In [1]:
```
import pandas as p
import numpy as n
import matplotlib.pyplot as plt
"%matplotlib inline"
import seaborn as s
import warnings
warnings.filterwarnings("ignore")
import matplotlib.pyplot as plt
```

In [2]:
```
data = p.read_clipboard()
data
```

Out[2]:

	EmpNumber	Age	Gender	EducationBackground	MaritalStatus	EmpDepartment	EmpJobRole
0	E1001007	40	Male	Life Sciences	Married	Sales	Sales Executive
1	E1001025	30	Male	Marketing	Divorced	Sales	Sales Executive
2	E1001054	52	Male	Marketing	Married	Sales	Manager
3	E1001059	25	Female	Medical	Single	Sales	Sales Executive
4	E1001064	34	Male	Other	Single	Sales	Sales Executive
...	...	...	...	...	...	...	...
81	E100724	33	Male	Marketing	Single	Sales	Sales Executive
82	E100891	31	Male	Life Sciences	Divorced	Sales	Manager
83	E100985	25	Male	Life Sciences	Married	Sales	Sales Executive
84	E100987	38	Female	Marketing	Single	Sales	Sales Executive
85	E100992	27	Female	Medical	Divorced	Sales	Sales Executive

86 rows × 28 columns

In [3]:
```
data.describe()
```

Out[3]:

	Age	DistanceFromHome	EmpEducationLevel	EmpEnvironmentSatisfaction	EmpHourly
count	86.000000	86.000000	86.000000	86.000000	86.00
mean	37.209302	8.918605	2.906977	3.151163	65.02
std	9.577076	8.569735	1.047438	0.926949	20.84
min	18.000000	1.000000	1.000000	1.000000	30.00
25%	31.000000	2.000000	2.000000	3.000000	46.00
50%	37.000000	6.500000	3.000000	3.000000	67.50
75%	43.000000	12.000000	4.000000	4.000000	79.00
max	60.000000	29.000000	5.000000	4.000000	100.00

In [4]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 86 entries, 0 to 85
Data columns (total 28 columns):
 #   Column                          Non-Null Count  Dtype 
---  ------                          --------------  ----- 
 0   EmpNumber                       86 non-null     object
 1   Age                             86 non-null     int64 
 2   Gender                          86 non-null     object
 3   EducationBackground             86 non-null     object
 4   MaritalStatus                   86 non-null     object
 5   EmpDepartment                   86 non-null     object
 6   EmpJobRole                      86 non-null     object
 7   BusinessTravelFrequency         86 non-null     object
 8   DistanceFromHome                86 non-null     int64 
 9   EmpEducationLevel               86 non-null     int64 
 10  EmpEnvironmentSatisfaction      86 non-null     int64 
 11  EmpHourlyRate                   86 non-null     int64 
 12  EmpJobInvolvement               86 non-null     int64 
 13  EmpJobLevel                     86 non-null     int64 
 14  EmpJobSatisfaction              86 non-null     int64 
 15  NumCompaniesWorked              86 non-null     int64 
 16  OverTime                        86 non-null     object
 17  EmpLastSalaryHikePercent        86 non-null     int64 
 18  EmpRelationshipSatisfaction     86 non-null     int64 
 19  TotalWorkExperienceInYears      86 non-null     int64 
 20  TrainingTimesLastYear           86 non-null     int64 
 21  EmpWorkLifeBalance              86 non-null     int64 
 22  ExperienceYearsAtThisCompany    86 non-null     int64 
 23  ExperienceYearsInCurrentRole    86 non-null     int64 
 24  YearsSinceLastPromotion         86 non-null     int64 
 25  YearsWithCurrManager            86 non-null     int64 
 26  Attrition                       86 non-null     object
 27  PerformanceRating               86 non-null     int64 
dtypes: int64(19), object(9)
memory usage: 18.9+ KB
```

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

Out[5]:
```
(86, 28)
```

In [6]:
```
data.head()
```

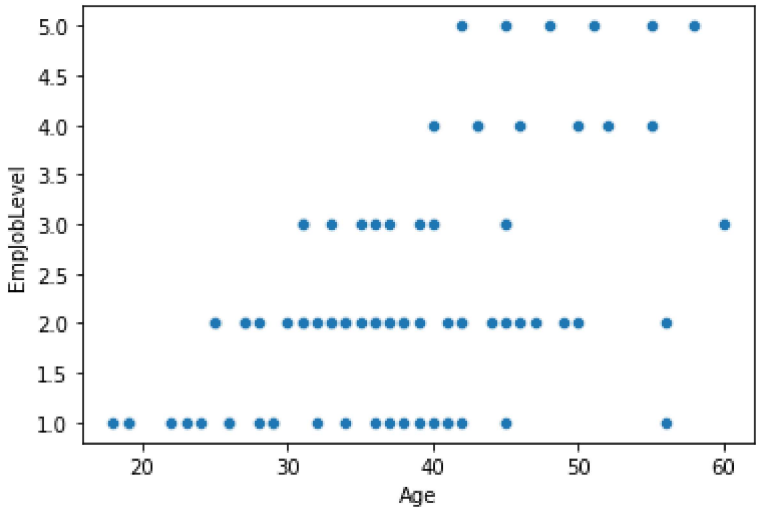
Out[6]:

	EmpNumber	Age	Gender	EducationBackground	MaritalStatus	EmpDepartment	EmpJobRole
0	E1001007	40	Male	Life Sciences	Married	Sales	Sales Executive
1	E1001025	30	Male	Marketing	Divorced	Sales	Sales Executive
2	E1001054	52	Male	Marketing	Married	Sales	Manager
3	E1001059	25	Female	Medical	Single	Sales	Sales Executive
4	E1001064	34	Male	Other	Single	Sales	Sales Executive

5 rows × 28 columns

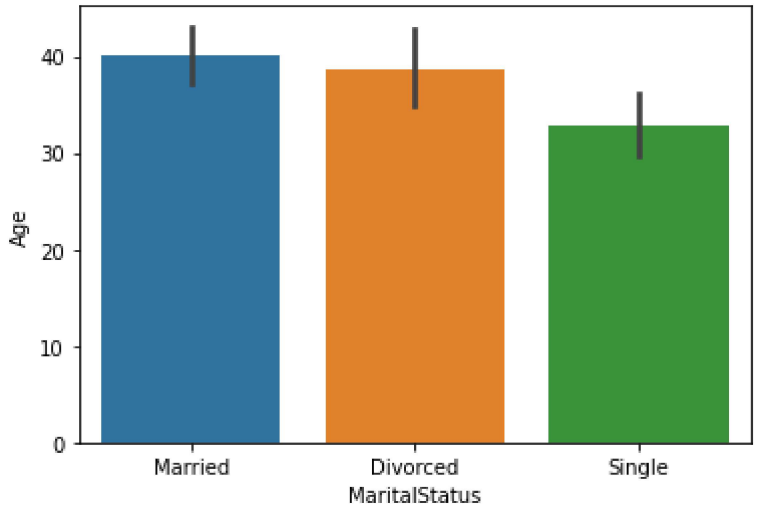
In [7]:
```
s.scatterplot(data=data,x="Age",y='EmpJobLevel')
```

Out[7]:
```
<AxesSubplot:xlabel='Age', ylabel='EmpJobLevel'>
```



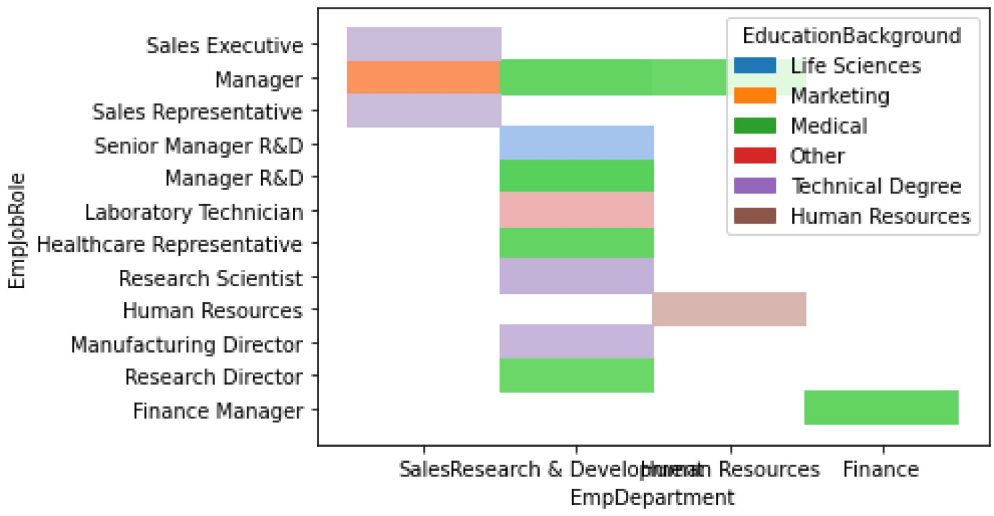
In [8]:
```
s.barplot(data=data,x='MaritalStatus',y='Age')
```

Out[8]:
```
<AxesSubplot:xlabel='MaritalStatus', ylabel='Age'>
```



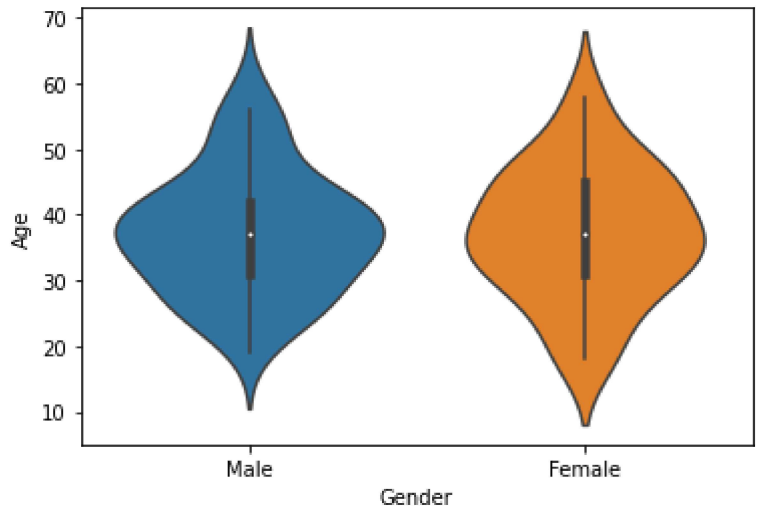
In [9]:
```
s.histplot(data=data,x='EmpDepartment',y='EmpJobRole',hue='EducationBackground')
```

Out[9]:
```
<AxesSubplot:xlabel='EmpDepartment', ylabel='EmpJobRole'>
```



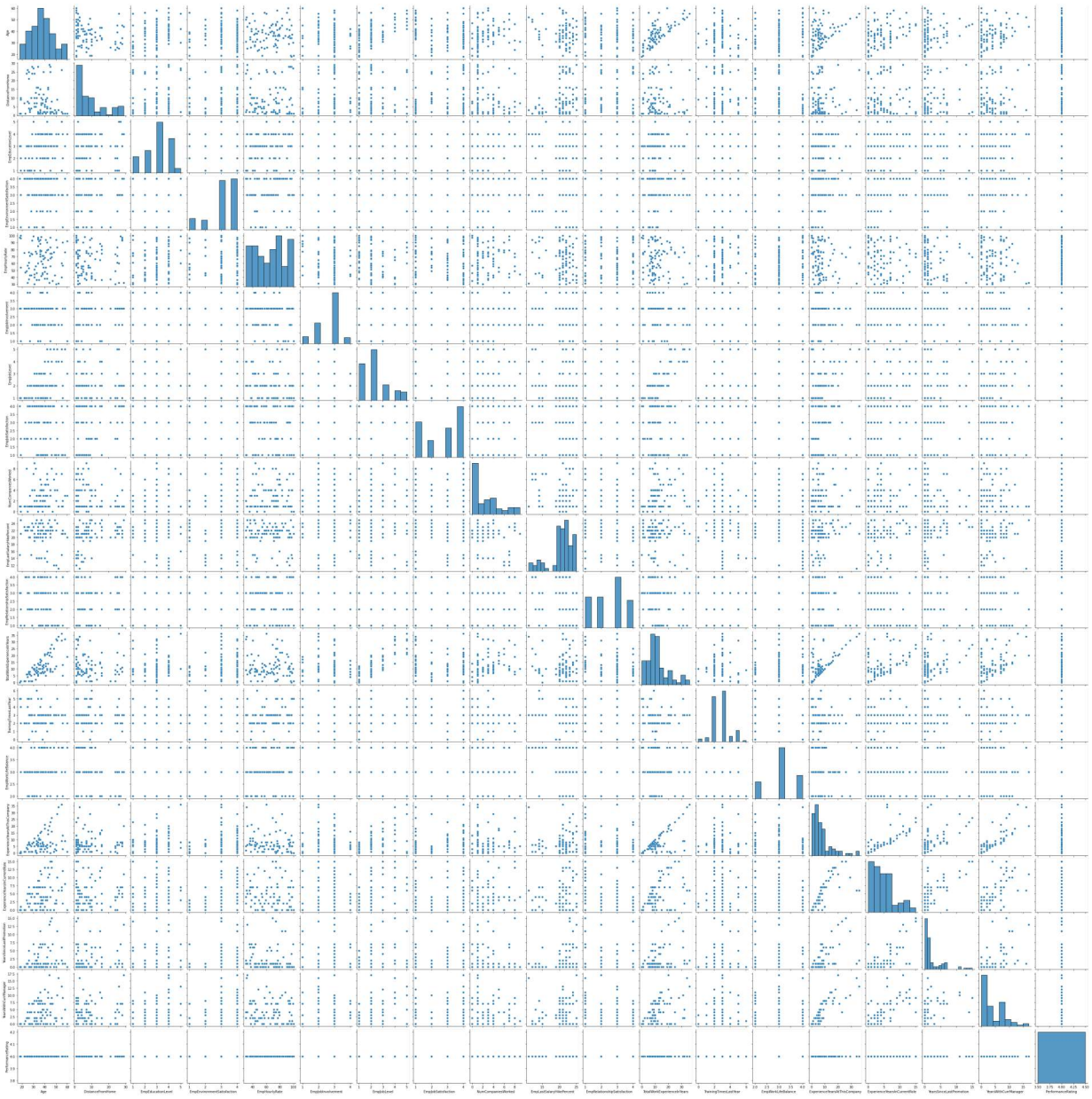
In [10]:
```
s.violinplot(data=data,x='Gender',y='Age',orient="v")
```

Out[10]:
```
<AxesSubplot:xlabel='Gender', ylabel='Age'>
```



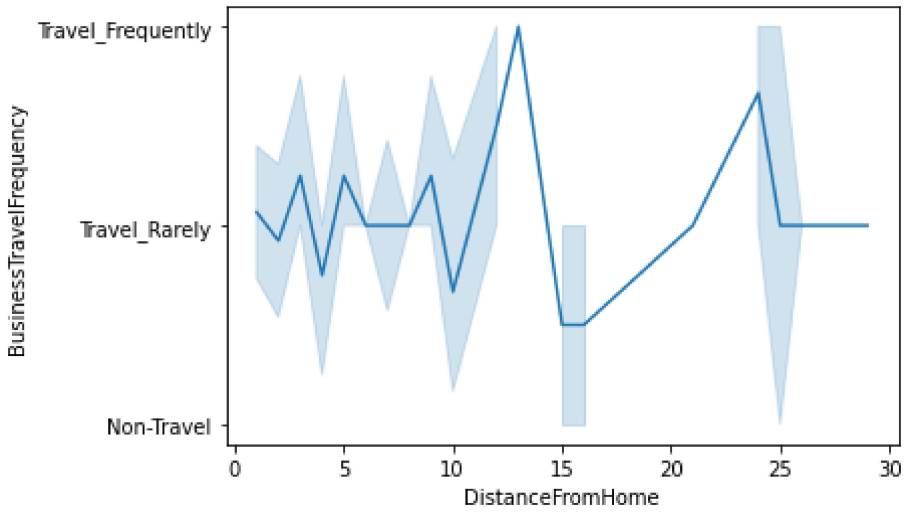
In [11]:
```
s.pairplot(data)
```

Out[11]:
```
<seaborn.axisgrid.PairGrid at 0x2d273e68a60>
```



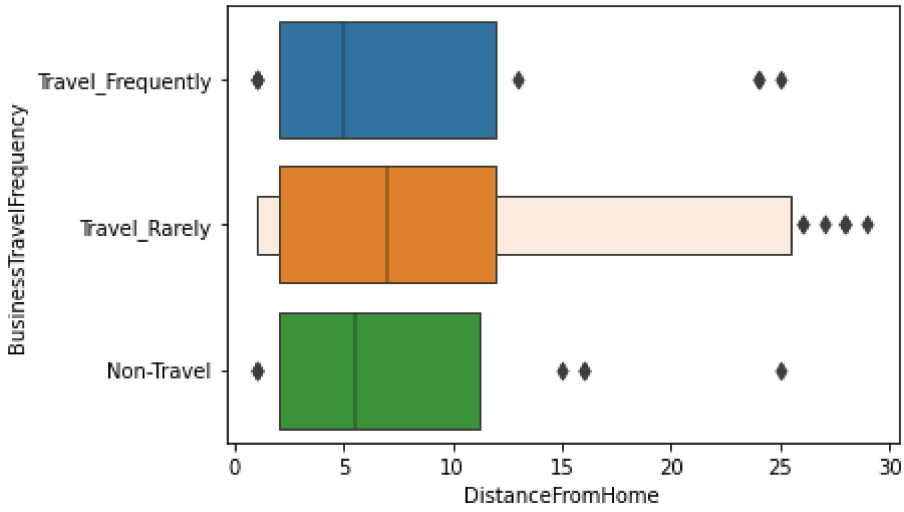
In [12]:
```
s.lineplot(data=data,x='DistanceFromHome',y='BusinessTravelFrequency')
```

Out[12]:
```
<AxesSubplot:xlabel='DistanceFromHome', ylabel='BusinessTravelFrequency'>
```



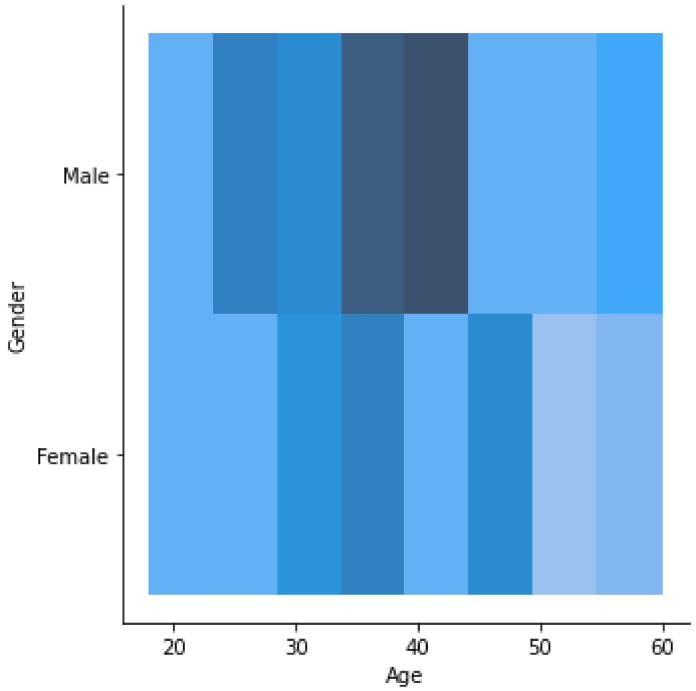
In [13]:
```
s.boxenplot(data=data,x='DistanceFromHome',y='BusinessTravelFrequency',orient='h'
```

Out[13]:
```
<AxesSubplot:xlabel='DistanceFromHome', ylabel='BusinessTravelFrequency'>
```



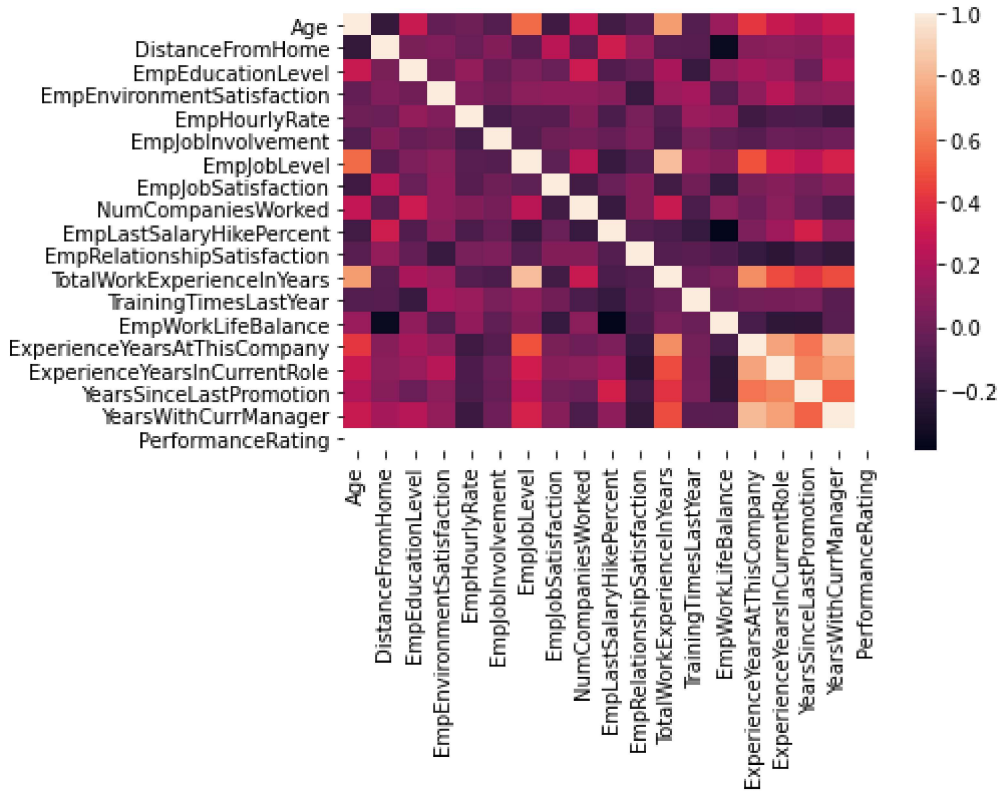
In [14]:
```
s.displot(data=data,x='Age',y='Gender')
```

Out[14]:
```
<seaborn.axisgrid.FacetGrid at 0x2d202b9d7f0>
```



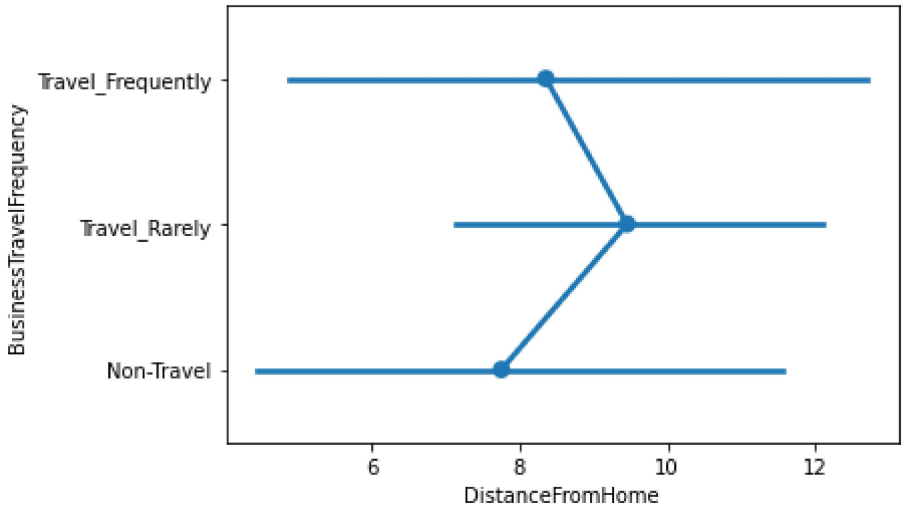
In [15]:
```
s.heatmap(data=data.corr())
```

Out[15]: <AxesSubplot:>



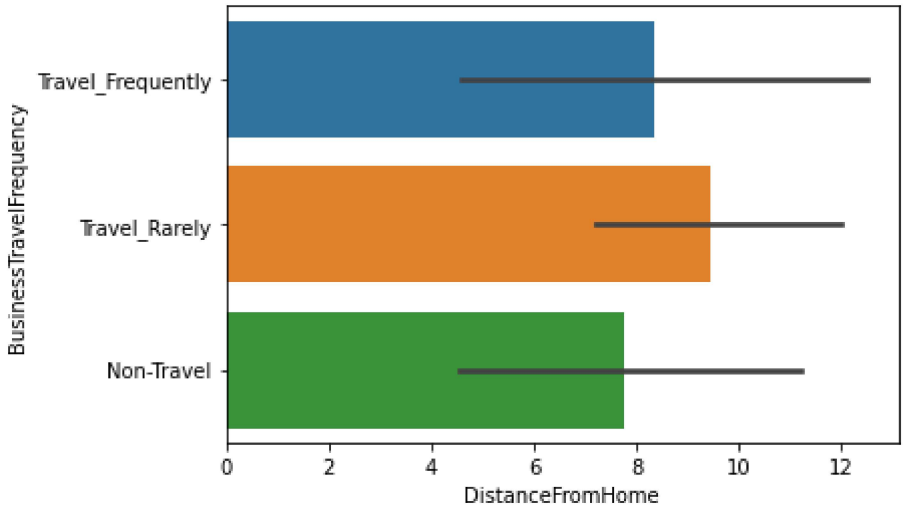
In [16]:
```
s.pointplot(data=data,x='DistanceFromHome',y='BusinessTravelFrequency')
```

Out[16]: <AxesSubplot:xlabel='DistanceFromHome', ylabel='BusinessTravelFrequency'>



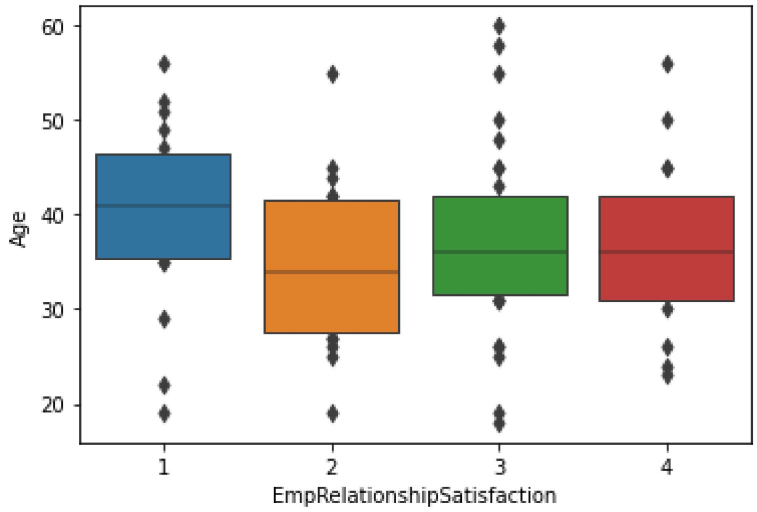
In [17]:
```
s.barplot(data=data,x='DistanceFromHome',y='BusinessTravelFrequency')
```

Out[17]: <AxesSubplot:xlabel='DistanceFromHome', ylabel='BusinessTravelFrequency'>



In [19]:
```
s.boxenplot(data=data,x='EmpRelationshipSatisfaction',y='Age')
```

Out[19]:
```
<AxesSubplot:xlabel='EmpRelationshipSatisfaction', ylabel='Age'>
```



In [ ]:
```

```