HR Attrition project - Juan Diego

This project will explain possible causes of why employees inside a company decide to quit their job.

In [2]:
```python
import pip
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

In [13]:
```python
Hrattr = 'HR attrition.csv'
df = pd.read_csv(Hrattr)
df
```

Out[13]:

	Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education	EducationField	EmployeeCount	EmployeeNumber	...	RelationshipSatisfaction	StandardHou
0	41	Yes	Travel_Rarely	1102	Sales	1	2	Life Sciences	1	1	...	1	8
1	49	No	Travel_Frequently	279	Research & Development	8	1	Life Sciences	1	2	...	4	8
2	37	Yes	Travel_Rarely	1373	Research & Development	2	2	Other	1	4	...	2	8
3	33	No	Travel_Frequently	1392	Research & Development	3	4	Life Sciences	1	5	...	3	8
4	27	No	Travel_Rarely	591	Research & Development	2	1	Medical	1	7	...	4	8
...	...	...	...	...	...	...	...	...	...	...	...	...	
1465	36	No	Travel_Frequently	884	Research & Development	23	2	Medical	1	2061	...	3	8
1466	39	No	Travel_Rarely	613	Research & Development	6	1	Medical	1	2062	...	1	8
1467	27	No	Travel_Rarely	155	Research & Development	4	3	Life Sciences	1	2064	...	2	8
1468	49	No	Travel_Frequently	1023	Sales	2	3	Medical	1	2065	...	4	8
1469	34	No	Travel_Rarely	628	Research & Development	8	3	Medical	1	2068	...	1	8

1470 rows × 35 columns

In [112…:
```python
df.shape
```

Out[112]:
```
(1470, 35)
```

In [158…:
```python
df.columns
```

Out[158]:
```
Index(['Age', 'Attrition', 'BusinessTravel', 'Department', 'DistanceFromHome',
       'Education', 'EducationField', 'EmployeeNumber',
       'EnvironmentSatisfaction', 'Gender', 'JobInvolvement', 'JobLevel',
       'JobRole', 'JobSatisfaction', 'MaritalStatus', 'MonthlyIncome',
       'NumCompaniesWorked', 'Over18', 'OverTime', 'PercentSalaryHike',
       'PerformanceRating', 'RelationshipSatisfaction', 'StockOptionLevel',
       'TotalWorkingYears', 'TrainingTimesLastYear', 'WorkLifeBalance',
       'YearsAtCompany', 'YearsInCurrentRole', 'YearsSinceLastPromotion',
       'YearsWithCurrManager'],
      dtype='object')
```

In [114…:
```python
df.dtypes
```

Out[114]:
```
Age                          int64
Attrition                   object
BusinessTravel              object
DailyRate                    int64
Department                  object
DistanceFromHome             int64
Education                    int64
EducationField              object
EmployeeCount                int64
EmployeeNumber               int64
EnvironmentSatisfaction      int64
Gender                      object
HourlyRate                   int64
JobInvolvement               int64
JobLevel                     int64
JobRole                     object
JobSatisfaction              int64
MaritalStatus               object
MonthlyIncome                int64
MonthlyRate                  int64
NumCompaniesWorked           int64
Over18                      object
OverTime                    object
PercentSalaryHike            int64
PerformanceRating            int64
RelationshipSatisfaction     int64
StandardHours                int64
StockOptionLevel             int64
TotalWorkingYears            int64
TrainingTimesLastYear        int64
WorkLifeBalance              int64
YearsAtCompany               int64
YearsInCurrentRole           int64
YearsSinceLastPromotion      int64
YearsWithCurrManager         int64
dtype: object
```

In [29]:
```python
df.isnull().sum()
```

Out[29]:
```
Age                         0
Attrition                   0
BusinessTravel              0
DailyRate                   0
Department                  0
DistanceFromHome            0
Education                   0
EducationField              0
EmployeeCount               0
EmployeeNumber              0
EnvironmentSatisfaction     0
Gender                      0
HourlyRate                  0
JobInvolvement              0
JobLevel                    0
JobRole                     0
JobSatisfaction             0
MaritalStatus               0
MonthlyIncome               0
MonthlyRate                 0
NumCompaniesWorked          0
Over18                      0
OverTime                    0
PercentSalaryHike           0
PerformanceRating           0
RelationshipSatisfaction    0
StandardHours               0
StockOptionLevel            0
TotalWorkingYears           0
TrainingTimesLastYear       0
WorkLifeBalance             0
YearsAtCompany              0
YearsInCurrentRole          0
YearsSinceLastPromotion     0
YearsWithCurrManager        0
dtype: int64
```

In [28]:
```python
statistics = df.describe()
print(statistics)
```

```
               Age    DailyRate  DistanceFromHome    Education  EmployeeCount  \
count  1470.000000  1470.000000       1470.000000  1470.000000         1470.0
mean     36.923810   802.485714          9.192517     2.912925            1.0
std       9.135373   403.509100          8.106864     1.024165            0.0
min      18.000000   102.000000          1.000000     1.000000            1.0
25%      30.000000   465.000000          2.000000     2.000000            1.0
50%      36.000000   802.000000          7.000000     3.000000            1.0
75%      43.000000  1157.000000         14.000000     4.000000            1.0
max      60.000000  1499.000000         29.000000     5.000000            1.0

       EmployeeNumber  EnvironmentSatisfaction   HourlyRate  JobInvolvement  \
count     1470.000000              1470.000000  1470.000000     1470.000000
mean      1024.865306                 2.721769    65.891156        2.729932
std        602.024335                 1.093082    20.329428        0.711561
min          1.000000                 1.000000    30.000000        1.000000
25%        491.250000                 2.000000    48.000000        2.000000
50%       1020.500000                 3.000000    66.000000        3.000000
75%       1555.750000                 4.000000    83.750000        3.000000
max       2068.000000                 4.000000   100.000000        4.000000

          JobLevel  ...  PerformanceRating  RelationshipSatisfaction  \
count  1470.000000  ...        1470.000000               1470.000000
mean      2.063946  ...           3.153741                  2.712245
std       1.106940  ...           0.360824                  1.081209
min       1.000000  ...           3.000000                  1.000000
25%       1.000000  ...           3.000000                  2.000000
50%       2.000000  ...           3.000000                  3.000000
75%       3.000000  ...           3.000000                  4.000000
max       5.000000  ...           4.000000                  4.000000

       StockOptionLevel  TotalWorkingYears  TrainingTimesLastYear  \
count       1470.000000        1470.000000            1470.000000
mean           0.793878          11.279592               2.799320
std            0.852077           7.780782               1.289271
min            0.000000           0.000000               0.000000
25%            0.000000           6.000000               2.000000
50%            1.000000          10.000000               3.000000
75%            1.000000          15.000000               3.000000
max            3.000000          40.000000               6.000000

       WorkLifeBalance  YearsAtCompany  YearsInCurrentRole  \
count      1470.000000     1470.000000         1470.000000
mean          2.761224        7.008163            4.229252
std           0.706476        6.126525            3.623137
min           1.000000        0.000000            0.000000
25%           2.000000        3.000000            2.000000
50%           3.000000        5.000000            3.000000
75%           3.000000        9.000000            7.000000
max           4.000000       40.000000           18.000000

       YearsSinceLastPromotion  YearsWithCurrManager
count              1470.000000           1470.000000
mean                  2.187755              4.123129
std                   3.222430              3.568136
min                   0.000000              0.000000
25%                   0.000000              2.000000
50%                   1.000000              3.000000
75%                   3.000000              7.000000
max                  15.000000             17.000000

[8 rows x 25 columns]
```

We can see through the descriptive statistics that there are columns that have unique values and doesnt help that much with the analysis. For example "Employeecount" That has value of 1, or "StandardHours", "DailyRate","HourlyRate "Over18"

In [14]:
```python
Dropcolumn ='Over18'
Dropcolumn ='DailyRate'
Dropcolumn ='HourlyRate'
Dropcolumn ='EmployeeCount'
Dropcolumn ='StandardHours'
df = df.drop(Dropcolumn,axis=1)
```

In [160…:
```python
df
```

Out[160]:

	Age	Attrition	BusinessTravel	Department	DistanceFromHome	Education	EducationField	EmployeeNumber	EnvironmentSatisfaction	Gender	...	PerformanceRating	Relationshi
0	41	Yes	Travel_Rarely	Sales	1	2	Life Sciences	1	2	Female	...	3	
1	49	No	Travel_Frequently	Research & Development	8	1	Life Sciences	2	3	Male	...	4	
2	37	Yes	Travel_Rarely	Research & Development	2	2	Other	4	4	Male	...	3	
3	33	No	Travel_Frequently	Research & Development	3	4	Life Sciences	5	4	Female	...	3	
4	27	No	Travel_Rarely	Research & Development	2	1	Medical	7	1	Male	...	3	
...	...	...	...	...	...	...	...	...	...	...	...	...	
1465	36	No	Travel_Frequently	Research & Development	23	2	Medical	2061	3	Male	...	3	
1466	39	No	Travel_Rarely	Research & Development	6	1	Medical	2062	4	Male	...	3	
1467	27	No	Travel_Rarely	Research & Development	4	3	Life Sciences	2064	2	Male	...	4	
1468	49	No	Travel_Frequently	Sales	2	3	Medical	2065	4	Male	...	3	
1469	34	No	Travel_Rarely	Research & Development	8	3	Medical	2068	2	Male	...	3	

1470 rows × 29 columns

In [34]:
```python
df
```

Out[34]:

	Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education	EducationField	EmployeeCount	EmployeeNumber	...	PerformanceRating	RelationshipSatisfa
0	41	Yes	Travel_Rarely	1102	Sales	1	2	Life Sciences	1	1	...	3	
1	49	No	Travel_Frequently	279	Research & Development	8	1	Life Sciences	1	2	...	4	
2	37	Yes	Travel_Rarely	1373	Research & Development	2	2	Other	1	4	...	3	
3	33	No	Travel_Frequently	1392	Research & Development	3	4	Life Sciences	1	5	...	3	
4	27	No	Travel_Rarely	591	Research & Development	2	1	Medical	1	7	...	3	
...	...	...	...	...	...	...	...	...	...	...	...	...	
1465	36	No	Travel_Frequently	884	Research & Development	23	2	Medical	1	2061	...	3	
1466	39	No	Travel_Rarely	613	Research & Development	6	1	Medical	1	2062	...	3	
1467	27	No	Travel_Rarely	155	Research & Development	4	3	Life Sciences	1	2064	...	4	
1468	49	No	Travel_Frequently	1023	Sales	2	3	Medical	1	2065	...	3	
1469	34	No	Travel_Rarely	628	Research & Development	8	3	Medical	1	2068	...	3	

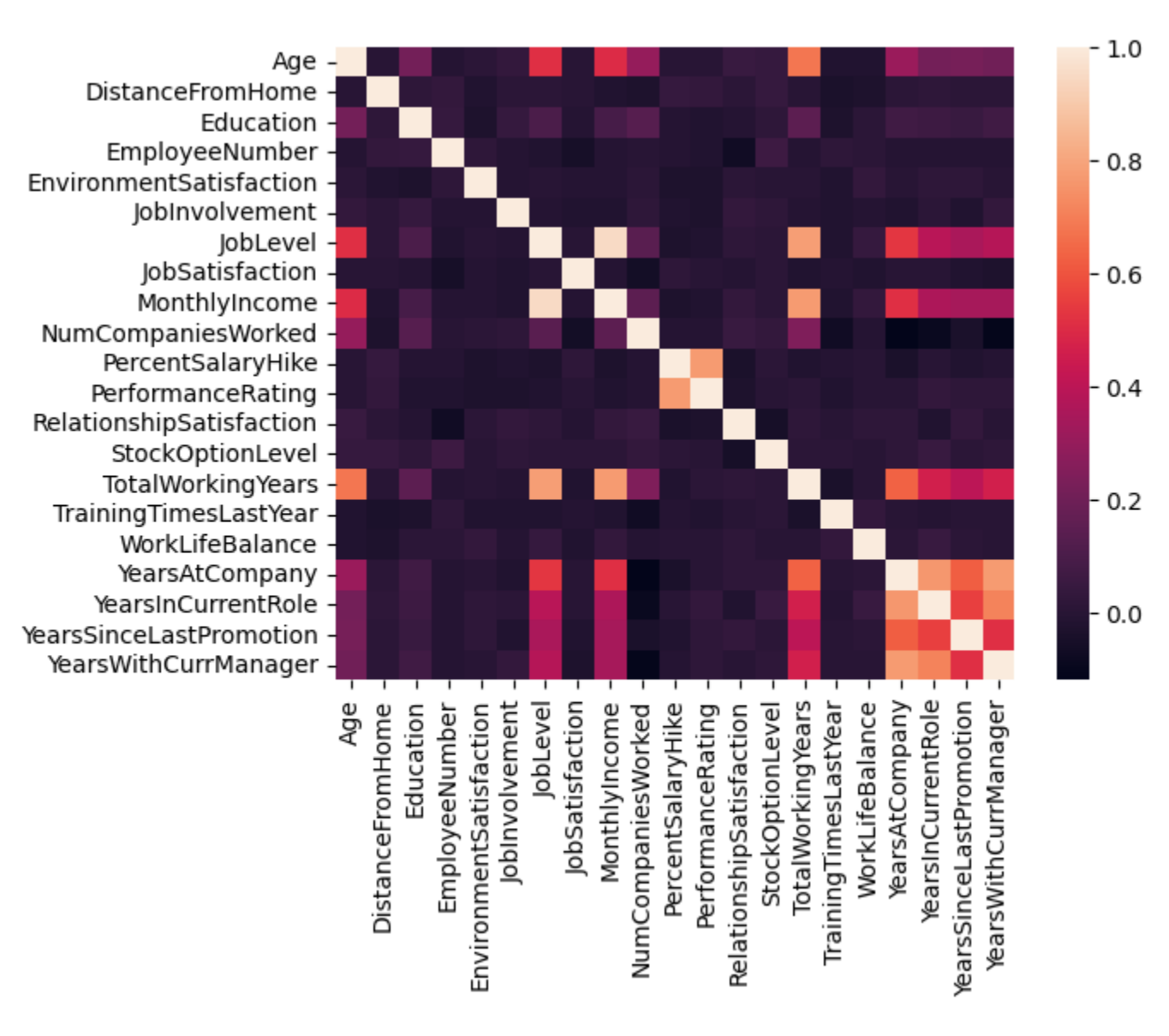
1470 rows × 34 columns

Correlation and possible explanations

We are going to create a correlation matrix in order to find possible patterns of what is most valuable in the company.

In [258…:
```python
sns.heatmap(df.corr().round(2))
```

Out[258]:
```
<AxesSubplot:>
```



Then, we proceed to create another column by categorizing the ages into 3 group ages

In [52]:
```python
def Age_range(Age):

    if Age >= 18 and Age <= 35:
        return '18 to 35'
    elif Age > 35 and Age <= 50:
        return '35 to 50'
    elif Age > 50:
        return '50 or more'
    else:
        return 'Unknown'

dfAttrition['AgeGroup'] = dfAttrition['Age'].apply(Age_range)
dfAttrition
```

```
C:\Users\juanm\AppData\Local\Temp\ipykernel_27240\2695766382.py:12: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  dfAttrition['AgeGroup'] = dfAttrition['Age'].apply(Age_range)
```

Out[52]:

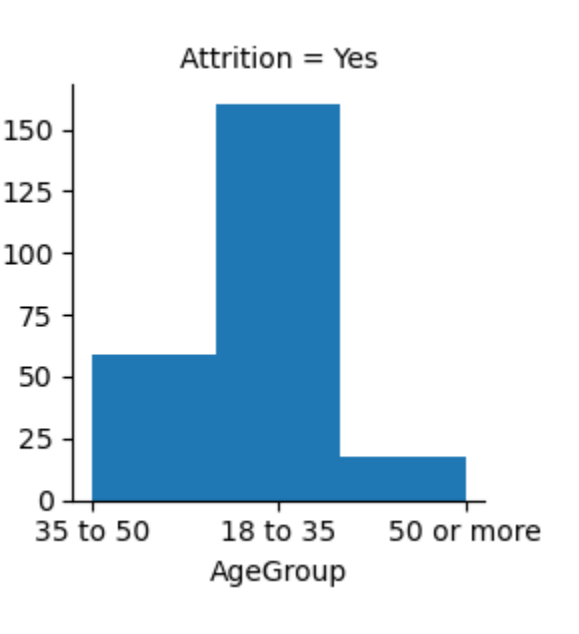
	Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education	EducationField	EmployeeCount	EmployeeNumber	...	RelationshipSatisfaction	StockOptionL
0	41	Yes	Travel_Rarely	1102	Sales	1	2	Life Sciences	1	1	...	1	
2	37	Yes	Travel_Rarely	1373	Research & Development	2	2	Other	1	4	...	2	
14	28	Yes	Travel_Rarely	103	Research & Development	24	3	Life Sciences	1	19	...	2	
21	36	Yes	Travel_Rarely	1218	Sales	9	4	Life Sciences	1	27	...	2	
24	34	Yes	Travel_Rarely	699	Research & Development	6	1	Medical	1	31	...	3	
...	...	...	...	...	...	...	...	...	...	...	...	...	
1438	23	Yes	Travel_Frequently	638	Sales	9	3	Marketing	1	2023	...	1	
1442	29	Yes	Travel_Rarely	1092	Research & Development	1	4	Medical	1	2027	...	2	
1444	56	Yes	Travel_Rarely	310	Research & Development	7	2	Technical Degree	1	2032	...	4	
1452	50	Yes	Travel_Frequently	878	Sales	1	4	Life Sciences	1	2044	...	4	
1461	50	Yes	Travel_Rarely	410	Sales	28	3	Marketing	1	2055	...	2	

237 rows × 35 columns

Histogram showing the attrition by age group

In [54]:
```python
agehist = sns.FacetGrid(dfAttrition, col='Attrition')
agehist.map(plt.hist, 'AgeGroup', bins=3)
```

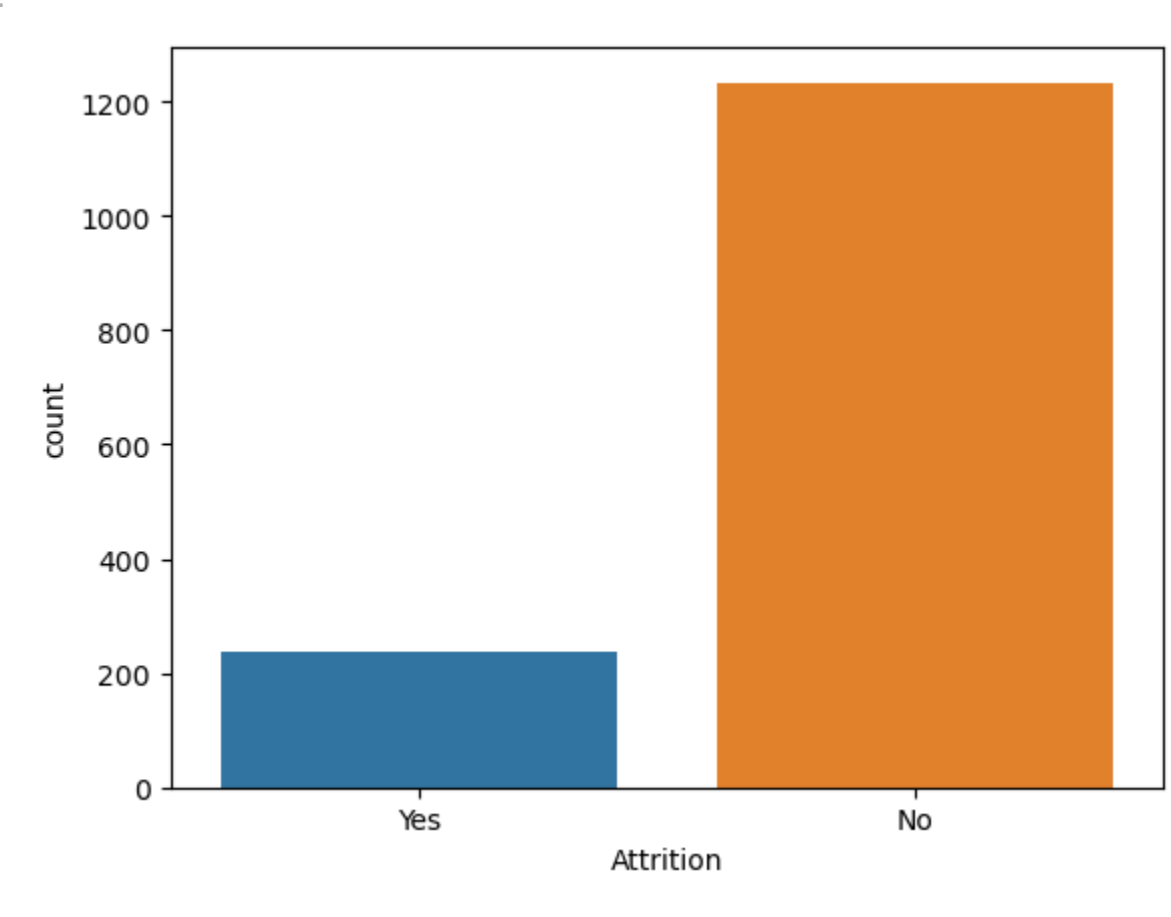
Out[54]:
```
<seaborn.axisgrid.FacetGrid at 0x19737230eb0>
```



Histogram showing the attrition by Gender

In [39]:
```python
sns.countplot(x = "Attrition", data = df)
```

Out[39]:
```
<AxesSubplot:xlabel='Attrition', ylabel='count'>
```



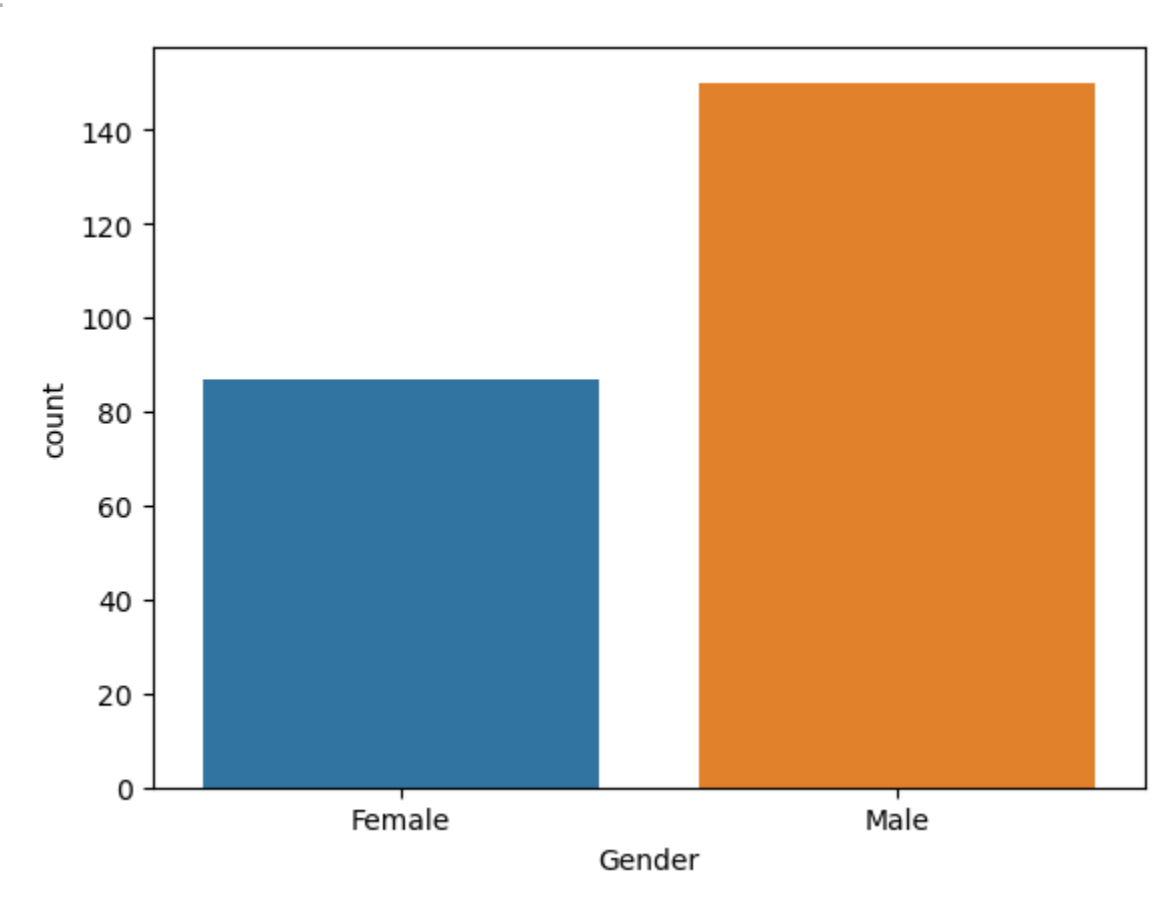
In [40]:
```python
(df.groupby('Attrition').Attrition.count() / df['Attrition'].count()) *100
```

Out[40]:
```
Attrition
No     83.877551
Yes    16.122449
Name: Attrition, dtype: float64
```

We can see that there is an Attrition of **16.12%** among all Employees.

In [41]:
```python
dfAttrition = df[df["Attrition"] == "Yes"]
dfNoAttrition = df[df["Attrition"] == "No"]
sns.countplot(x = "Gender", data = dfAttrition)
```

Out[41]:
```
<AxesSubplot:xlabel='Gender', ylabel='count'>
```



In [70]:
```python
pd.crosstab([dfAttrition["Department"], dfAttrition["Gender"]], dfAttrition["JobSatisfaction"],
    margins = True).style.background_gradient(cmap = "summer_r")
```

Out[70]:

Department	Gender \ JobSatisfaction	1	2	3	4	All
Human Resources	Female	1	2	2	1	6
	Male	4	0	1	1	6
Research & Development	Female	13	11	11	8	43
	Male	25	13	32	20	90
Sales	Female	7	11	13	7	38
	Male	16	9	14	15	54
All		66	46	73	52	237

We can clearly see low satisfaction jobs within the categories. Research and Development department has critical negative ratings and is increasing considerably the attritions metric. Human resources apparently is doing good and Sales can do better also. From this finding we are going to give a further look to R&D

In [64]:
```python
pd.crosstab([dfAttrition["Department"], dfAttrition["Gender"]], dfAttrition["OverTime"],
    margins = True).style.background_gradient(cmap = "summer_r")
```

Out[64]:

Department	Gender \ OverTime	No	Yes	All
Human Resources	Female	3	3	6
	Male	4	2	6
Research & Development	Female	19	24	43
	Male	40	50	90
Sales	Female	18	20	38
	Male	26	28	54
All		110	127	237

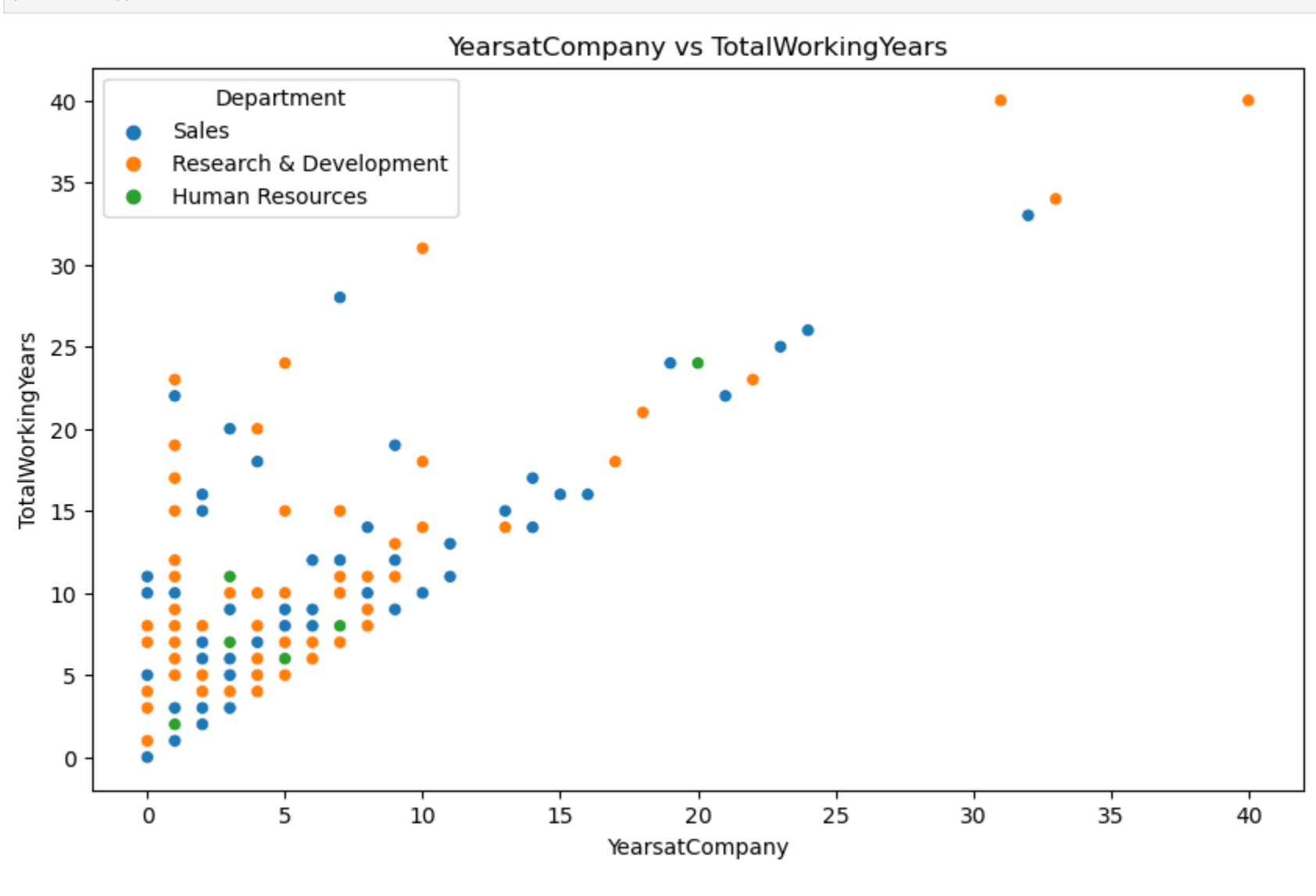
In [18]:
```python
pd.crosstab([dfAttrition["Department"], dfAttrition["Gender"]], dfAttrition["WorkLifeBalance"],
    margins = True).style.background_gradient(cmap = "summer_r")
```

Out[18]:

Department	Gender \ WorkLifeBalance	1	2	3	4	All
Human Resources	Female	0	2	4	0	6
	Male	0	0	5	1	6
Research & Development	Female	4	7	24	8	43
	Male	15	25	44	6	90
Sales	Female	3	10	19	6	38
	Male	3	14	31	6	54
All		25	58	127	27	237

Also, the percentage of employees quiting tend to have worked less than 5 years in the company. This means that most of the attritions that are explained by Junior - Mid Seniors professionals in which their desires are not fulfilled and want to try new experiences

In [43]:
```python
plt.figure(figsize=(10, 6))
sns.scatterplot(x='YearsAtCompany', y='TotalWorkingYears', data=dfAttrition, hue='Department')
plt.title('YearsatCompany vs TotalWorkingYears')
plt.xlabel('YearsatCompany')
plt.ylabel('TotalWorkingYears')
plt.legend(title='Department')
plt.show()
```



Monthly Income

We can clearly see that there is a considerable difference is the average monthly income comparing the cluster that quits more the job (18 to35)

In [72]:
```python
mean_salary = dfAttrition.groupby('AgeGroup')['MonthlyIncome'].mean()
mean_salary
```

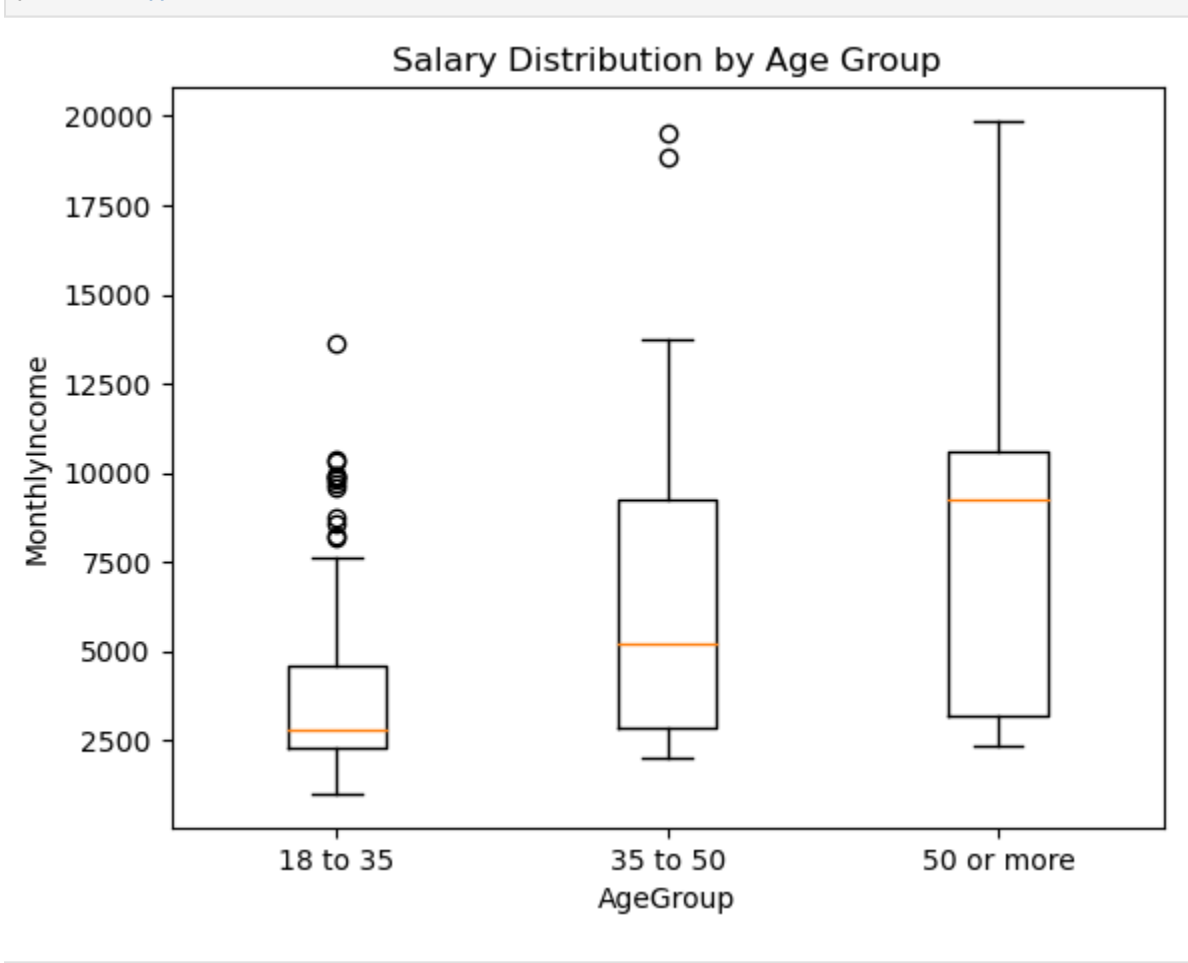
Out[72]:
```
AgeGroup
18 to 35      3734.975000
35 to 50      6389.101695
50 or more    8888.222222
Name: MonthlyIncome, dtype: float64
```

It can be inferred that in order to retain talent is needed to give rewards, bonuses or even promotions. We proceed to analyze the results with a boxplot to see the distribution of the data. By just looking it we can see a type of skewness in the salary.

In [82]:
```python
plt.boxplot([dfAttrition[dfAttrition['AgeGroup'] == '18 to 35']['MonthlyIncome'],
             dfAttrition[dfAttrition['AgeGroup'] == '35 to 50']['MonthlyIncome'],
             dfAttrition[dfAttrition['AgeGroup'] == '50 or more']['MonthlyIncome']],
            labels=['18 to 35', '35 to 50', '50 or more'])

# Set labels and title
plt.xlabel('AgeGroup')
plt.ylabel('MonthlyIncome')
plt.title('Salary Distribution by Age Group')

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



In [20]:
```python
dfAttrition.groupby("OverTime").OverTime.count()/ dfAttrition['OverTime'].count() *100
```

Out[20]:
```
OverTime
No     46.413502
Yes    53.586498
Name: OverTime, dtype: float64
```

In [ ]:
```
Moreover, we can see that the employees that tend to quit their jobs are the ones who don´t have promotions. Is recommended to dig in into this case
```

In [66]:
```python
dfAttrition.groupby('YearsSinceLastPromotion').YearsSinceLastPromotion.count()
```

Out[66]:
```
YearsSinceLastPromotion
0     110
1      49
2      27
3       9
4       5
5       2
6       6
7      16
9       4
10      1
11      2
13      2
14      1
15      3
Name: YearsSinceLastPromotion, dtype: int64
```

Recommendations

Implement policies such as remote work options, flexible hours, or compressed workweeks to provide employees with more control over their schedules.

Monitor employee workloads to ensure they are manageable and realistic. Consider redistributing tasks or hiring additional resources if necessary.

Provide a clear career progression plan for researchers, with opportunities for advancement and skill development. Invest in the professional development of researchers by providing opportunities for training, attending conferences, and publishing their work.

Conduct regular surveys or feedback sessions to understand the specific challenges faced by researchers in the department. Actively listen to their concerns and suggestions

Offer performance-based bonuses or incentives tied to individual or team research achievements.

Create pathways for career advancement within the Research department. This can include opportunities for promotions, access to leadership training programs

Implement a recognition program that recognizes researchers' accomplishments and contributions. Celebrate milestones in research, successful projects, and publications. Regular feedback and constructive feedback can boost morale and job satisfaction.