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

In [3]:
```
df = pd.read_csv("D:\HR Employee Attrition.csv")
```

In [4]:
```
df.head(5)
```

Out[4]:

StandardHours	StockOptionLevel	TotalWorkingYears	TrainingTimesLastYear	WorkLifeBalance
80	0	8	0	1
80	1	10	3	3
80	0	7	3	3
80	0	8	3	3
80	1	6	3	3

In [5]:
```
df.shape
```

Out[5]:
```
(1470, 35)
```

In [6]:
```
df.isnull().sum()
```

Out[6]:
```
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 [7]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1470 entries, 0 to 1469
Data columns (total 35 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   Age                       1470 non-null   int64
 1   Attrition                 1470 non-null   object
 2   BusinessTravel            1470 non-null   object
 3   DailyRate                 1470 non-null   int64
 4   Department                1470 non-null   object
 5   DistanceFromHome          1470 non-null   int64
 6   Education                 1470 non-null   int64
 7   EducationField            1470 non-null   object
 8   EmployeeCount             1470 non-null   int64
 9   EmployeeNumber            1470 non-null   int64
 10  EnvironmentSatisfaction   1470 non-null   int64
 11  Gender                    1470 non-null   object
 12  HourlyRate                1470 non-null   int64
 13  JobInvolvement            1470 non-null   int64
 14  JobLevel                  1470 non-null   int64
 15  JobRole                   1470 non-null   object
 16  JobSatisfaction           1470 non-null   int64
 17  MaritalStatus             1470 non-null   object
 18  MonthlyIncome             1470 non-null   int64
 19  MonthlyRate               1470 non-null   int64
 20  NumCompaniesWorked        1470 non-null   int64
 21  Over18                    1470 non-null   object
 22  OverTime                  1470 non-null   object
 23  PercentSalaryHike         1470 non-null   int64
 24  PerformanceRating         1470 non-null   int64
 25  RelationshipSatisfaction  1470 non-null   int64
 26  StandardHours             1470 non-null   int64
 27  StockOptionLevel          1470 non-null   int64
 28  TotalWorkingYears         1470 non-null   int64
 29  TrainingTimesLastYear     1470 non-null   int64
 30  WorkLifeBalance           1470 non-null   int64
 31  YearsAtCompany            1470 non-null   int64
 32  YearsInCurrentRole        1470 non-null   int64
 33  YearsSinceLastPromotion   1470 non-null   int64
 34  YearsWithCurrManager      1470 non-null   int64
dtypes: int64(26), object(9)
memory usage: 402.1+ KB
```

In [8]:
```
df.columns
```

Out[8]:
```
Index(['Age', 'Attrition', 'BusinessTravel', 'DailyRate', 'Department',
       'DistanceFromHome', 'Education', 'EducationField', 'EmployeeCoun
t',
       'EmployeeNumber', 'EnvironmentSatisfaction', 'Gender', 'HourlyRat
e',
       'JobInvolvement', 'JobLevel', 'JobRole', 'JobSatisfaction',
       'MaritalStatus', 'MonthlyIncome', 'MonthlyRate', 'NumCompaniesWork
ed',
       'Over18', 'OverTime', 'PercentSalaryHike', 'PerformanceRating',
       'RelationshipSatisfaction', 'StandardHours', 'StockOptionLevel',
       'TotalWorkingYears', 'TrainingTimesLastYear', 'WorkLifeBalance',
       'YearsAtCompany', 'YearsInCurrentRole', 'YearsSinceLastPromotion',
       'YearsWithCurrManager'],
      dtype='object')
```

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

Out[9]:

ıtion	EmployeeCount	EmployeeNumber	EnvironmentSatisfaction	HourlyRate	JobInvolvement
)000	1470.0	1470.000000	1470.000000	1470.000000	1470.000000
2925	1.0	1024.865306	2.721769	65.891156	2.729932
1165	0.0	602.024335	1.093082	20.329428	0.711561
)000	1.0	1.000000	1.000000	30.000000	1.000000
)000	1.0	491.250000	2.000000	48.000000	2.000000
)000	1.0	1020.500000	3.000000	66.000000	3.000000
)000	1.0	1555.750000	4.000000	83.750000	3.000000
)000	1.0	2068.000000	4.000000	100.000000	4.000000

In [14]:
```
df.fillna(method='ffill', inplace=True)
df
```

Out[14]:

ttrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education	EducationFie
Yes	Travel_Rarely	1102	Sales	1	2	Life Scienc
No	Travel_Frequently	279	Research & Development	8	1	Life Scienc
Yes	Travel_Rarely	1373	Research & Development	2	2	Oth
No	Travel_Frequently	1392	Research & Development	3	4	Life Scienc
No	Travel_Rarely	591	Research & Development	2	1	Medic
...	...	...	...	...	...	
No	Travel_Frequently	884	Research & Development	23	2	Medic
No	Travel_Rarely	613	Research & Development	6	1	Medic
No	Travel_Rarely	155	Research & Development	4	3	Life Scienc
No	Travel_Frequently	1023	Sales	2	3	Medic
No	Travel_Rarely	628	Research & Development	8	3	Medic

35 columns

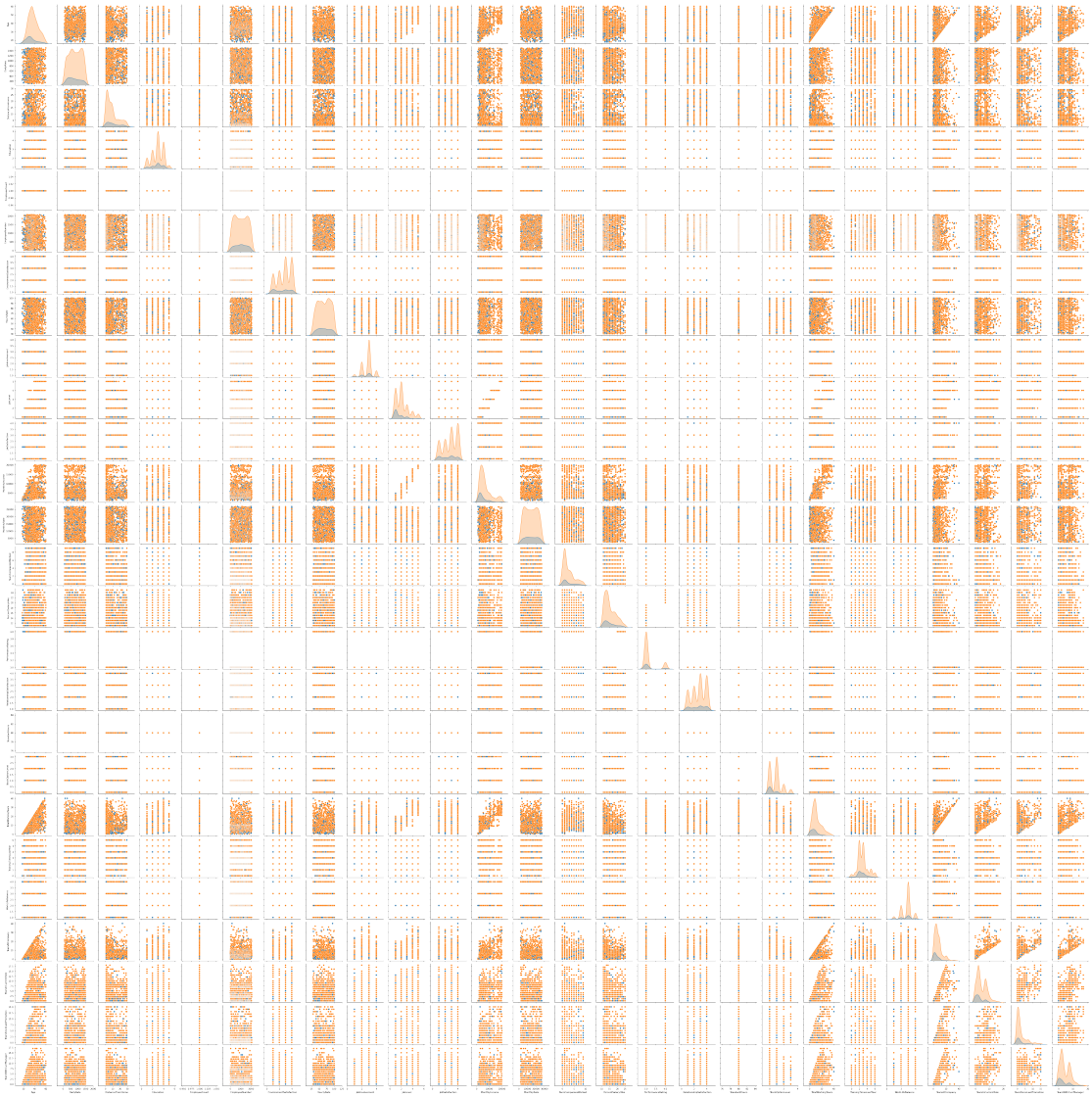
In [18]:
```
duplicates = df[df.duplicated()]
print(duplicates)
```

```
Empty DataFrame
Columns: [Age, Attrition, BusinessTravel, DailyRate, Department, Distance
FromHome, Education, EducationField, EmployeeCount, EmployeeNumber, Envir
onmentSatisfaction, Gender, HourlyRate, JobInvolvement, JobLevel, JobRol
e, JobSatisfaction, MaritalStatus, MonthlyIncome, MonthlyRate, NumCompani
esWorked, Over18, OverTime, PercentSalaryHike, PerformanceRating, Relatio
nshipSatisfaction, StandardHours, StockOptionLevel, TotalWorkingYears, Tr
ainingTimesLastYear, WorkLifeBalance, YearsAtCompany, YearsInCurrentRole,
YearsSinceLastPromotion, YearsWithCurrManager]
Index: []

[0 rows x 35 columns]
```

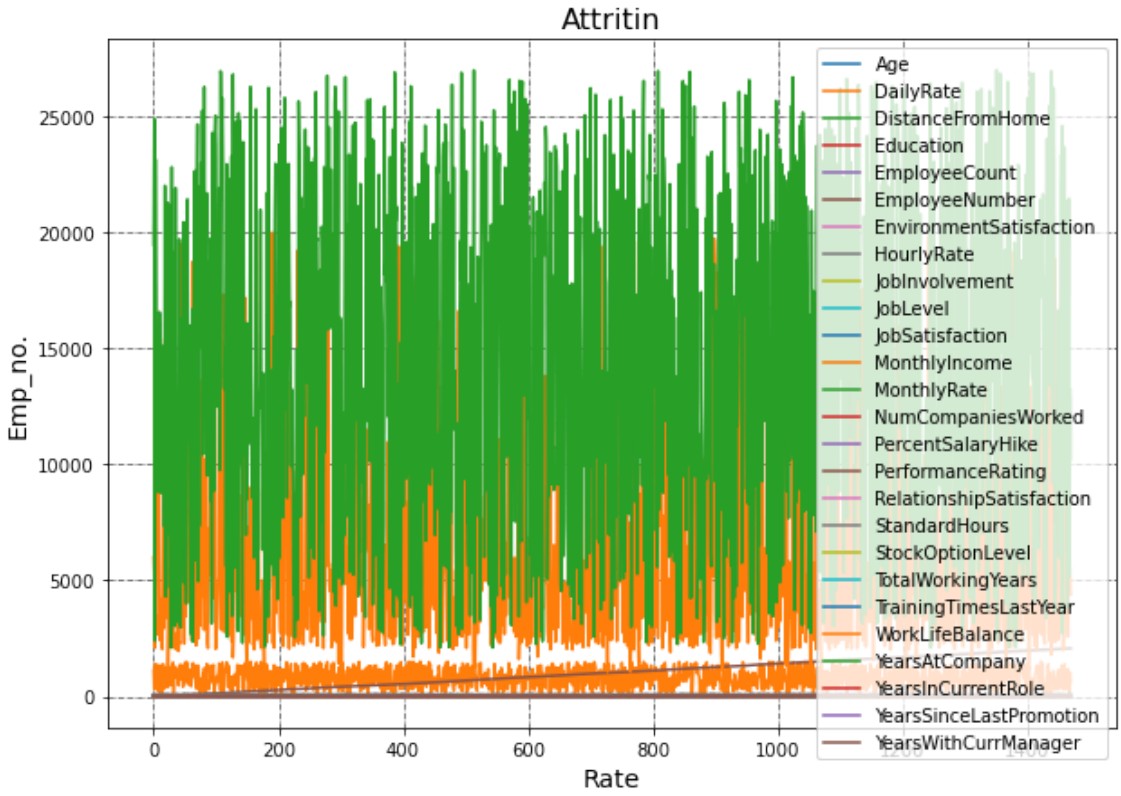
In [19]:
```
df.drop_duplicates(inplace=True)
```

In [31]:
```
sns.pairplot(df, hue= 'Attrition')
```

Out[31]:
```
<seaborn.axisgrid.PairGrid at 0x135501e2d90>
```

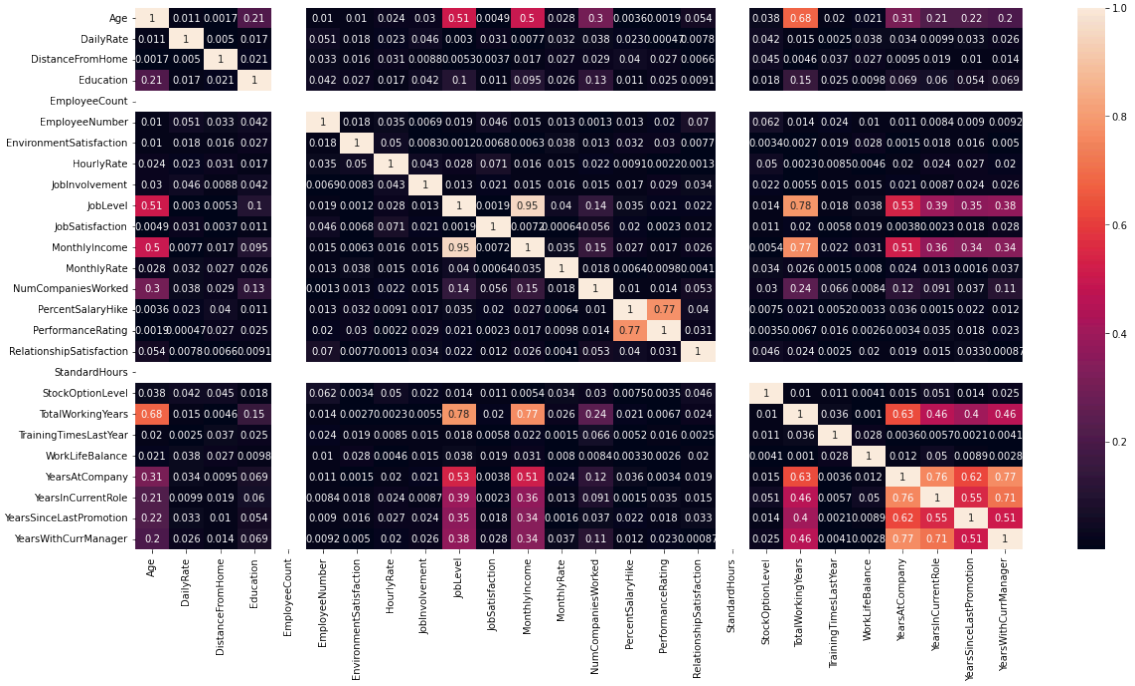


In [33]:
```
# Plot all the close prices
df.plot(figsize=(10, 7))

# Show the legend
plt.legend()

# Define the label for the title of the figure
plt.title("Attritin", fontsize=16)

# Define the labels for x-axis and y-axis
plt.ylabel('Emp_no.', fontsize=14)
plt.xlabel('Rate', fontsize=14)

# Plot the grid lines
plt.grid(which="major", color='k', linestyle='-.', linewidth=0.5)
plt.show()
```



In [40]:
```
plt.figure(figsize=(20, 10))
sns.heatmap(df.corr().abs(),  annot=True)
```

Out[40]: <AxesSubplot:>



In [41]:
```
# Encode categorical variables
data_encoded = pd.get_dummies(df, drop_first=True)
data_encoded
```

Out[41]:

	Age	DailyRate	DistanceFromHome	Education	EmployeeCount	EmployeeNumber
0	41	1102	1	2	1	1
1	49	279	8	1	1	2
2	37	1373	2	2	1	4
3	33	1392	3	4	1	5
4	27	591	2	1	1	7
...	...	...	...	...	...	...
1465	36	884	23	2	1	2061
1466	39	613	6	1	1	2062
1467	27	155	4	3	1	2064
1468	49	1023	2	3	1	2065

In [ ]:

In [ ]: