Menu

Python 3

Not Trusted

- File
  - New NotebookDropdown
    - Python 3
  - Open...
  -
  - Make a Copy...
  - Save as...
  - Rename...
  - Save and Checkpoint`Ctrl-S`
  -
  - Revert to CheckpointDropdown
    - Wednesday, 25 August 2021 19:16
  -
  - Print Preview
  - Download asDropdown
    - AsciiDoc (.asciidoc)
    - HTML (.html)
    - LaTeX (.tex)
    - Markdown (.md)
    - Notebook (.ipynb)
    - PDF via LaTeX (.pdf)
    - reST (.rst)
    - Python (.py)
    - Reveal.js slides (.slides.html)
    - PDF via pyppeteer (.html)
  - Deploy as
  -
  - Trust Notebook
  -
  - Close and Halt
- Edit
  - Cut Cells`x`
  - Copy Cells`c`
  - Paste Cells Above`Shift-V`
  - Paste Cells Below`v`
  - Paste Cells & Replace
  - Delete Cells`D,D`
  - Undo Delete Cells`z`
  -
  - Split Cell`Ctrl-Shift-Minus`
  - Merge Cell Above
  - Merge Cell Below
  -
  - Move Cell Up
  - Move Cell Down
  -
  - Edit Notebook Metadata
  -
  - Find and Replace
  -
  - Cut Cell Attachments
  - Copy Cell Attachments
  - Paste Cell Attachments
  -
  - Insert Image
- View
  - Toggle Header
  - Toggle Toolbar
  - Toggle Line Numbers`Shift-L`
  - Cell Toolbar
    - None
    - Edit Metadata
    - Raw Cell Format
    - Slideshow
    - Attachments
    - Tags
- Insert
  - Insert Cell Above`A`
  - Insert Cell Below`B`
- Cell
  - Run Cells`Ctrl-Enter`
  - Run Cells and Select Below`Shift-Enter`
  - Run Cells and Insert Below`Alt-Enter`
  - Run All
  - Run All Above

- Run All Below
  -
- Cell Type
  - Code**Y**
  - Markdown**M**
  - Raw NBConvert**R**
  -
- Current Outputs
  - Toggle**O**
  - Toggle Scrolling**Shift-O**
  - Clear
- All Output
  - Toggle
  - Toggle Scrolling
  - Clear
- Kernel
  - Interrupt**I,I**
  - Restart**0,0**
  - Restart & Clear Output
  - Restart & Run All
  - Reconnect
  - Shutdown
  -
  - Change kernel
    - Python 3
- Widgets
  - Save Notebook Widget State
  - Clear Notebook Widget State
  - Download Widget State
  - Embed Widgets
- Help
  - User Interface Tour
  - Keyboard Shortcuts**H**
  - Edit Keyboard Shortcuts
  -
  - Notebook Help
  - Markdown
  -
  - Python Reference
  - IPython Reference
  - NumPy Reference
  - SciPy Reference
  - Matplotlib Reference
  - SymPy Reference
  - pandas Reference
  -
  - About

Run

Code

In [35]:

```
#import the libralies
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
%reload_ext autoreload
%autoreload
```

In [8]:

x

```
#load the the data files
#display the data in a table format
df=pd.read_csv("WA_Fn-UseC_-HR-Employee-Attrition.csv")
pd.set_option("display.max_columns", None)
df.head(5)
```

Out[8]:

	Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education	EducationField	EmployeeCount	EmployeeNumbe

	Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education	EducationField	EmployeeCount	EmployeeNumbe
0	41	Yes	Travel_Rarely	1102	Sales	1	2	Life Sciences	1	1
1	49	No	Travel_Frequently	279	Research & Development	8	1	Life Sciences	1	2
2	37	Yes	Travel_Rarely	1373	Research & Development	2	2	Other	1	4
3	33	No	Travel_Frequently	1392	Research & Development	3	4	Life Sciences	1	5
4	27	No	Travel_Rarely	591	Research & Development	2	1	Medical	1	7

In [4]:

```
df.describe()
```

Out[4]:

	Age	DailyRate	DistanceFromHome	Education	EmployeeCount	EmployeeNumber	EnvironmentSatisfaction	HourlyRat
count	1470.000000	1470.000000	1470.000000	1470.000000	1470.0	1470.000000	1470.000000	1470.00000
mean	36.923810	802.485714	9.192517	2.912925	1.0	1024.865306	2.721769	65.891156
std	9.135373	403.509100	8.106864	1.024165	0.0	602.024335	1.093082	20.329428
min	18.000000	102.000000	1.000000	1.000000	1.0	1.000000	1.000000	30.000000
25%	30.000000	465.000000	2.000000	2.000000	1.0	491.250000	2.000000	48.000000
50%	36.000000	802.000000	7.000000	3.000000	1.0	1020.500000	3.000000	66.000000
75%	43.000000	1157.000000	14.000000	4.000000	1.0	1555.750000	4.000000	83.750000
max	60.000000	1499.000000	29.000000	5.000000	1.0	2068.000000	4.000000	100.000000

In [5]:

xxxxxxxxxx

```
df.shape
```

Out[5]:

```
(1470, 35)
```

In [6]:

```
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 [7]:

```
df.isnull().values.any()
```

Out[7]:

```
False
```

In [27]:

```
df=pd.read_csv("WA_Fn-UseC_-HR-Employee-Attrition.csv")

pd.set_option("display.max_columns", None)

df.head(5)
```
Out[27]:

	Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education	EducationField	EmployeeCount	EmployeeNumbe
0	41	Yes	Travel_Rarely	1102	Sales	1	2	Life Sciences	1	1
1	49	No	Travel_Frequently	279	Research & Development	8	1	Life Sciences	1	2
2	37	Yes	Travel_Rarely	1373	Research & Development	2	2	Other	1	4
3	33	No	Travel_Frequently	1392	Research & Development	3	4	Life Sciences	1	5
4	27	No	Travel_Rarely	591	Research & Development	2	1	Medical	1	7

In [42]:

```
categorical_df = df.drop(['Age','DailyRate','Department','WorkLifeBalance','JobInvolvement','DistanceFromHome','Education','EmployeeCount','EmployeeNumber',

categorical_df.head()
```
Out[42]:

	Attrition	BusinessTravel	EducationField	Gender	JobRole	MaritalStatus	OverTime
0	Yes	Travel_Rarely	Life Sciences	Female	Sales Executive	Single	Yes
1	No	Travel_Frequently	Life Sciences	Male	Research Scientist	Married	No
2	Yes	Travel_Rarely	Other	Male	Laboratory Technician	Single	Yes
3	No	Travel_Frequently	Life Sciences	Female	Research Scientist	Married	Yes
4	No	Travel_Rarely	Medical	Male	Laboratory Technician	Married	No

In [57]:

```
df["Attrition"] = df["Attrition"].astype('category')

df["BusinessTravel"] = df["BusinessTravel"].astype('category')

df["Gender"] = df["Gender"].astype('category')

df["EducationField"] = df["EducationField"].astype('category')

df["JobRole"] = df["JobRole"].astype('category')

df["MaritalStatus"] = df["MaritalStatus"].astype('category')

df["OverTime"]=df["OverTime"].astype("category")

df["Department"]=df["Department"].astype("category")

df.dtypes
```
Out[57]:

```
Age                         int64
Attrition                category
BusinessTravel           category
DailyRate                   int64
Department               category
DistanceFromHome            int64
Education                   int64
EducationField           category
EmployeeCount               int64
EmployeeNumber              int64
EnvironmentSatisfaction     int64
Gender                   category
HourlyRate                  int64
JobInvolvement              int64
JobLevel                    int64
JobRole                  category
JobSatisfaction             int64
MaritalStatus            category
MonthlyIncome               int64
MonthlyRate                 int64
NumCompaniesWorked          int64
Over18                     object
OverTime                 category
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 [58]:

```
categorical_df = df.drop(['Age','DailyRate','WorkLifeBalance','JobInvolvement','DistanceFromHome','Education','EmployeeCount','EmployeeNumber','EnvironmentS
```

```
categorical_df.head()
```

Out[58]:

	Attrition	BusinessTravel	Department	EducationField	Gender	JobRole	MaritalStatus	OverTime
0	Yes	Travel_Rarely	Sales	Life Sciences	Female	Sales Executive	Single	Yes
1	No	Travel_Frequently	Research & Development	Life Sciences	Male	Research Scientist	Married	No
2	Yes	Travel_Rarely	Research & Development	Other	Male	Laboratory Technician	Single	Yes
3	No	Travel_Frequently	Research & Development	Life Sciences	Female	Research Scientist	Married	Yes
4	No	Travel_Rarely	Research & Development	Medical	Male	Laboratory Technician	Married	No

In [132]:

```
x


#label encoding

categorical_df["Attrition_cat"]= categorical_df["Attrition"].cat.codes

categorical_df["BusinessTravel_cat"]=categorical_df["BusinessTravel"].cat.codes

categorical_df["OverTime_cat"]=categorical_df["OverTime"].cat.codes

categorical_df["MaritalStatus_cat"]= categorical_df["MaritalStatus"].cat.codes

categorical_df["JobRole_cat"]=categorical_df["JobRole"].cat.codes

categorical_df["Gender_cat"] = categorical_df["Gender"].cat.codes

categorical_df["Department_cat"] = categorical_df["Department"].cat.codes

categorical_df["EducationField_cat"] = categorical_df["EducationField"].cat.codes

categorical_df.head()
```

Out[132]:

	Attrition	BusinessTravel	Department	EducationField	Gender	JobRole	MaritalStatus	OverTime	Attrition_cat	BusinessTravel_cat	Ov
0	Yes	Travel_Rarely	Sales	Life Sciences	Female	Sales Executive	Single	Yes	1	2	1
1	No	Travel_Frequently	Research & Development	Life Sciences	Male	Research Scientist	Married	No	0	1	0
2	Yes	Travel_Rarely	Research & Development	Other	Male	Laboratory Technician	Single	Yes	1	2	1
3	No	Travel_Frequently	Research & Development	Life Sciences	Female	Research Scientist	Married	Yes	0	1	1
4	No	Travel_Rarely	Research & Development	Medical	Male	Laboratory Technician	Married	No	0	2	0

In [77]:

```
plt.subplots(figsize=(20,15))

sns.set(font_scale=0.8)

sns.heatmap(df.corr(),annot=True,cmap="viridis")
```

Out[77]:

```
<AxesSubplot:>
```
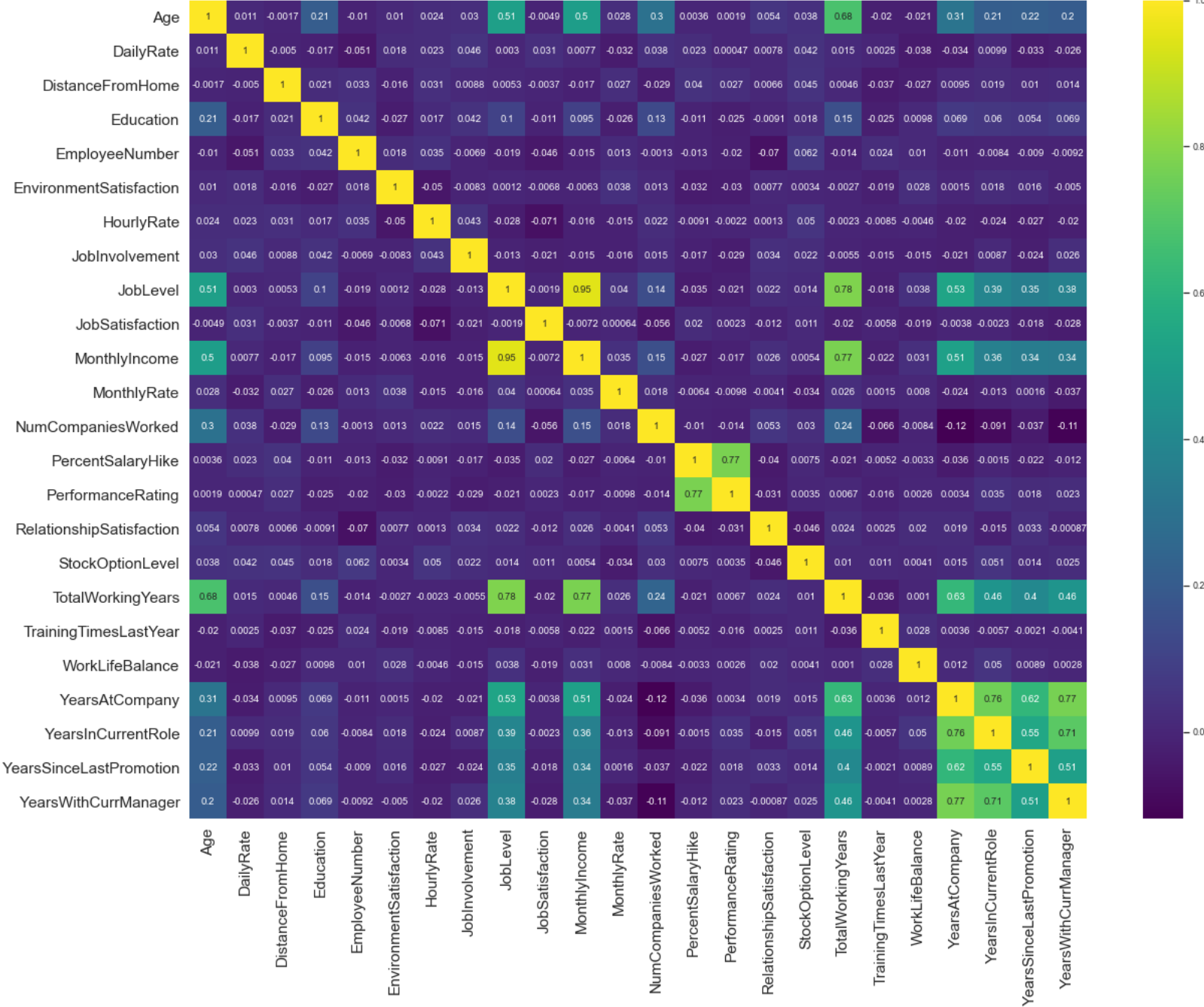
xxxxxxxxxx

There is a good correlation between the variables,joblevel and monthly Income.

There is a strong correlation betweeen the variables;Monthly income and Total working years,Years with current manager and Years in current role,Years with

Type *Markdown* and LaTeX: ^{2}

In [ ]:

```
print()
```
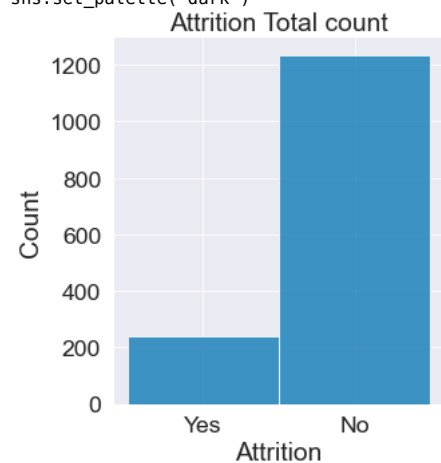
xxxxxxxxxx

```
# factors that lead to employee attrition
```

Uncover the factors that lead to employee attrition¶

In [203]:

```
sns.set_palette("colorblind")
sns.displot(x="Attrition",data=df,)
```

```
plt.title("Attrition Total count")
```

```
sns.set_palette("dark")
```


In [201]:

```
x
```

```
sns.displot(data=df, x="Attrition", hue="MaritalStatus", col="MaritalStatus")
```

```
plt.title("A comparison between Marital Status and Attrition")
```

```
sns.set_palette("Spectral")
```
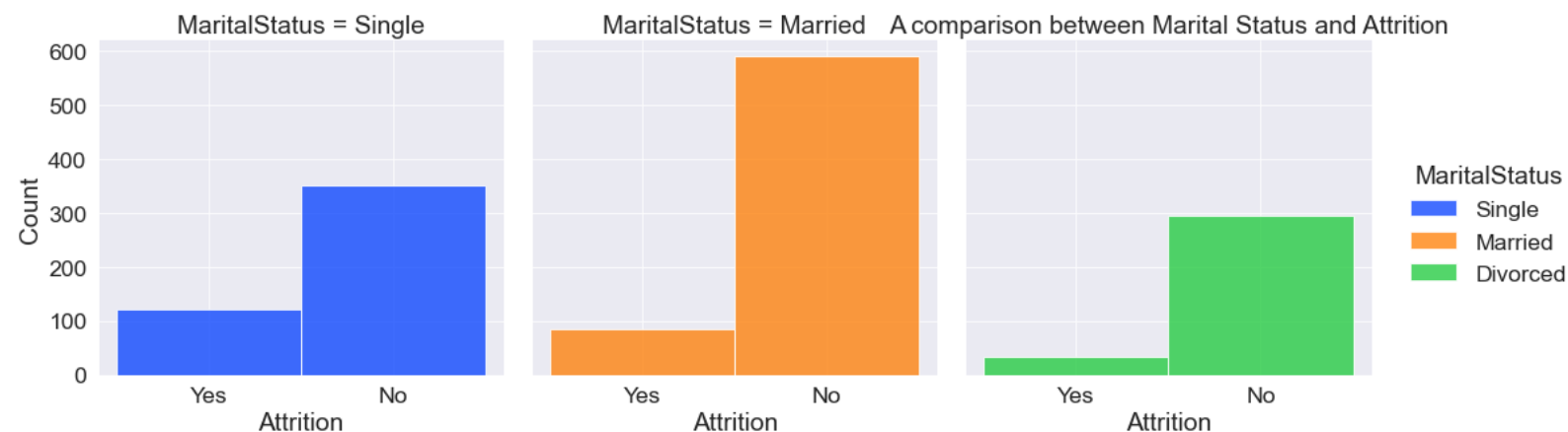

In [197]:

```
xxxxxxxxxx
```

```
g = sns.displot(data=df, x="Attrition", y="RelationshipSatisfaction", col="RelationshipSatisfaction",height=(5))
```

```
g.set_axis_labels("Attrition", "RelationshipSatisfaction")
```

```
sns.set_palette("coolwarm")
```


In [190]:

```
x
```
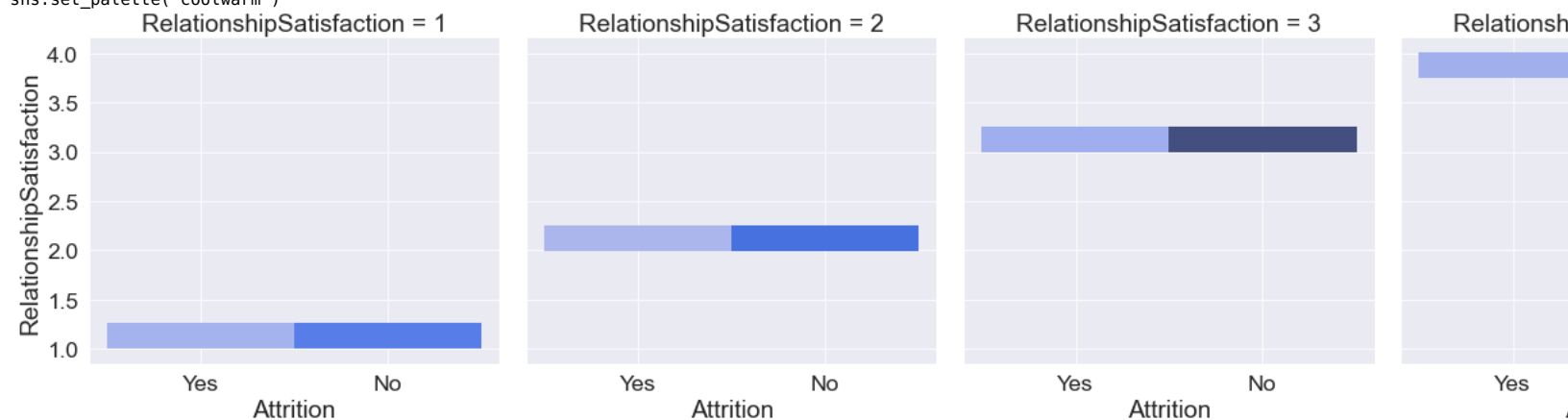
```
df =df=pd.read_csv("WA_Fn-UseC_-HR-Employee-Attrition.csv")
```

```
g=sns.catplot(x="Attrition", y="Education", data=df,kind="box")
```

```
plt.title("A comparison between Education and Attrition")
```

```
sns.set_palette("coolwarm")
```
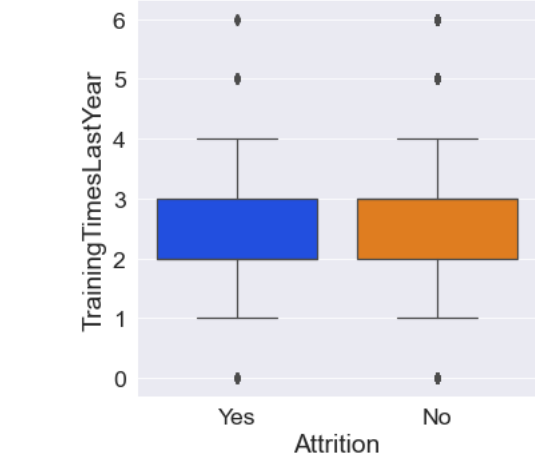
A comparison between Education and Attrition
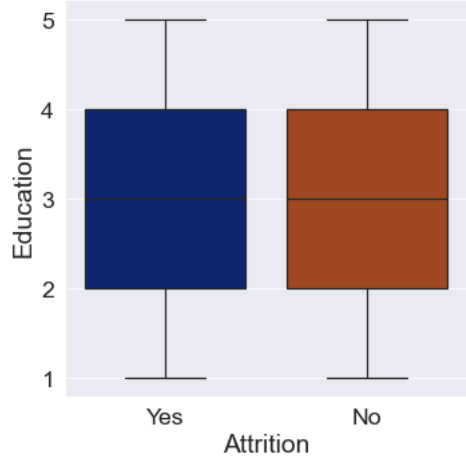
In [188]:

```
df =df=pd.read_csv("WA_Fn-UseC_-HR-Employee-Attrition.csv")
sns.catplot(x="Attrition", y="TrainingTimesLastYear", data=df,kind="box")
plt.title("A comparison between Training times last yearand Attrition")
sns.set_palette("coolwarm")
```

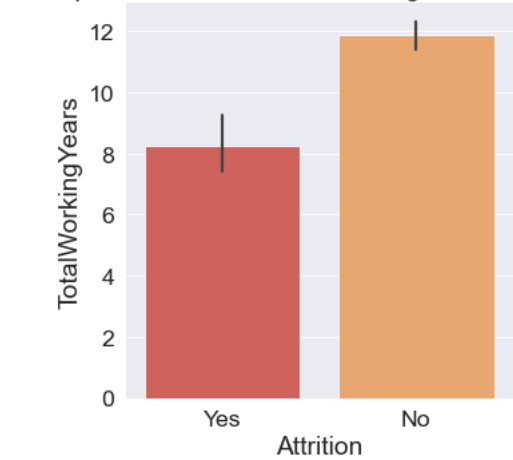A comparison between Training times last yearand Attrition
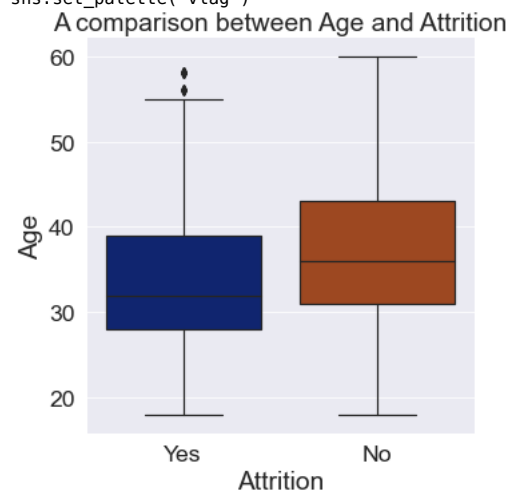

In [181]:

```
x
```

```
df =df=pd.read_csv("WA_Fn-UseC_-HR-Employee-Attrition.csv")
sns.catplot(x="Attrition", y="TotalWorkingYears", data=df,kind="bar")
plt.title("A comparison between Total working Years and Attrition")
sns.set_palette("coolwarm")
```

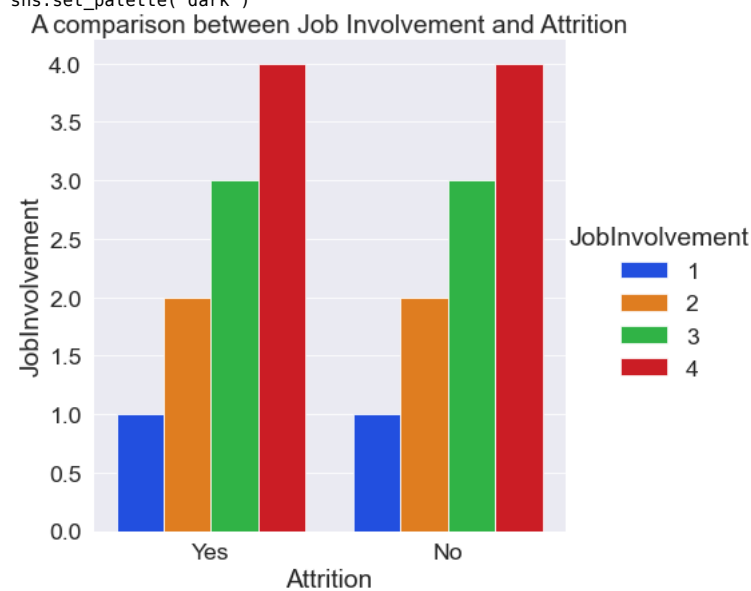A comparison between Total working Years and Attrition

In [183]:

```
df =df=pd.read_csv("WA_Fn-UseC_-HR-Employee-Attrition.csv")

sns.catplot(x="Attrition", y="Age", data=df,kind="box")

plt.title("A comparison between Age and Attrition")

sns.set_palette("vlag")
```
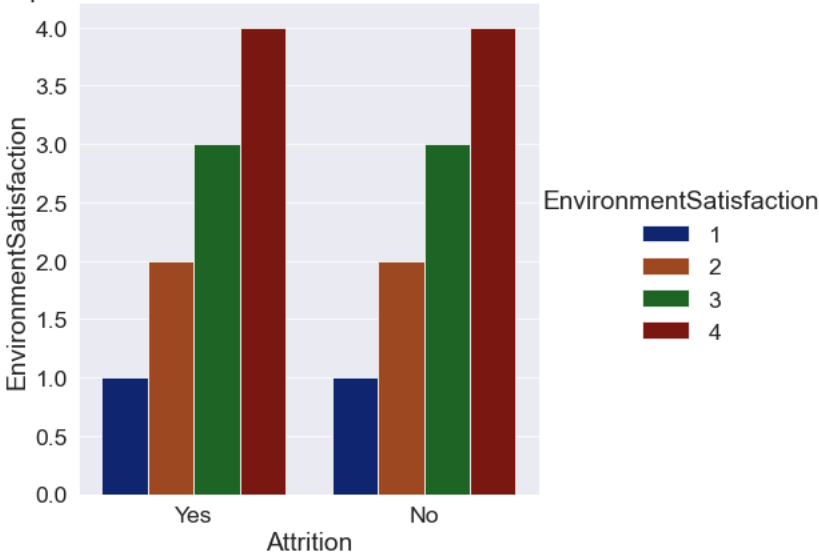

A comparison between Age and Attrition

```
x

df =df=pd.read_csv("WA_Fn-UseC_-HR-Employee-Attrition.csv")

sns.catplot(x="Attrition", y="JobInvolvement",hue="JobInvolvement",data=df,kind="bar",height=6)

plt.title("A comparison between Job Involvement and Attrition")

sns.set_palette("dark")
```


A comparison between Job Involvement and Attrition

```
df =df=pd.read_csv("WA_Fn-UseC_-HR-Employee-Attrition.csv")

sns.catplot(x="Attrition", y="EnvironmentSatisfaction",hue="EnvironmentSatisfaction",data=df,kind="bar",height=6)

plt.title("A comparison between Enviromental Satisfaction and Attrition")

sns.set_palette("bright")
```

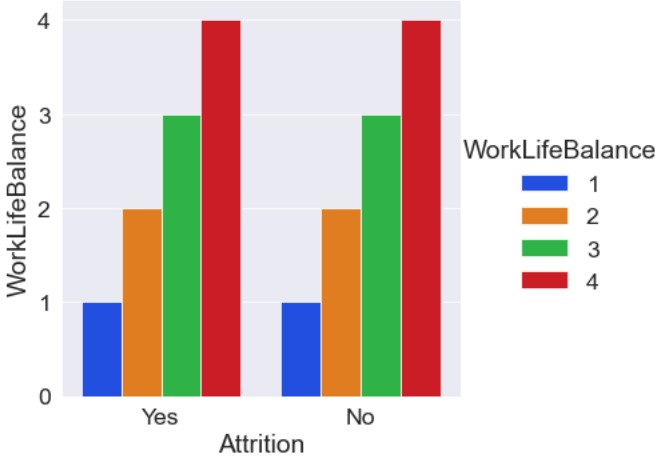## A comparison between Enviromental Satisfaction and Attrition

In [ ]:

In [168]:

```
df=pd.read_csv("WA_Fn-UseC_-HR-Employee-Attrition.csv")
sns.catplot(x="Attrition", y="WorkLifeBalance",hue="WorkLifeBalance",data=df,kind="bar")
plt.title("A comparison between Work life  balance and Attrition")
sns.set_palette("bright")
```

A comparison between Work life balance and Attrition


xxxxxxxxxx

Distance from home by job role and attrition
Type *Markdown* and LaTeX: 2

In [158]:

```
#A  breakdown of distance from home by job role and attrition
import matplotlib.pyplot as plt
df=pd.read_csv("WA_Fn-UseC_-HR-Employee-Attrition.csv")
sns.catplot(x="JobRole", y="DistanceFromHome",hue="Attrition",data=df,kind='box',height=10)
sns.set_palette("bright")
plt.title(" A  breakdown of distance from home by job role and attrition")
plt.xticks(rotation=45)
```
Out[158]:

```
(array([0, 1, 2, 3, 4, 5, 6, 7, 8]),
 [Text(0, 0, 'Sales Executive'),
  Text(1, 0, 'Research Scientist'),
  Text(2, 0, 'Laboratory Technician'),
  Text(3, 0, 'Manufacturing Director'),
  Text(4, 0, 'Healthcare Representative'),
  Text(5, 0, 'Manager'),
  Text(6, 0, 'Sales Representative'),
  Text(7, 0, 'Research Director'),
  Text(8, 0, 'Human Resources')])
```

A breakdown of distance from home by job role and attrition
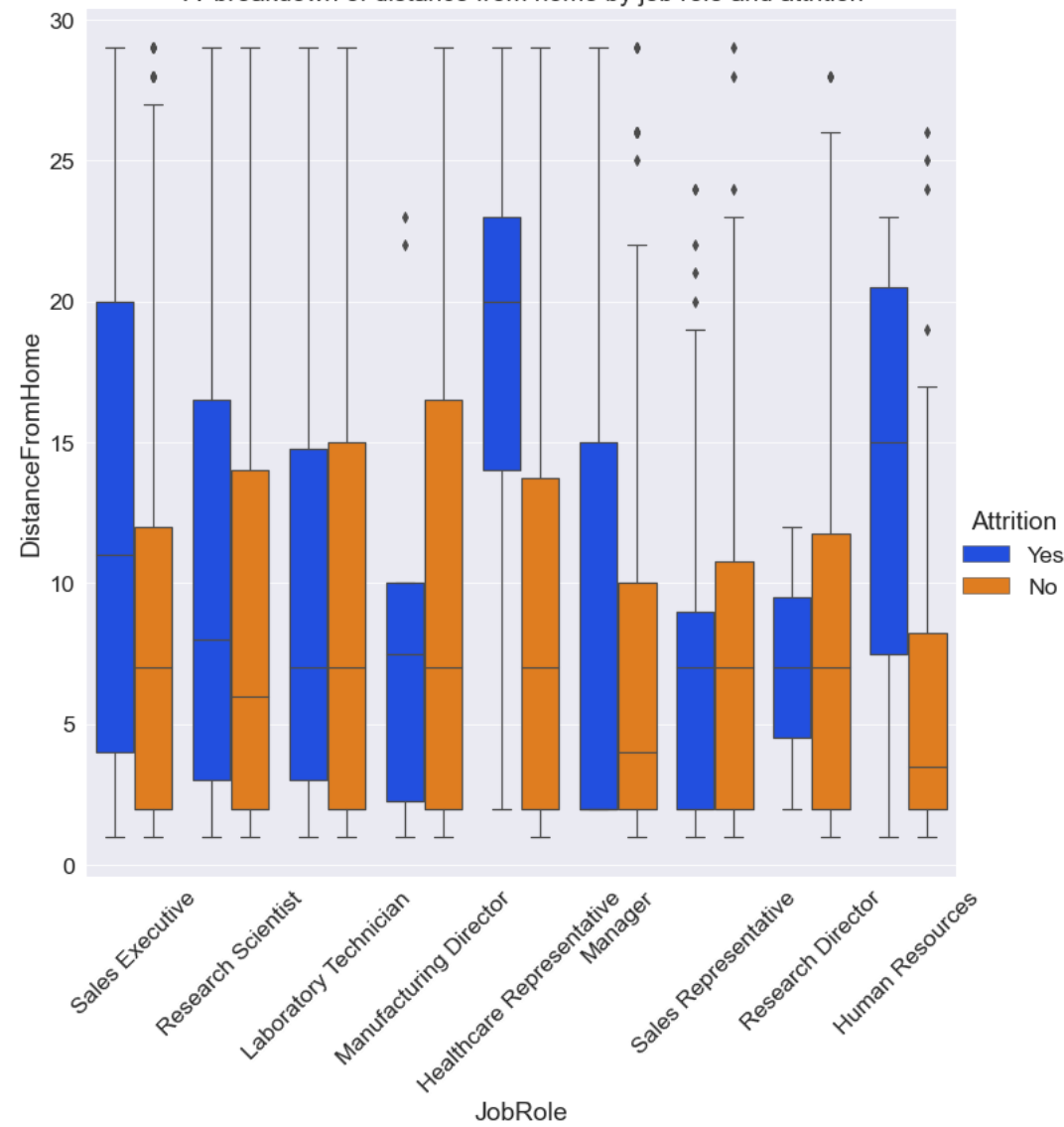
In [110]:

```
x

import matplotlib.pyplot as plt

df=pd.read_csv("WA_Fn-UseC_-HR-Employee-Attrition.csv")

plt.figure(figsize=(6, 4))

sns.set_context("paper", font_scale=2)

sns.catplot(x="Education", y="MonthlyIncome",hue="Attrition",data=df,kind='boxen',height=10)

plt.title(" A Comparison between Monthly income by Education and Attrition")

sns.set_palette("colorblind")

plt.xticks(rotation=90)
```
Out[110]:

```
(array([0, 1, 2, 3, 4]),
 [Text(0, 0, '1'),
  Text(1, 0, '2'),
  Text(2, 0, '3'),
  Text(3, 0, '4'),
  Text(4, 0, '5')])
```

<Figure size 432x288 with 0 Axes>

A Comparison between Monthly income by Education and Attrition
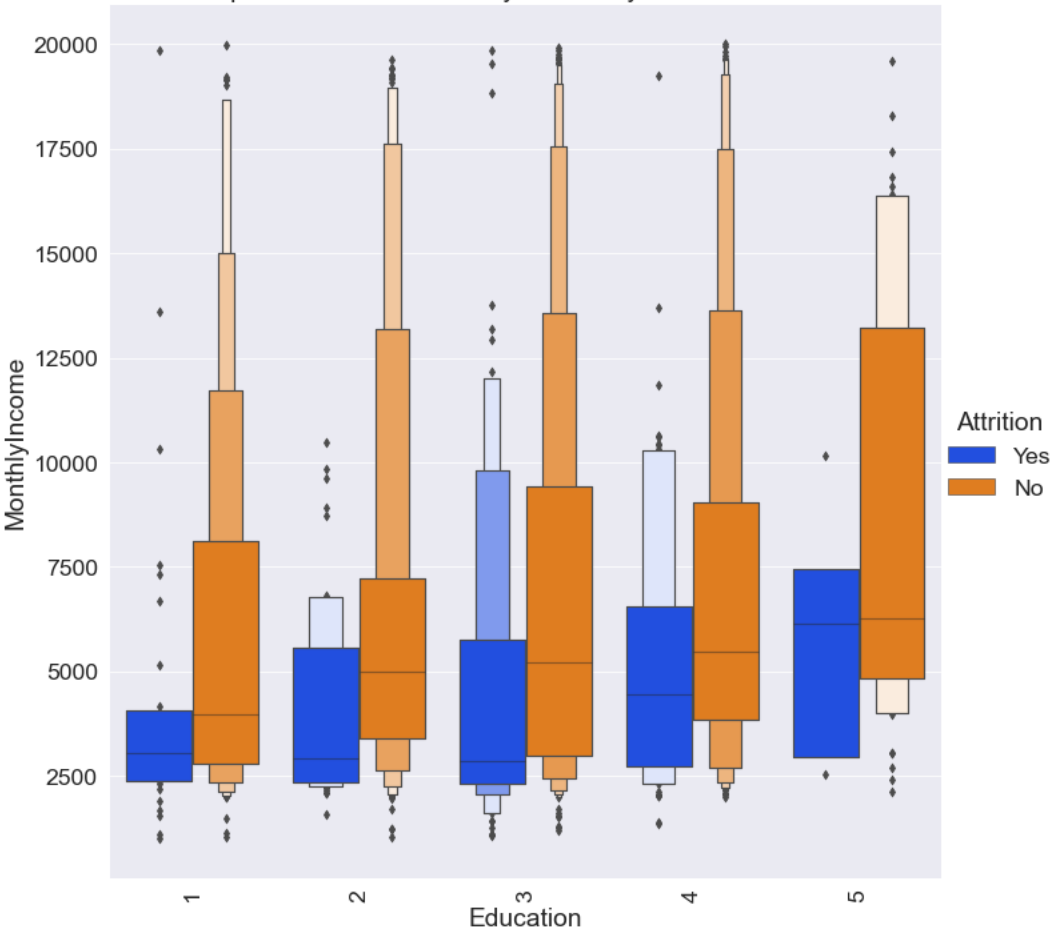
In [144]:

```
x

new_df=categorical_df

new_df.head()
```
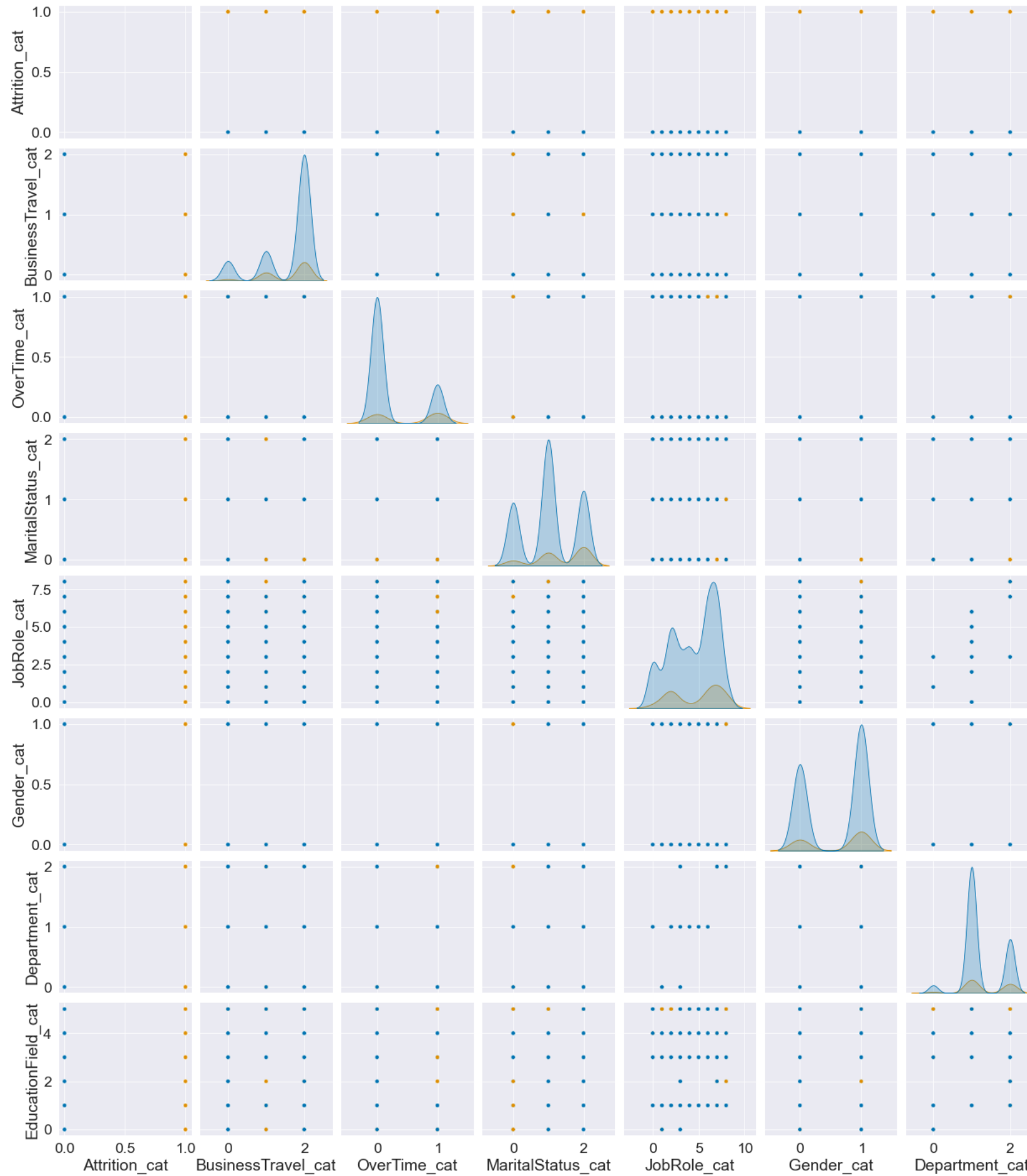
Out[144]:

	Attrition	BusinessTravel	Department	EducationField	Gender	JobRole	MaritalStatus	OverTime	Attrition_cat	BusinessTravel_cat	Ov
0	Yes	Travel_Rarely	Sales	Life Sciences	Female	Sales Executive	Single	Yes	1	2	1
1	No	Travel_Frequently	Research & Development	Life Sciences	Male	Research Scientist	Married	No	0	1	0
2	Yes	Travel_Rarely	Research & Development	Other	Male	Laboratory Technician	Single	Yes	1	2	1
3	No	Travel_Frequently	Research & Development	Life Sciences	Female	Research Scientist	Married	Yes	0	1	1
4	No	Travel_Rarely	Research & Development	Medical	Male	Laboratory Technician	Married	No	0	2	0

In [148]:

```
#Question 3

sns.pairplot(new_df, hue="Attrition")
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:306: UserWarning: Dataset has 0 variance; skipping density estimate.
  warnings.warn(msg, UserWarning)
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:306: UserWarning: Dataset has 0 variance; skipping density estimate.
  warnings.warn(msg, UserWarning)
```

Out[148]:

```
<seaborn.axisgrid.PairGrid at 0x1be6a40bfd0>
```