In [5]:
```
import pandas as pd

# Load Excel file into a DataFrame
df = pd.read_csv('D:\ENTRI CORE ASSIGNMENT\python\myexcel - myexcel.csv.csv')

# Display the first few rows of the DataFrame
print(df.head())
```

```
              Name            Team  Number Position  Age  Height  Weight  \
0   Avery Bradley  Boston Celtics       0       PG   25  06-Feb     180
1     Jae Crowder  Boston Celtics      99       SF   25  06-Jun     235
2    John Holland  Boston Celtics      30       SG   27  06-May     205
3     R.J. Hunter  Boston Celtics      28       SG   22  06-May     185
4   Jonas Jerebko  Boston Celtics       8       PF   29  06-Oct     231

              College     Salary
0               Texas  7730337.0
1           Marquette  6796117.0
2  Boston University        NaN
3       Georgia State  1148640.0
4                 NaN  5000000.0
```

In [6]:
```
import pandas as pd
import numpy as np
```

In [7]:
```
# Replace "height" column with random numbers between 150 and 180
df['Height'] = np.random.randint(150, 181, size=len(df))
```

In [8]:
```
# Save the updated DataFrame back to Excel
df.to_csv('updated_data.csv', index=False)  # Replace with your desired file path
```

In [9]:
```
# Print the first few rows to verify
print(df.head())
```

```
              Name            Team  Number Position  Age  Height  Weight  \
0   Avery Bradley  Boston Celtics       0       PG   25     166     180
1     Jae Crowder  Boston Celtics      99       SF   25     172     235
2    John Holland  Boston Celtics      30       SG   27     170     205
3     R.J. Hunter  Boston Celtics      28       SG   22     179     185
4   Jonas Jerebko  Boston Celtics       8       PF   29     163     231

              College     Salary
0               Texas  7730337.0
1           Marquette  6796117.0
2  Boston University        NaN
3       Georgia State  1148640.0
4                 NaN  5000000.0
```

1. Determine the distribution of employees across each team and calculate the percentage split relative to the total number of employees.

In [10]:
```
df1 = pd.DataFrame(df)

# Step 1: Get the count of employees in each team
team_distribution = df['Team'].value_counts().reset_index()
team_distribution.columns = ['Team', 'employee_count']
```

```
# Step 2: Calculate the total number of employees
total_employees = df1.shape[0]
```

In [11]:
```
total_employees
```

Out[11]:
```
458
```

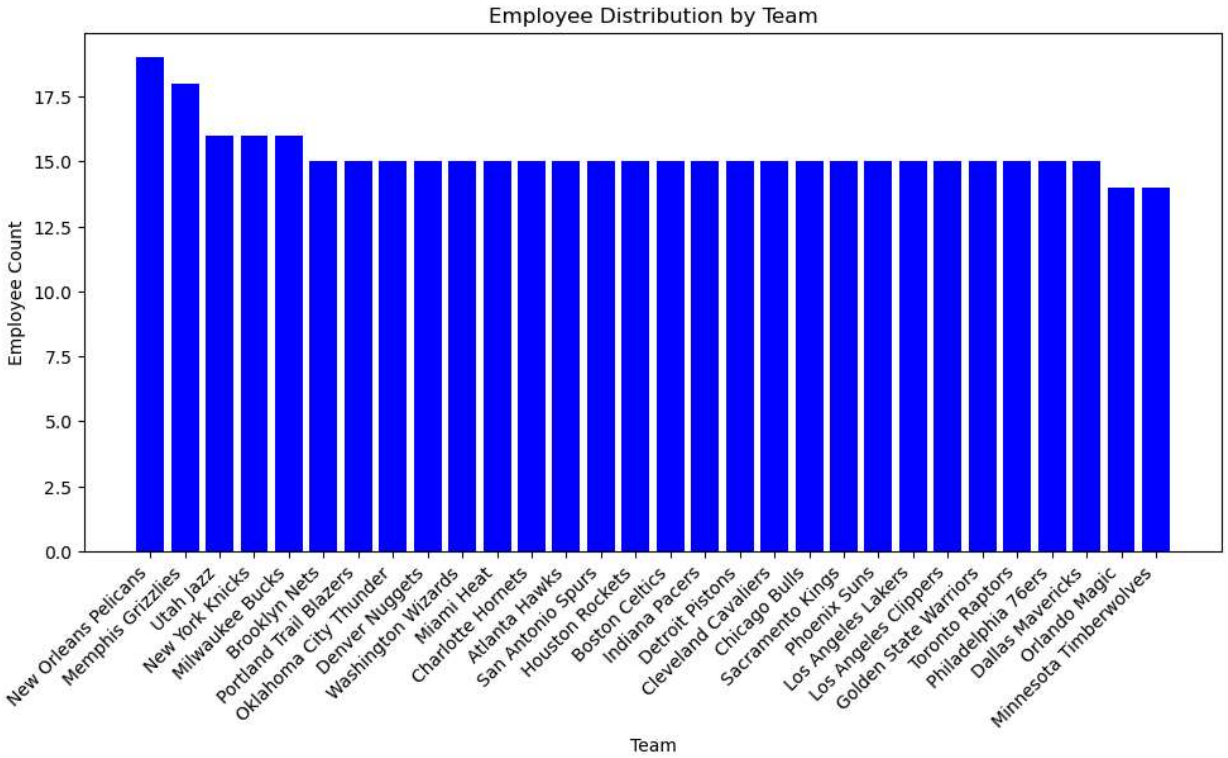
In [12]:
```
df
```

Out[12]:

	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0	PG	25	166	180	Texas	7730337.0
1	Jae Crowder	Boston Celtics	99	SF	25	172	235	Marquette	6796117.0
2	John Holland	Boston Celtics	30	SG	27	170	205	Boston University	NaN
3	R.J. Hunter	Boston Celtics	28	SG	22	179	185	Georgia State	1148640.0
4	Jonas Jerebko	Boston Celtics	8	PF	29	163	231	NaN	5000000.0
...	...	...	...	...	...	...	...	...	...
453	Shelvin Mack	Utah Jazz	8	PG	26	154	203	Butler	2433333.0
454	Raul Neto	Utah Jazz	25	PG	24	151	179	NaN	900000.0
455	Tibor Pleiss	Utah Jazz	21	C	26	171	256	NaN	2900000.0
456	Jeff Withey	Utah Jazz	24	C	26	173	231	Kansas	947276.0
457	Priyanka	Utah Jazz	34	C	25	162	231	Kansas	947276.0

458 rows × 9 columns

In [13]:
```
import matplotlib.pyplot as plt
# Plotting the bar graph
plt.figure(figsize=(10, 6))
plt.bar(team_distribution['Team'], team_distribution['employee_count'], color='blue')
plt.xlabel('Team')
# Adjusting the x-axis labels
plt.xticks(rotation=45, ha='right')

# Adjusting the layout
plt.tight_layout()
plt.ylabel('Employee Count')
plt.title('Employee Distribution by Team')
plt.show()
```



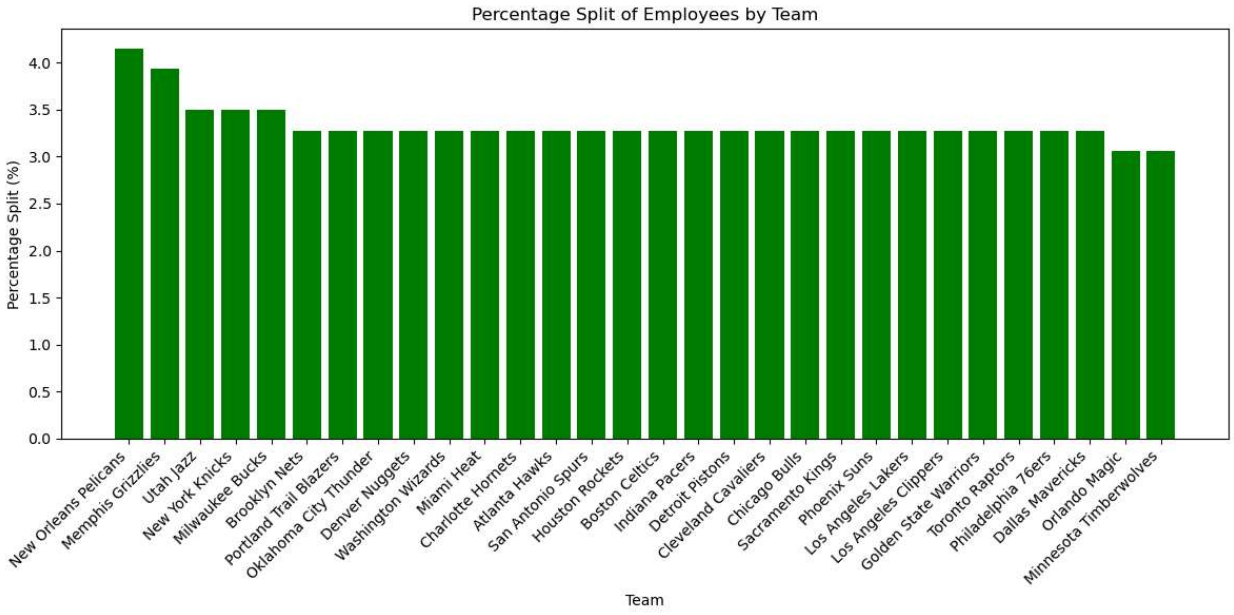
In [ ]:
```
Employment count is higher over the team New Orleans Pelicans. Minnesota Timberwolves
```

In [14]:
```
# Step 3: Compute the percentage split for each team
team_distribution['percentage_split'] = (team_distribution['employee_count'] / total_e
```

In [15]:
```
# Display the result
print(team_distribution)
```

	Team	employee_count	percentage_split
0	New Orleans Pelicans	19	4.148472
1	Memphis Grizzlies	18	3.930131
2	Utah Jazz	16	3.493450
3	New York Knicks	16	3.493450
4	Milwaukee Bucks	16	3.493450
5	Brooklyn Nets	15	3.275109
6	Portland Trail Blazers	15	3.275109
7	Oklahoma City Thunder	15	3.275109
8	Denver Nuggets	15	3.275109
9	Washington Wizards	15	3.275109
10	Miami Heat	15	3.275109
11	Charlotte Hornets	15	3.275109
12	Atlanta Hawks	15	3.275109
13	San Antonio Spurs	15	3.275109
14	Houston Rockets	15	3.275109
15	Boston Celtics	15	3.275109
16	Indiana Pacers	15	3.275109
17	Detroit Pistons	15	3.275109
18	Cleveland Cavaliers	15	3.275109
19	Chicago Bulls	15	3.275109
20	Sacramento Kings	15	3.275109
21	Phoenix Suns	15	3.275109
22	Los Angeles Lakers	15	3.275109
23	Los Angeles Clippers	15	3.275109
24	Golden State Warriors	15	3.275109
25	Toronto Raptors	15	3.275109
26	Philadelphia 76ers	15	3.275109
27	Dallas Mavericks	15	3.275109
28	Orlando Magic	14	3.056769
29	Minnesota Timberwolves	14	3.056769

In [16]:
```
# Plotting the bar graph for percentage split
plt.figure(figsize=(12, 6))
plt.bar(team_distribution['Team'], team_distribution['percentage_split'], color='greer
plt.xlabel('Team')
plt.ylabel('Percentage Split (%)')
plt.title('Percentage Split of Employees by Team')

# Adjusting the x-axis labels
plt.xticks(rotation=45, ha='right')

# Adjusting the layout
plt.tight_layout()

# Showing the plot
plt.show()
```



New Orleans Pelicans has higher percentage split%.

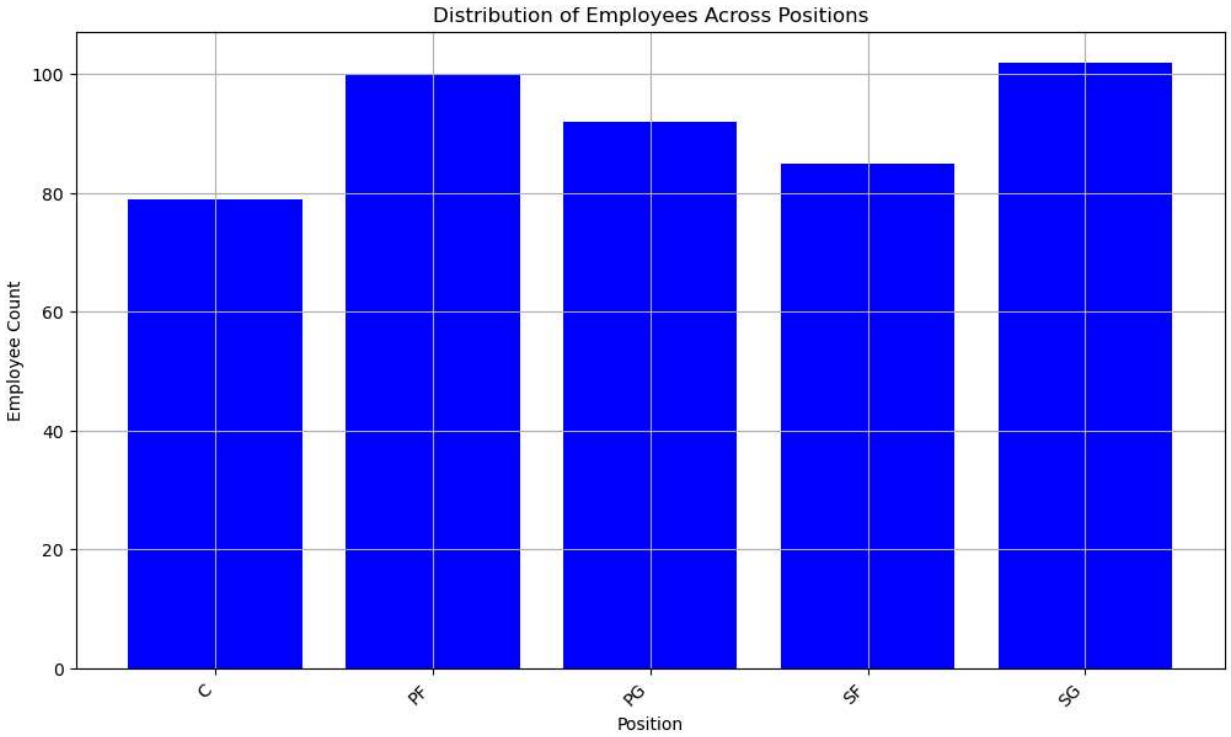
1. Segregate employees based on their positions within the company. (2 marks)

In [17]:
```
# Group by position and count the number of employees in each position
position_distribution = df.groupby('Position').size().reset_index(name='employee_count
```

In [18]:
```
# Display the result
print(position_distribution)
```

```
  Position  employee_count
0        C              79
1       PF             100
2       PG              92
3       SF              85
4       SG             102
```

In [19]:
```
import matplotlib.pyplot as plt
# Plotting the bar graph
plt.figure(figsize=(10, 6))
plt.bar(position_distribution['Position'], position_distribution['employee_count'], co
plt.xlabel('Position')
# Adjusting the x-axis labels
plt.xticks(rotation=45, ha='right')

# Adjusting the layout
plt.tight_layout()
plt.ylabel('Employee Count')

plt.title('Distribution of Employees Across Positions')
plt.grid(True)

plt.show()
```



Employment count is higher over SG position.

1. Identify the predominant age group among employees. (2 marks)

In [23]:
```
bins = [20, 30, 40, 50]  # Adjust these bins as necessary
labels = ['20-29', '30-39', '40-49']
```

In [24]:
```
# Categorize employees into these age groups
df['age_group'] = pd.cut(df['Age'], bins=bins, labels=labels, right=False)
```

In [25]:
```
df['age_group']
```

Out[25]:
```
0      20-29
1      20-29
2      20-29
3      20-29
4      20-29
       ...
453    20-29
454    20-29
455    20-29
456    20-29
457    20-29
Name: age_group, Length: 458, dtype: category
Categories (3, object): ['20-29' < '30-39' < '40-49']
```

In [26]:
```
df['Age']
```

Out[26]:
```
0      25
1      25
2      27
3      22
4      29
       ..
453    26
454    24
455    26
456    26
457    25
Name: Age, Length: 458, dtype: int64
```

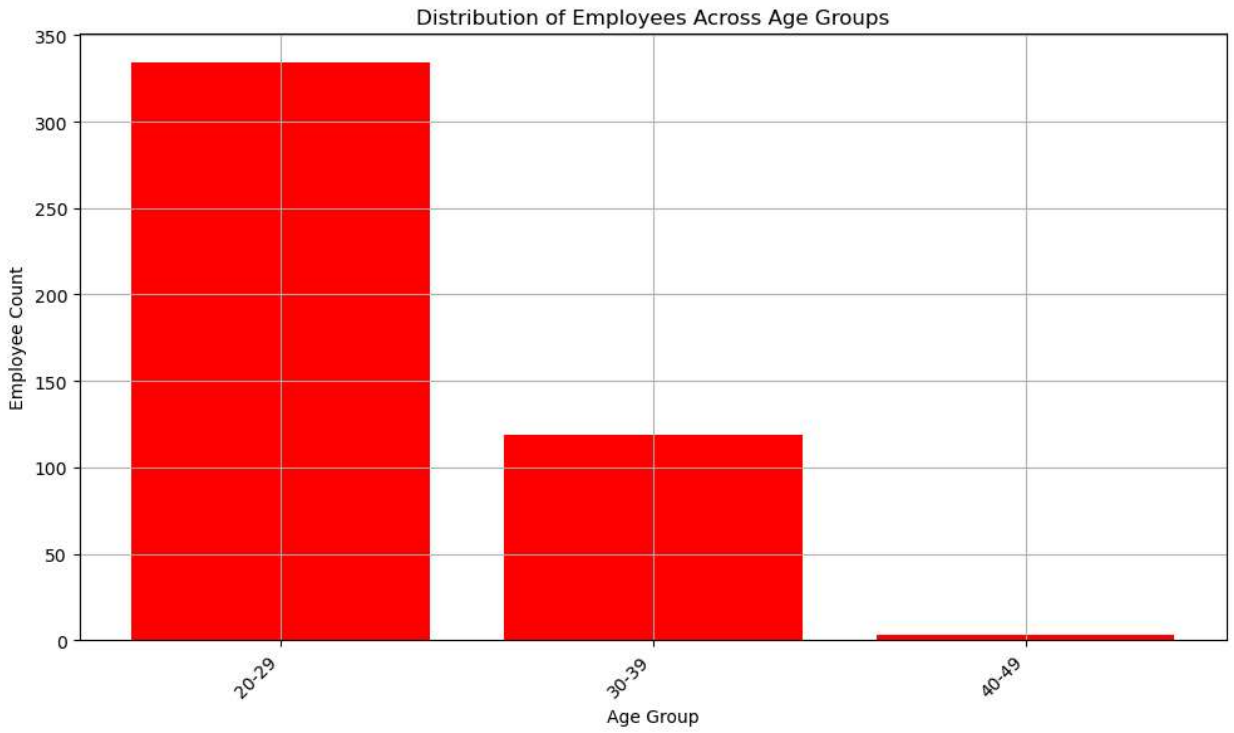
In [27]:
```
age_group_distribution = df['age_group'].value_counts().reset_index()
age_group_distribution.columns = ['age_group', 'employee_count']
```

In [28]:
```
predominant_age_group = age_group_distribution.loc[age_group_distribution['employee_co
```

In [29]:
```
predominant_age_group
```

Out[29]:
```
age_group         20-29
employee_count      334
Name: 0, dtype: object
```

In [30]:
```
import matplotlib.pyplot as plt

# Plotting the bar graph
plt.figure(figsize=(10, 6))

# Use the DataFrame columns directly
plt.bar(age_group_distribution['age_group'], age_group_distribution['employee_count'],

plt.xlabel('Age Group')
plt.ylabel('Employee Count')
plt.title('Distribution of Employees Across Age Groups')

# Adjusting the x-axis labels
plt.xticks(rotation=45, ha='right')

# Adjusting the layout
plt.tight_layout()

# Adding grid lines
plt.grid(True)

# Showing the plot
plt.show()
```



In [ ]:
```
Employment count is higer over the age group 40-49.
```

1. Discover which team and position have the highest salary expenditure. (2 marks)

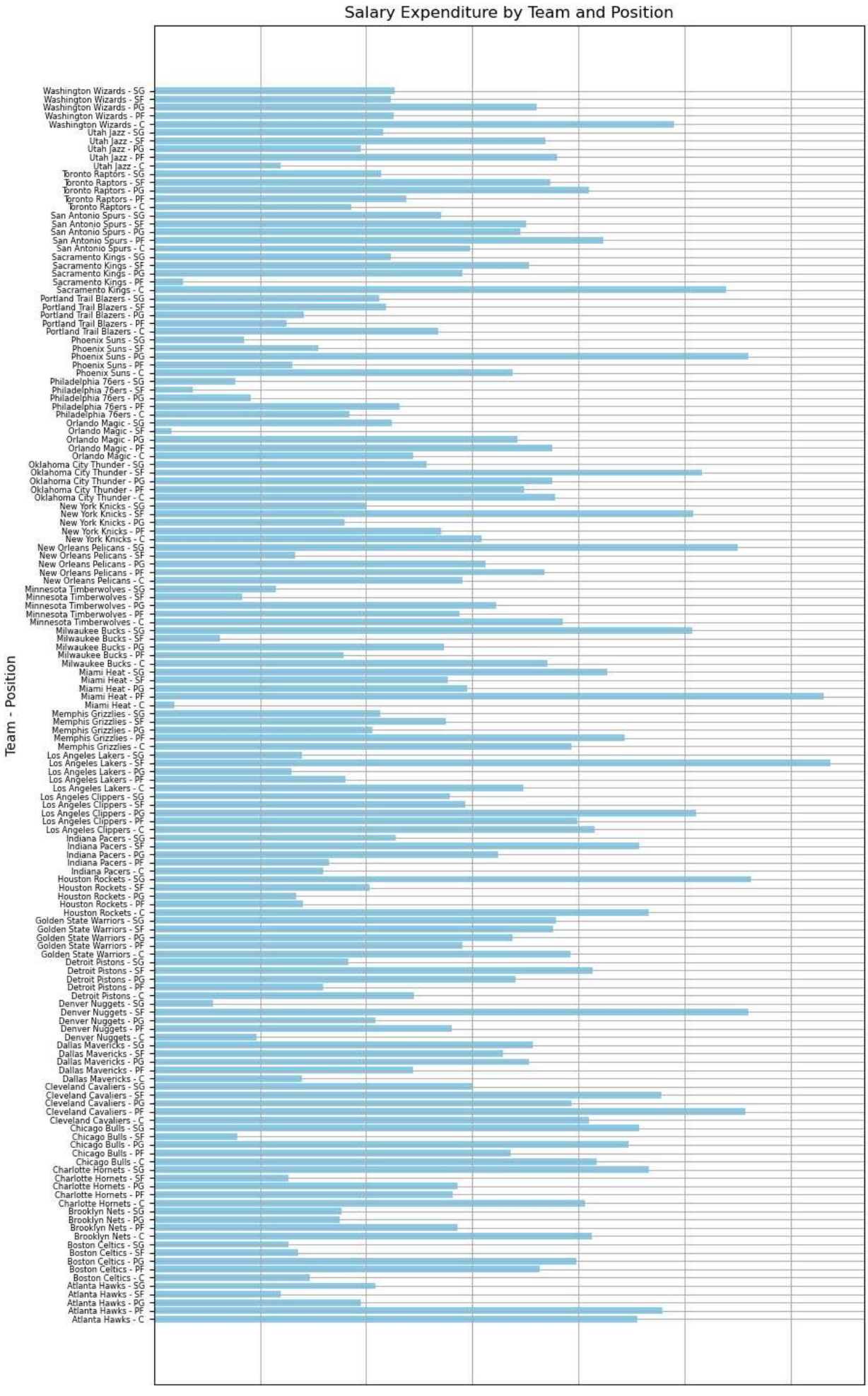
In [31]:
```
salary_expenditure = df.groupby(['Team', 'Position'])['Salary'].sum().reset_index()
```

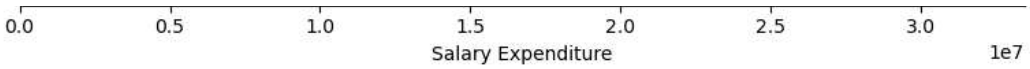
In [32]:
```
# Identify the team and position with the highest salary expenditure
highest_salary_expenditure = salary_expenditure.loc[salary_expenditure['Salary'].idxma
```

In [33]:
```
highest_salary_expenditure
```

Out[33]:
```
Team        Los Angeles Lakers
Position                    SF
Salary              31866445.0
Name: 67, dtype: object
```

In [45]:
```
# Group by Team and Position, and sum the Salary
salary_expenditure = df.groupby(['Team', 'Position'])['Salary'].sum().reset_index()

# Plotting with rotated labels
plt.figure(figsize=(10, 15))
bars = plt.barh(salary_expenditure['Team'] + ' - ' + salary_expenditure['Position'], s
plt.xlabel('Salary Expenditure')
plt.ylabel('Team - Position')
plt.title('Salary Expenditure by Team and Position')
plt.grid(True)
plt.tight_layout()

# Rotate y-axis labels
# Reduce y-axis tick label font size
plt.tick_params(axis='y', labelsize=6)

plt.show()
```


Salary Expenditure by Team and Position
Team - Position
Washington Wizards - SG
Washington Wizards - SF
Washington Wizards - PG
Washington Wizards - PF
Washington Wizards - C
Utah Jazz - SG
Utah Jazz - SF
Utah Jazz - PG
Utah Jazz - PF
Utah Jazz - C
Toronto Raptors - SG
Toronto Raptors - SF
Toronto Raptors - PG
Toronto Raptors - PF
Toronto Raptors - C
San Antonio Spurs - SG
San Antonio Spurs - SF
San Antonio Spurs - PG
San Antonio Spurs - PF
San Antonio Spurs - C
Sacramento Kings - SG
Sacramento Kings - SF
Sacramento Kings - PG
Sacramento Kings - PF
Sacramento Kings - C
Portland Trail Blazers - SG
Portland Trail Blazers - SF
Portland Trail Blazers - PG
Portland Trail Blazers - PF
Portland Trail Blazers - C
Phoenix Suns - SG
Phoenix Suns - SF
Phoenix Suns - PG
Phoenix Suns - PF
Phoenix Suns - C
Philadelphia 76ers - SG
Philadelphia 76ers - SF
Philadelphia 76ers - PG
Philadelphia 76ers - PF
Philadelphia 76ers - C
Orlando Magic - SG
Orlando Magic - SF
Orlando Magic - PG
Orlando Magic - PF
Orlando Magic - C
Oklahoma City Thunder - SG
Oklahoma City Thunder - SF
Oklahoma City Thunder - PG
Oklahoma City Thunder - PF
Oklahoma City Thunder - C
New York Knicks - SG
New York Knicks - SF
New York Knicks - PG
New York Knicks - PF
New York Knicks - C
New Orleans Pelicans - SG
New Orleans Pelicans - SF
New Orleans Pelicans - PG
New Orleans Pelicans - PF
New Orleans Pelicans - C
Minnesota Timberwolves - SG
Minnesota Timberwolves - SF
Minnesota Timberwolves - PG
Minnesota Timberwolves - PF
Minnesota Timberwolves - C
Milwaukee Bucks - SG
Milwaukee Bucks - SF
Milwaukee Bucks - PG
Milwaukee Bucks - PF
Milwaukee Bucks - C
Miami Heat - SG
Miami Heat - SF
Miami Heat - PG
Miami Heat - PF
Miami Heat - C
Memphis Grizzlies - SG
Memphis Grizzlies - SF
Memphis Grizzlies - PG
Memphis Grizzlies - PF
Memphis Grizzlies - C
Los Angeles Lakers - SG
Los Angeles Lakers - SF
Los Angeles Lakers - PG
Los Angeles Lakers - PF
Los Angeles Lakers - C
Los Angeles Clippers - SG
Los Angeles Clippers - SF
Los Angeles Clippers - PG
Los Angeles Clippers - PF
Los Angeles Clippers - C
Indiana Pacers - SG
Indiana Pacers - SF
Indiana Pacers - PG
Indiana Pacers - PF
Indiana Pacers - C
Houston Rockets - SG
Houston Rockets - SF
Houston Rockets - PG
Houston Rockets - PF
Houston Rockets - C
Golden State Warriors - SG
Golden State Warriors - SF
Golden State Warriors - PG
Golden State Warriors - PF
Golden State Warriors - C
Detroit Pistons - SG
Detroit Pistons - SF
Detroit Pistons - PG
Detroit Pistons - PF
Detroit Pistons - C
Denver Nuggets - SG
Denver Nuggets - SF
Denver Nuggets - PG
Denver Nuggets - PF
Denver Nuggets - C
Dallas Mavericks - SG
Dallas Mavericks - SF
Dallas Mavericks - PG
Dallas Mavericks - PF
Dallas Mavericks - C
Cleveland Cavaliers - SG
Cleveland Cavaliers - SF
Cleveland Cavaliers - PG
Cleveland Cavaliers - PF
Cleveland Cavaliers - C
Chicago Bulls - SG
Chicago Bulls - SF
Chicago Bulls - PG
Chicago Bulls - PF
Chicago Bulls - C
Charlotte Hornets - SG
Charlotte Hornets - SF
Charlotte Hornets - PG
Charlotte Hornets - PF
Charlotte Hornets - C
Brooklyn Nets - SG
Brooklyn Nets - PG
Brooklyn Nets - PF
Brooklyn Nets - C
Boston Celtics - SG
Boston Celtics - SF
Boston Celtics - PG
Boston Celtics - PF
Boston Celtics - C
Atlanta Hawks - SG
Atlanta Hawks - SF
Atlanta Hawks - PG
Atlanta Hawks - PF
Atlanta Hawks - C



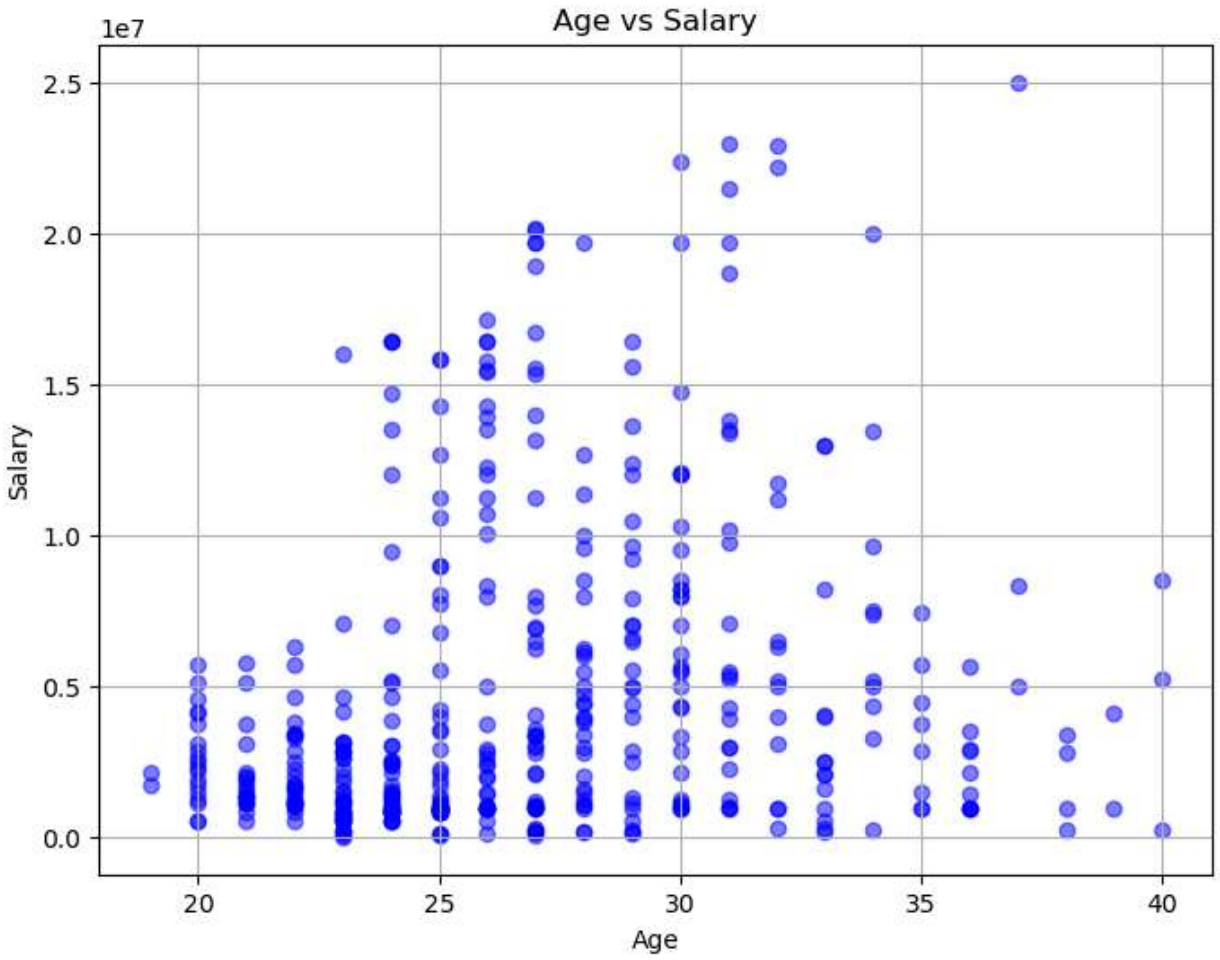
In [ ]:
```
Team - Los Angeles Lakers Position - SF, has highest Salary - 31866445.0
```

1. Investigate if there's any correlation between age and salary, and represent it visually. (2 marks)

In [33]:
```
# Calculate correlation between age and salary
correlation = df['Age'].corr(df['Salary'])
print("Correlation between age and salary:", correlation)
```

```
Correlation between age and salary: 0.21400941226570974
```

In [39]:
```
import matplotlib.pyplot as plt
# Visualize the correlation
plt.figure(figsize=(8, 6))
plt.scatter(df['Age'], df['Salary'], color='blue', alpha=0.5)
plt.title('Age vs Salary')
plt.xlabel('Age')
plt.ylabel('Salary')
plt.grid(True)
plt.show()
```



Its found that the age and salary has no significant correlation.

In [ ]: