAffinity Solutions / Marketing...	2.9	Affinity Solutions\n2.9	New York, NY	51 to 200 employees	1

In [12]:
```
# There is no need for the Location column anymore since we separated the contents
# Thus, "Location" will be dropped and the state and city column will be rearranged

data.drop('Location', axis=1)

cols = ['Job Title', 'Salary Estimate', 'Rating', 'Company Name', 'State', 'City',
       'Size', 'Founded', 'Type of ownership', 'Industry', 'Sector', 'Revenue',
       'hourly', 'same_state', 'age']

data = data[cols]
```

In [13]:
```
# The Location column has been dropped and the State and City columns are found in
data.head()
```

Out[13]:

	Job Title	Salary Estimate	Rating	Company Name	State	City	Size	Founded	Typ owne
0	Data Scientist	$53K-$91K (Glassdoor est.)	3.8	Tecolote Research\n3.8	NM	Albuquerque	501 to 1000 employees	1973	Compa Pr
1	Healthcare Data Scientist	$63K-$112K (Glassdoor est.)	3.4	University of Maryland Medical System\n3.4	MD	Linthicum	10000+ employees	1984	C Organiz
2	Data Scientist	$80K-$90K (Glassdoor est.)	4.8	KnowBe4\n4.8	FL	Clearwater	501 to 1000 employees	2010	Compa Pr
3	Data Scientist	$56K-$97K (Glassdoor est.)	3.8	PNNL\n3.8	WA	Richland	1001 to 5000 employees	1965	Governı
4	Data Scientist	$86K-$143K (Glassdoor est.)	2.9	Affinity Solutions\n2.9	NY	New York	51 to 200 employees	1998	Compa Pr

Next, the 'Job Title' column will be cleaned in order for it to be of use for EDA.

In [14]:
```
# There are 264 unique job titles

data['Job Title'].nunique()
```

Out[14]: 264

The goal will be to reduce the number of unique jobs by simplifying the Job Title.

For instance, if a job title has the words "Data Scientist" in it, and the full Job Title is "Healthcare Data Scientist," then

the new title will be just "Data Scientist"

In [15]:
```
pd.set_option("display.max_rows", None)
data['Job Title'].value_counts()
```

Out[15]:
```
Data Scientist                                                  131
Data Engineer                                                    53
Senior Data Scientist                                            34
Data Analyst                                                     15
Senior Data Engineer                                             14
Senior Data Analyst                                              12
Lead Data Scientist                                               8
Marketing Data Analyst                                            6
Sr. Data Engineer                                                 6
Machine Learning Engineer                                         5
Principal Data Scientist                                          5
R&D Specialist/ Food Scientist                                    4
Medical Laboratory Scientist                                      4
Research Scientist                                                4
Senior Research Scientist-Machine Learning                        4
MED TECH/LAB SCIENTIST- SOUTH COASTAL LAB                         4
Analytics Manager - Data Mart                                     4
Food Scientist - Developer                                        4
Staff Scientist-Downstream Process Development                    4
Sr. Data Engineer - Contract-to-Hire (Java)                       4
Sr. Scientist Method Development                                  3
Project Scientist - Auton Lab, Robotics Institute                 3
Scientist, Molecular/Cellular Biologist                           3
Associate Scientist, LC/MS Biologics                              3
Revenue Analytics Manager                                         3
IT - Data Engineer II                                             3
Research Scientist, Immunology - Cancer Biology                   3
Scientist - Biomarker and Flow Cytometry                          3
```

Senior Scientist - Regulatory Submissions	3
ENVIRONMENTAL ENGINEER/SCIENTIST	3
Senior Data Science Systems Engineer	3
Senior Insurance Data Scientist	3
Associate Data Analyst- Graduate Development Program	3
Scientist - Analytical Services	3
Principal Scientist, Chemistry & Immunology	3
Director II, Data Science - GRM Actuarial	3
Scientist/Senior Scientist, Autoimmune	3
Clinical Data Analyst	3
Staff Data Engineer	3
Consultant - Analytics Consulting	3
Lead Data Engineer	3
Principal Scientist, Hematology	3
Staff Machine Learning Engineer	3
Software Engineer - Data Visualization	3
Senior Data Scientist - R&D Oncology	3
Principal Data Scientist (Computational Chemistry)	3
Senior Scientist (Neuroscience)	3
Machine Learning Engineer - Regulatory	3
Scientist, Bacteriology	3
Associate Director, Platform and DevOps- Data Engineering and Aritifical Intelligence	3
Clinical Laboratory Scientist	3
Information Security Data Analyst	3
Sr. Data Analyst	3
Data Science Manager	3
Scientist	3
Staff Scientist- Upstream PD	2

Sr Expert Data Science, Advanced Visual Analytics (Associate level)	2
Sr Data Analyst - IT	2
Associate Machine Learning Engineer / Data Scientist May 2020 Undergrad	2
Senior Scientist - Biostatistician	2
Senior Data Scientist Oncology	2
Associate Principal Scientist, Pharmacogenomics	2
Data Scientist - Systems Engineering	2
Data Engineer - Consultant (Charlotte Based)	2
Data Analyst 1, full-time contract worker for up to 12 months	2
Scientist, Immuno-Oncology	2
Products Data Analyst II	2
Lead Data Analyst	2
Data Science Engineer - Mobile	2
IT Associate Data Analyst	2
Scientist, Pharmacometrics	2
Business Data Analyst	2
Principal, Data Science - Advanced Analytics	2
Data Science Project Manager	2
Sr Scientist, Immuno-Oncology - Oncology	2
Data Analytics Project Manager	2
Scientist Manufacturing - Kentucky BioProcessing	2
Director - Data, Privacy and AI Governance	2
Staff BI and Data Engineer	2
Associate Data Engineer	2
Research Scientist – Security and Privacy	2
Market Data Analyst	2
Product Engineer – Data Science	2
Computational Chemist/Data Scientist	2

Director Data Science	2
Senior Research Analytical Scientist-Non-Targeted Analysis	2
Systems Engineer II - Data Analyst	2
Product Engineer – Spatial Data Science and Statistical Analysis	2
Managing Data Scientist/ML Engineer	2
Technology-Minded, Data Professional Opportunities	2
Salesforce Analytics Consultant	2
Scientist Manufacturing Pharma - Kentucky BioProcessing	2
Sr Data Engineer (Sr BI Developer)	2
Associate, Data Science, Internal Audit	2
Associate Environmental Scientist - Wildlife Biologist	2
Data Modeler - Data Solutions Engineer	2
Marketing Data Analyst, May 2020 Undergrad	2
Corporate Risk Data Analyst (SQL Based) - Milwaukee or	2
Senior Manager, Epidemiologic Data Scientist	2
Enterprise Architect, Data	2
Lead Big Data Engineer	2
Sr Software Engineer (Data Scientist)	2
Senior Scientist - Toxicologist - Product Integrity (Stewardship)	2
Sr. Data Engineer (ETL Developer)	2
Senior Data Scientist Artificial Intelligence	2
Analytics - Business Assurance Data Analyst	2
Associate Director/Director, Safety Scientist	2
Senior Scientist, Cell Pharmacology/Assay Development	2
Data Analyst Senior	2
Lead Data Engineer (Python)	2
Senior Data Scientist 4 Artificial Intelligence	2
Medical Lab Scientist - MLT	2

Senior Operations Data Analyst, Call Center Operations	2
Director II, Data Science - GRS Predictive Analytics	2
RESEARCH COMPUTER SCIENTIST - RESEARCH ENGINEER - SR. COMPUTER SCIENTIST - SOFTWARE DEVELOPMENT	2
Geospatial Software Developer and Data Scientist	2
Senior LiDAR Data Scientist	2
Big Data Engineer	2
Medical Lab Scientist	2
Senior Data & Machine Learning Scientist	2
Radar Data Analyst	2
VP, Data Science	2
Machine Learning Research Scientist	2
Excel / VBA / SQL Data Analyst	2
Sr. Data Scientist II	2
Sr. Scientist, Quantitative Translational Sciences	2
MED TECH/LAB SCIENTIST - LABORATORY	2
Principal Data Scientist with over 10 years experience	2
Principal Scientist - Immunologist	2
Sr. Scientist - Digital & Image Analysis/Computational Pathology	2
Digital Marketing & ECommerce Data Analyst	2
Analytics Manager	2
Scientist, Analytical Development	2
Risk and Analytics IT, Data Scientist	2
Senior Scientist - Neuroscience	2
Data Engineer 5 - Contract (Remote)	2
Staff Data Scientist	2
Data Scientist (Actuary, FSA or ASA)	2
Pricipal Scientist Molecular and cellular biologist	2
Sr. Data Scientist - Analytics, Personalized Healthcare (PHC)	2

BI & Platform Analytics Manager
2
PL Actuarial-Lead Data Scientist
2
PV Scientist
2
Data Scientist - Algorithms & Inference
2
Data Scientist - Quantitative
2
Data Scientist, Office of Data Science
2
College Hire - Data Scientist - Open to December 2019 Graduates
2
Senior Risk Data Scientist
2
Staff Data Scientist - Technology
2
Data Scientist / Machine Learning Expert
2
Clinical Data Scientist
2
Associate Data Analyst
2
Digital Health Data Scientist
2
Senior Data Scientist / Machine Learning
2
Data Scientist in Artificial Intelligence Early Career
2
Data Scientist - Health Data Analytics
2
Customer Data Scientist
2
Data Scientist SR
2
Data Scientist - Alpha Insights
1
Senior Data Scientist - Algorithms
1
Data Engineer - ETL
1
Data Modeler (Analytical Systems)
1
Data Science Analyst
1
Data Scientist in Translational Medicine
1
Data Analyst 2 (Missionary Department)
1
Supply Chain Data Analyst
1
Spectral Scientist/Engineer
1
Web Data Analyst
1

Data Scientist - Sales	1
Scientist I/II, Biology	1
Data Engineer I	1
Senior Data Scientist – Visualization, Novartis AI Innovation Lab	1
Product Manager/Data Evangelist	1
Insurance Financial Data Analyst	1
Senior Data Analyst/Scientist	1
Scientist – Cancer Discovery, Molecular Assay	1
Associate Scientist / Sr. Associate Scientist, Antibody Discovery	1
Data Architect / Data Modeler	1
Jr. Data Scientist	1
Data Scientist (Warehouse Automation)	1
Scientist - CVRM Metabolism - in vivo pharmacology	1
Sr. Data Engineer | Big Data SaaS Pipeline	1
Assistant Director/Director, Office of Data Science	1
Manager, Safety Scientist, Medical Safety & Risk Management	1
Software Engineer Staff Scientist: Human Language Technologies	1
Clinical Scientist, Clinical Development	1
Quality Control Scientist III- Analytical Development	1
Senior Engineer, Data Management Engineering	1
Principal Research Scientist/Team Lead, Medicinal Chemistry - Oncology	1
Senior Health Data Analyst, Star Ratings	1
Foundational Community Supports Data Analyst	1
Research Scientist, Machine Learning Department	1
Research Scientist / Principal Research Scientist - Multiphysical Systems	1
Data Analyst Chemist - Quality System Contractor	1
Research Scientist or Senior Research Scientist - Computer Vision	1
Senior Quantitative Analyst	1

Senior Formulations Scientist II	1
Director, Precision Medicine Clinical Biomarker Scientist	1
Associate Research Scientist I (Protein Expression and Production)	1
Software Engineer (Data Scientist/Software Engineer) - SISW - MG	1
Data Scientist Manager	1
Manager of Data Science	1
Data Engineering Analyst	1
Software Data Engineer - College	1
Sr. Scientist II	1
Data Analyst, Performance Partnership	1
Junior Data Analyst	1
Senior Data Scientist Statistics	1
Senior Spark Engineer (Data Science)	1
Senior Research Statistician- Data Scientist	1
Business Data Analyst, SQL	1
Medical Technologist / Clinical Laboratory Scientist	1
Associate Data Scientist/Computer Scientist	1
Business Intelligence Analyst / Developer	1
System and Data Analyst	1
Data & Analytics Consultant (NYC)	1
Big Data Engineer - Chicago - Future Opportunity	1
Survey Data Analyst	1
Lead Health Data Analyst - Front End	1
Healthcare Data Scientist	1
Customer Data Scientist/Sales Engineer	1
Data Operations Lead	1
RESEARCH SCIENTIST - BIOLOGICAL SAFETY	1
Principal Data Engineer, Data Platform & Insights	1

Senior Data Scientist: Causal & Predictive analytics AI Innovation Lab
1
Program/Data Analyst
1
SQL Data Engineer
1
Associate Scientist/Scientist, Process Analytical Technology - Small Molecule Anal
ytical Chemistry 1
Staff Scientist
1
Data Engineer, Data Engineering and Artifical Intelligence
1
CONSULTANT– DATA ANALYTICS GROUP
1
Data Scientist, Senior
1
Sr. Data Scientist, Cyber-Security LT Contract
1
MongoDB Data Engineer II
1
Data Scientist - Bioinformatics
1
Principal Machine Learning Scientist
1
Data Analyst / Scientist
1
Data Scientist - Research
1
R&D Data Analysis Scientist
1
Analytics Consultant
1
Director, Data Science
1
R&D Sr Data Scientist
1
Customer Data Scientist/Sales Engineer (Bay
1
Jr. Business Data Analyst
1
Data Management Specialist
1
E-Commerce Data Analyst
1
Data Engineer I - Azure
1
Insurance Data Scientist
1
Data Modeler
1
Data Scientist, Rice University
1
Senior Research Scientist - Embedded System Development for DevOps
1
Financial Data Analyst
1

```
Ag Data Scientist                                                                  1
Data Scientist II                                                                  1
Project Scientist                                                                  1
Data Analytics Manager                                                             1
Senior Machine Learning (ML) Engineer / Data Scientist - Cyber Security Analytics  1
Associate Scientist                                                                1
Scientist 2, QC Viral Vector                                                       1
Data Scientist/ML Engineer                                                         1
Sr. Data Scientist                                                                 1
Data Engineer 4 - Contract                                                         1
Data Analyst - Asset Management                                                    1
Machine Learning Engineer (NLP)                                                    1
Name: Job Title, dtype: int64
```

In [16]:
```
from collections import Counter

ex = data['Job Title'].tolist()

ex_count = Counter(ex)

print(ex_count.most_common(15)) #15 of the most common job Titles
```

```
[('Data Scientist', 131), ('Data Engineer', 53), ('Senior Data Scientist', 34),
('Data Analyst', 15), ('Senior Data Engineer', 14), ('Senior Data Analyst', 12),
('Lead Data Scientist', 8), ('Marketing Data Analyst', 6), ('Sr. Data Engineer',
6), ('Machine Learning Engineer', 5), ('Principal Data Scientist', 5), ('Research
Scientist', 4), ('Medical Laboratory Scientist', 4), ('R&D Specialist/ Food Scient
ist', 4), ('Senior Research Scientist-Machine Learning', 4)]
```

We see from the list above that most of the Job Titles contain the phrases, "Data Scientist," "Data Engineer," and "Data Analyst".

Any job title that have these phrases will be simplified with that phrase.

In [17]:
```
data['Job Title'].head(10)
```

Out[17]:
```
0               Data Scientist
1    Healthcare Data Scientist
2               Data Scientist
3               Data Scientist
4               Data Scientist
5               Data Scientist
6               Data Scientist
7               Data Scientist
8           Research Scientist
9               Data Scientist
Name: Job Title, dtype: object
```

In [18]:
```
jobTitles = []

for i in range(len(data['Job Title'])):
    if 'data scientist' in data['Job Title'][i].lower():
        jobTitles.append('Data Scientist')
    elif 'data analyst' in data['Job Title'][i].lower():
        jobTitles.append('Data Analyst')
    elif 'data engineer' in data['Job Title'][i].lower():
        jobTitles.append('Data Engineer')
    elif 'scientist' in data['Job Title'][i].lower():
        jobTitles.append('Scientist')
    elif 'machine learning' in data['Job Title'][i].lower():
        jobTitles.append('Machine Learning Engineer')
    elif 'manager' in data['Job Title'][i].lower():
        jobTitles.append('Manager')
    else:
        jobTitles.append('Other')
```

The 'Job Title' Column will be simplified to 7 job titles. Data Scientist, Data Analyst, Data Engineer, Scientist, Machine

Learning Engineer, Manager, and Other. The Scientist title refers to any sort of scientist position, such as a Biology Scientist or Research Scientist. The Manager refers to managerial positions. Finally, the Other title refers to any other job titles that do not meet the conditions in the code.

In [19]:
```
jobTitles[0:10]
```

Out[19]:
```
['Data Scientist',
 'Data Scientist',
 'Data Scientist',
 'Data Scientist',
 'Data Scientist',
 'Data Scientist',
 'Data Scientist',
 'Data Scientist',
 'Scientist',
 'Data Scientist']
```

In [20]:
```
pd.Series(i for i in jobTitles).nunique() #264 unique values down to 7
```

Out[20]:
```
7
```

In [21]:
```
ex2 = jobTitles

ex2_count = Counter(ex2)

print(ex2_count.most_common())
```

```
[('Data Scientist', 279), ('Scientist', 157), ('Data Engineer', 119), ('Data Analy
st', 99), ('Other', 55), ('Manager', 21), ('Machine Learning Engineer', 12)]
```

In [22]:
```
df_jobTitles = pd.DataFrame(jobTitles, columns=['Job Title Simplified'])


data = pd.concat([data, df_jobTitles],axis=1)
```

In [23]:
```
data.drop('Job Title', axis=1)

cols = ['Job Title Simplified','Salary Estimate', 'Rating', 'Company Name', 'State'
       'City', 'Size', 'Founded', 'Type of ownership', 'Industry', 'Sector',
       'Revenue', 'hourly', 'same_state', 'age']

data = data[cols]
```

In [24]:
```
data.head()
```

Out[24]:

	Job Title Simplified	Salary Estimate	Rating	Company Name	State	City	Size	Founded	Typ owne
0	Data Scientist	$53K-$91K (Glassdoor est.)	3.8	Tecolote Research\n3.8	NM	Albuquerque	501 to 1000 employees	1973	Comp Pr
1	Data Scientist	$63K-$112K (Glassdoor est.)	3.4	University of Maryland Medical System\n3.4	MD	Linthicum	10000+ employees	1984	Organiz
2	Data Scientist	$80K-$90K (Glassdoor est.)	4.8	KnowBe4\n4.8	FL	Clearwater	501 to 1000 employees	2010	Comp Pr
3	Data Scientist	$56K-$97K (Glassdoor est.)	3.8	PNNL\n3.8	WA	Richland	1001 to 5000 employees	1965	Govern
4	Data Scientist	$86K-$143K (Glassdoor est.)	2.9	Affinity Solutions\n2.9	NY	New York	51 to 200 employees	1998	Comp Pr

Cleaning the 'Salary Estimate' column

When cleaning the 'Salary Estimate' column there were instances were text in the string were not uniform with the other string values.

Thus, multiple conditions were created in the for loop to deal with these instances. Additionally, the numbers from the string were converted

into integer values. Furthermore, the integer values were then used to calculate the minimum salary and maximum salary in thousands of USD .

In [25]:
```
import re
```

In [26]:
```
min_salary = []
max_salary = []

#Note: the 260 used to multiply the hourly rate was the number of working days in 2

for i in range(len(data['Salary Estimate'])):
    if 'Per Hour' in data['Salary Estimate'][i]:
        if 'Employer Provided Salary' in data['Salary Estimate'][i]:
            min_salary.append((int(re.split(r'[-$P]', data['Salary Estimate'][i])[2
            max_salary.append((int(re.split(r'[-$P]', data['Salary Estimate'][i])[4
        else:
            min_salary.append((int(re.split(r'[-$P]', data['Salary Estimate'][i])[1
            max_salary.append((int(re.split(r'[-$P]', data['Salary Estimate'][i])[3
    elif 'Employer Provided Salary' in data['Salary Estimate'][i]:
        min_salary.append((int(re.split(r'[$K-]',data['Salary Estimate'][i])[1])*10
        max_salary.append((int(re.split(r'[$K-]',data['Salary Estimate'][i])[4])*10

    else:
        min_salary.append((int(re.split(r'[-K$]',data['Salary Estimate'][i])[1]) *
        max_salary.append((int(re.split(r'[-K$]',data['Salary Estimate'][i])[4]) *
```

In [27]:
```
#Concatenating the two lists, min_salary and max_salary, to the dataframe.
df_min_salary = pd.DataFrame(min_salary, columns=['Min_Salary'])
df_max_salary = pd.DataFrame(max_salary, columns=['Max_Salary'])

data = pd.concat([data, df_min_salary, df_max_salary],axis=1)
```

In [28]:
```
data.head()
```

Out[28]:

	Job Title Simplified	Salary Estimate	Rating	Company Name	State	City	Size	Founded	Typ owne
0	Data Scientist	$53K−91K (Glassdoor est.)	3.8	Tecolote Research\n3.8	NM	Albuquerque	501 to 1000 employees	1973	Comp Pr
1	Data Scientist	$63K−112K (Glassdoor est.)	3.4	University of Maryland Medical System\n3.4	MD	Linthicum	10000+ employees	1984	Organiz
2	Data Scientist	$80K−90K (Glassdoor est.)	4.8	KnowBe4\n4.8	FL	Clearwater	501 to 1000 employees	2010	Comp Pr
3	Data Scientist	$56K−97K (Glassdoor est.)	3.8	PNNL\n3.8	WA	Richland	1001 to 5000 employees	1965	Govern
4	Data Scientist	$86K−143K (Glassdoor est.)	2.9	Affinity Solutions\n2.9	NY	New York	51 to 200 employees	1998	Comp Pr

In [29]:
```
#Dropping and reorganizing columns of the dataframe.

data.drop('Salary Estimate',axis=1)

cols = ['Job Title Simplified','Min_Salary', 'Max_Salary', 'Rating', 'Company Name'
       'City', 'Size', 'Founded', 'Type of ownership', 'Industry', 'Sector',
       'Revenue', 'hourly', 'same_state', 'age']

data = data[cols]
```

In [30]:
```
data.head()
```

Out[30]:

	Job Title Simplified	Min_Salary	Max_Salary	Rating	Company Name	State	City	Size	Fou
0	Data Scientist	53000	91000	3.8	Tecolote Research\n3.8	NM	Albuquerque	501 to 1000 employees	
1	Data Scientist	63000	112000	3.4	University of Maryland Medical System\n3.4	MD	Linthicum	10000+ employees	
2	Data Scientist	80000	90000	4.8	KnowBe4\n4.8	FL	Clearwater	501 to 1000 employees	
3	Data Scientist	56000	97000	3.8	PNNL\n3.8	WA	Richland	1001 to 5000 employees	
4	Data Scientist	86000	143000	2.9	Affinity Solutions\n2.9	NY	New York	51 to 200 employees	

The next column to be cleaned will be the company name

In [31]:
```
# The primary thing that needs to be cleaned in the company name column is the '\n'
# to it.
data['Company Name'] =  data['Company Name'].apply(lambda x: x.split('\n')[0])
```

In [32]:
```
data.head()
```

Out[32]:

	Job Title Simplified	Min_Salary	Max_Salary	Rating	Company Name	State	City	Size	Founde
0	Data Scientist	53000	91000	3.8	Tecolote Research	NM	Albuquerque	501 to 1000 employees	197
1	Data Scientist	63000	112000	3.4	University of Maryland Medical System	MD	Linthicum	10000+ employees	198
2	Data Scientist	80000	90000	4.8	KnowBe4	FL	Clearwater	501 to 1000 employees	201
3	Data Scientist	56000	97000	3.8	PNNL	WA	Richland	1001 to 5000 employees	196
4	Data Scientist	86000	143000	2.9	Affinity Solutions	NY	New York	51 to 200 employees	199

The next column to be cleaned will be the Size column. The size column

will be divided into two new columns.

Minimum_Size and Maximum_Size

In [33]:
```
min_employees = []
max_employees = []

for i in range(len(data['Size'])):
    if '+' in data['Size'][i]:
        min_employees.append(int(re.split(r'[+]',data['Size'][i])[0]))
        max_employees.append(int(re.split(r'[+]',data['Size'][i])[0]))
    elif 'to' in data['Size'][i]:
        min_employees.append(int(re.split(r'[toe]',data['Size'][i])[0]))
        max_employees.append(int(re.split(r'[toe]',data['Size'][i])[2]))
    else:
        min_employees.append(np.NaN)
        max_employees.append(np.NaN)
```

In [34]:
```
df_min_employees = pd.DataFrame(min_employees, columns=['Min_Employees'])
df_max_employees = pd.DataFrame(max_employees, columns = ['Max_Employees'])

data = pd.concat([data, df_min_employees, df_max_employees], axis =1)
```

In [35]:
```
data.columns
```

Out[35]:
```
Index(['Job Title Simplified', 'Min_Salary', 'Max_Salary', 'Rating',
       'Company Name', 'State', 'City', 'Size', 'Founded', 'Type of ownership',
       'Industry', 'Sector', 'Revenue', 'hourly', 'same_state', 'age',
       'Min_Employees', 'Max_Employees'],
      dtype='object')
```

In [36]:
```
data.drop('Size', axis=1)

cols = ['Job Title Simplified', 'Min_Salary', 'Max_Salary', 'Rating',
       'Company Name', 'State', 'City', 'Min_Employees', 'Max_Employees', 'Founded'
       'Industry', 'Sector', 'Revenue', 'hourly', 'same_state', 'age']

data = data[cols]
```

In [37]:
```
data[data['Type of ownership'] == 'College / University' ]
```

Out[37]:

	Job Title Simplified	Min_Salary	Max_Salary	Rating	Company Name	State	City	Min_Employees	I
143	Scientist	81000	167000	2.6	Software Engineering Institute	PA	Pittsburgh	501.0	
181	Scientist	81000	159000	2.6	Software Engineering Institute	PA	Pittsburgh	501.0	
199	Scientist	81000	167000	2.6	Software Engineering Institute	PA	Pittsburgh	501.0	
276	Scientist	81000	159000	2.6	Software Engineering Institute	PA	Pittsburgh	501.0	
343	Scientist	81000	167000	2.6	Software Engineering Institute	PA	Pittsburgh	501.0	
371	Scientist	56000	91000	2.6	Software Engineering Institute	PA	Pittsburgh	501.0	
412	Data Scientist	82000	129000	3.7	Applied Research Laboratories	TX	Austin	501.0	
504	Scientist	81000	167000	2.6	Software Engineering Institute	PA	Pittsburgh	501.0	
557	Scientist	56000	91000	2.6	Software Engineering Institute	PA	Pittsburgh	501.0	
620	Data Scientist	82000	129000	3.7	Applied Research Laboratories	TX	Austin	501.0	
638	Scientist	71000	144000	2.6	Software Engineering Institute	PA	Pittsburgh	501.0	
680	Scientist	81000	167000	2.6	Software Engineering Institute	PA	Pittsburgh	501.0	
739	Scientist	56000	91000	2.6	Software Engineering Institute	PA	Pittsburgh	501.0	

Next column to be cleaned is the 'Type of ownership' column. The goal is just to simply the categories in this column.

In [38]:
```
own = []
```

```
for i in range(len(data['Type of ownership'])):
    if 'Company' in data['Type of ownership'][i]:
        own.append(re.split(r'[-]',data['Type of ownership'][i])[1].strip())
    elif 'Other' in data['Type of ownership'][i]:
        own.append('Other')
    elif 'School' in data['Type of ownership'][i]:
        own.append('Other')
    elif 'Unknown' in data['Type of ownership'][i]:
        own.append('Other')
    elif '-1' in data['Type of ownership'][i]:
        own.append('Other')
    elif 'Subsidiary' in data['Type of ownership'][i]:
        own.append('Subsidiary')
    else:
        own.append(data['Type of ownership'][i])
```

In [39]:
```
df_own = pd.DataFrame(own, columns=['Ownership Type'])


data = pd.concat([data, df_own], axis = 1)
```

In [40]:
```
data.drop('Type of ownership', axis=1)

cols = ['Job Title Simplified', 'Min_Salary', 'Max_Salary', 'Rating',
       'Company Name', 'State', 'City', 'Min_Employees', 'Max_Employees',
       'Founded', 'Ownership Type','Industry', 'Sector', 'Revenue',
       'hourly', 'same_state', 'age']

data = data[cols]
```

In [41]:
```
data.head()
```

Out[41]:

	Job Title Simplified	Min_Salary	Max_Salary	Rating	Company Name	State	City	Min_Employees	Ma
0	Data Scientist	53000	91000	3.8	Tecolote Research	NM	Albuquerque	501.0	
1	Data Scientist	63000	112000	3.4	University of Maryland Medical System	MD	Linthicum	10000.0	
2	Data Scientist	80000	90000	4.8	KnowBe4	FL	Clearwater	501.0	
3	Data Scientist	56000	97000	3.8	PNNL	WA	Richland	1001.0	
4	Data Scientist	86000	143000	2.9	Affinity Solutions	NY	New York	51.0	

The final column to be cleaned will be the "Revenue" column. If the

column specifies a value range, such as 2 to 5 billion dollars, then the highest value will be kept.

In [42]:
```
data['Revenue'][0:10]
```

Out[42]:
```
0           $50 to $100 million (USD)
1              $2 to $5 billion (USD)
2          $100 to $500 million (USD)
3    $500 million to $1 billion (USD)
4             Unknown / Non-Applicable
5              $1 to $2 billion (USD)
6             Unknown / Non-Applicable
7             $25 to $50 million (USD)
8    $500 million to $1 billion (USD)
9          $100 to $500 million (USD)
Name: Revenue, dtype: object
```

In [43]:
```
rev = []

for i in range(len(data['Revenue'])):
    if 'to' in data['Revenue'][i]:
        if 'million' in data['Revenue'][i]:
            if 'billion' in data['Revenue'][i]:
                rev.append(int(re.split(r'[$tomb]', data['Revenue'][i])[6].strip())
            else:
                rev.append(int(re.split(r'[$tom]', data['Revenue'][i])[4].strip())
        elif 'billion' in data['Revenue'][i]:
            rev.append(int(re.split(r'[$tob]', data['Revenue'][i])[4].strip()) * 10
    elif '+' in data['Revenue'][i]:
        rev.append(int(re.split(r'[$+]', data['Revenue'][i])[1].strip()) * 10000000
    else:
        rev.append(np.NaN)
```

In [44]:
```
df_rev = pd.DataFrame(rev, columns = ['Revenue (USD)'])
data = pd.concat([data, df_rev], axis=1)
```

In [45]:
```
data.columns

cols = ['Job Title Simplified', 'Min_Salary', 'Max_Salary', 'Rating',
       'Company Name', 'State', 'City', 'Min_Employees', 'Max_Employees',
       'Founded', 'Ownership Type', 'Industry', 'Sector', 'Revenue (USD)',
       'hourly','same_state', 'age']

data = data[cols]
```

In [46]:
```
data.head()
```

Out[46]:

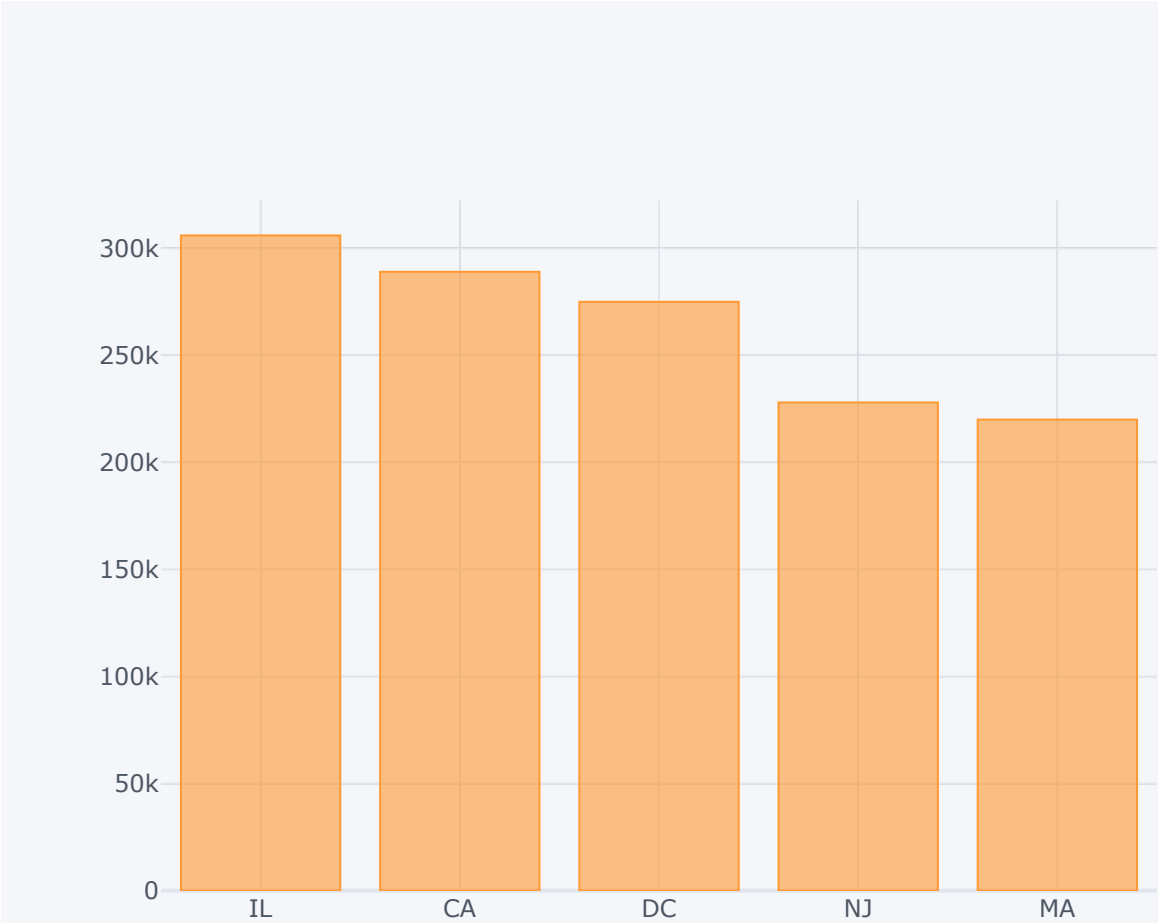
	Job Title Simplified	Min_Salary	Max_Salary	Rating	Company Name	State	City	Min_Employees	Ma
0	Data Scientist	53000	91000	3.8	Tecolote Research	NM	Albuquerque	501.0	
1	Data Scientist	63000	112000	3.4	University of Maryland Medical System	MD	Linthicum	10000.0	
2	Data Scientist	80000	90000	4.8	KnowBe4	FL	Clearwater	501.0	
3	Data Scientist	56000	97000	3.8	PNNL	WA	Richland	1001.0	
4	Data Scientist	86000	143000	2.9	Affinity Solutions	NY	New York	51.0	

Exploratory Data Analysis (EDA)

Finding the Top 5 states that have jobs with the highest paying salary

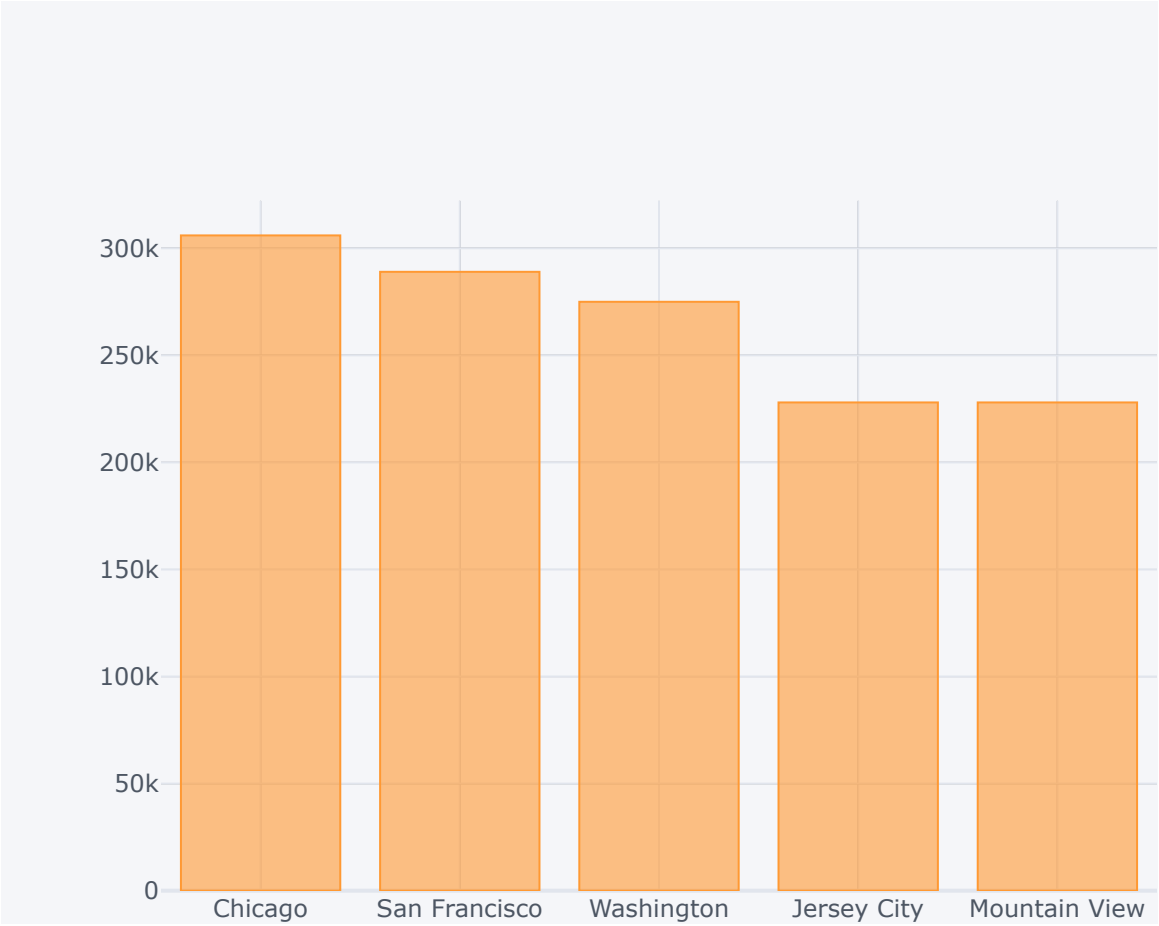
In [47]:
```
import cufflinks as cf
cf.go_offline()
cf.set_config_file(offline=False, world_readable=True)
```

In [48]:
```
data.groupby(['State'])['Max_Salary'].max().nlargest(5).iplot(kind='bar')
```



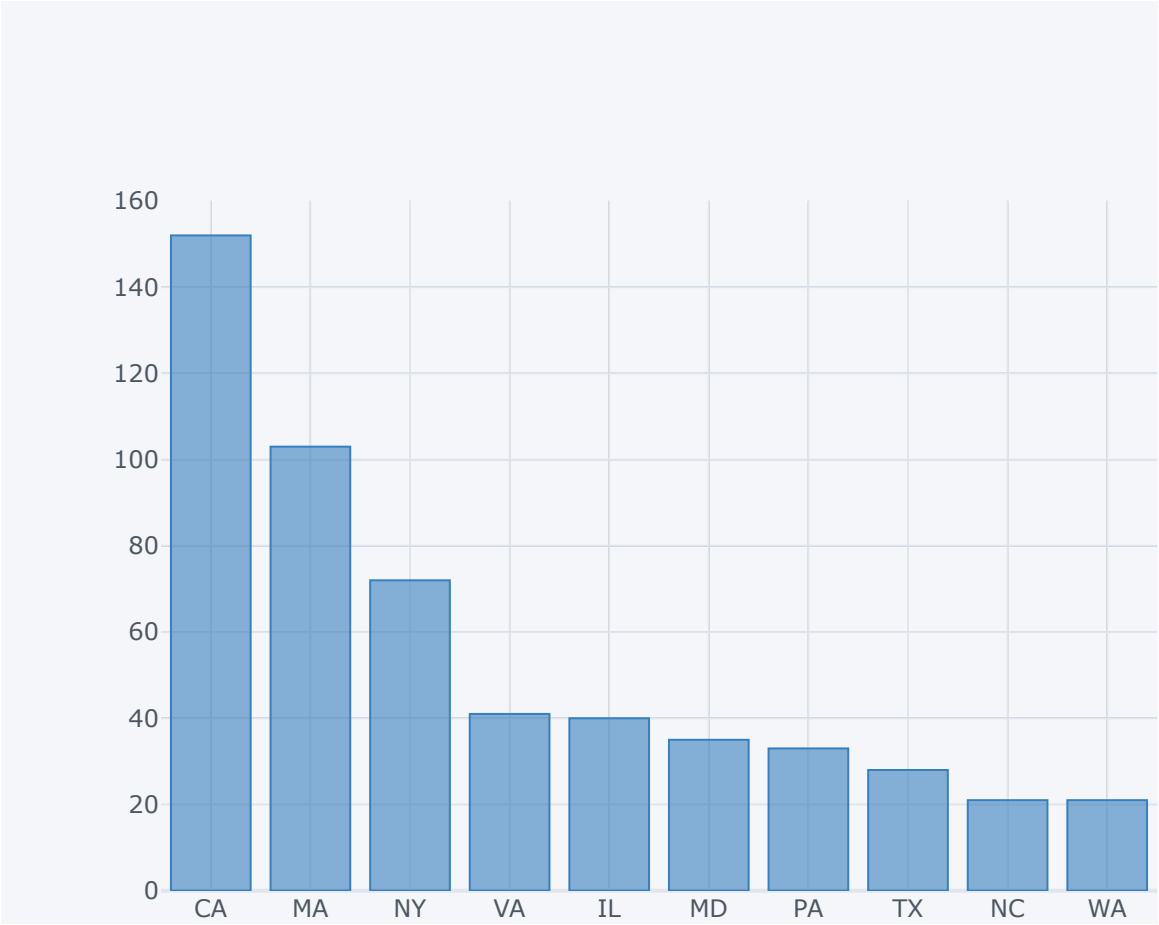
Determining which top 5 cities offer the highest paying technology jobs.

In [49]:
```
data.groupby('City')['Max_Salary'].max().nlargest(5).iplot(kind='bar')
```



We see that for most of the top 5 cities with the highest paying job, are within their respective state. However, Mountain View, a city in California, is not within massachusetts.

In [50]:
```
data['State'].value_counts().nlargest(10).iplot(kind='bar', color='blue')
```



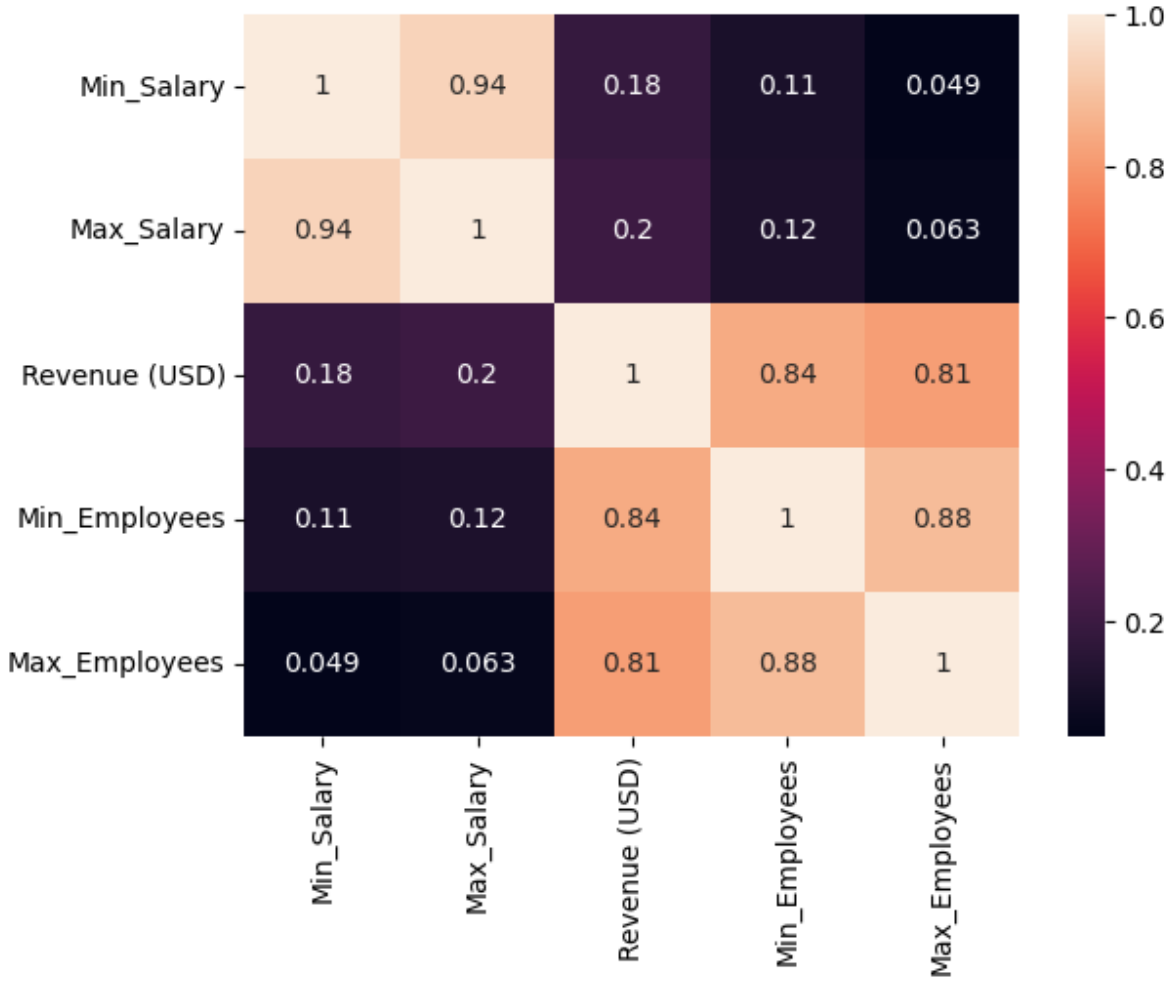
We see from the above plot, that roughly 44% of all tech related job postings come from the states, California, Massachusetts, and New York.

Now, the columns, 'Min_Salary,' 'Max_Salary,' 'Revenue (USD),' 'Min_Employees,' and 'Max_Employees', will be checked to see for any high correlations between the variables. Primarily, to see if the salary columns have any correlations with revenue and the number of employees that a company has.

In [51]:
```
df2 = data[['Min_Salary','Max_Salary','Revenue (USD)','Min_Employees','Max_Employee
```

In [52]:
```
sns.heatmap(df2.corr(),annot=True)
```

Out[52]:
```
<AxesSubplot: >
```



We see from the above heatmap that the salary columns are only highly correlated with one another and the other three columns, 'Revenue (USD),' 'Min_Employees,' 'Max_Employees,' are highly correlated with one another. It can be deduced that more employees are correlated to higher revenue due to more workers being present to do work.

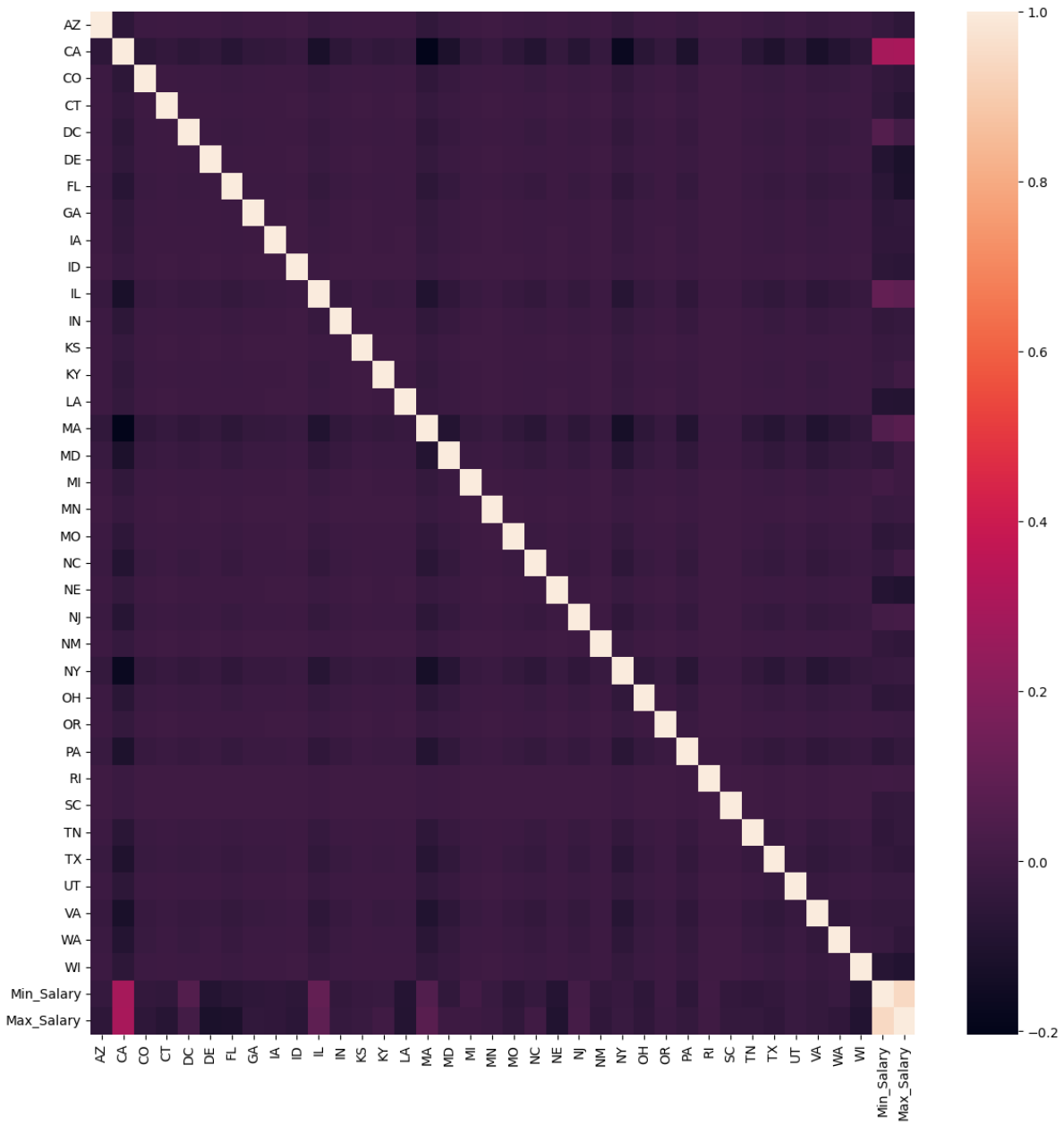
In [53]:
```
df3 = data[['Job Title Simplified','State', 'Sector','Min_Salary', 'Max_Salary']].c
```

In [54]:
```
states = pd.get_dummies(data['State'],drop_first=True)
```

In [55]:
```
df3_state = pd.concat([states,data[['Min_Salary','Max_Salary']]],axis=1)
```

In [56]:
```
plt.figure(figsize=(14,14))
sns.heatmap(df3_state.corr())
```

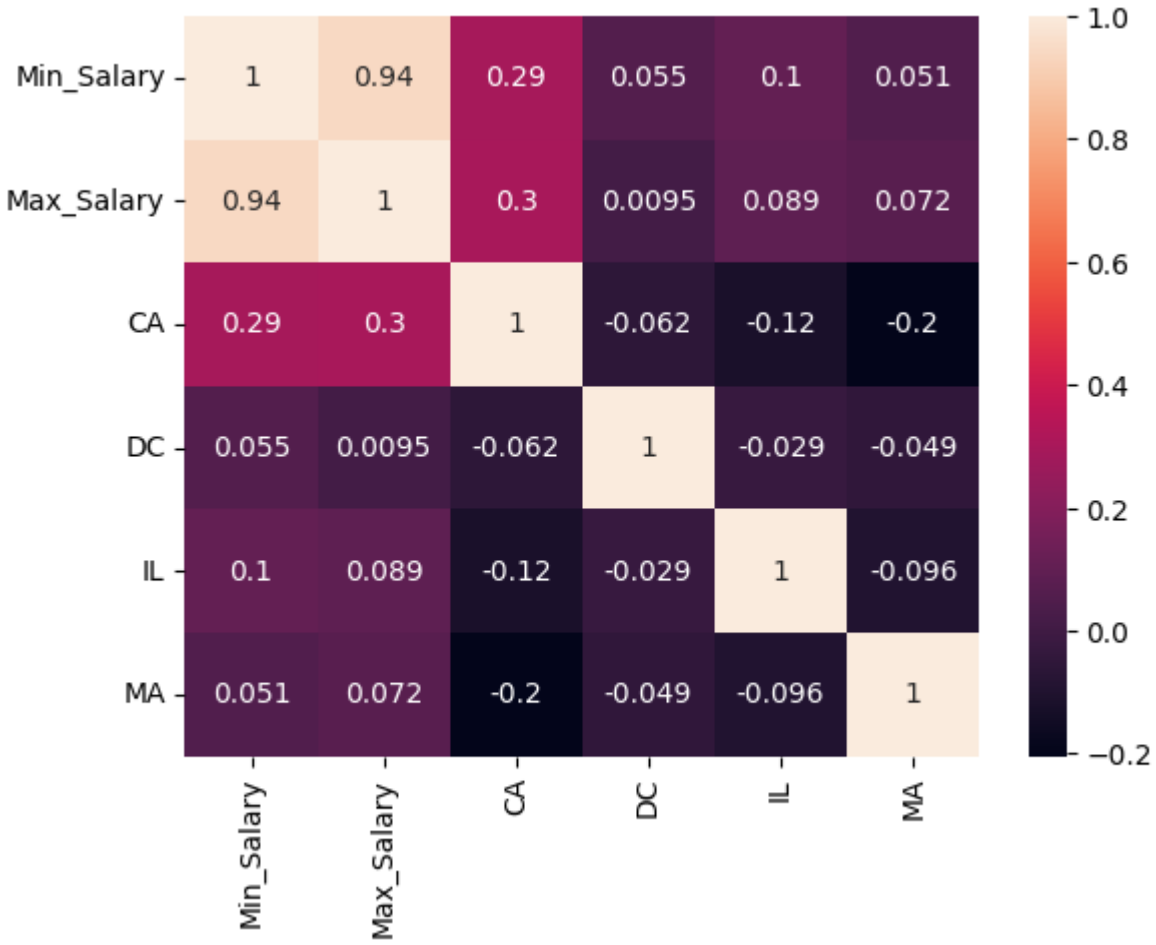
Out[56]:
```
<AxesSubplot: >
```



We see from the above heatmap that the states that have the highest correlation with the salary columns are California, the capital (DC) , Illinois, and Massachusetts

In [57]:
```
sns.heatmap(df3_state[['Min_Salary','Max_Salary','CA','DC','IL','MA']].corr(),annot
```

Out[57]:
```
<AxesSubplot: >
```



By taking a closer look at the four states and their correlation with the salary columns, we see that states are still not highly correlated with the salary columns. California is the highest correlated state with the salary columns.

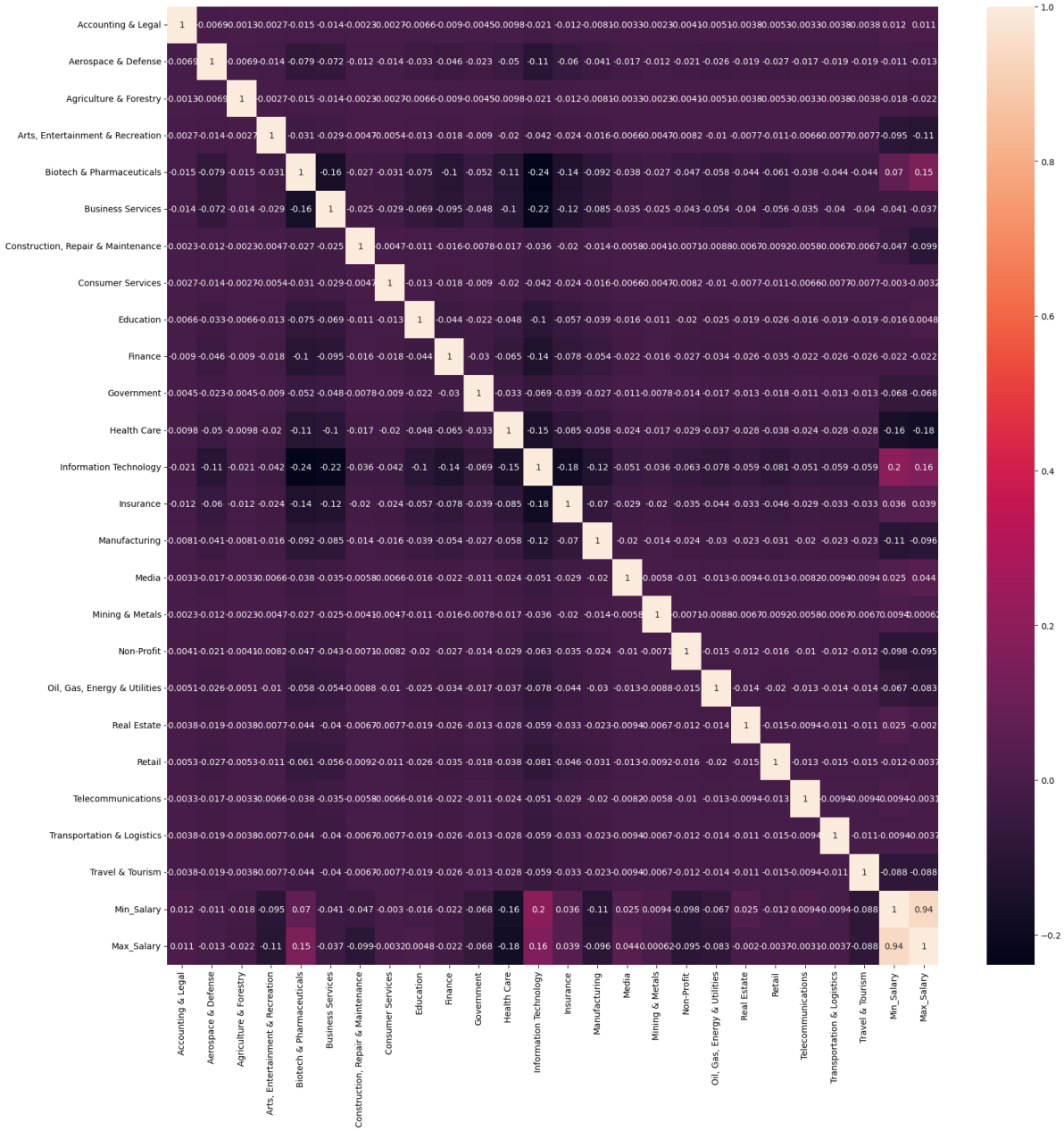
Now, the correlation between Sector and the salary columns will be checked

In [58]:
```
sector = pd.get_dummies(data['Sector'],drop_first=True)
```

In [59]:
```
df4_sector = pd.concat([sector,data[['Min_Salary','Max_Salary']]],axis=1)
```

In [60]:
```
plt.figure(figsize=(20,20))
sns.heatmap(df4_sector.corr(),annot=True)
```

Out[60]:
```
<AxesSubplot: >
```



We see from the above heatmap that there are no sectors that are highly correlated with any of the two salary columns. The sector that is highly correlated with the two salary columns is the Information Technology

Finally, the correlation between the Job Title and the salary's column will be looked at.

In [61]:
```
job_titles = pd.get_dummies(data['Job Title Simplified'],drop_first=True)
```

In [62]:
```
df5_titles = pd.concat([job_titles,data[['Min_Salary','Max_Salary']]],axis=1)
```

In [63]:
```
sns.heatmap(df5_titles.corr(),annot=True)
```

Out[63]: <AxesSubplot: >

Based on the above heatmap,we see again that the Job Title categories are not highly correlated with the salaries columns. However, we do see that the Job Title, 'Data Scientist,' is the most highly correlated value to both of the salaries columns.

In [64]:
```
data['Job Title Simplified'].value_counts()
```

Out[64]:
```
Data Scientist                279
Scientist                     157
Data Engineer                 119
Data Analyst                   99
Other                          55
Manager                        21
Machine Learning Engineer      12
Name: Job Title Simplified, dtype: int64
```

We see that most of the entries in the data set are for Data Scientist positions. Thus, it can be deduced that most of the higher paying salaries are for Data Scientist roles.

To conclude, the dataset needed to be extensively cleaned in order to be usable for Exploratory Data Analysis (EDA). After the dataset was cleaned, EDA was conducted in order to find any patterns with the dataset. Plots were created to quickly identify which states and cities offered tech positions with the

highest paying salaries. Furthermore, the variables of interest were lowly correlated with the target salary variables. Furthermore, there is insufficient data to deduce the tech jobs that have the highest paying salaries due to most entries being for the Data Scientist Position (this was the case also for the other variables that were looked at).