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

In [2]:
```
salaries_df=pd.read_csv("salary_data_states_corrected.csv")
```

In [4]:
```
"""
Looking by each Case_Status to compare the distribution ratio of each Job_Title_Subgroup
to see if there's any commonalities/Preference for admission
"""
```

Out[4]:
```
"\nLooking by each Case_Status to compare the distribution ratio of each Job_Title_Subgroup \nto see if there's any commonalities/Preference for admission\n"
```

In [5]:
```
salaries_df["CASE_STATUS"].value_counts()
```

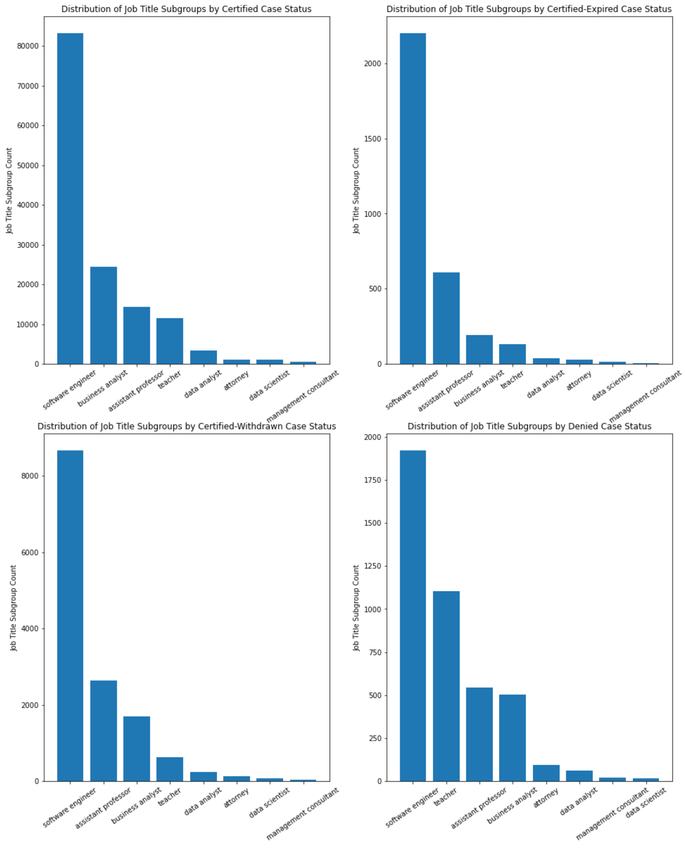
Out[5]:
```
certified              140031
certified-withdrawn     14146
withdrawn                5602
denied                   4273
certified-expired        3226
Name: CASE_STATUS, dtype: int64
```

In [6]:
```
salaries_df["JOB_TITLE_SUBGROUP"].value_counts()
```

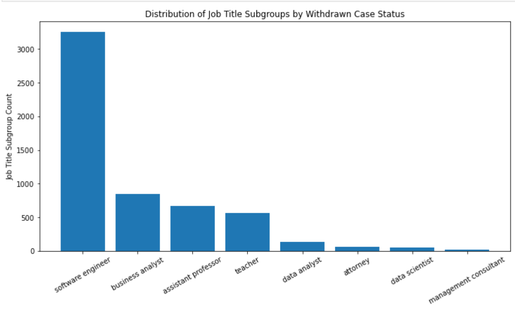
Out[6]:
```
software engineer        99364
business analyst         27811
assistant professor      18866
teacher                  13912
data analyst              3840
attorney                  1488
data scientist            1227
management consultant      770
Name: JOB_TITLE_SUBGROUP, dtype: int64
```

In [7]:
```
cert_df=salaries_df.groupby("CASE_STATUS").get_group('certified')
cert_exp_df=salaries_df.groupby("CASE_STATUS").get_group('certified-expired')
cert_with_df=salaries_df.groupby("CASE_STATUS").get_group('certified-withdrawn')
denied_df=salaries_df.groupby("CASE_STATUS").get_group('denied')
withdrawn_df=salaries_df.groupby("CASE_STATUS").get_group('withdrawn')
```

In [43]:
```
fig, (ax) = plt.subplots(nrows=2,ncols=2,figsize=(16, 20)) #canvas for both bar graphs

 #variables for the the datasets
cert_count=cert_df["JOB_TITLE_SUBGROUP"].value_counts()
cert_exp_count=cert_exp_df["JOB_TITLE_SUBGROUP"].value_counts()
cert_with_count=cert_with_df["JOB_TITLE_SUBGROUP"].value_counts()
denied_count=denied_df["JOB_TITLE_SUBGROUP"].value_counts()


ax[0,0].bar(cert_count.index, cert_count.values) #Plotting the data with the variables
ax[0,1].bar(cert_exp_count.index, cert_exp_count.values)
ax[1,0].bar(cert_with_count.index, cert_with_count.values)
ax[1,1].bar(denied_count.index, denied_count.values)

ax[0,0].tick_params(axis="x", labelrotation=35) #Bargraph customization and Titling
ax[0,1].tick_params(axis="x", labelrotation=35)
ax[1,0].tick_params(axis="x", labelrotation=35)
ax[1,1].tick_params(axis="x", labelrotation=35)

ax[0,0].set_ylabel("Job Title Subgroup Count")
ax[0,1].set_ylabel("Job Title Subgroup Count")
ax[1,0].set_ylabel("Job Title Subgroup Count")
ax[1,1].set_ylabel("Job Title Subgroup Count")

ax[0,0].set_title("Distribution of Job Title Subgroups by Certified Case Status")
ax[0,1].set_title("Distribution of Job Title Subgroups by Certified-Expired Case Status");
ax[1,0].set_title("Distribution of Job Title Subgroups by Certified-Withdrawn Case Status");
ax[1,1].set_title("Distribution of Job Title Subgroups by Denied Case Status");
```



In [42]:
```
fig, (bx) = plt.subplots(ncols=1,figsize=(12, 6))

withdrawn_count=withdrawn_df["JOB_TITLE_SUBGROUP"].value_counts()

bx.bar(withdrawn_count.index, withdrawn_count.values)

bx.tick_params(axis="x", labelrotation=30)

bx.set_ylabel("Job Title Subgroup Count")

bx.set_title("Distribution of Job Title Subgroups by Withdrawn Case Status");
```



In [44]:
```
"""
Software engineer's have the highest application and case status which makes sense for the technical work.
For the certified it seem's business analyst rank higher then Assitant professor & teachers but rank lower in the
certified withdrawn/expired and especially in denied showing a preference for them.
"""
```

Out[44]:
```
"\nSoftware engineer's have the highest application and case status which makes sense for the technical work.\nFor the certified it seem's business analyst rank higher then Assitant professor & teachers but rank lower in the \n\n"
```

In [ ]:
```
```