Analysing employee details in company ABC

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

In [2]:
```
#reading the excel data into dataframe
df=pd.read_excel("C:/Users/Elitebook/Downloads/myexcel.xlsx")
```

In [3]:
```
#displaying top rows
df.head()
```

Out[3]:

	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

Data Preprocessing

In [3]:
```
df.describe
```

In [4]:
```
<bound method NDFrame.describe of                Name                Team  Number Position  Age              Height  \
0       Avery Bradley      Boston Celtics       0       PG   25 2023-02-06 00:00:00
1          Jae Crowder      Boston Celtics      99       SF   25 2023-06-06 00:00:00
2         John Holland      Boston Celtics      30       SG   27 2023-05-06 00:00:00
3          R.J. Hunter      Boston Celtics      28       SG   22 2023-05-06 00:00:00
4        Jonas Jerebko      Boston Celtics       8       PF   29 2023-10-06 00:00:00
..                 ...                 ...     ...      ...  ...                 ...
453     Shelvin Mack          Utah Jazz       8       PG   26 2023-03-06 00:00:00
454        Raul Neto          Utah Jazz      25       PG   24 2023-01-06 00:00:00
455     Tibor Pleiss          Utah Jazz      21        C   26 2023-03-07 00:00:00
456     Jeff Withey          Utah Jazz      24        C   26                  7-0
457         Priyanka          Utah Jazz      34        C   25 2023-03-07 00:00:00

       Weight            College      Salary
0         180              Texas   7730337.0
1         235          Marquette   6796117.0
2         205   Boston University         NaN
3         185      Georgia State   1148640.0
4         231                NaN   5000000.0
..        ...                ...         ...
453       203             Butler   2433333.0
454       179                NaN    900000.0
455       256                NaN   2900000.0
456       231             Kansas    947276.0
457       231             Kansas    947276.0

[458 rows x 9 columns]>
```

In [4]:
```
#details about columns
df.info()
```

In [5]:
```
<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    446 non-null    float64
dtypes: float64(1), int64(3), object(5)
memory usage: 32.3+ KB
```

In [6]:
```
df["Weight"] = np.random.randint(150, 181, size=df.shape[0])
df.head()
```

Out[6]:

	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0	PG	25	156	180	Texas	7730337.0
1	Jae Crowder	Boston Celtics	99	SF	25	178	235	Marquette	6796117.0
2	John Holland	Boston Celtics	30	SG	27	175	205	Boston University	NaN
3	R.J. Hunter	Boston Celtics	28	SG	22	158	185	Georgia State	1148640.0
4	Jonas Jerebko	Boston Celtics	8	PF	29	168	231	NaN	5000000.0

In [7]:
```
# drop unwanted column
df = df.drop(['Number','Height','Weight'],axis=1)
df.head()
```

Out[7]:

	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0	PG	25	Texas	7730337.0		
1	Jae Crowder	Boston Celtics	SF	25	Marquette	6796117.0			
2	John Holland	Boston Celtics	SG	27	Boston University	NaN			
3	R.J. Hunter	Boston Celtics	SG	22	Georgia State	1148640.0			
4	Jonas Jerebko	Boston Celtics	PF	29	NaN	5000000.0			

In [8]:
```
#replacing NaN salary values
df['Salary'] = df['Salary'].fillna(0)
df.head()
```

Out[8]:

	Name	Team	Number	Position	Age	College	Salary
0	Avery Bradley	Boston Celtics	PG	25	Texas	7730337.0	
1	Jae Crowder	Boston Celtics	SF	25	Marquette	6796117.0	
2	John Holland	Boston Celtics	SG	27	Boston University	0.0	
3	R.J. Hunter	Boston Celtics	SG	22	Georgia State	1148640.0	
4	Jonas Jerebko	Boston Celtics	PF	29	NaN	5000000.0	

In [9]:
```
df.describe()
```

Out[9]:

	Age	Salary
count	458.000000	4.580000e+02
mean	26.936468	4.717870e+06
std	4.404128	5.215322e+06
min	19.000000	0.000000e+00
25%	24.000000	1.000000e+06
50%	26.000000	2.647980e+06
75%	30.000000	6.325915e+06
max	40.000000	2.500000e+07

1.How many are there in each Team and the percentage splitting with respect to the total employees.

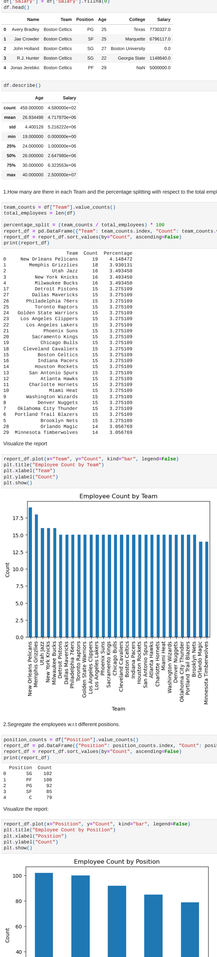
In [23]:
```
team_counts = df["Team"].value_counts()
total_employees = len(df)

percentage_split = (team_counts / total_employees) * 100
report_df = pd.DataFrame({"Team": team_counts.index, "Count": team_counts.values, "Percentage": percentage_split.values})
report_df = report_df.sort_values(by="Count", ascending=False)
print(report_df)
```

```
                  Team  Count  Percentage
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          Detroit Pistons     15    3.275109
6          Dallas Mavericks     15    3.275109
7          Philadelphia 76ers     15    3.275109
8          Toronto Raptors     15    3.275109
9    Golden State Warriors     15    3.275109
10     Los Angeles Clippers     15    3.275109
11       Los Angeles Lakers     15    3.275109
12            Phoenix Suns     15    3.275109
13       Sacramento Kings     15    3.275109
14        Chicago Bulls     15    3.275109
15     Cleveland Cavaliers     15    3.275109
16          Boston Celtics     15    3.275109
17          Indiana Pacers     15    3.275109
18         Houston Rockets     15    3.275109
19       San Antonio Spurs     15    3.275109
20          Atlanta Hawks     15    3.275109
21       Charlotte Hornets     15    3.275109
22          Miami Heat     15    3.275109
23      Washington Wizards     15    3.275109
24          Denver Nuggets     15    3.275109
25      Oklahoma City Thunder     15    3.275109
26    Portland Trail Blazers     15    3.275109
27          Brooklyn Nets     15    3.275109
28         Orlando Magic     14    3.056769
29    Minnesota Timberwolves     14    3.056769
```

Visualize the report

In [24]:
```
report_df.plot(x="Team", y="Count", kind="bar", legend=False)
plt.title("Employee Count by Team")
plt.xlabel("Team")
plt.ylabel("Count")
plt.show()
```



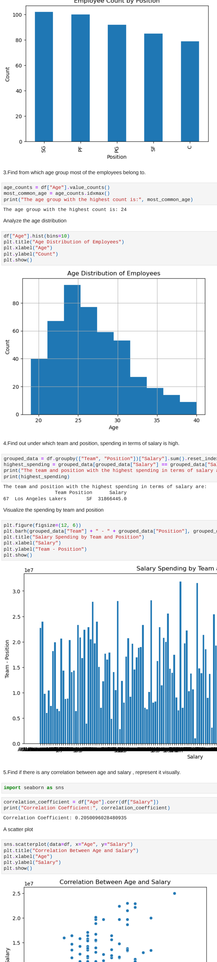
2.Segregate the employees w.r.t different positions.

In [25]:
```
position_counts = df["Position"].value_counts()
report_df = pd.DataFrame({"Position": position_counts.index, "Count": position_counts.values})
report_df = report_df.sort_values(by="Count", ascending=False)
print(report_df)
```

```
  Position  Count
0       SG    102
1       PF    100
2        PG    92
3        SF    89
4         C    75
```

Visualize the report:

In [26]:
```
report_df.plot(x="Position", y="Count", kind="bar", legend=False)
plt.title("Employee Count by Position")
plt.xlabel("Position")
plt.ylabel("Count")
plt.show()
```



3.Find from which age group most of the employees belong to.

In [27]:
```
age_counts = df["Age"].value_counts()
most_common_age = age_counts.idxmax()
print("The age group with the highest count is:", most_common_age)
```

```
The age group with the highest count is: 24
```

Analyze the age distribution

In [29]:
```
df["Age"].hist(bins=10)
plt.title("Age Distribution of Employees")
plt.xlabel("Age")
plt.ylabel("Count")
plt.show()
```


4.Find out under which team and position, spending in terms of salary is high.

In [30]:
```
grouped_data = df.groupby(["Team", "Position"])["Salary"].sum().reset_index()
highest_spending = grouped_data[grouped_data["Salary"] == grouped_data["Salary"].max()]
print("The team and position with the highest spending in terms of salary are:")
print(highest_spending)
```

```
The team and position with the highest spending in terms of salary are:
                Team Position     Salary
67  Los Angeles Lakers       SF  31866445.0
```

Visualize the spending by team and position

In [31]:
```
plt.figure(figsize=(12, 6))
plt.bar(grouped_data["Team"] + " " + grouped_data["Position"], grouped_data["Salary"])
plt.title('Salary Spending by Team and Position')
plt.xlabel("Salary")
plt.ylabel("Team - Position")
plt.show()
```


5.Find if there is any correlation between age and salary , represent it visually.

In [32]:
```
import seaborn as sns
```

In [33]:
```
correlation_coefficient = df["Age"].corr(df["Salary"])
print("Correlation Coefficient:", correlation_coefficient)
```

```
Correlation Coefficient: 0.2050096002848935
```

A scatter plot

In [34]:
```
sns.scatterplot(data=df, x="Age", y="Salary")
plt.title("Correlation Between Age and Salary")
plt.xlabel("Age")
plt.ylabel("Salary")
plt.show()
```