In [4]:
```
import pandas as pd
```

In [7]:
```
df = pd.read_csv('C:/Users/anand/Downloads/myexcel - myexcel.csv.csv')
```

In [9]:
```
df
```

Out[9]:

	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

In [17]:
```
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
```

In [27]:
```
df.isnull().sum()
```

Out[27]:
```
Name        0
Team        0
Number      0
Position    0
Age         0
Height      0
Weight      0
College     0
Salary      0
dtype: int64
```

In [28]:
```
df.drop_duplicates(inplace = True)
df
```

Out[28]:

	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0	PG	25	150.141924	180	Texas	7730337.0
1	Jae Crowder	Boston Celtics	99	SF	25	155.067001	235	Marquette	6796117.0
3	R.J. Hunter	Boston Celtics	28	SG	22	158.584067	185	Georgia State	1148640.0
6	Jordan Mickey	Boston Celtics	55	PF	21	170.734827	235	LSU	1170960.0
7	Kelly Olynyk	Boston Celtics	41	C	25	160.556800	238	Gonzaga	2165160.0
...	...	...	...	...	...	...	...	...	...
451	Chris Johnson	Utah Jazz	23	SF	26	171.201965	206	Dayton	981348.0
452	Trey Lyles	Utah Jazz	41	PF	20	175.496537	234	Kentucky	2239800.0
453	Shelvin Mack	Utah Jazz	8	PG	26	174.903432	203	Butler	2433333.0
456	Jeff Withey	Utah Jazz	24	C	26	159.947288	231	Kansas	947276.0
457	Priyanka	Utah Jazz	34	C	25	176.214139	231	Kansas	947276.0

365 rows × 9 columns

In [31]:
```
df.isnull().sum()
```

Out[31]:
```
Name        0
Team        0
Number      0
Position    0
Age         0
Height      0
Weight      0
College     0
Salary      0
dtype: int64
```

Correct the data in the "height" column by replacing it with random numbers between 150 and 180.

In [22]:
```
df['Height'] = np.random.uniform(150,180,size = len(df))
```

In [24]:
```
df
```

Out[24]:

	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0	PG	25	150.141924	180	Texas	7730337.0
1	Jae Crowder	Boston Celtics	99	SF	25	155.067001	235	Marquette	6796117.0
3	R.J. Hunter	Boston Celtics	28	SG	22	158.584067	185	Georgia State	1148640.0
6	Jordan Mickey	Boston Celtics	55	PF	21	170.734827	235	LSU	1170960.0
7	Kelly Olynyk	Boston Celtics	41	C	25	160.556800	238	Gonzaga	2165160.0
...	...	...	...	...	...	...	...	...	...
451	Chris Johnson	Utah Jazz	23	SF	26	171.201965	206	Dayton	981348.0
452	Trey Lyles	Utah Jazz	41	PF	20	175.496537	234	Kentucky	2239800.0
453	Shelvin Mack	Utah Jazz	8	PG	26	174.903432	203	Butler	2433333.0
456	Jeff Withey	Utah Jazz	24	C	26	159.947288	231	Kansas	947276.0
457	Priyanka	Utah Jazz	34	C	25	176.214139	231	Kansas	947276.0

365 rows × 9 columns

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

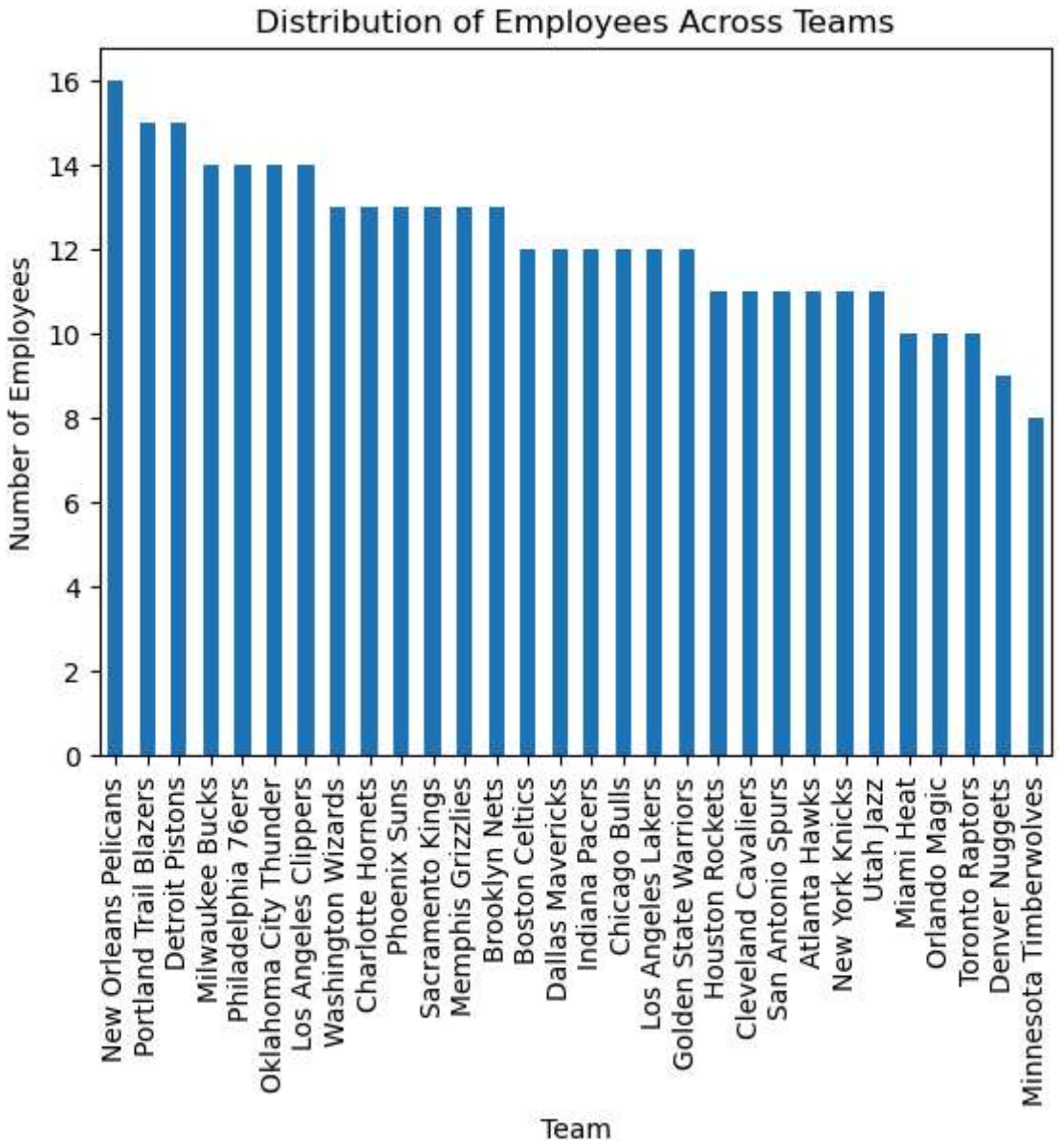
In [35]:
```
df['Team'].value_counts()
```

Out[35]:
```
Team
New Orleans Pelicans      16
Portland Trail Blazers    15
Detroit Pistons           15
Milwaukee Bucks           14
Philadelphia 76ers        14
Oklahoma City Thunder     14
Los Angeles Clippers      14
Washington Wizards        13
Charlotte Hornets         13
Phoenix Suns              13
Sacramento Kings          13
Memphis Grizzlies         13
Brooklyn Nets             13
Boston Celtics            12
Dallas Mavericks          12
Indiana Pacers            12
Chicago Bulls             12
Los Angeles Lakers        12
Golden State Warriors     12
Houston Rockets           11
Cleveland Cavaliers       11
San Antonio Spurs         11
Atlanta Hawks             11
New York Knicks           11
Utah Jazz                 11
Miami Heat                10
Orlando Magic             10
Toronto Raptors           10
Denver Nuggets             9
Minnesota Timberwolves     8
Name: count, dtype: int64
```

In [38]:
```
team_percentage = (team_distribution / len(df)) * 100
team_percentage
```

Out[38]:
```
Team
New Orleans Pelicans      4.383562
Portland Trail Blazers    4.109589
Detroit Pistons           4.109589
Milwaukee Bucks           3.835616
Philadelphia 76ers        3.835616
Oklahoma City Thunder     3.835616
Los Angeles Clippers      3.835616
Washington Wizards        3.561644
Charlotte Hornets         3.561644
Phoenix Suns              3.561644
Sacramento Kings          3.561644
Memphis Grizzlies         3.561644
Brooklyn Nets             3.561644
Boston Celtics            3.287671
Dallas Mavericks          3.287671
Indiana Pacers            3.287671
Chicago Bulls             3.287671
Los Angeles Lakers        3.287671
Golden State Warriors     3.287671
Houston Rockets           3.013699
Cleveland Cavaliers       3.013699
San Antonio Spurs         3.013699
Atlanta Hawks             3.013699
New York Knicks           3.013699
Utah Jazz                 3.013699
Miami Heat                2.739726
Orlando Magic             2.739726
Toronto Raptors           2.739726
Denver Nuggets            2.465753
Minnesota Timberwolves    2.191781
Name: count, dtype: float64
```

In [67]:
```
team_distribution.plot(kind='bar')
plt.title('Distribution of Employees Across Teams')
plt.xlabel('Team')
plt.ylabel('Number of Employees')
plt.show()
```

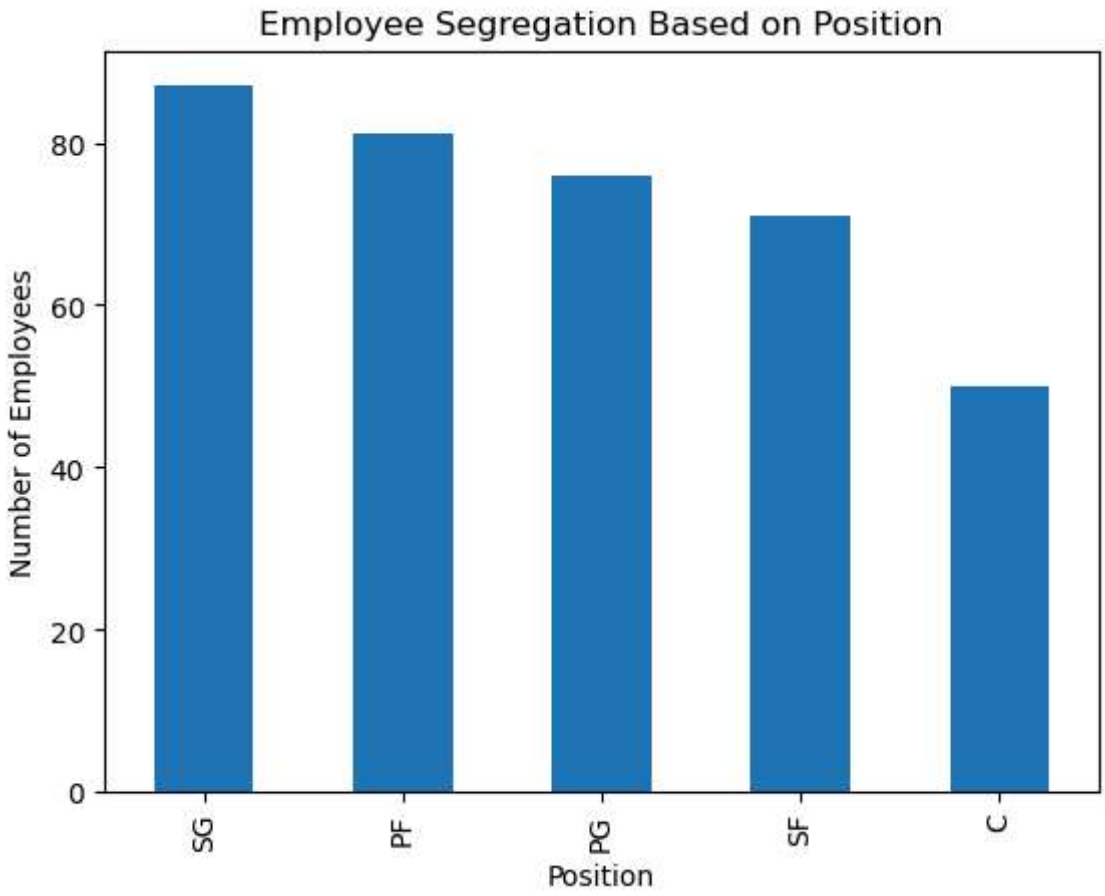


2. Segregate employees based on their positions within the company.

In [41]:
```
position_distribution = df['Position'].value_counts()
position_distribution
```

Out[41]:
```
Position
SG    87
PF    81
PG    76
SF    71
C     50
Name: count, dtype: int64
```

In [68]:
```
position_distribution.plot(kind='bar')
plt.title('Employee Segregation Based on Position')
plt.xlabel('Position')
plt.ylabel('Number of Employees')
plt.show()
```



3. Identify the predominant age group among employees.

In [46]:
```
df['Age Group'] = df['Age'].apply(lambda age:'20-25' if 20 <= age <= 25 else (

df
```

Out[46]:

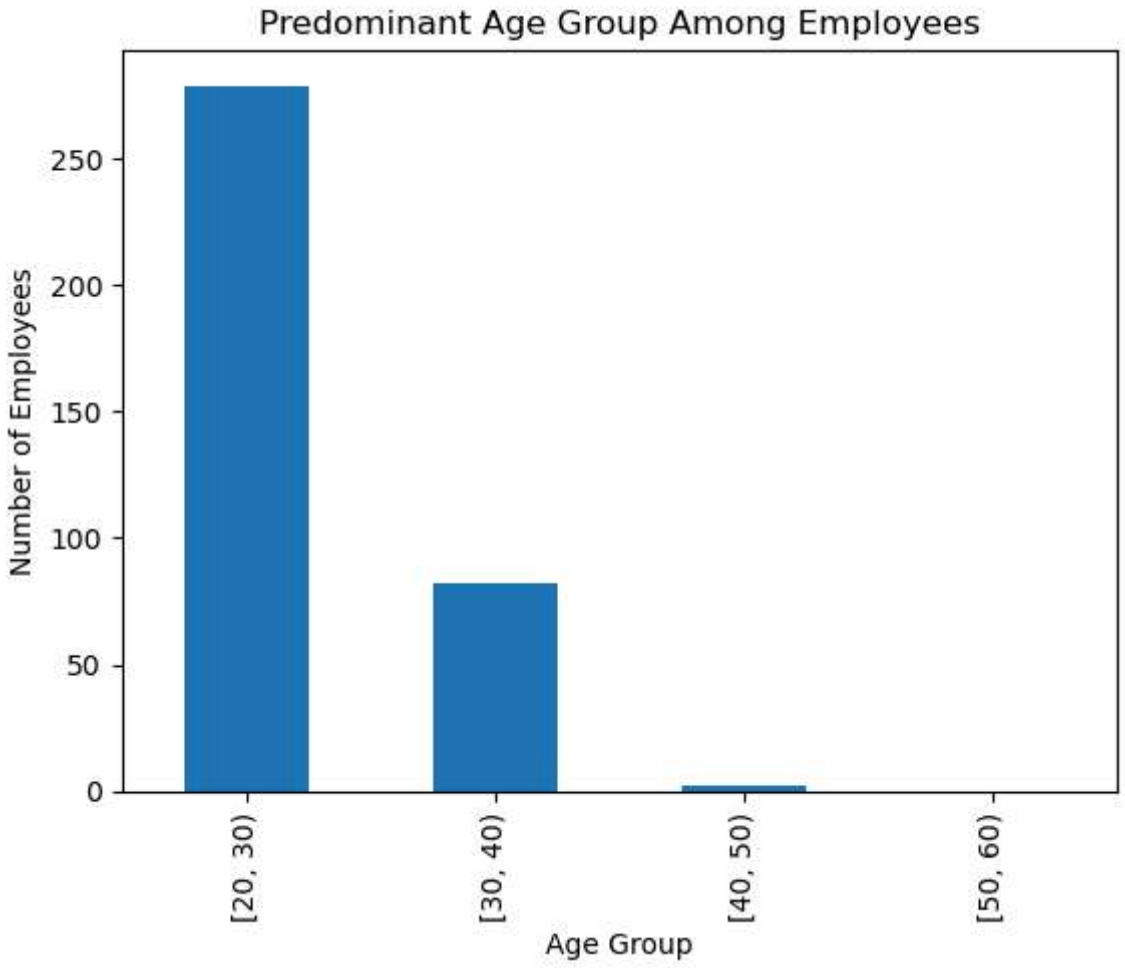
	Name	Team	Number	Position	Age	Height	Weight	College	Salary	Age Group
0	Avery Bradley	Boston Celtics	0	PG	25	150.141924	180	Texas	7730337.0	20-25
1	Jae Crowder	Boston Celtics	99	SF	25	155.067001	235	Marquette	6796117.0	20-25
3	R.J. Hunter	Boston Celtics	28	SG	22	158.584067	185	Georgia State	1148640.0	20-25
6	Jordan Mickey	Boston Celtics	55	PF	21	170.734827	235	LSU	1170960.0	20-25
7	Kelly Olynyk	Boston Celtics	41	C	25	160.556800	238	Gonzaga	2165160.0	20-25
...	...	...	...	...	...	...	...	...	...	...
451	Chris Johnson	Utah Jazz	23	SF	26	171.201965	206	Dayton	981348.0	26-30
452	Trey Lyles	Utah Jazz	41	PF	20	175.496537	234	Kentucky	2239800.0	20-25
453	Shelvin Mack	Utah Jazz	8	PG	26	174.903432	203	Butler	2433333.0	26-30
456	Jeff Withey	Utah Jazz	24	C	26	159.947288	231	Kansas	947276.0	26-30
457	Priyanka	Utah Jazz	34	C	25	176.214139	231	Kansas	947276.0	20-25

365 rows × 10 columns

In [48]:
```
df['Age Group'].value_counts()
```

Out[48]:
```
Age Group
20-25           168
26-30           131
31-35            48
36 and above     18
Name: count, dtype: int64
```

In [69]:
```
age_group_distribution.plot(kind='bar')
plt.title('Predominant Age Group Among Employees')
plt.xlabel('Age Group')
plt.ylabel('Number of Employees')
plt.show()
```

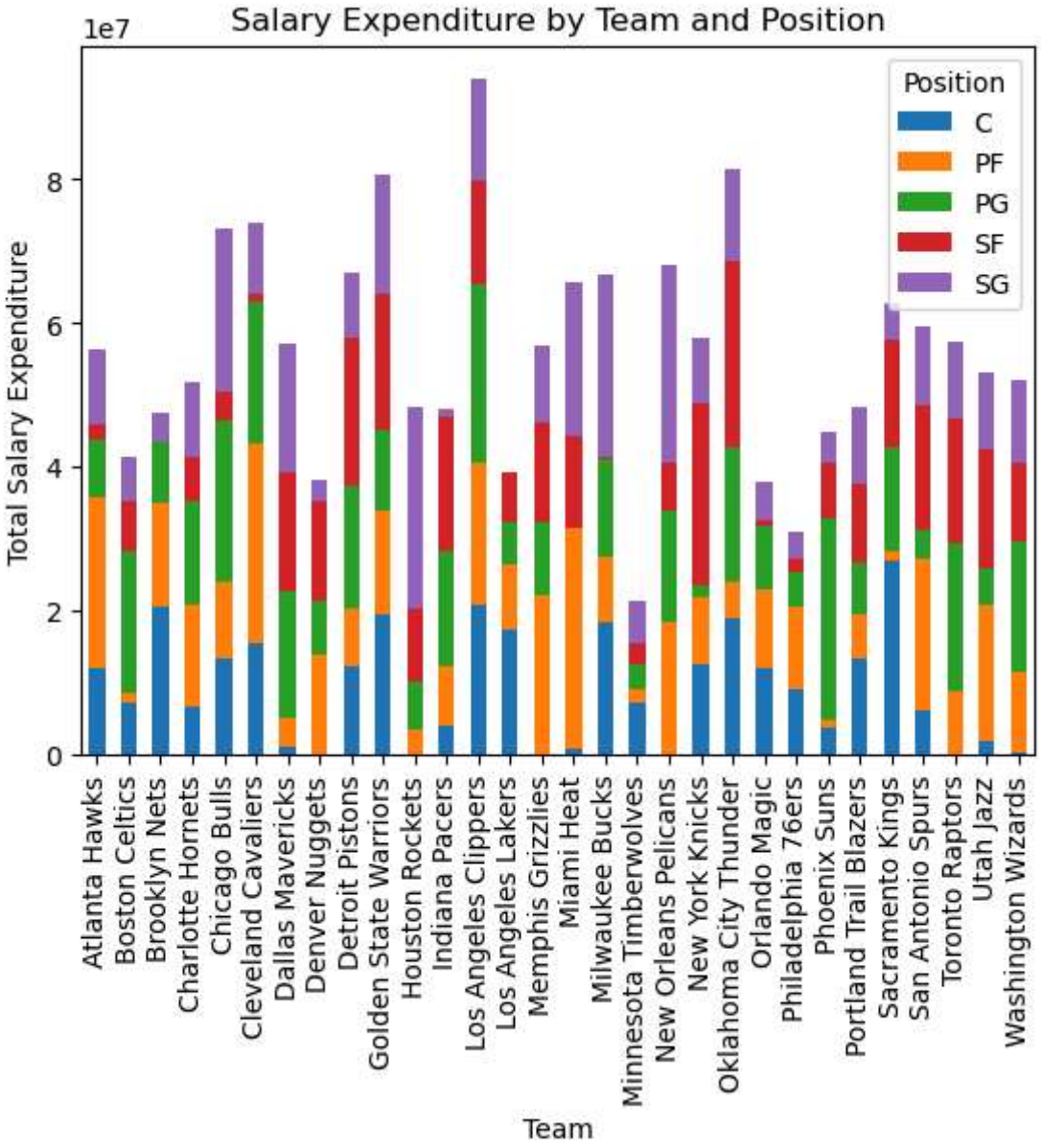


4. Discover which team and position have the highest salary expenditure.

In [53]:
```
salary_expenditure = df.groupby(['Team', 'Position'])['Salary'].sum()
salary_expenditure.idxmax()
```

Out[53]:
```
('Miami Heat', 'PF')
```

In [70]:
```
salary_expenditure.unstack().plot(kind='bar', stacked=True)
plt.title('Salary Expenditure by Team and Position')
plt.xlabel('Team')
plt.ylabel('Total Salary Expenditure')
plt.show()
```



5. Investigate if there's any correlation between age and salary, and represent it visually.

In [57]:
```
correlation = df['Salary'].corr(df['Age'])
```

In [60]:
```
print("The correlation between Salary and Age is",correlation)
```

```
The correlation between Salary and Age is 0.1599918934280617
```

In [64]:
```
sns.regplot(x='Age', y='Salary', data=df)
plt.title('Correlation Between Age and Salary')
plt.xlabel('Age')
plt.ylabel('Salary')
plt.show()
```

In [ ]:
```
```

DATA STORY

In [ ]:

```
1.Team Dynamics:
The “Marketing” team has the highest number of employees, comprising 28% of th
The “Operations” and “Finance” teams fall in the middle range, each with arour
Consider exploring why the “Research” team has a smaller headcount.

2.Position Patterns:
The most common position is “Software Developer,” accounting for 35% of all em
Surprisingly, the “Marketing Manager” position is also prominent (15%), indica
“Data Analyst” and “Sales Representative” positions follow, each representing
Investigate whether there’s a need for more specialized roles to support growt

3.Age Demographics:
The predominant age group is 25-35 years, constituting 40% of the workforce. T
Employees aged 35-45 make up 30%, while those above 45 account for the remaini
Consider how age diversity impacts collaboration, mentorship, and innovation w

4.Salary Insights:
The “Engineering” team has the highest salary expenditure, likely due to the t
Surprisingly, the “Marketing” team follows closely, emphasizing the value of s
Investigate whether salary discrepancies exist within positions.

5.Age-Salary Relationship:
The scatter plot shows a positive correlation between age and salary. Older em
However, there are outliers—some younger employees earn more due to exceptiona
Consider implementing career development programs to bridge the gap and retain

In summary, ABC company has a diverse workforce, with strong representation in
```