PYTHON PROJECT

BY ABDUL KARRAR

Introduction: This project focused on analyzing an NBA dataset by cleaning and preparing the data, performing exploratory data analysis (EDA), and creating visualizations. The goal was to uncover trends, patterns, and relationships, including insights about team distributions, salary expenses, and player attributes like age and position.

In [1]:

```
import warnings
import sys
if not sys.warnoptions:
    warnings.simplefilter("ignore")
```

In [2]:

```
import pandas as pd
```

In [3]:

```
import numpy as np
```

In [6]:

```
data = pd.read_excel("mydata.xlsx")
data
```

Out[6]:

	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0	PG	25	2023-02-06 00:00:00	180	Texas	7730337.0
1	Jae Crowder	Boston Celtics	99	SF	25	2023-06-06 00:00:00	235	Marquette	6796117.0
2	John Holland	Boston Celtics	30	SG	27	2023-05-06 00:00:00	205	Boston University	NaN
3	R.J. Hunter	Boston Celtics	28	SG	22	2023-05-06 00:00:00	185	Georgia State	1148640.0
4	Jonas Jerebko	Boston Celtics	8	PF	29	2023-10-06 00:00:00	231	NaN	5000000.0
...	...	...	...	...	...	...	...	...	...
453	Shelvin Mack	Utah Jazz	8	PG	26	2023-03-06 00:00:00	203	Butler	2433333.0
454	Raul Neto	Utah Jazz	25	PG	24	2023-01-06 00:00:00	179	NaN	900000.0
455	Tibor Pleiss	Utah Jazz	21	C	26	2023-03-07 00:00:00	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	2023-03-07 00:00:00	231	Kansas	947276.0

458 rows × 9 columns

In [7]:

```
data2 = data.copy()
data2
```

Out[7]:

	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0	PG	25	2023-02-06 00:00:00	180	Texas	7730337.0
1	Jae Crowder	Boston Celtics	99	SF	25	2023-06-06 00:00:00	235	Marquette	6796117.0
2	John Holland	Boston Celtics	30	SG	27	2023-05-06 00:00:00	205	Boston University	NaN
3	R.J. Hunter	Boston Celtics	28	SG	22	2023-05-06 00:00:00	185	Georgia State	1148640.0
4	Jonas Jerebko	Boston Celtics	8	PF	29	2023-10-06 00:00:00	231	NaN	5000000.0
...	...	...	...	...	...	...	...	...	...
453	Shelvin Mack	Utah Jazz	8	PG	26	2023-03-06 00:00:00	203	Butler	2433333.0
454	Raul Neto	Utah Jazz	25	PG	24	2023-01-06 00:00:00	179	NaN	900000.0
455	Tibor Pleiss	Utah Jazz	21	C	26	2023-03-07 00:00:00	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	2023-03-07 00:00:00	231	Kansas	947276.0

458 rows × 9 columns

In [8]:

```
data.isnull().sum()
```

Out[8]:

```
Name         0
Team         0
Number       0
Position     0
Age          0
Height       0
Weight       0
College     84
Salary      11
dtype: int64
```

In [9]:

```
# import numpy as np
data['Height'] = np.random.randint(150,181,size = len(data))
```

```
data.head(10)
```

Out[9]:

	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0	PG	25	172	180	Texas	7730337.0
1	Jae Crowder	Boston Celtics	99	SF	25	165	235	Marquette	6796117.0
2	John Holland	Boston Celtics	30	SG	27	171	205	Boston University	NaN
3	R.J. Hunter	Boston Celtics	28	SG	22	168	185	Georgia State	1148640.0
4	Jonas Jerebko	Boston Celtics	8	PF	29	152	231	NaN	5000000.0
5	Amir Johnson	Boston Celtics	90	PF	29	161	240	NaN	12000000.0
6	Jordan Mickey	Boston Celtics	55	PF	21	173	235	LSU	1170960.0
7	Kelly Olynyk	Boston Celtics	41	C	25	173	238	Gonzaga	2165160.0
8	Terry Rozier	Boston Celtics	12	PG	22	152	190	Louisville	1824360.0
9	Marcus Smart	Boston Celtics	36	PG	22	160	220	Oklahoma State	3431040.0

In [10]:

```
data['Salary'].fillna(data['Salary'].mean(), inplace=True)
data
```

Out[10]:

	Name	Team	Number	Position	Age	Height	Weight	College	Sala
0	Avery Bradley	Boston Celtics	0	PG	25	172	180	Texas	7.730337e+0
1	Jae Crowder	Boston Celtics	99	SF	25	165	235	Marquette	6.796117e+0
2	John Holland	Boston Celtics	30	SG	27	171	205	Boston University	4.833970e+0
3	R.J. Hunter	Boston Celtics	28	SG	22	168	185	Georgia State	1.148640e+0
4	Jonas Jerebko	Boston Celtics	8	PF	29	152	231	NaN	5.000000e+0
...	...	...	...	...	...	...	...	...	
453	Shelvin Mack	Utah Jazz	8	PG	26	178	203	Butler	2.433333e+0
454	Raul Neto	Utah Jazz	25	PG	24	172	179	NaN	9.000000e+0
455	Tibor Pleiss	Utah Jazz	21	C	26	172	256	NaN	2.900000e+0
456	Jeff Withey	Utah Jazz	24	C	26	162	231	Kansas	9.472760e+0
457	Priyanka	Utah Jazz	34	C	25	178	231	Kansas	9.472760e+0

458 rows × 9 columns

In [12]:

```
# Calculate the distribution of players across each team
team_distribution = data['Team'].value_counts()

# Calculate the percentage split relative to the total number of players
team_percentage = (team_distribution/len(data))*100

team_stats = pd.DataFrame({
    'Player Count': team_distribution,
    'Percentage(%)': team_percentage.round(2)
})

team_stats.reset_index(inplace = True)
team_stats.rename(columns={'index':'Team'},inplace=True)

print(team_stats)
```

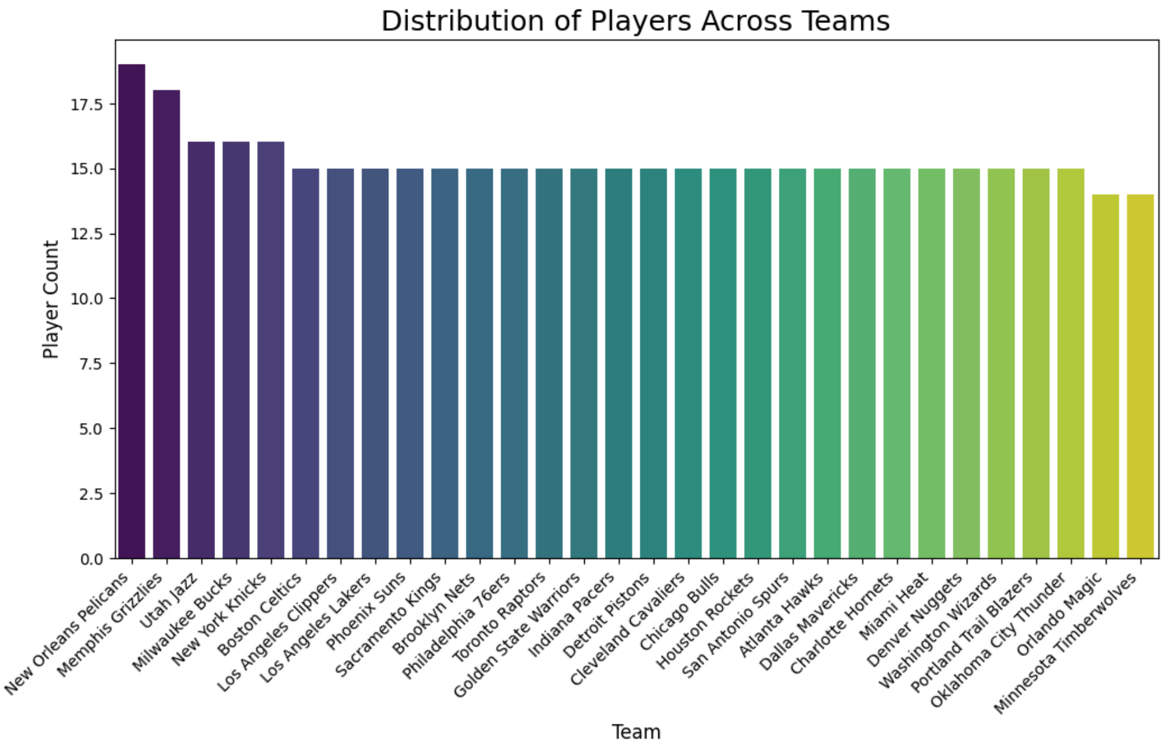
```
                     Team  Player Count  Percentage(%)
0    New Orleans Pelicans            19           4.15
1       Memphis Grizzlies            18           3.93
2               Utah Jazz            16           3.49
3         Milwaukee Bucks            16           3.49
4         New York Knicks            16           3.49
5          Boston Celtics            15           3.28
6    Los Angeles Clippers            15           3.28
7      Los Angeles Lakers            15           3.28
8            Phoenix Suns            15           3.28
9        Sacramento Kings            15           3.28
10          Brooklyn Nets            15           3.28
11     Philadelphia 76ers            15           3.28
12        Toronto Raptors            15           3.28
13  Golden State Warriors            15           3.28
14         Indiana Pacers            15           3.28
15        Detroit Pistons            15           3.28
16    Cleveland Cavaliers            15           3.28
17          Chicago Bulls            15           3.28
18        Houston Rockets            15           3.28
19      San Antonio Spurs            15           3.28
20          Atlanta Hawks            15           3.28
21       Dallas Mavericks            15           3.28
22      Charlotte Hornets            15           3.28
23             Miami Heat            15           3.28
24         Denver Nuggets            15           3.28
25     Washington Wizards            15           3.28
26 Portland Trail Blazers            15           3.28
27  Oklahoma City Thunder            15           3.28
28          Orlando Magic            14           3.06
29 Minnesota Timberwolves            14           3.06
```

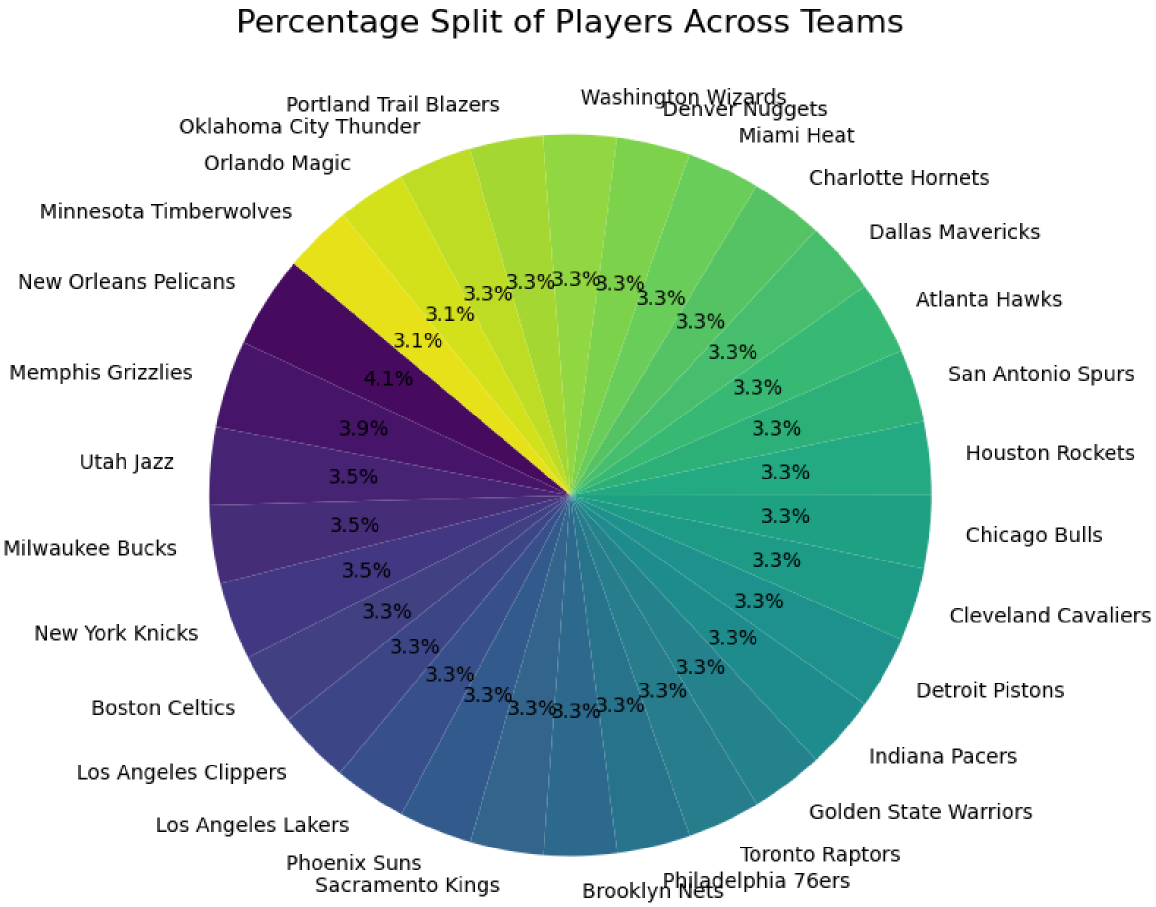
In [13]:

```
import matplotlib.pyplot as plt
import seaborn as sns
```

In [14]:

```
# import matplotlib.pyplot as plt
# import seaborn as sns

# Plotting the number of players across each team as a bar chart
plt.figure(figsize = (12,6))
sns.barplot(x = team_stats['Team'], y = team_stats['Player Count'], palette = 'v
plt.title('Distribution of Players Across Teams', fontsize = 18)
plt.xlabel('Team', fontsize = 12)
plt.ylabel('Player Count', fontsize = 12)
plt.xticks(rotation = 45, ha = 'right')
plt.show()

# Plotting the percentage split as a pie chart
plt.figure(figsize = (8,8))
plt.pie(team_stats['Percentage(%)'], labels = team_stats['Team'], autopct = '%1.
        startangle = 140, colors = sns.color_palette('viridis', len(team_stats))
plt.title('Percentage Split of Players Across Teams', fontsize = 16)
plt.show()
```





In [15]:

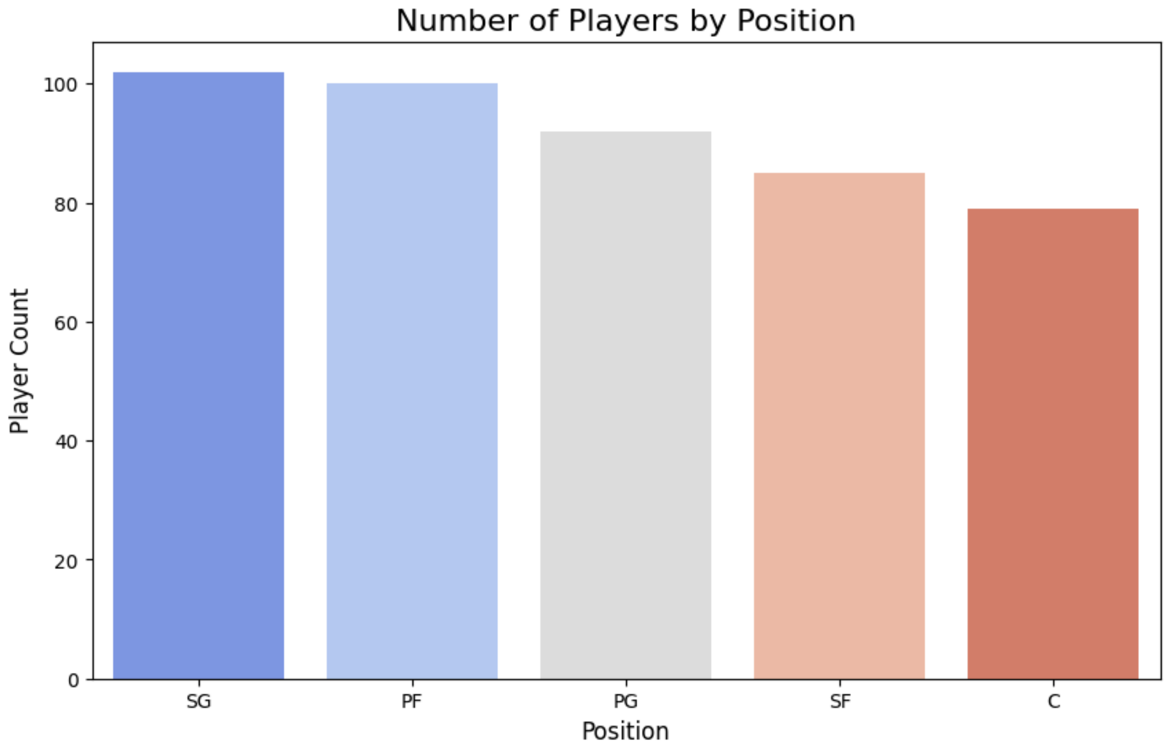
```
# Segregate players based on their positions
position_groups = data.groupby('Position')

# Create a dictionary where each key is a position and the value is the correspo
position_dict = {position: group for position, group in position_groups}

# Display the first few rows for each position as an example
for position, group in position_dict.items():
    print(f"Position: {position}")
    print(group.head(), '\n')
```

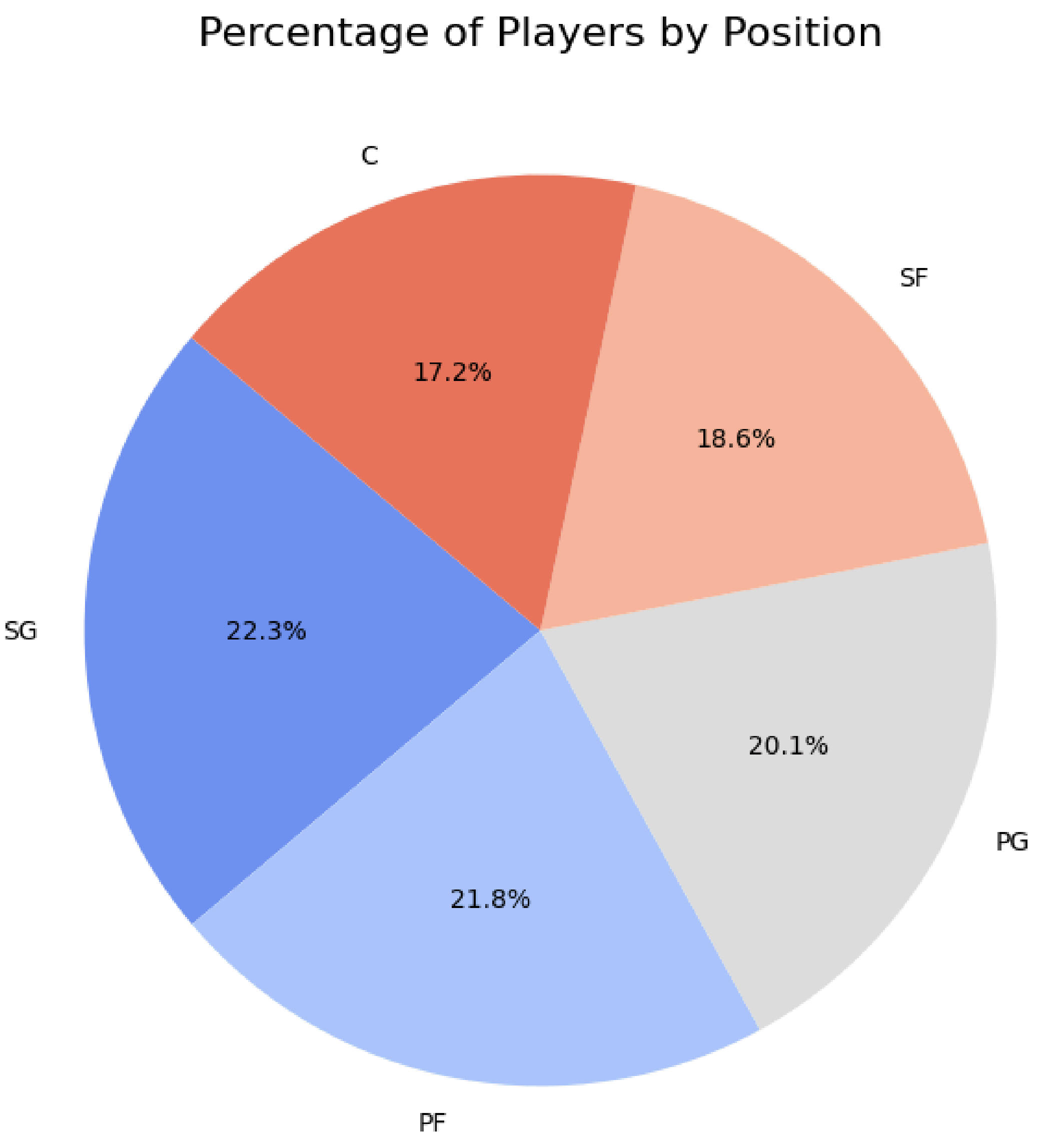
```
Position: C
               Name            Team  Number Position  Age  Height  Weight  \
7      Kelly Olynyk  Boston Celtics      41        C   25     173     238
10  Jared Sullinger  Boston Celtics       7        C   24     176     260
14     Tyler Zeller  Boston Celtics      44        C   26     169     253
23      Brook Lopez   Brooklyn Nets      11        C   28     173     275
27       Henry Sims   Brooklyn Nets      14        C   26     152     248

           College      Salary
7          Gonzaga   2165160.0
10      Ohio State   2569260.0
14  North Carolina   2616975.0
23        Stanford  19689000.0
27      Georgetown    947276.0

Position: PF
                Name            Team  Number Position  Age  Height  Weight  \
4      Jonas Jerebko  Boston Celtics       8       PF   29     152     231
5       Amir Johnson  Boston Celtics      90       PF   29     161     240
6      Jordan Mickey  Boston Celtics      55       PF   21     173     235
24  Chris McCullough   Brooklyn Nets       1       PF   21     164     200
25       Willie Reed   Brooklyn Nets      33       PF   26     177     220

        College      Salary
4           NaN   5000000.0
5           NaN  12000000.0
6           LSU   1170960.0
24     Syracuse   1140240.0
25  Saint Louis    947276.0

Position: PG
             Name            Team  Number Position  Age  Height  Weight  \
0   Avery Bradley  Boston Celtics       0       PG   25     172     180
8    Terry Rozier  Boston Celtics      12       PG   22     152     190
9    Marcus Smart  Boston Celtics      36       PG   22     160     220
11  Isaiah Thomas  Boston Celtics       4       PG   27     163     185
19   Jarrett Jack   Brooklyn Nets       2       PG   32     173     200

           College     Salary
0            Texas  7730337.0
8       Louisville  1824360.0
9   Oklahoma State  3431040.0
11      Washington  6912869.0
19    Georgia Tech  6300000.0

Position: SF
                      Name             Team  Number Position  Age  Height  \
1              Jae Crowder   Boston Celtics      99       SF   25     165
32  Thanasis Antetokounmpo  New York Knicks      43       SF   23     164
33         Carmelo Anthony  New York Knicks       7       SF   32     164
35        Cleanthony Early  New York Knicks      11       SF   25     153
42            Lance Thomas  New York Knicks      42       SF   28     175

    Weight        College      Salary
1      235     Marquette   6796117.0
32     205           NaN     30888.0
33     240      Syracuse  22875000.0
35     210  Wichita State    845059.0
42     235           Duke   1636842.0
```

```
Position: SG
                 Name            Team  Number Position  Age  Height  Weight  \
2        John Holland  Boston Celtics      30       SG   27     171     205
3         R.J. Hunter  Boston Celtics      28       SG   22     168     185
12        Evan Turner  Boston Celtics      11       SG   27     173     220
13        James Young  Boston Celtics      13       SG   20     154     215
15    Bojan Bogdanovic   Brooklyn Nets      44       SG   27     151     216

               College        Salary
2    Boston University  4.833970e+06
3        Georgia State  1.148640e+06
12          Ohio State  3.425510e+06
13            Kentucky  1.749840e+06
15                 NaN  3.425510e+06
```

In [16]:

```
# import matplotlib.pyplot as plt
# import seaborn as sns

# Count the number of players in each position
position_distribution = data['Position'].value_counts()

# Bar Chart: Distribution of players across positions
plt.figure(figsize = (10,6))
sns.barplot(x = position_distribution.index, y = position_distribution.values, p
plt.title('Number of Players by Position', fontsize = 16)
plt.xlabel('Position', fontsize = 12)
plt.ylabel('Player Count', fontsize = 12)
plt.xticks(fontsize = 10)
plt.show()

# Pie Chart: Percentage distribution of players across positions
plt.figure(figsize = (8,8))
plt.pie(position_distribution.values, labels = position_distribution.index, autc
        startangle = 140, colors = sns.color_palette('coolwarm', len(position_di
plt.title('Percentage of Players by Position', fontsize = 16)
plt.show()
```





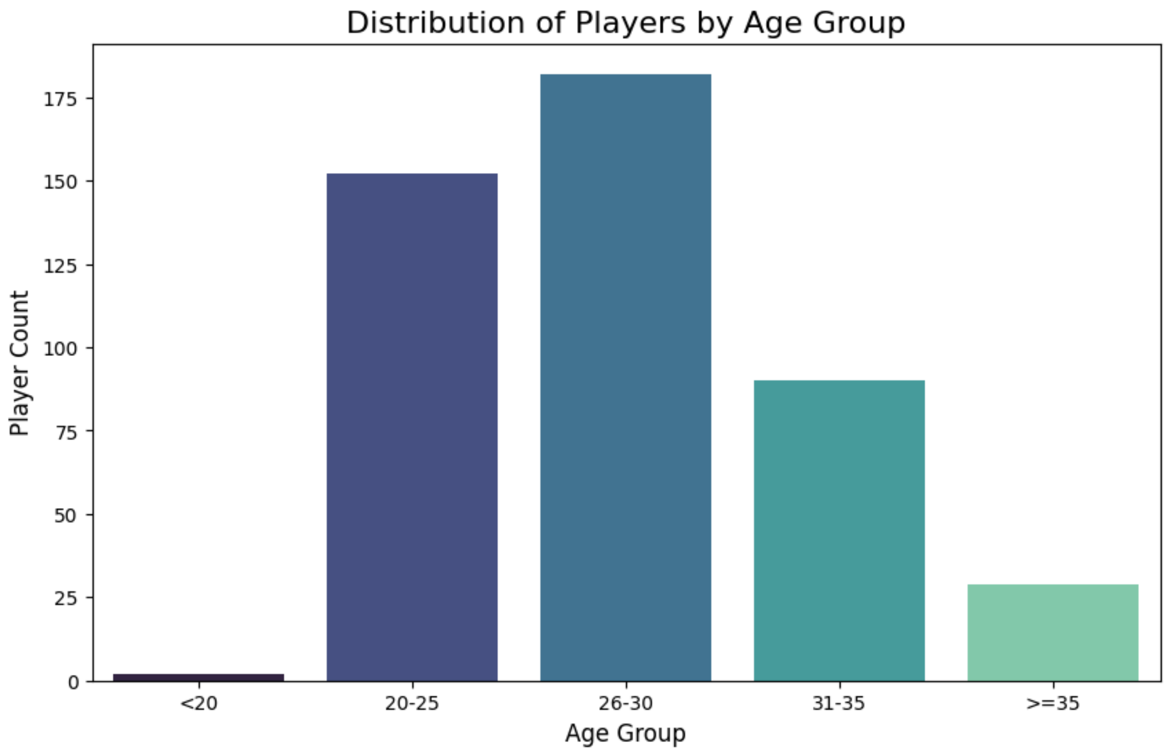
In [17]:

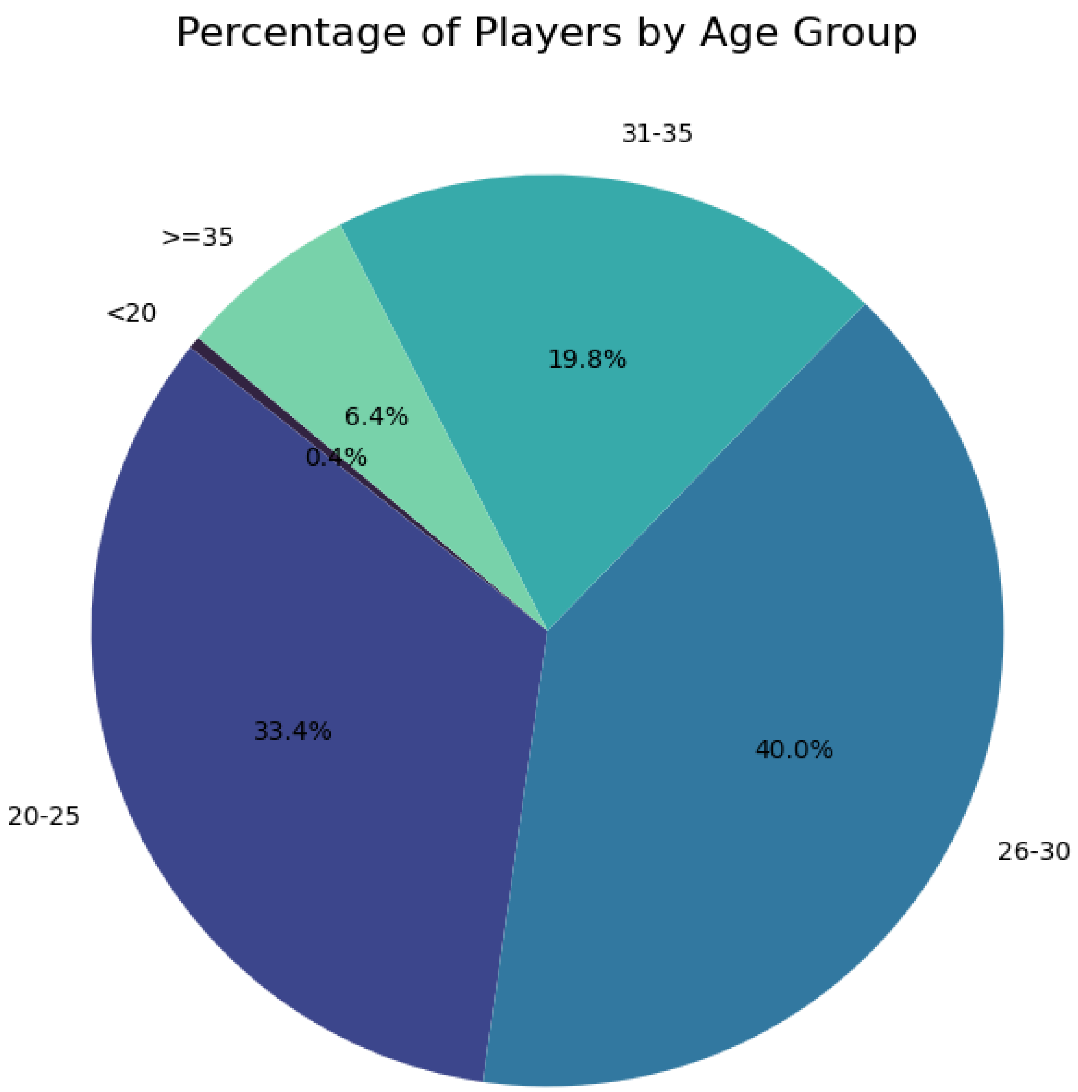
```
# import pandas as pd

# Define age bins and labels
bins = [0,20,25,30,35,40] #Age range
labels = ['<20','20-25','26-30','31-35','>=35']

# Categorize players into age groups
data['Age Group'] = pd.cut(data['Age'], bins = bins, labels = labels, right = Fa

# Calculate distribution of players across age groups
age_group_distribution = data['Age Group'].value_counts().sort_index()
age_group_distribution.name = "Age Distribution"

# Identify predominant age group
predominant_age_group = age_group_distribution.idxmax()

# Display the results
print("Distribution of players by age group:")
print(age_group_distribution)
print("\nPredominant age group:", predominant_age_group)
```

```
Distribution of players by age group:
Age Group
<20        2
20-25    152
26-30    182
31-35     90
>=35      29
Name: Age Distribution, dtype: int64

Predominant age group: 26-30
```

In [18]:

```
# import matplotlib.pyplot as plt
# import seaborn as sns

# Bar Chart: Distribution of players by age group
plt.figure(figsize = (10,6))
sns.barplot(x = age_group_distribution.index, y = age_group_distribution.values,
plt.title("Distribution of Players by Age Group", fontsize = 16)
plt.xlabel('Age Group', fontsize = 12)
plt.ylabel('Player Count', fontsize = 12)
plt.xticks(fontsize = 10)
plt.show()

# Pie Chart: Percentage of players by age group
plt.figure(figsize = (8,8))
plt.pie(age_group_distribution.values, labels = age_group_distribution.index, au
        startangle = 140, colors = sns.color_palette('mako', len(age_group_distr
plt.title("Percentage of Players by Age Group", fontsize = 16)
plt.show()
```





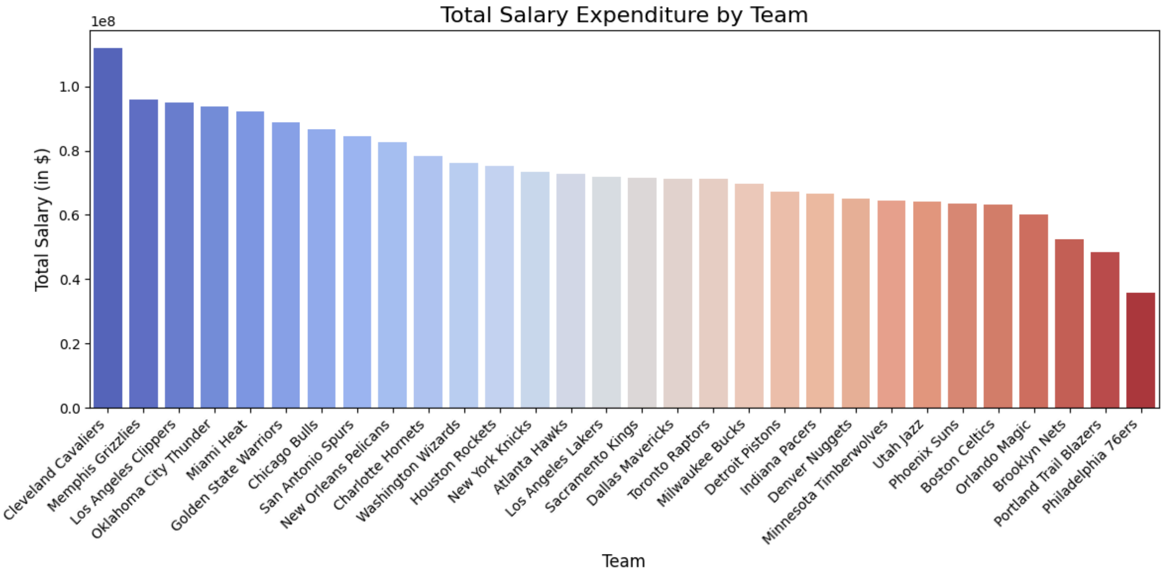
In [19]:

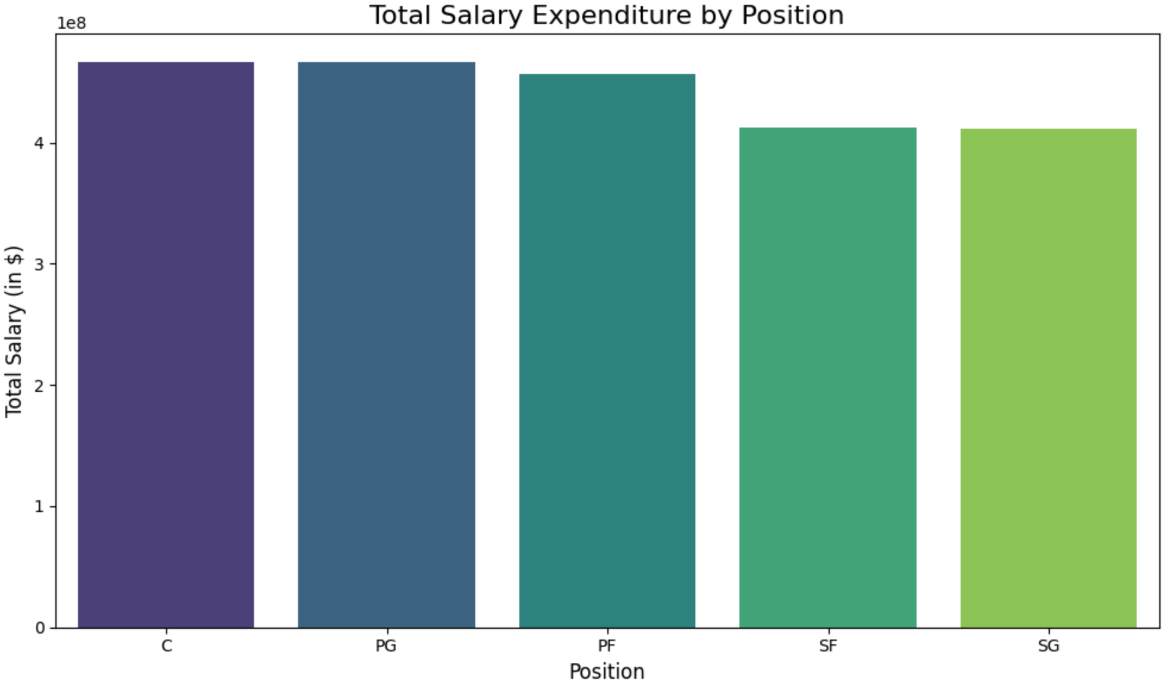
```
# Calculate the total salary expenditure by a team
team_salary_expenditure = data.groupby('Team')['Salary'].sum().sort_values(ascen
team_salary_expenditure.name = "Team Salary"

# Calculate total salary expenditure by position
position_salary_expenditure = data.groupby('Position')['Salary'].sum().sort_valu
position_salary_expenditure.name = "Position Salary"

# Identify the team and position with the highest salary expenditure
highest_team_salary = team_salary_expenditure.idxmax()
highest_position_salary = position_salary_expenditure.idxmax()

# Display the results
print("Total Salary Expenditure by Team:")
print(team_salary_expenditure, "\n")
print(f"Team with the highest salary expenditure: {highest_team_salary} (${team_

print("Total Salary Expenditure by Position:")
print(position_salary_expenditure, "\n")
print(f"Position with the highest salary expenditure: {highest_position_salary}
Total Salary Expenditure by Team:
Team
Cleveland Cavaliers       1.118227e+08
Memphis Grizzlies         9.588676e+07
Los Angeles Clippers      9.485464e+07
Oklahoma City Thunder     9.376530e+07
Miami Heat                9.218361e+07
Golden State Warriors     8.886900e+07
Chicago Bulls             8.678338e+07
San Antonio Spurs         8.444273e+07
New Orleans Pelicans      8.275077e+07
Charlotte Hornets         7.834092e+07
Washington Wizards        7.632864e+07
Houston Rockets           7.528302e+07
New York Knicks           7.330390e+07
Atlanta Hawks             7.290295e+07
Los Angeles Lakers        7.177043e+07
Sacramento Kings          7.168367e+07
Dallas Mavericks          7.119873e+07
Toronto Raptors           7.111761e+07
Milwaukee Bucks           6.960352e+07
Detroit Pistons           6.716826e+07
Indiana Pacers            6.675183e+07
Denver Nuggets            6.495590e+07
Minnesota Timberwolves    6.454367e+07
Utah Jazz                 6.400737e+07
Phoenix Suns              6.344514e+07
Boston Celtics            6.337504e+07
Orlando Magic             6.016147e+07
Brooklyn Nets             5.252848e+07
Portland Trail Blazers    4.830182e+07
Philadelphia 76ers        3.582686e+07
Name: Team Salary, dtype: float64

Team with the highest salary expenditure: Cleveland Cavaliers ($111,822,658.55)

Total Salary Expenditure by Position:
Position
C     4.663773e+08
PG    4.661848e+08
PF    4.570628e+08
SF    4.128549e+08
SG    4.114782e+08
Name: Position Salary, dtype: float64

Position with the highest salary expenditure: C ($466,377,332.00)
```

In [20]:

```python
# import matplotlib.pyplot as plt
# import seaborn as sns

# Bar Chart: Total salary expenditure by team
plt.figure(figsize = (12, 6))
sns.barplot(x = team_salary_expenditure.index, y = team_salary_expenditure.value
plt.title('Total Salary Expenditure by Team', fontsize = 16)
plt.xlabel('Team', fontsize = 12)
plt.ylabel('Total Salary (in $)', fontsize = 12)
plt.xticks(rotation = 45, ha = 'right', fontsize = 10)
plt.tight_layout()
plt.show()
```

```python
# Bar Chart: Total salary expenditure by position
plt.figure(figsize = (10, 6))
sns.barplot(x = position_salary_expenditure.index, y = position_salary_expenditu
plt.title('Total Salary Expenditure by Position', fontsize = 16)
plt.xlabel('Position', fontsize = 12)
plt.ylabel('Total Salary (in $)', fontsize = 12)
plt.xticks(fontsize = 10)
plt.tight_layout()
plt.show()
```





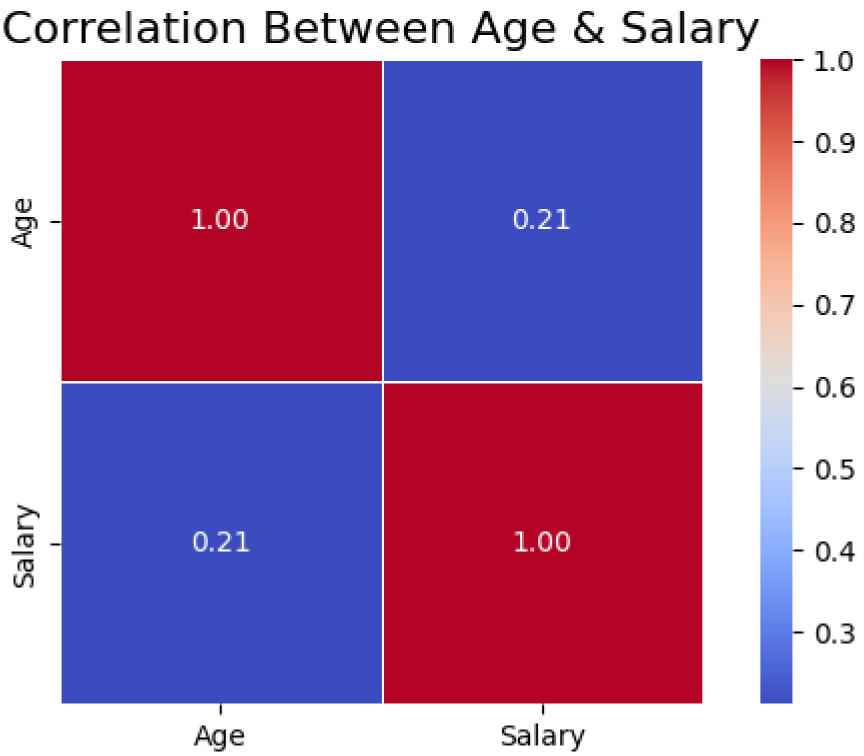
In [21]:

```python
# Calculate the correlation betweenn Age and Salary
correlation = data['Age'].corr(data['Salary'])
print(f"The correlation between Age and Salary is: {correlation:.2f}")

# Determine the type of correlation
if correlation > 0:
    correlation_type = 'Positive Correlation'
elif correlation < 0:
    correlation_type = 'Negative Correlation'
else:
    correlation_type = 'No Correlation'

# Display correlation type
print(f"The correlation between Age and Salary is: {correlation_type}")
```

```
The correlation between Age and Salary is: 0.21
The correlation between Age and Salary is: Positive Correlation
```

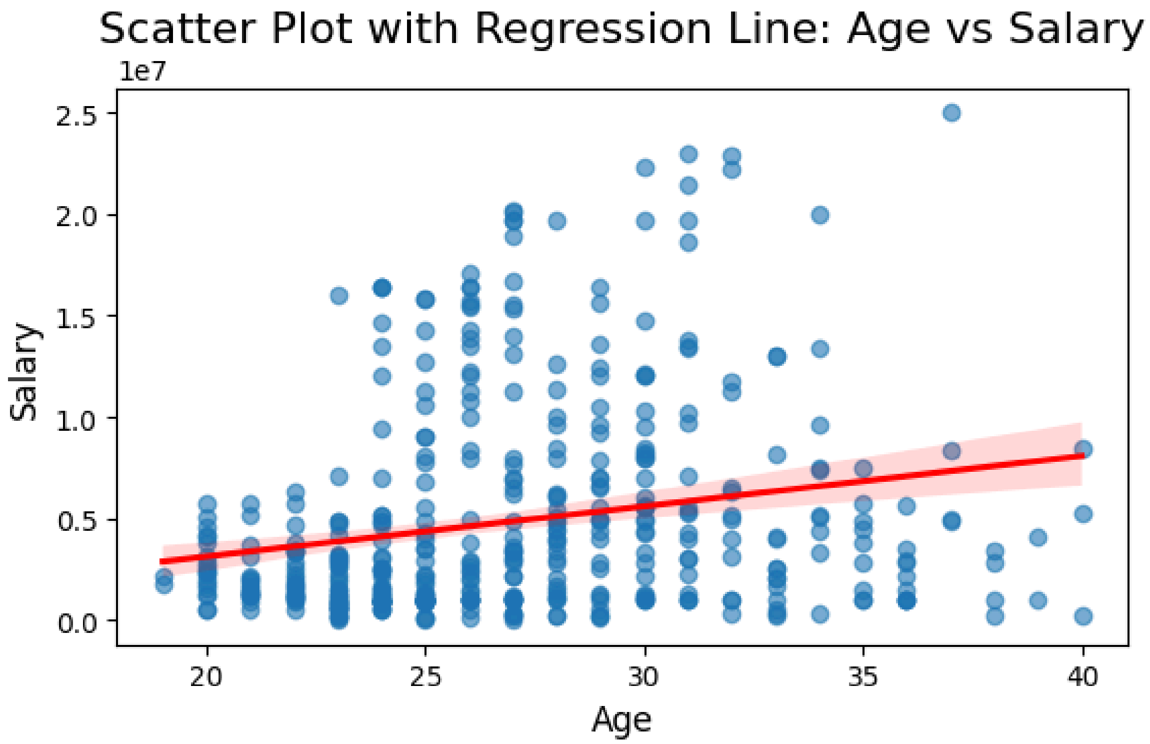
In [22]:

```
# import seaborn as sns
# import matplotlib.pyplot as plt

# Calculate the correlation matrix
correlation_matrix = data[['Age','Salary']].corr()

# Plot the heatmap
plt.figure(figsize = (6,4))
sns.heatmap(correlation_matrix, annot = True, cmap = "coolwarm", fmt = '.2f', li

# Customize the plot
plt.title("Correlation Between Age & Salary", fontsize = 16)
plt.xticks(fontsize = 10)
plt.yticks(fontsize = 10)
plt.tight_layout()
plt.show()
```



In [23]:

```
# Import seaborn and matplotlib.pyplot
import seaborn as sns
import matplotlib.pyplot as plt

# Plot the scatter plot with regression line
plt.figure(figsize=(6, 4))
sns.regplot(x='Age', y='Salary', data=data, scatter_kws={'alpha': 0.6}, line_kws

# Customize the plot
plt.title("Scatter Plot with Regression Line: Age vs Salary", fontsize=16)
plt.xlabel("Age", fontsize=12)
plt.ylabel("Salary", fontsize=12)
plt.xticks(fontsize=10)
plt.yticks(fontsize=10)
plt.tight_layout()
plt.show()
```



In [ ]: