Seaborn

```python
import seaborn as sns
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np

mydata={ 'Name' : ['Ram','Sam','Joe','Asha'],
         'Age'  : [23,22,26,47],
        'Salary' : [34000,2000,30600,5000],
        'Exp':[2,1,3,10],
       }

pd1=pd.DataFrame(mydata)
print(pd1.head())

    Name  Age  Salary  Exp
0    Ram   23   34000    2
1    Sam   22    2000    1
2    Joe   26   30600    3
3   Asha   47    5000   10
```
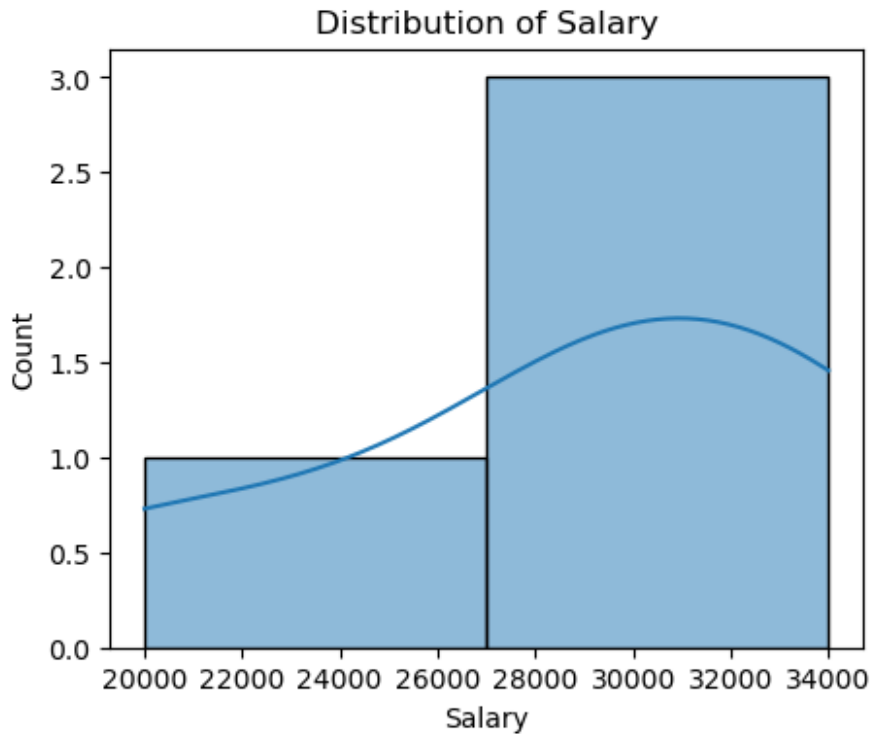
1.Histogram

```python
plt.figure(figsize=(5,4))
sns.histplot(pd1['Salary'],kde=True,bins=2)
plt.title('Distribution of Salary')
plt.show()

C:\Users\DELL\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119:
FutureWarning: use_inf_as_na option is deprecated and will be removed
in a future version. Convert inf values to NaN before operating
instead.
  with pd.option_context('mode.use_inf_as_na', True):
```

Distribution of Salary

1. Negative Skew,Larger Salary found in below average
   2.No outlier Detected 3.Majority Salary values found between 27000 to 34000 4.The average salary value is around 31000

Correlation Matrix(Heat Map)

```
#Step1 : filter numerical dat
ndf=pd1.select_dtypes(include=['number'])
ndf.head()

    Age  Salary  Exp
0    23   34000    2
1    22    2000    1
2    26   30600    3
3    47    5000   10

#Step 2: Correlation graph
plt.figure(figsize=(6,5))
sns.heatmap(ndf.corr(),cmap='coolwarm',annot=True)
plt.title('Correlation Between Age,Salary and Experience')
plt.show()
```
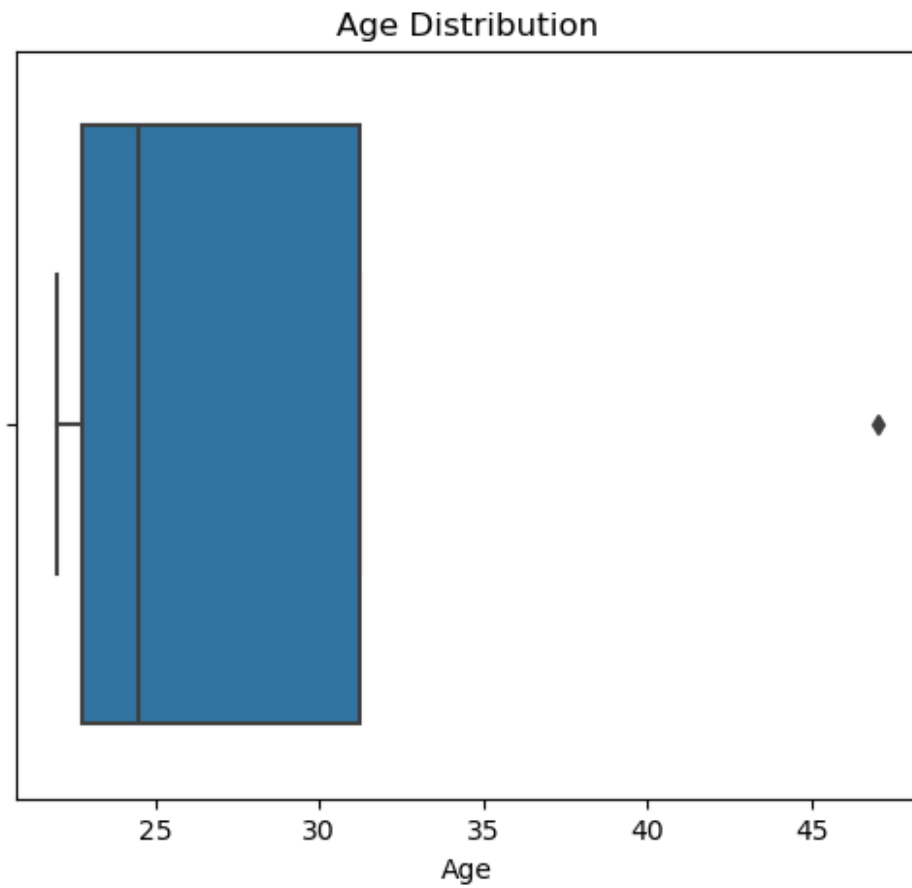
Correlation Between Age,Salary and Experience

1. Age and Experience are most correlated values.
2. Age and Salary are not correlated values

Box plot

```
plt.figure(figsize=(6,5))
sns.boxplot(x=pd1['Age'])
plt.title('Age Distribution')
plt.show()
```

Age Distribution

```python
pd2={ 'Temp':[21,47,39,22,31,33,29,26,27,25,49,46]}

plt.figure(figsize=(6,5))
sns.boxplot(x=pd2['Temp'])
plt.title('Temperature Distribution')
plt.show()
```

Temperature Distribution
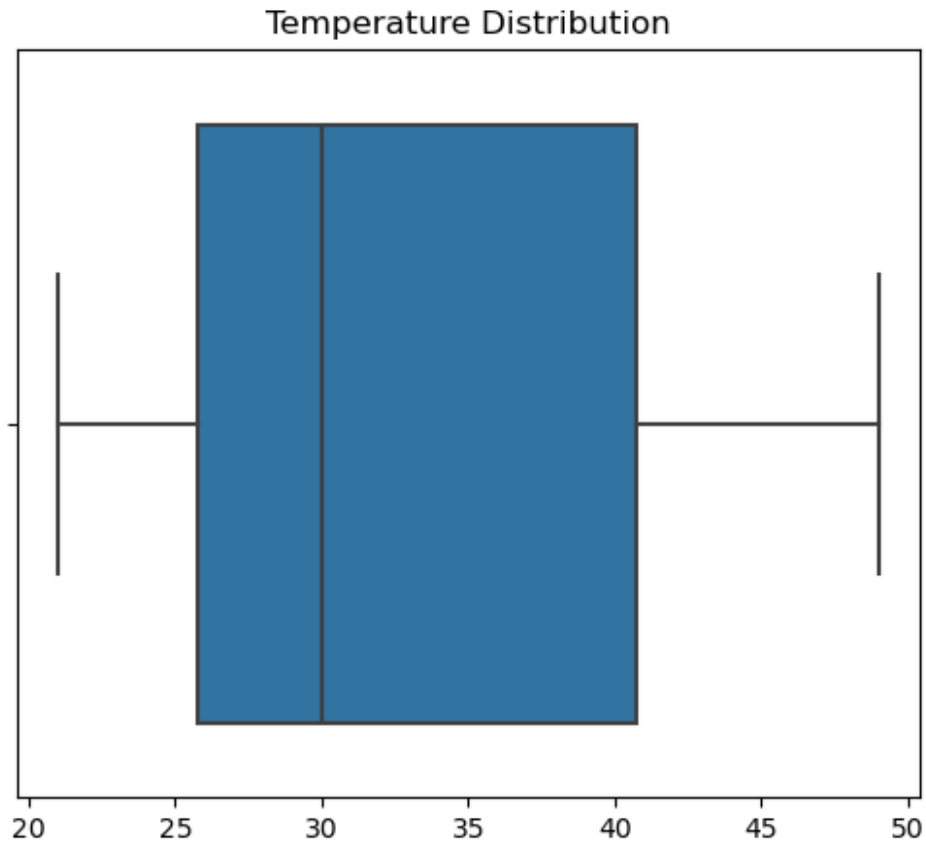
1.Average age values is around 30. 2.There is no abnormal value. 3.The lower bound is around 21 and upper bound is around 49.

Count Plot

```python
mydata1={ 'Name' : ['Ram','Sam','Joe','Asha'],
         'Age'   : [23,22,26,47],
        'Salary' : [34000,2000,30600,5000],
        'Exp':[2,3,3,10],
        'G' : ['M','F','M','M']
      }
df2=pd.DataFrame(mydata1)

plt.figure(figsize=(6,5))
sns.countplot(x=df2['Exp'],palette='bright',hue=df2['G'])
plt.title('Count of Exp')
plt.show()
```

Count of Exp
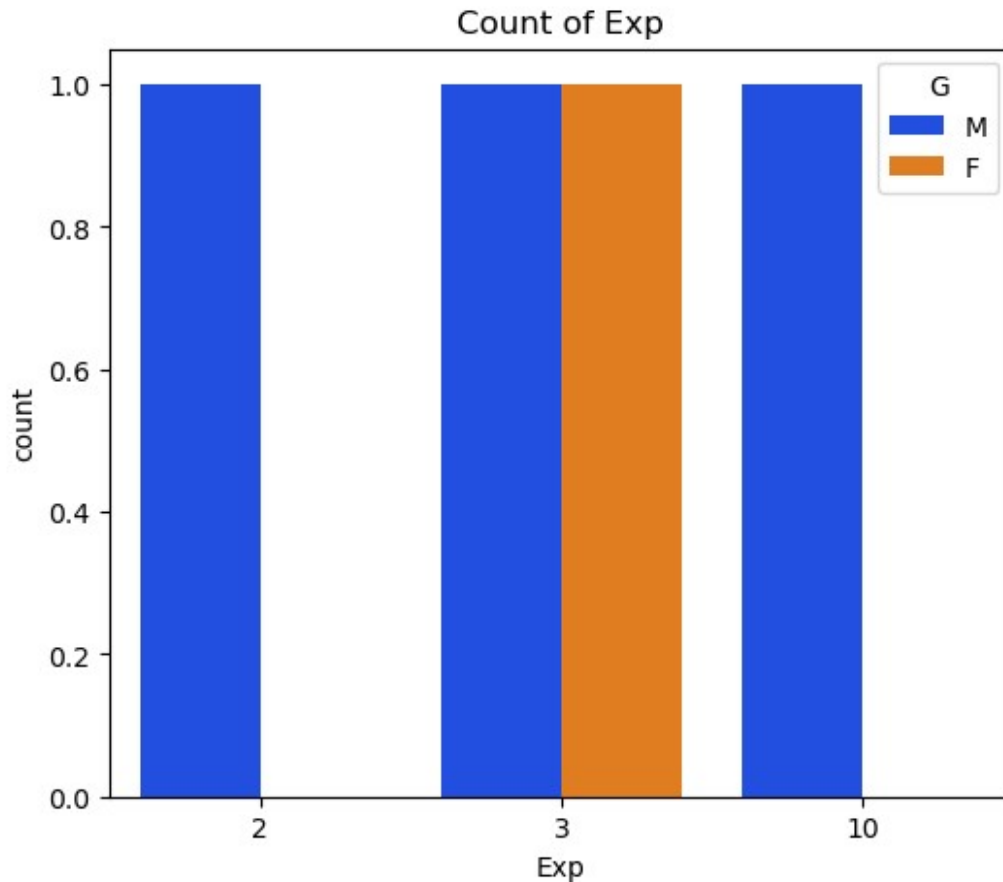
1.The male gender is given by blue color and female by yellow. 2.There is one value of experience whose value is 2. 3.There is one male and female whose experience value is 3. 4.There is only one male with experience 10.

Pair Plot

```
sns.pairplot(df2,hue='G')

C:\Users\DELL\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119:
FutureWarning: use_inf_as_na option is deprecated and will be removed
in a future version. Convert inf values to NaN before operating
instead.
  with pd.option_context('mode.use_inf_as_na', True):
C:\Users\DELL\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119:
FutureWarning: use_inf_as_na option is deprecated and will be removed
in a future version. Convert inf values to NaN before operating
instead.
  with pd.option_context('mode.use_inf_as_na', True):
C:\Users\DELL\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119:
FutureWarning: use_inf_as_na option is deprecated and will be removed
in a future version. Convert inf values to NaN before operating
instead.
  with pd.option_context('mode.use_inf_as_na', True):
```
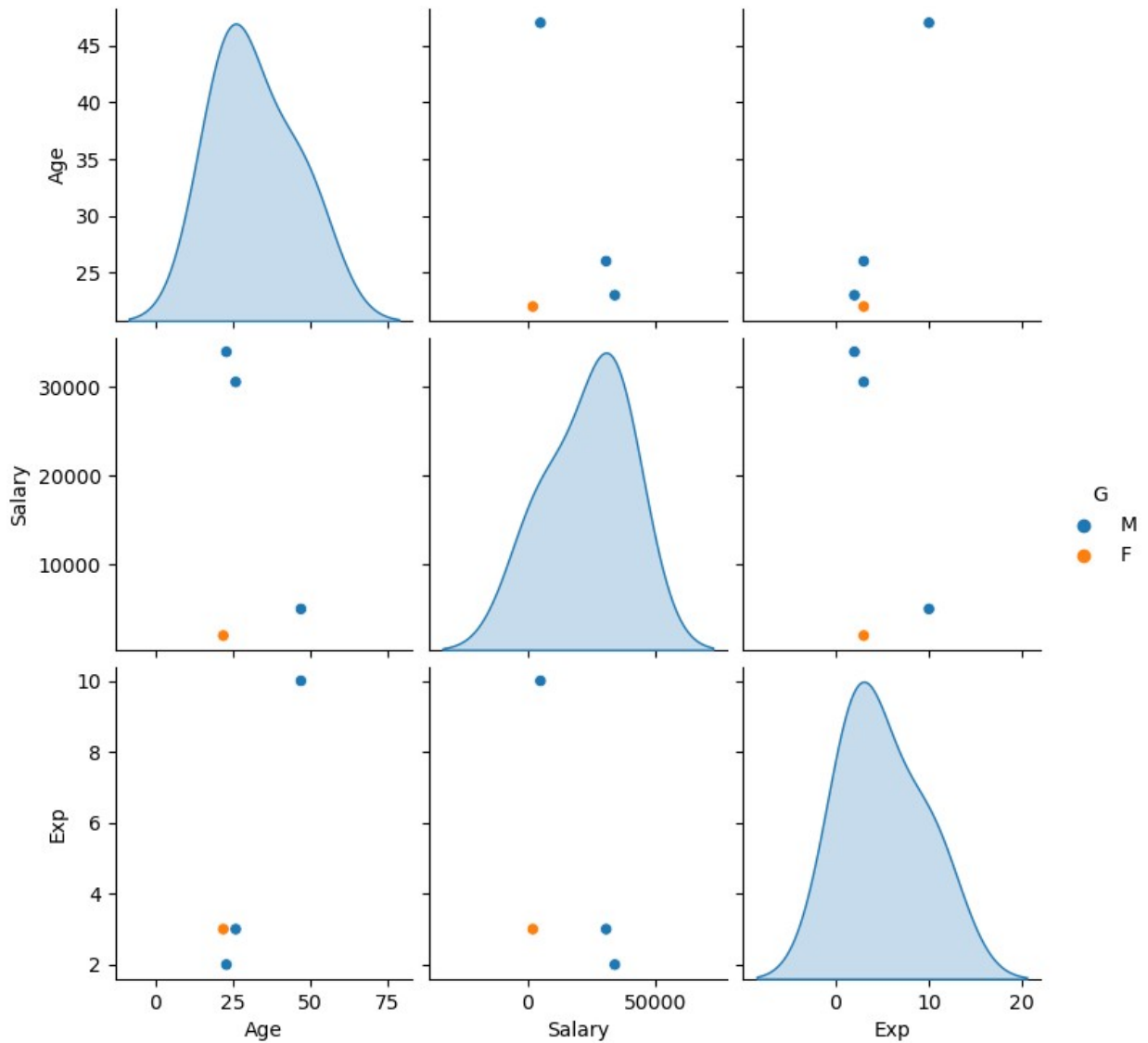
```
<seaborn.axisgrid.PairGrid at 0x202360ae450>
```


1.The Age-Age , Salary-Salary , Experience-Experience are highly related. 2.The male and female gender is differentiated by colors.