

```
[1]: #project
```

```
[2]: #import necessary modules
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
[3]: df=pd.read_csv('myexcel.csv')
df
```

[3]:									
	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0	PG	25	06-Feb	180	Texas	7730337.0
1	Jae Crowder	Boston Celtics	99	SF	25	06-Jun	235	Marquette	6796117.0
2	John Holland	Boston Celtics	30	SG	27	06-May	205	Boston University	NaN
3	R.J. Hunter	Boston Celtics	28	SG	22	06-May	185	Georgia State	1148640.0
4	Jonas Jerebko	Boston Celtics	8	PF	29	06-Oct	231	NaN	5000000.0
...
453	Shelvin Mack	Utah Jazz	8	PG	26	06-Mar	203	Butler	2433333.0
454	Raul Neto	Utah Jazz	25	PG	24	06-Jan	179	NaN	900000.0

455	Tibor Pleiss	Utah Jazz	21	C	26	07-Mar	256	NaN	2900000.0
456	Jeff Withey	Utah Jazz	24	C	26	7-0	231	Kansas	947276.0
457	Priyanka	Utah Jazz	34	C	25	07-Mar	231	Kansas	947276.0

458 rows × 9 columns

```
[4]: df.head()
```

[4]:									
	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0	PG	25	06-Feb	180	Texas	7730337.0
1	Jae Crowder	Boston Celtics	99	SF	25	06-Jun	235	Marquette	6796117.0
2	John Holland	Boston Celtics	30	SG	27	06-May	205	Boston University	NaN
3	R.J. Hunter	Boston Celtics	28	SG	22	06-May	185	Georgia State	1148640.0
4	Jonas Jerebko	Boston Celtics	8	PF	29	06-Oct	231	NaN	5000000.0

```
[5]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 458 entries, 0 to 457
Data columns (total 9 columns):
#   Column      Non-Null Count  Dtype
---  -
0   Name        458 non-null   object
```



```
1 Team      458 non-null object
2 Number    458 non-null int64
3 Position  458 non-null object
4 Age       458 non-null int64
5 Height    458 non-null object
6 Weight    458 non-null int64
7 College   374 non-null object
8 Salary    447 non-null float64
dtypes: float64(1), int64(3), object(5)
memory usage: 32.3+ KB
```

[]:

```
[6]: df['Height'] = np.random.randint(150, 181, size=len(df)) #Replace height with random values between 150 and 180.
df.head()
```

[6]:		Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0	PG	25	178	180		Texas	7730337.0
1	Jae Crowder	Boston Celtics	99	SF	25	169	235		Marquette	6796117.0
2	John Holland	Boston Celtics	30	SG	27	159	205	Boston University		NaN
3	R.J. Hunter	Boston Celtics	28	SG	22	153	185		Georgia State	1148640.0
4	Jonas Jerebko	Boston Celtics	8	PF	29	151	231		NaN	5000000.0

```
[7]: sns.set(style="whitegrid")
```

```
[8]: df = df.fillna({'Salary': df.Salary.median()}) #filling null salary values with median
df.head()
```

[8]:									
	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0	PG	25	178	180	Texas	7730337.0
1	Jae Crowder	Boston Celtics	99	SF	25	169	235	Marquette	6796117.0
2	John Holland	Boston Celtics	30	SG	27	159	205	Boston University	2836186.0
3	R.J. Hunter	Boston Celtics	28	SG	22	153	185	Georgia State	1148640.0
4	Jonas Jerebko	Boston Celtics	8	PF	29	151	231	NaN	5000000.0

```
[9]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 458 entries, 0 to 457
Data columns (total 9 columns):
#   Column          Non-Null Count  Dtype
---  -
0   Name            458 non-null   object
1   Team            458 non-null   object
2   Number          458 non-null   int64
3   Position        458 non-null   object
4   Age             458 non-null   int64
5   Height          458 non-null   int32
6   Weight          458 non-null   int64
7   College         374 non-null   object
8   Salary          458 non-null   float64
```



```
8   Salary      458 non-null    float64
dtypes: float64(1), int32(1), int64(3), object(4)
memory usage: 30.5+ KB
```

```
[10]: df_clean=df.fillna({'College':'Unknown'})
df #filling null college values with "Unknown".
```

[10]:									
	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0	PG	25	178	180	Texas	7730337.0
1	Jae Crowder	Boston Celtics	99	SF	25	169	235	Marquette	6796117.0
2	John Holland	Boston Celtics	30	SG	27	159	205	Boston University	2836186.0
3	R.J. Hunter	Boston Celtics	28	SG	22	153	185	Georgia State	1148640.0
4	Jonas Jerebko	Boston Celtics	8	PF	29	151	231	NaN	5000000.0
...
453	Shelvin Mack	Utah Jazz	8	PG	26	175	203	Butler	2433333.0
454	Raul Neto	Utah Jazz	25	PG	24	153	179	NaN	900000.0
455	Tibor Pleiss	Utah Jazz	21	C	26	165	256	NaN	2900000.0
456	Jeff Withey	Utah Jazz	24	C	26	173	231	Kansas	947276.0
457	Priyanka	Utah Jazz	34	C	25	156	231	Kansas	947276.0

458 rows × 9 columns

```
[11]:
```

```
df_clean.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 458 entries, 0 to 457
Data columns (total 9 columns):
#   Column      Non-Null Count  Dtype
---  -
0   Name        458 non-null   object
1   Team        458 non-null   object
2   Number      458 non-null   int64
3   Position    458 non-null   object
4   Age         458 non-null   int64
5   Height      458 non-null   int32
6   Weight      458 non-null   int64
7   College     458 non-null   object
8   Salary      458 non-null   float64
dtypes: float64(1), int32(1), int64(3), object(4)
memory usage: 30.5+ KB
```

[12]:

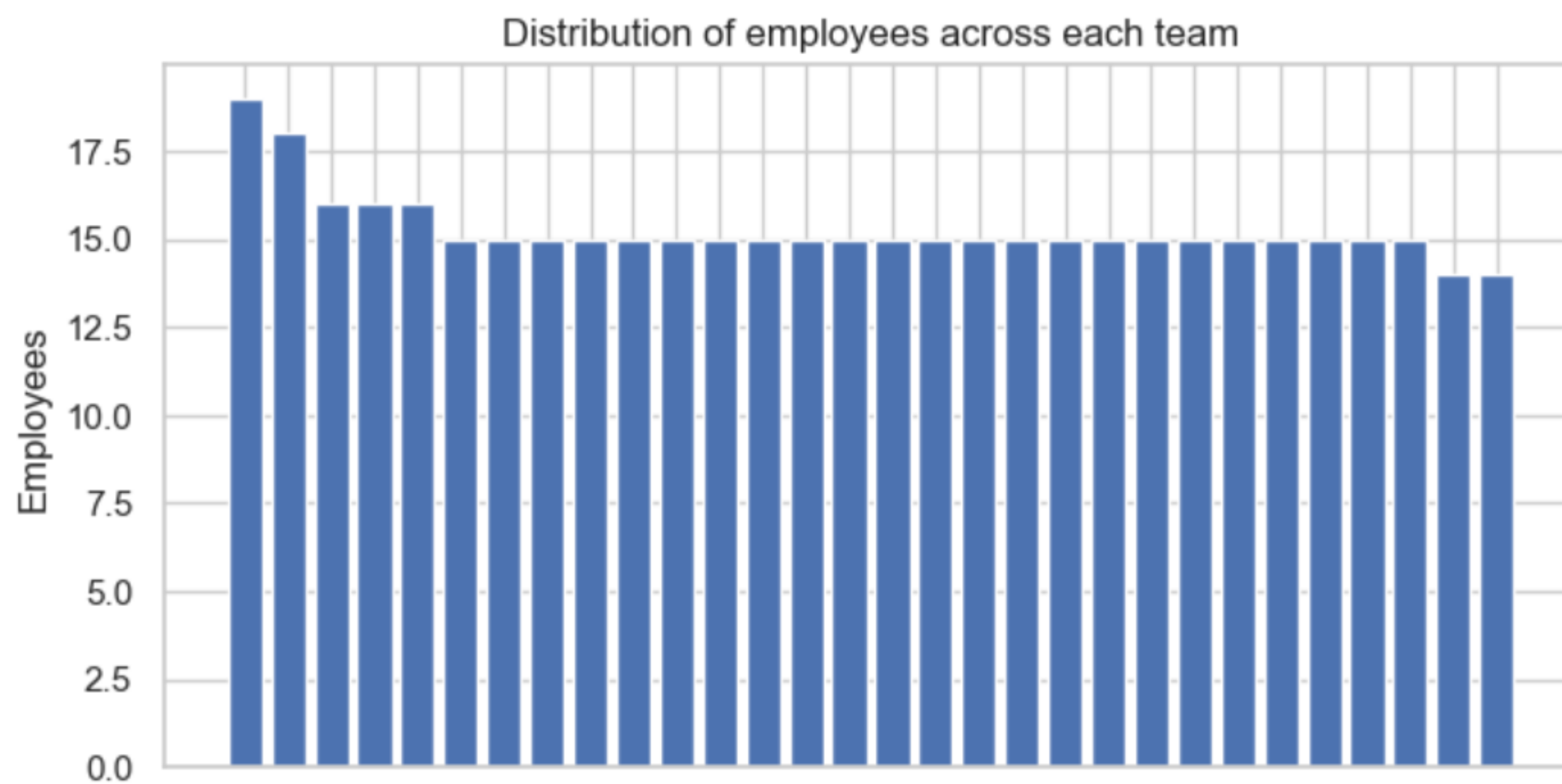
```
Team_Count=df_clean['Team'].value_counts()
print(Team_Count)
```

Team	
New Orleans Pelicans	19
Memphis Grizzlies	18
Utah Jazz	16
New York Knicks	16
Milwaukee Bucks	16
Brooklyn Nets	15

New Orleans Pelicans	19
Memphis Grizzlies	18
Utah Jazz	16
New York Knicks	16
Milwaukee Bucks	16
Brooklyn Nets	15
Portland Trail Blazers	15
Oklahoma City Thunder	15
Denver Nuggets	15
Washington Wizards	15
Miami Heat	15
Charlotte Hornets	15
Atlanta Hawks	15
San Antonio Spurs	15
Houston Rockets	15
Boston Celtics	15
Indiana Pacers	15
Detroit Pistons	15
Cleveland Cavaliers	15
Chicago Bulls	15
Sacramento Kings	15
Phoenix Suns	15
Los Angeles Lakers	15
Los Angeles Clippers	15
Golden State Warriors	15
Toronto Raptors	15
Philadelphia 76ers	15
Dallas Mavericks	15
Orlando Magic	14
Minnesota Timberwolves	14

New Orleans Pelicans	19
Memphis Grizzlies	18
Utah Jazz	16
New York Knicks	16
Milwaukee Bucks	16
Brooklyn Nets	15
Portland Trail Blazers	15
Oklahoma City Thunder	15
Denver Nuggets	15
Washington Wizards	15
Miami Heat	15
Charlotte Hornets	15
Atlanta Hawks	15
San Antonio Spurs	15
Houston Rockets	15
Boston Celtics	15
Indiana Pacers	15
Detroit Pistons	15
Cleveland Cavaliers	15
Chicago Bulls	15
Sacramento Kings	15
Phoenix Suns	15
Los Angeles Lakers	15
Los Angeles Clippers	15
Golden State Warriors	15
Toronto Raptors	15
Philadelphia 76ers	15
Dallas Mavericks	15
Orlando Magic	14
Minnesota Timberwolves	14


```
[13]: plt.figure(figsize=(8,4))
plt.bar(Team_Count.index, Team_Count.values)
plt.title('Distribution of employees across each team')
plt.xlabel('Team')
plt.ylabel('Employees')
plt.xticks(rotation=90)
plt.show()
```



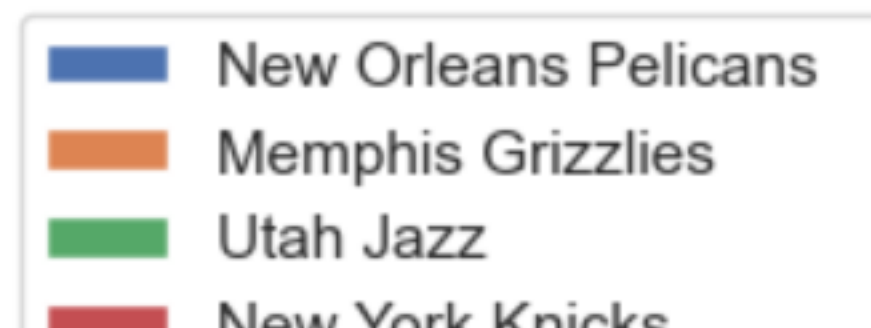


	Count	Percentage
Team		
New Orleans Pelicans	19	4.148472
Memphis Grizzlies	18	3.930131
Utah Jazz	16	3.493450

Indiana Pacers	15	3.275109
Detroit Pistons	15	3.275109
Cleveland Cavaliers	15	3.275109
Chicago Bulls	15	3.275109
Sacramento Kings	15	3.275109
Phoenix Suns	15	3.275109
Los Angeles Lakers	15	3.275109
Los Angeles Clippers	15	3.275109
Golden State Warriors	15	3.275109
Toronto Raptors	15	3.275109
Philadelphia 76ers	15	3.275109
Dallas Mavericks	15	3.275109
Orlando Magic	14	3.056769
Minnesota Timberwolves	14	3.056769

```
[16]: plt.figure(figsize=(7, 7))
plt.pie(perc_split_df['Percentage'], autopct='%1.1f%%')
plt.title('Percentage of Employees by Team')
plt.axis('equal') # Equal aspect ratio ensures the pie chart is circular
plt.legend(perc_split_df.index, loc='best', bbox_to_anchor=(1, 1))
plt.tight_layout()
plt.show()
```

Percentage of Employees by Team



ABC-Company-Data-Analysis/ABC company Main Project

Home

ABC company Main Project

localhost:8889/notebooks/python/ABC company%20 Main Project .ipynb

90%

Trusted

Jupyter

ABC company Main Project

Last Checkpoint: 12 days ago

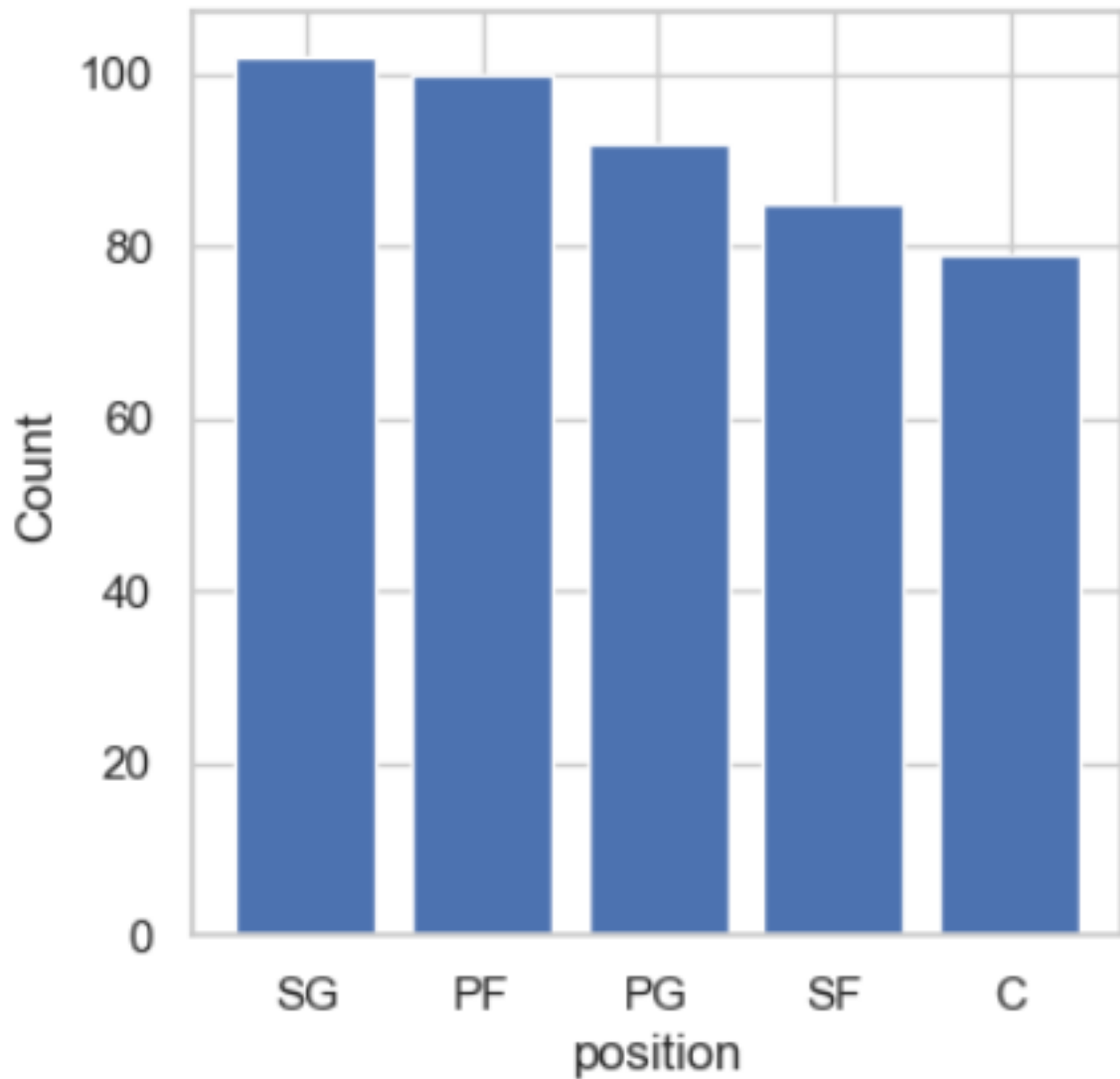
File Edit View Run Kernel Settings Help

Python 3 (ipykernel)

SG 102
PF 100
PG 92
SF 85
C 79
Name: count, dtype: int64

[18]: plt.figure(figsize=(4,4))
plt.bar(Position_Count.index, Position_Count.values)
plt.title('Distribution of Positions')
plt.xlabel('position')
plt.ylabel('Count')
plt.show()

Distribution of Positions



position	Count
SG	102
PF	100
PG	92
SF	85
C	79

[19]: df_clean['Age'].min()

[19]: 19

1

Windows Taskbar

10:58 PM 15-11-2024

ABC-Company-Data-Analysis/ABC company Main Project

Home

ABC company Main Project

localhost:8889/notebooks/python/ABC company%20 Main Project .ipynb

90%

Trusted

JupyterLabPython 3 (ipykernel)

ABC company Main ProjectLast Checkpoint: 12 days ago

FileEditViewRunKernelSettingsHelp

Code

position

0

SGPFPGSFC

[19]: df_clean['Age'].min()

[19]: 19

[20]: df_clean['Age'].max()

[20]: 40

[21]: bins = [18, 26, 31, 36, 40] # Define age intervals to determine the age group
labels = ['18-25', '26-30', '31-35', '36-40']

[22]: df_clean['Age Group'] = pd.cut(df_clean['Age'], bins=bins, labels=labels, right=False)

[23]: df_clean.head()

[23]:

	Name	Team	Number	Position	Age	Height	Weight	College	Salary	Age Group
0	Avery Bradley	Boston Celtics	0	PG	25	178	180	Texas	7730337.0	18-25
1	Jae Crowder	Boston Celtics	99	SF	25	169	235	Marquette	6796117.0	18-25
2	John Holland	Boston Celtics	30	SG	27	159	205	Boston University	2836186.0	26-30
3	R.J. Hunter	Boston Celtics	28	SG	22	153	185	Georgia State	1148640.0	18-25
4	Jonas Jerebko	Boston Celtics	8	PF	29	151	231	Unknown	5000000.0	26-30

[24]: age_group_counts=df_clean['Age Group'].value_counts()

[25]: age_group_counts

[25]: Age Group
18-25200

1

myhp

ENG IN

10:58 PM15-11-2024

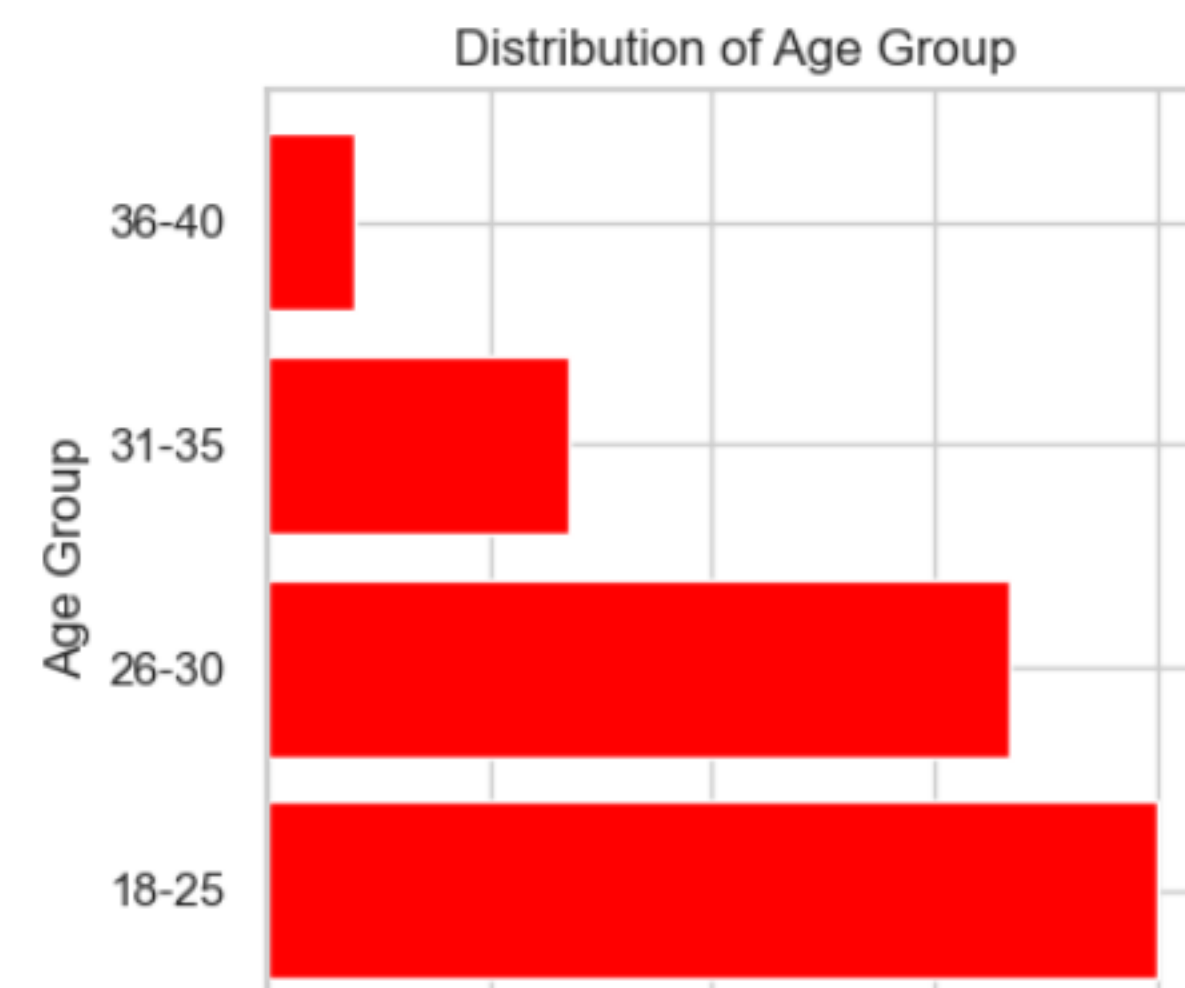

```
[25]: age_group_counts
```

```
[25]: Age Group
      18-25    200
      26-30    167
      31-35     68
      36-40     20
      Name: count, dtype: int64
```

```
[26]: predominant_age_group = age_group_counts.idxmax()
      predominant_count = age_group_counts.max()
      print(f"{predominant_age_group} is the predominant age group with {predominant_count} count")
```

18-25 is the predominant age group with 200 count

```
[27]: plt.figure(figsize=(4,4))
plt.barh(age_group_counts.index, age_group_counts.values, color='red')
plt.title('Distribution of Age Group')
plt.ylabel('Age Group')
plt.xlabel('Count')
plt.show()
```



ABC-Company-Data-Analysis/ABC company Main Project

Home

ABC company Main Project

localhost:8889/notebooks/python/ABC company%20 Main Project .ipynb

90%

Trusted

JupyterLabPython 3 (ipykernel)

FileEditViewRunKernelSettingsHelp

Code

36-40

31-35

26-30

18-25

Age Group

Count

050100150200

[28]: #To determine this we find the sum of salary after grouping Team and Position by Salary Expenditure using groupby function

[29]: grouped_salary=df_clean.groupby(['Team','Position'])['Salary'].sum()
grouped_salary.head(10)

[29]: TeamPosition
Atlanta HawksC22756250.0
PF23952268.0
PG9763400.0
SF6000000.0
SG10431032.0
Boston CelticsC7351395.0
PF18170960.0
PG19898606.0
SF6796117.0
SG9160176.0
Name: Salary, dtype: float64

[30]: highest_salary=grouped_salary.max()
highest salary group=grouped salary.idxmax()

1

Windows Taskbar

10:59 PM 15-11-2024

Count

```
[28]: #To determine this we find the sum of salary after grouping Team and Position by Salary Expenditure using groupby function
```

```
[29]: grouped_salary=df_clean.groupby(['Team','Position'])['Salary'].sum()
      grouped_salary.head(10)
```

```
[29]: Team Position
Atlanta Hawks C 22756250.0
PF 23952268.0
PG 9763400.0
SF 6000000.0
SG 10431032.0
Boston Celtics C 7351395.0
PF 18170960.0
PG 19898606.0
SF 6796117.0
SG 9160176.0
Name: Salary, dtype: float64
```

```
[30]: highest_salary=grouped_salary.max()
highest_salary_group=grouped_salary.idxmax()
print(f"The team and position have the highest salary expenditure is { highest_salary_group} with salary {highest_salary}")

The team and position have the highest salary expenditure is ('Los Angeles Lakers', 'SF') with salary 31866445.0
```

```
[31]: Top_grouped_salary = grouped_salary.sort_values(ascending=False).head(15) #sorting grouped salary in descending order
      Top_grouped_salary
```

[31]:	Team	Position	
	Los Angeles Lakers	SF	31866445.0
	Miami Heat	PF	31538671.0
	Houston Rockets	SG	28122883.0
	Phoenix Suns	PG	28002998.0
	Denver Nuggets	SF	27982771.0
	Cleveland Cavaliers	PF	27882029.0
	New Orleans Pelicans	SG	27489643.0
	Sacramento Kings	C	26950230.0
	Oklahoma City Thunder	SF	25798862.0
	Los Angeles Clippers	PG	25527217.0
	New York Knicks	SF	25387789.0

The team and position have the highest salary expenditure is ('Los Angeles Lakers', 'SF') with salary 31866445.0

```
[31]: Top_grouped_salary = grouped_salary.sort_values(ascending=False).head(15) #sorting grouped salary in descending order
      Top_grouped_salary
```

```
[31]: Team Position
Los Angeles Lakers SF 31866445.0
Miami Heat PF 31538671.0
Houston Rockets SG 28122883.0
Phoenix Suns PG 28002998.0
Denver Nuggets SF 27982771.0
Cleveland Cavaliers PF 27882029.0
New Orleans Pelicans SG 27489643.0
Sacramento Kings C 26950230.0
Oklahoma City Thunder SF 25798862.0
Los Angeles Clippers PG 25527217.0
New York Knicks SF 25387789.0
Milwaukee Bucks SG 25380316.0
Memphis Grizzlies PF 25014640.0
Washington Wizards C 24490429.0
Atlanta Hawks PF 23952268.0
Name: Salary, dtype: float64
```

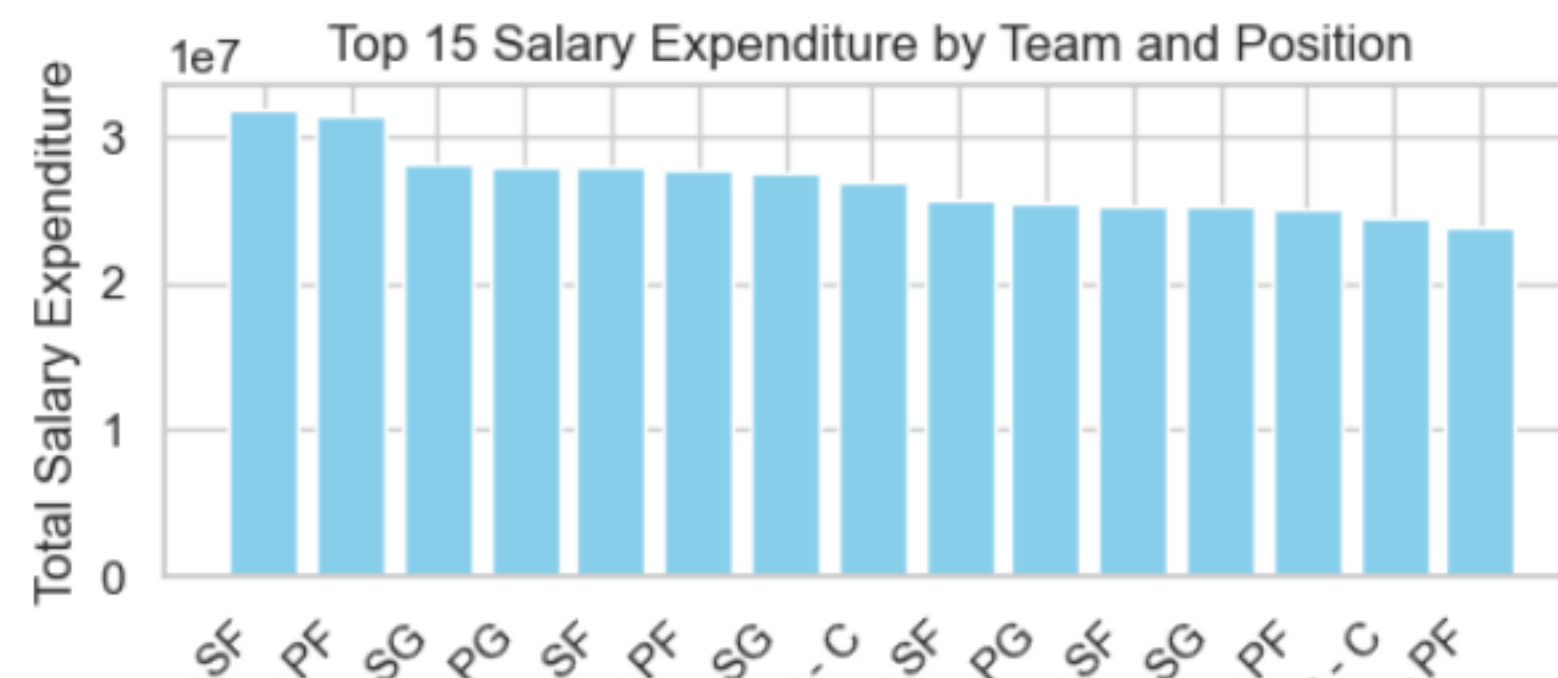
```
[32]: index_labels = [f"{Team} - {Position}" for Team, Position in Top_grouped_salary.index]
      #making a single index from the multi index data
      index_labels
```

```
[32]: ['Los Angeles Lakers - SF',
        'Miami Heat - PF',
        'Houston Rockets - SG',
        'Phoenix Suns - PG',
        'Denver Nuggets - SF',
        'Cleveland Cavaliers - PF',
        'New Orleans Pelicans - SG',
        'Sacramento Kings - C',
        'Oklahoma City Thunder - SF',
        'Los Angeles Clippers - PG',
        'New York Knicks - SF',
        'Milwaukee Bucks - SG',
        'Memphis Grizzlies - PF',
        'Washington Wizards - C',
        'Atlanta Hawks - PF']
```


index_labels

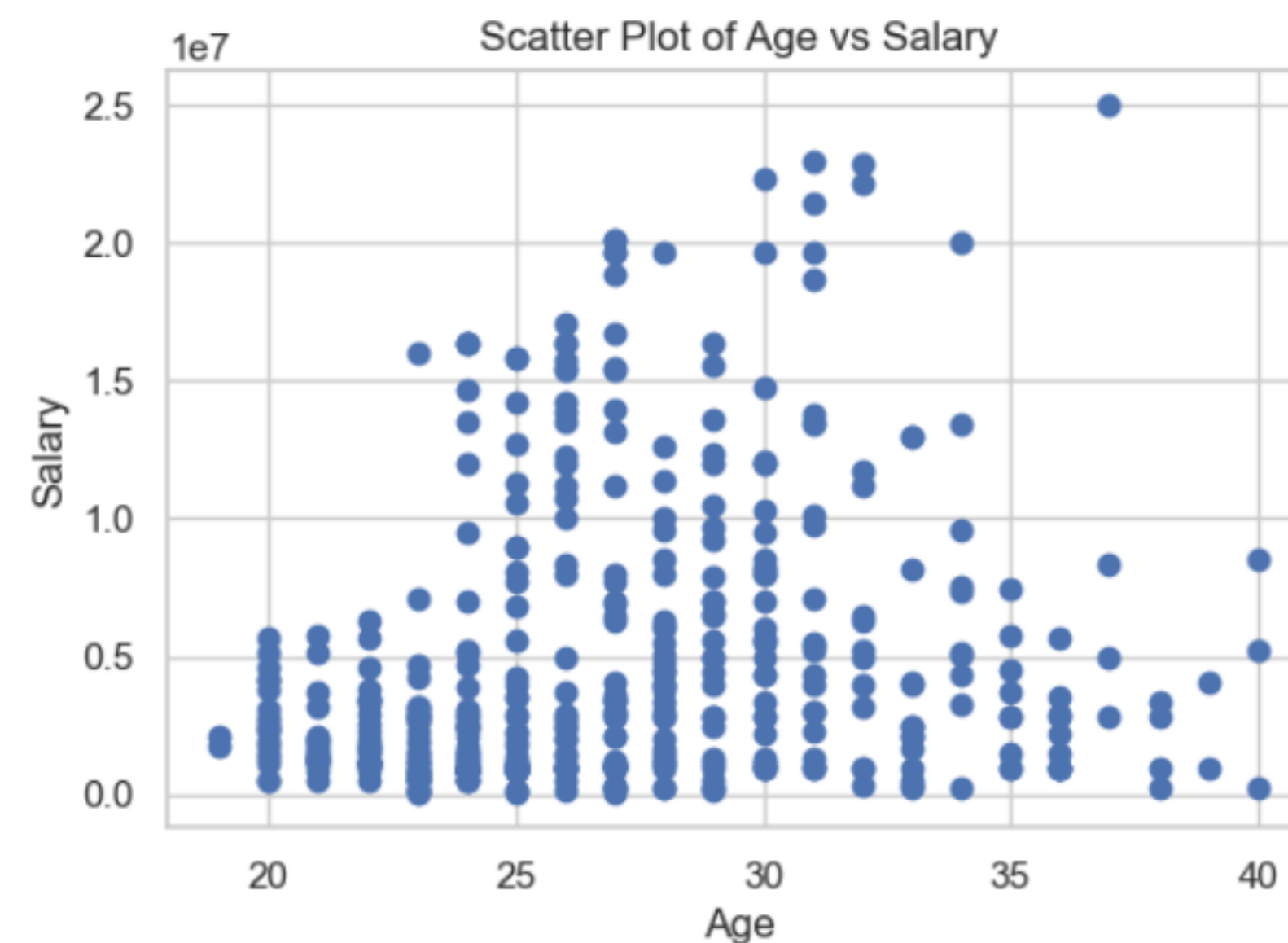
```
[32]: ['Los Angeles Lakers - SF',
        'Miami Heat - PF',
        'Houston Rockets - SG',
        'Phoenix Suns - PG',
        'Denver Nuggets - SF',
        'Cleveland Cavaliers - PF',
        'New Orleans Pelicans - SG',
        'Sacramento Kings - C',
        'Oklahoma City Thunder - SF',
        'Los Angeles Clippers - PG',
        'New York Knicks - SF',
        'Milwaukee Bucks - SG',
        'Memphis Grizzlies - PF',
        'Washington Wizards - C',
        'Atlanta Hawks - PF']
```

```
[33]: plt.figure(figsize=(6, 4))
plt.bar(index_labels, Top_grouped_salary.values, color='skyblue')
plt.title('Top 15 Salary Expenditure by Team and Position')
plt.xlabel('Team and Position')
plt.ylabel('Total Salary Expenditure')
plt.xticks(rotation=45, ha='right') # Rotate labels for readability
plt.tight_layout()
plt.show()
```



```
plt.ylabel('Salary')
```

```
[34]: Text(0, 0.5, 'Salary')
```



```
print(f"Correlation between Age and Salary: {correlation:.2f}")
```

Correlation between Age and Salary: 0.21