Import important library

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

Upload and Read Data

In [6]:
```
df = pd.read_csv(r"C:\Users\meanu\Downloads\salary dataset based on country and race\Salary_Data_Based_country_and_race.csv")
```

Get Top 5 Data

In [7]:
```
df.head()
```

Out[7]:

	Unnamed: 0	Age	Gender	Education Level	Job Title	Years of Experience	Salary	Country	Race
0	0	32.0	Male	Bachelor's	Software Engineer	5.0	90000.0	UK	White
1	1	28.0	Female	Master's	Data Analyst	3.0	65000.0	USA	Hispanic
2	2	45.0	Male	PhD	Senior Manager	15.0	150000.0	Canada	White
3	3	36.0	Female	Bachelor's	Sales Associate	7.0	60000.0	USA	Hispanic
4	4	52.0	Male	Master's	Director	20.0	200000.0	USA	Asian

Know the shape of data

In [8]:
```
df.shape
```

Out[8]:
```
(6704, 9)
```

Get bottom 5 data

In [9]:

```
df.tail()
```

Out[9]:

	Unnamed: 0	Age	Gender	Education Level	Job Title	Years of Experience	Salary	Country	Race
6699	6699	49.0	Female	PhD	Director of Marketing	20.0	200000.0	UK	Mixed
6700	6700	32.0	Male	High School	Sales Associate	3.0	50000.0	Australia	Australian
6701	6701	30.0	Female	Bachelor's Degree	Financial Manager	4.0	55000.0	China	Chinese
6702	6702	46.0	Male	Master's Degree	Marketing Manager	14.0	140000.0	China	Korean
6703	6703	26.0	Female	High School	Sales Executive	1.0	35000.0	Canada	Black

Describe the dataset

In [10]:

```
df.describe
```

Out[10]:

```
<bound method NDFrame.describe of       Unnamed: 0   Age  Gender     Education Level               Job Title  \
0              0  32.0    Male         Bachelor's      Software Engineer
1              1  28.0  Female           Master's           Data Analyst
2              2  45.0    Male                PhD         Senior Manager
3              3  36.0  Female         Bachelor's        Sales Associate
4              4  52.0    Male           Master's               Director
...          ...   ...     ...                ...                    ...
6699        6699  49.0  Female                PhD  Director of Marketing
6700        6700  32.0    Male        High School        Sales Associate
6701        6701  30.0  Female  Bachelor's Degree      Financial Manager
6702        6702  46.0    Male    Master's Degree      Marketing Manager
6703        6703  26.0  Female        High School        Sales Executive

      Years of Experience    Salary    Country        Race
0                     5.0   90000.0         UK       White
1                     3.0   65000.0        USA    Hispanic
2                    15.0  150000.0     Canada       White
3                     7.0   60000.0        USA    Hispanic
4                    20.0  200000.0        USA       Asian
...                   ...       ...        ...         ...
6699                 20.0  200000.0         UK       Mixed
6700                  3.0   50000.0  Australia  Australian
6701                  4.0   55000.0      China     Chinese
6702                 14.0  140000.0      China      Korean
6703                  1.0   35000.0     Canada       Black

[6704 rows x 9 columns]>
```

Get the Unique element from the Job Title data

In [12]:

```
df['Job Title'].unique()
```

```
Out[12]: array(['Software Engineer', 'Data Analyst', 'Senior Manager',
                'Sales Associate', 'Director', 'Marketing Analyst',
                'Product Manager', 'Sales Manager', 'Marketing Coordinator',
                'Senior Scientist', 'Software Developer', 'HR Manager',
                'Financial Analyst', 'Project Manager', 'Customer Service Rep',
                'Operations Manager', 'Marketing Manager', 'Senior Engineer',
                'Data Entry Clerk', 'Sales Director', 'Business Analyst',
                'VP of Operations', 'IT Support', 'Recruiter', 'Financial Manager',
                'Social Media Specialist', 'Software Manager', 'Junior Developer',
                'Senior Consultant', 'Product Designer', 'CEO', 'Accountant',
                'Data Scientist', 'Marketing Specialist', 'Technical Writer',
                'HR Generalist', 'Project Engineer', 'Customer Success Rep',
                'Sales Executive', 'UX Designer', 'Operations Director',
                'Network Engineer', 'Administrative Assistant',
                'Strategy Consultant', 'Copywriter', 'Account Manager',
                'Director of Marketing', 'Help Desk Analyst',
                'Customer Service Manager', 'Business Intelligence Analyst',
                'Event Coordinator', 'VP of Finance', 'Graphic Designer',
                'UX Researcher', 'Social Media Manager', 'Director of Operations',
                'Senior Data Scientist', 'Junior Accountant',
                'Digital Marketing Manager', 'IT Manager',
                'Customer Service Representative', 'Business Development Manager',
                'Senior Financial Analyst', 'Web Developer', 'Research Director',
                'Technical Support Specialist', 'Creative Director',
                'Senior Software Engineer', 'Human Resources Director',
                'Content Marketing Manager', 'Technical Recruiter',
                'Sales Representative', 'Chief Technology Officer',
                'Junior Designer', 'Financial Advisor', 'Junior Account Manager',
                'Senior Project Manager', 'Principal Scientist',
                'Supply Chain Manager', 'Senior Marketing Manager',
                'Training Specialist', 'Research Scientist',
                'Junior Software Developer', 'Public Relations Manager',
                'Operations Analyst', 'Product Marketing Manager',
                'Senior HR Manager', 'Junior Web Developer',
                'Senior Project Coordinator', 'Chief Data Officer',
                'Digital Content Producer', 'IT Support Specialist',
                'Senior Marketing Analyst', 'Customer Success Manager',
                'Senior Graphic Designer', 'Software Project Manager',
                'Supply Chain Analyst', 'Senior Business Analyst',
                'Junior Marketing Analyst', 'Office Manager', 'Principal Engineer',
                'Junior HR Generalist', 'Senior Product Manager',
                'Junior Operations Analyst', 'Senior HR Generalist',
                'Sales Operations Manager', 'Senior Software Developer',
                'Junior Web Designer', 'Senior Training Specialist',
                'Senior Research Scientist', 'Junior Sales Representative',
                'Junior Marketing Manager', 'Junior Data Analyst',
                'Senior Product Marketing Manager', 'Junior Business Analyst',
                'Senior Sales Manager', 'Junior Marketing Specialist',
                'Junior Project Manager', 'Senior Accountant', 'Director of Sales',
                'Junior Recruiter', 'Senior Business Development Manager',
                'Senior Product Designer', 'Junior Customer Support Specialist',
                'Senior IT Support Specialist', 'Junior Financial Analyst',
                'Senior Operations Manager', 'Director of Human Resources',
                'Junior Software Engineer', 'Senior Sales Representative',
                'Director of Product Management', 'Junior Copywriter',
                'Senior Marketing Coordinator', 'Senior Human Resources Manager',
                'Junior Business Development Associate', 'Senior Account Manager',
                'Senior Researcher', 'Junior HR Coordinator',
                'Director of Finance', 'Junior Marketing Coordinator', nan,
                'Junior Data Scientist', 'Senior Operations Analyst',
                'Senior Human Resources Coordinator', 'Senior UX Designer',
                'Junior Product Manager', 'Senior Marketing Specialist',
                'Senior IT Project Manager', 'Senior Quality Assurance Analyst',
                'Director of Sales and Marketing', 'Senior Account Executive',
                'Director of Business Development', 'Junior Social Media Manager',
                'Senior Human Resources Specialist', 'Senior Data Analyst',
                'Director of Human Capital', 'Junior Advertising Coordinator',
                'Junior UX Designer', 'Senior Marketing Director',
                'Senior IT Consultant', 'Senior Financial Advisor',
                'Junior Business Operations Analyst',
                'Junior Social Media Specialist',
                'Senior Product Development Manager', 'Junior Operations Manager',
                'Senior Software Architect', 'Junior Research Scientist',
                'Senior Financial Manager', 'Senior HR Specialist',
                'Senior Data Engineer', 'Junior Operations Coordinator',
                'Director of HR', 'Senior Operations Coordinator',
                'Junior Financial Advisor', 'Director of Engineering',
                'Software Engineer Manager', 'Back end Developer',
                'Senior Project Engineer', 'Full Stack Engineer',
                'Front end Developer', 'Developer', 'Front End Developer',
                'Director of Data Science', 'Human Resources Coordinator',
                'Junior Sales Associate', 'Human Resources Manager',
                'Juniour HR Generalist', 'Juniour HR Coordinator',
                'Digital Marketing Specialist', 'Receptionist',
                'Marketing Director', 'Social M', 'Social Media Man',
                'Delivery Driver'], dtype=object)
```

Get the unique element from the Education label data

In [14]:

```
df['Education Level'].unique()
```

Out[14]:

```
array(["Bachelor's", "Master's", 'PhD', nan, "Bachelor's Degree",
       "Master's Degree", 'High School', 'phD'], dtype=object)
```

Data cleaning

In [15]:

```
df.isna()
```

Out[15]:

	Unnamed: 0	Age	Gender	Education Level	Job Title	Years of Experience	Salary	Country	Race
0	False	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...	...
6699	False	False	False	False	False	False	False	False	False
6700	False	False	False	False	False	False	False	False	False
6701	False	False	False	False	False	False	False	False	False
6702	False	False	False	False	False	False	False	False	False
6703	False	False	False	False	False	False	False	False	False

6704 rows × 9 columns

In [17]:

```
df.isna().sum()
```

```
Out[17]: Unnamed: 0             0
         Age                    2
         Gender                 2
         Education Level        3
         Job Title              2
         Years of Experience    3
         Salary                 5
         Country                0
         Race                   0
         dtype: int64
```

Total salary count

```
In [20]: df.Salary.value_counts()
```

```
Out[20]: 140000.0    287
         120000.0    282
         160000.0    276
         55000.0     251
         60000.0     231
                    ...
         150534.0      1
         68732.0       1
         187951.0      1
         137336.0      1
         178284.0      1
         Name: Salary, Length: 444, dtype: int64
```

```
In [21]: df.iloc[0,1]
```

```
Out[21]: 32.0
```

```
In [22]: df.iloc[5555,5]
```

```
Out[22]: 2.0
```

Arrange identical data into groups

```
In [23]: df.groupby('Salary').max()
```

```
C:\Users\meanu\AppData\Local\Temp\ipykernel_26772\1156658275.py:1: FutureWarning: Dropping invalid columns in DataFrameGroupBy.m
ax is deprecated. In a future version, a TypeError will be raised. Before calling .max, select only columns which should be vali
d for the function.
  df.groupby('Salary').max()
```

Out[23]:

	Unnamed: 0	Age	Gender	Job Title	Years of Experience	Country	Race
Salary							
350.0	259	29.0	Male	Junior Business Operations Analyst	1.5	USA	Hispanic
500.0	4633	31.0	Female	Junior HR Coordinator	4.0	USA	Asian
550.0	1890	25.0	Female	Front end Developer	1.0	UK	Mixed
579.0	2654	23.0	Male	Software Engineer Manager	1.0	UK	Mixed
25000.0	6254	33.0	Male	Sales Associate	1.0	USA	White
...	...	...	...	...	...	...	...
220000.0	4132	49.0	Male	Director of Data Science	22.0	USA	White
225000.0	4257	50.0	Male	Data Scientist	23.0	USA	White
228000.0	4397	49.0	Male	Marketing Manager	23.0	Canada	White
240000.0	4381	51.0	Male	Data Scientist	24.0	USA	White
250000.0	5001	52.0	Male	Financial Manager	25.0	Canada	Black

444 rows × 7 columns

In [24]:

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

Out[24]:

```
array(['White', 'Hispanic', 'Asian', 'Korean', 'Chinese', 'Australian',
       'Welsh', 'African American', 'Mixed', 'Black'], dtype=object)
```

In [26]:

```
df.groupby('Years of Experience').max()
```

```
C:\Users\meanu\AppData\Local\Temp\ipykernel_26772\2003821097.py:1: FutureWarning: Dropping invalid columns in DataFrameGroupBy.m
ax is deprecated. In a future version, a TypeError will be raised. Before calling .max, select only columns which should be vali
d for the function.
  df.groupby('Years of Experience').max()
```

Out[26]:

	Unnamed: 0	Age	Gender	Job Title	Salary	Country	Race
Years of Experience							
0.0	6254	25.0	Male	Software Engineer Manager	55538.0	USA	White
0.5	114	23.0	Female	Junior Marketing Analyst	35000.0	USA	White
1.0	6703	33.0	Male	Web Developer	119836.0	USA	White
1.5	310	29.0	Male	Junior UX Designer	50000.0	USA	White
2.0	6694	36.0	Other	Web Developer	125000.0	USA	White
3.0	6700	36.0	Male	Web Developer	180000.0	USA	White
4.0	6701	34.0	Male	Web Developer	182000.0	USA	White
5.0	6687	36.0	Male	Web Developer	180000.0	USA	White
6.0	6698	37.0	Male	Web Developer	180000.0	USA	White
7.0	6695	37.0	Male	Web Developer	185000.0	USA	White
8.0	6681	45.0	Other	Web Developer	190000.0	USA	White
9.0	6691	39.0	Male	Software Project Manager	195000.0	USA	White
10.0	6677	42.0	Male	Web Developer	195000.0	USA	White
11.0	6587	44.0	Male	Software Manager	198000.0	USA	White
12.0	6580	47.0	Male	Training Specialist	196000.0	USA	White
13.0	6690	46.0	Male	Strategy Consultant	197000.0	USA	White
14.0	6702	54.0	Other	Software Engineer Manager	195000.0	USA	White
15.0	6679	50.0	Male	Software Engineer Manager	210000.0	USA	White
16.0	6688	57.0	Male	Software Engineer Manager	220000.0	USA	White
17.0	6585	58.0	Male	Software Engineer Manager	200000.0	USA	White
18.0	6489	60.0	Male	Supply Chain Manager	210000.0	USA	White
19.0	6697	62.0	Male	VP of Operations	210000.0	USA	White
20.0	6699	62.0	Male	Software Engineer Manager	220000.0	USA	White

	Unnamed: 0	Age	Gender	Job Title	Salary	Country	Race
Years of Experience							
21.0	5001	51.0	Male	Software Engineer Manager	250000.0	USA	White
22.0	4513	51.0	Male	Supply Chain Analyst	220000.0	USA	White
23.0	4397	52.0	Male	Software Engineer Manager	228000.0	USA	White
24.0	4381	52.0	Male	Software Engineer Manager	250000.0	USA	White
25.0	3067	54.0	Male	Software Engineer Manager	250000.0	USA	White
26.0	3126	52.0	Male	Software Engineer Manager	194638.0	USA	White
27.0	3047	58.0	Male	Software Engineer Manager	190596.0	USA	White
28.0	3120	55.0	Male	Software Engineer Manager	193964.0	USA	White
29.0	3041	55.0	Other	Software Engineer Manager	194778.0	USA	White
30.0	3104	57.0	Male	Software Engineer Manager	186321.0	USA	Welsh
31.0	2632	56.0	Other	Software Engineer Manager	197354.0	UK	White
32.0	3084	54.0	Male	Software Engineer Manager	195270.0	USA	White
33.0	2515	60.0	Female	Software Engineer Manager	191790.0	UK	White
34.0	2501	60.0	Female	Software Engineer Manager	188651.0	China	Korean

Drop the duplicate values

In [27]:

```
df= df.drop_duplicates()
```

Getting information about the dataset

In [28]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 6704 entries, 0 to 6703
Data columns (total 9 columns):
 #   Column               Non-Null Count  Dtype  
---  ------               --------------  -----  
 0   Unnamed: 0           6704 non-null   int64  
 1   Age                  6702 non-null   float64
 2   Gender               6702 non-null   object 
 3   Education Level      6701 non-null   object 
 4   Job Title            6702 non-null   object 
 5   Years of Experience  6701 non-null   float64
 6   Salary               6699 non-null   float64
 7   Country              6704 non-null   object 
 8   Race                 6704 non-null   object 
dtypes: float64(3), int64(1), object(5)
memory usage: 523.8+ KB
```

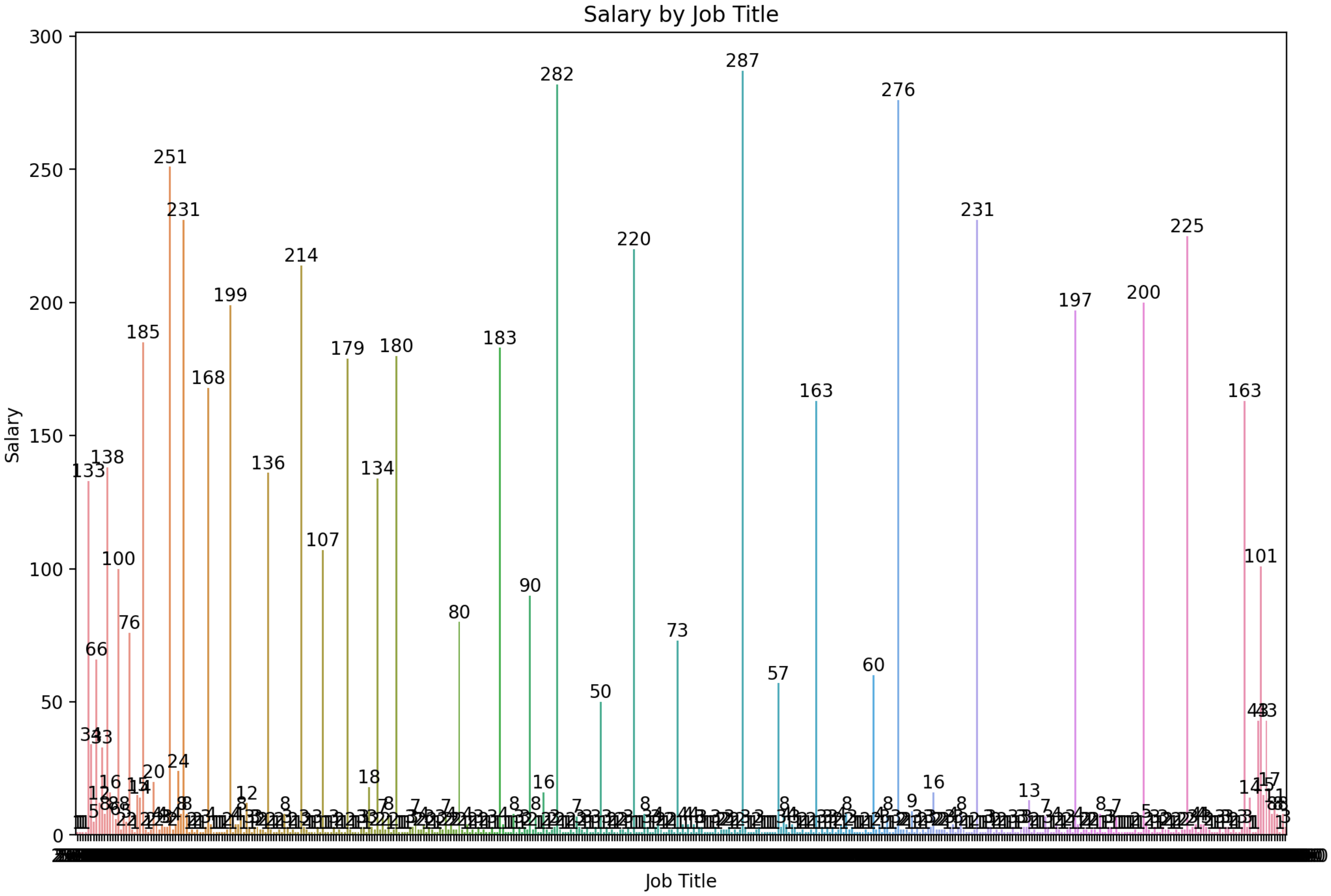
In [31]:

```python
Salary_counts = df['Salary'].value_counts()
```

Data visualization using matplotlib and seaborn

In [32]:

```python
plt.figure(figsize=(12, 8), dpi = 200)
ax = sns.barplot(x = Salary_counts.index, y = Salary_counts.values, width = 0.7)
for bars in ax.containers:
    ax.bar_label(bars)
plt.xlabel('Job Title')
plt.ylabel('Salary')
plt.title('Salary by Job Title')

plt.show()
```

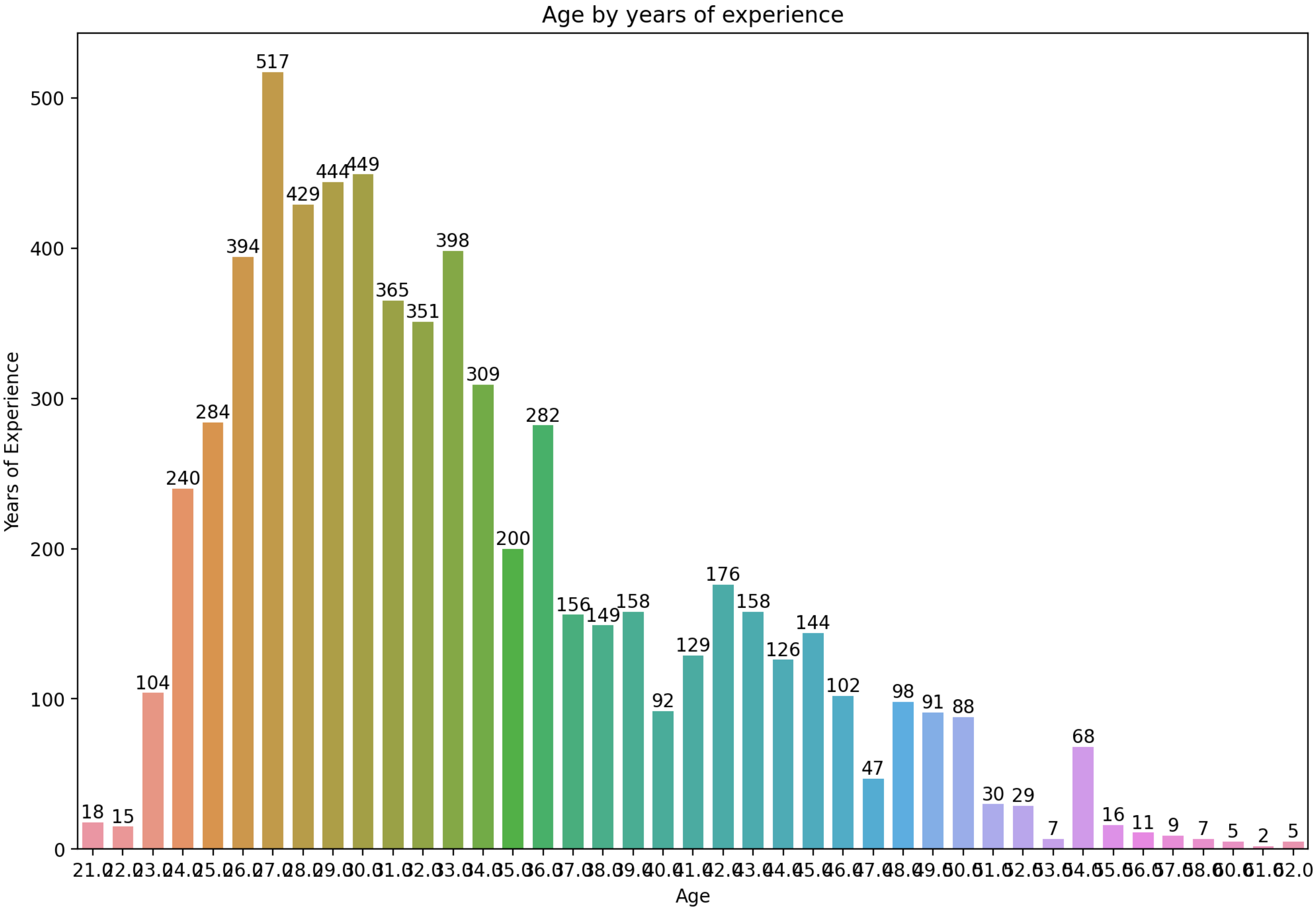


```
In [33]: Age_counts = df['Age'].value_counts()
         Age_counts
```

```
Out[33]: 27.0    517
         30.0    449
         29.0    444
         28.0    429
         33.0    398
         26.0    394
         31.0    365
         32.0    351
         34.0    309
         25.0    284
         36.0    282
         24.0    240
         35.0    200
         42.0    176
         43.0    158
         39.0    158
         37.0    156
         38.0    149
         45.0    144
         41.0    129
         44.0    126
         23.0    104
         46.0    102
         48.0     98
         40.0     92
         49.0     91
         50.0     88
         54.0     68
         47.0     47
         51.0     30
         52.0     29
         21.0     18
         55.0     16
         22.0     15
         56.0     11
         57.0      9
         53.0      7
         58.0      7
         62.0      5
         60.0      5
         61.0      2
         Name: Age, dtype: int64
```

In [35]:

```
plt.figure(figsize=(12,8), dpi = 200)

ax = sns.barplot(x = Age_counts.index, y = Age_counts.values, width = 0.7)

for bars in ax.containers:
    ax.bar_label(bars)
plt.xlabel('Age')
plt.ylabel('Years of Experience')
plt.title('Age by years of experience')

plt.show()
```



In [36]:

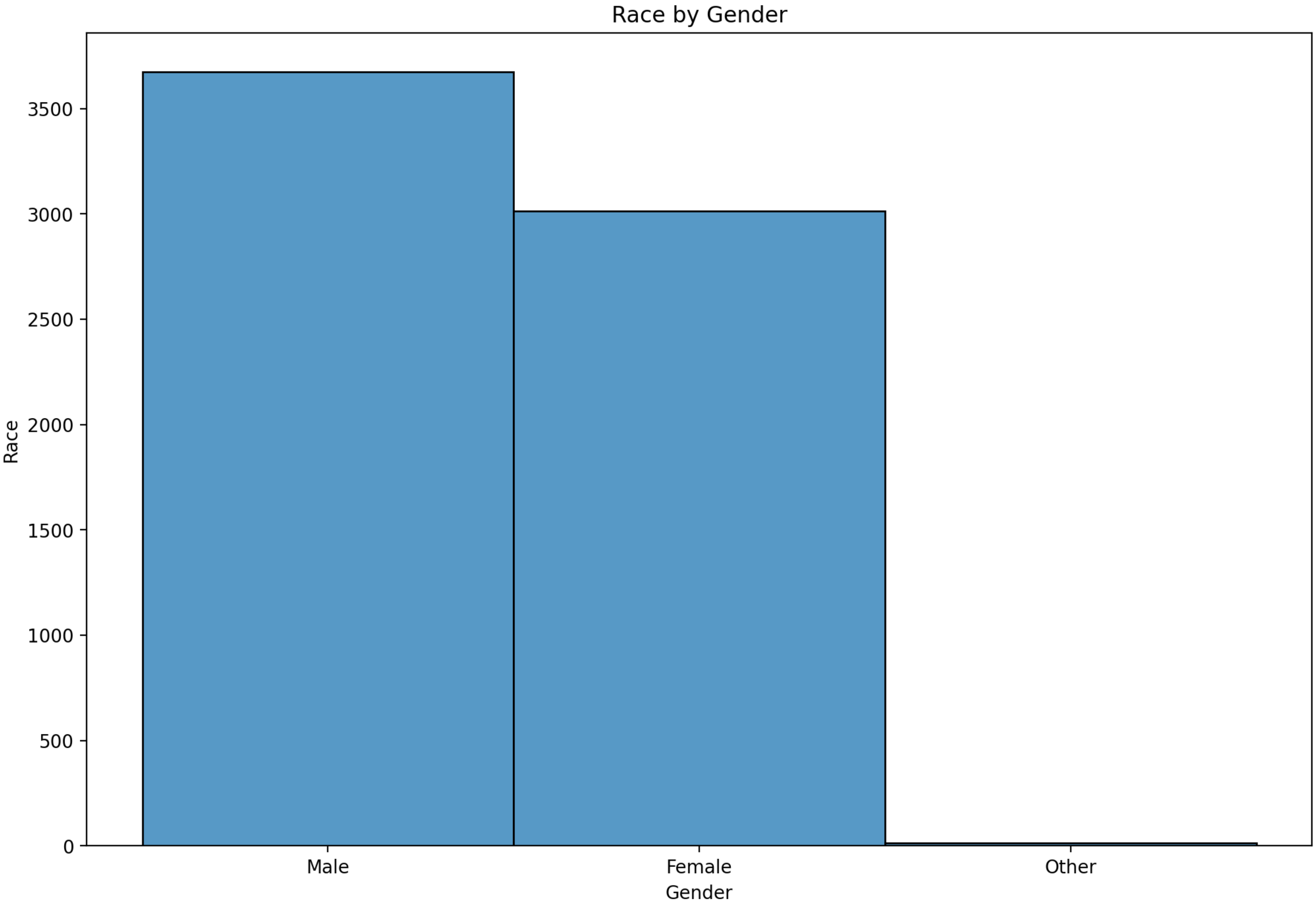
```
Gender_counts = df['Gender'].value_counts()
Gender_counts
```

Out[36]:

```
Male      3674
Female    3014
Other       14
Name: Gender, dtype: int64
```

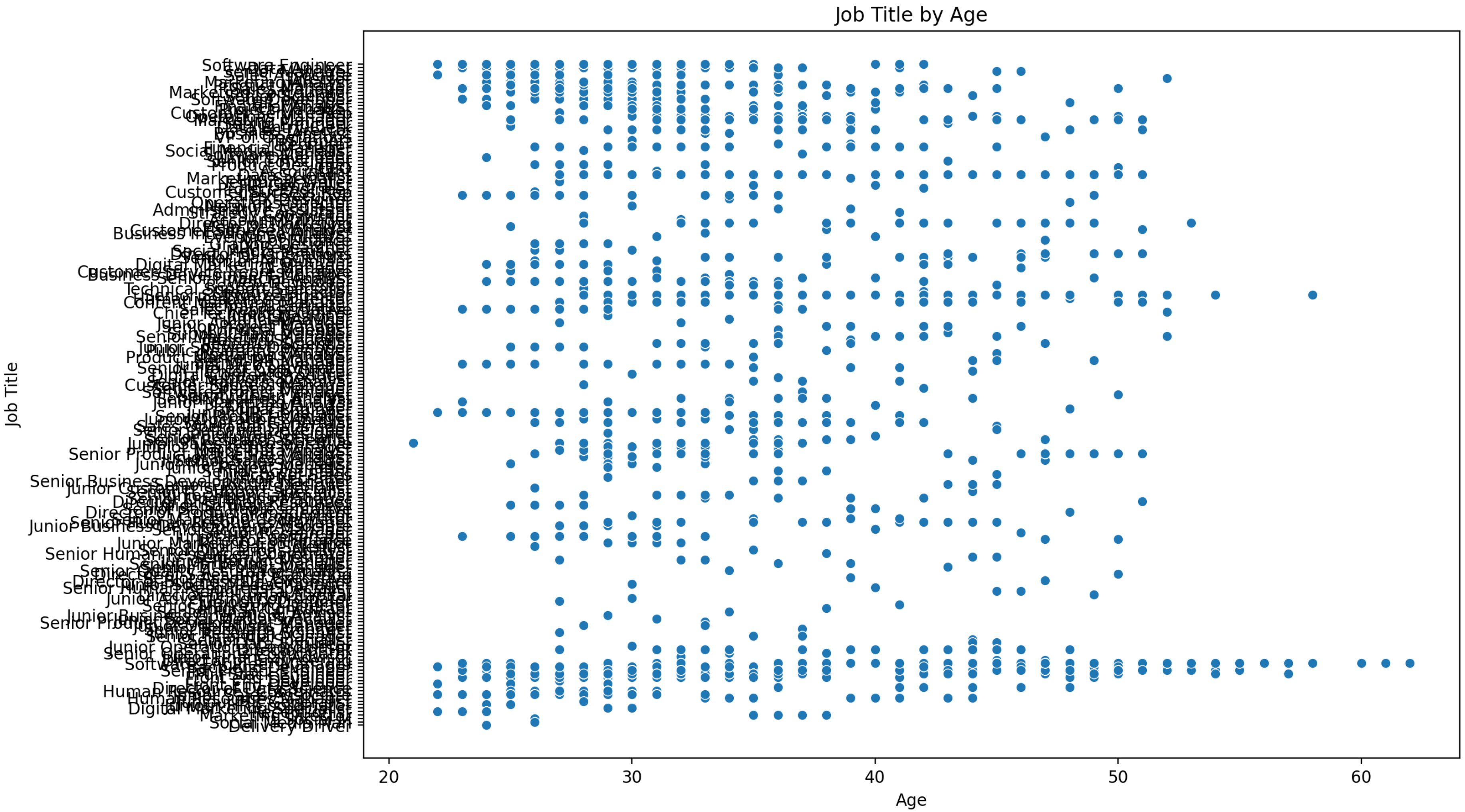
In [38]:

```
plt.figure(figsize= (12,8), dpi = 200)
sns.histplot(df['Gender'])

plt.xlabel('Gender')
plt.ylabel('Race')
plt.title('Race by Gender')

plt.show()
```



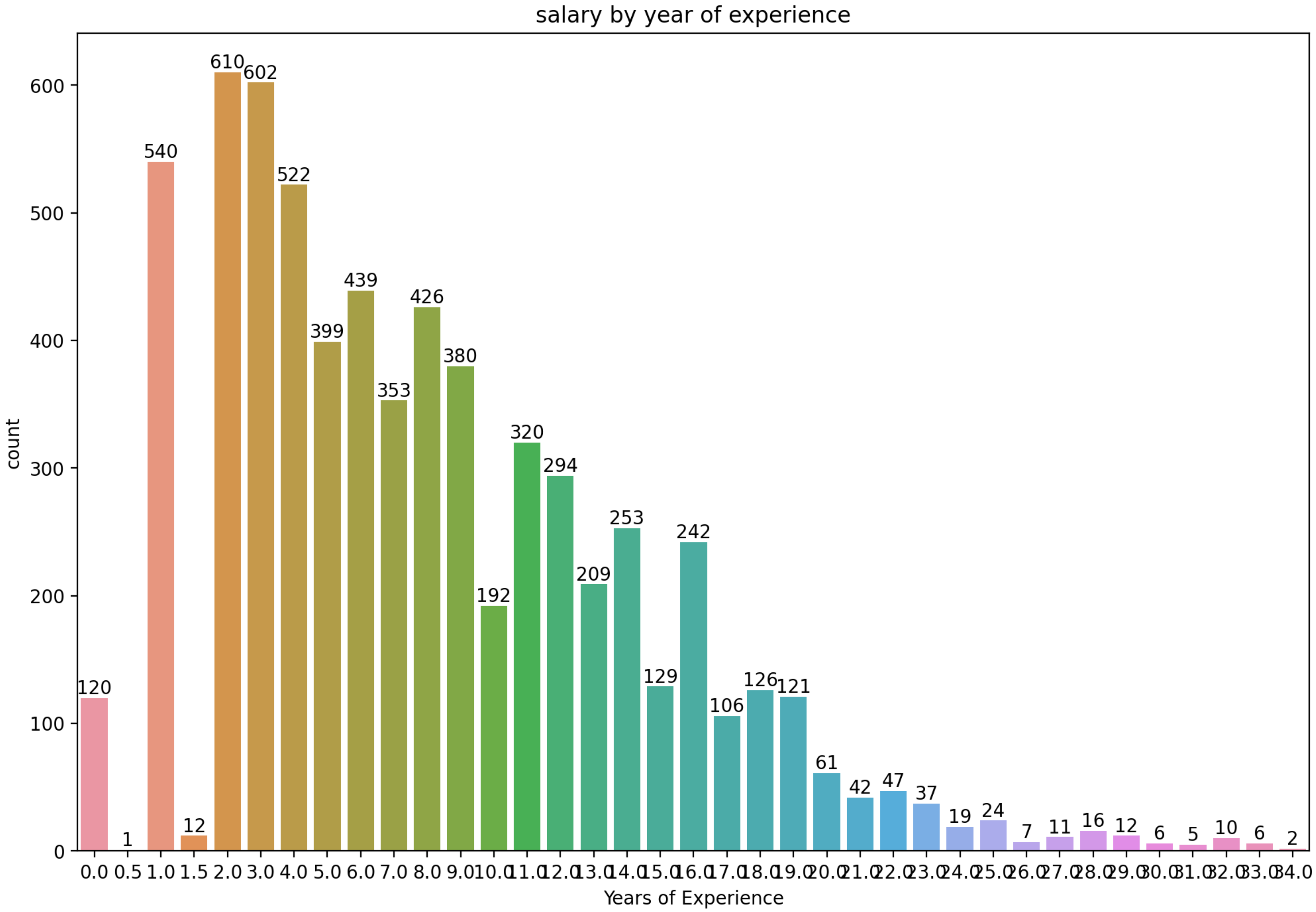
In [40]:
```
plt.figure(figsize= (12,8), dpi = 200)

ax = sns.scatterplot(x= 'Age', y = 'Job Title' , data = df)
```

```
for scatters in ax.containers:
    ax.scatter_label(scatters)

plt.xlabel('Age')
plt.ylabel('Job Title')
plt.title('Job Title by Age')

plt.show()
```



In [41]:
```
plt.figure(figsize = (12,8), dpi = 200)
```

```
ax = sns.countplot(data = df, x = 'Years of Experience')
ax.bar_label(ax.containers[0])
plt.title('salary by year of experience')

plt.show()
```


salary by year of experience
count
0
100
200
300
400
500
600
120
1
540
12
610
602
522
399
439
353
426
380
192
320
294
209
253
129
242
106
126
121
61
42
47
37
19
24
7
11
16
12
6
5
10
6
2
0.0
0.5
1.0
1.5
2.0
3.0
4.0
5.0
6.0
7.0
8.0
9.0
10.0
11.0
12.0
13.0
14.0
15.0
16.0
17.0
18.0
19.0
20.0
21.0
22.0
23.0
24.0
25.0
26.0
27.0
28.0
29.0
30.0
31.0
32.0
33.0
34.0
Years of Experience

In [ ]: