Lab 8- Logistic Regression

In [ ]:
```
N01419700
Harini Rajarathinam
```

Import Libraries

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

Get the Data

**Read in the Import Libraries.csv file **

In [2]:
```
data = pd.read_csv(r'C:\Users\rcher\Documents\Humber work\Semester 2\Intro to Dat
data.head()
```

Out[2]:

	User ID	Gender	Age	EstimatedSalary	Purchased
0	15624510	Male	19	19000	0
1	15810944	Male	35	20000	0
2	15668575	Female	26	43000	0
3	15603246	Female	27	57000	0
4	15804002	Male	19	76000	0

Check the head of data

In [3]:
```
data.isnull().sum()
```

Out[3]:
```
User ID            0
Gender             0
Age                0
EstimatedSalary    0
Purchased          0
dtype: int64
```

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

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 400 entries, 0 to 399
Data columns (total 5 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   User ID          400 non-null    int64
 1   Gender           400 non-null    object
 2   Age              400 non-null    int64
 3   EstimatedSalary  400 non-null    int64
 4   Purchased        400 non-null    int64
dtypes: int64(4), object(1)
memory usage: 15.8+ KB
```

In [5]:
```
data.describe()
```

Out[5]:

	User ID	Age	EstimatedSalary	Purchased
count	4.000000e+02	400.000000	400.000000	400.000000
mean	1.569154e+07	37.655000	69742.500000	0.357500
std	7.165832e+04	10.482877	34096.960282	0.479864
min	1.556669e+07	18.000000	15000.000000	0.000000
25%	1.562676e+07	29.750000	43000.000000	0.000000
50%	1.569434e+07	37.000000	70000.000000	0.000000
75%	1.575036e+07	46.000000	88000.000000	1.000000
max	1.581524e+07	60.000000	150000.000000	1.000000

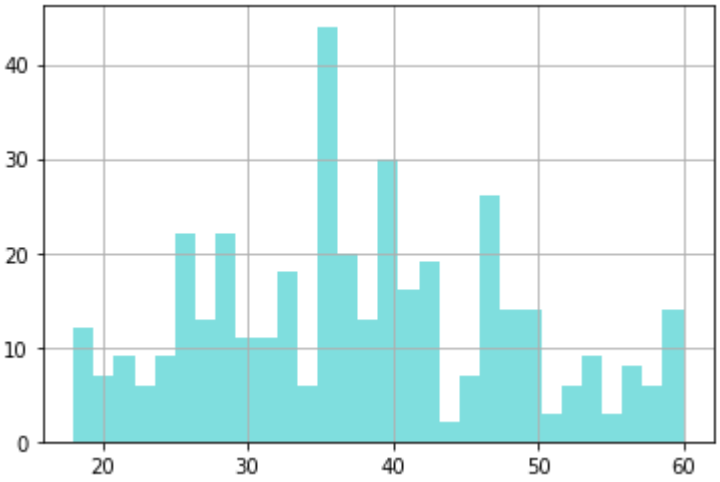
In [10]:
```
#Write the code to show the result(below)
```

Out[10]:

	User ID	Age	EstimatedSalary	Purchased
count	4.000000e+02	400.000000	400.000000	400.000000
mean	1.569154e+07	37.655000	69742.500000	0.357500
std	7.165832e+04	10.482877	34096.960282	0.479864
min	1.556669e+07	18.000000	15000.000000	0.000000
25%	1.562676e+07	29.750000	43000.000000	0.000000
50%	1.569434e+07	37.000000	70000.000000	0.000000
75%	1.575036e+07	46.000000	88000.000000	1.000000
max	1.581524e+07	60.000000	150000.000000	1.000000

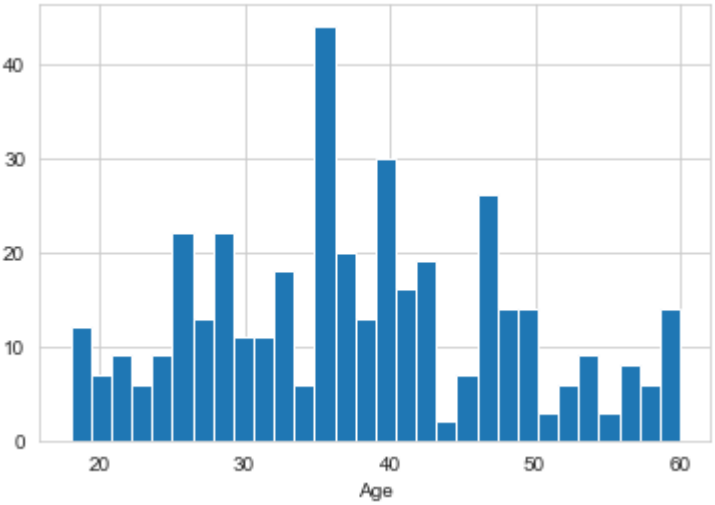
In [8]:
```
data['Age'].hist(bins=30, color='c', alpha=0.5)
```

Out[8]: <AxesSubplot:>



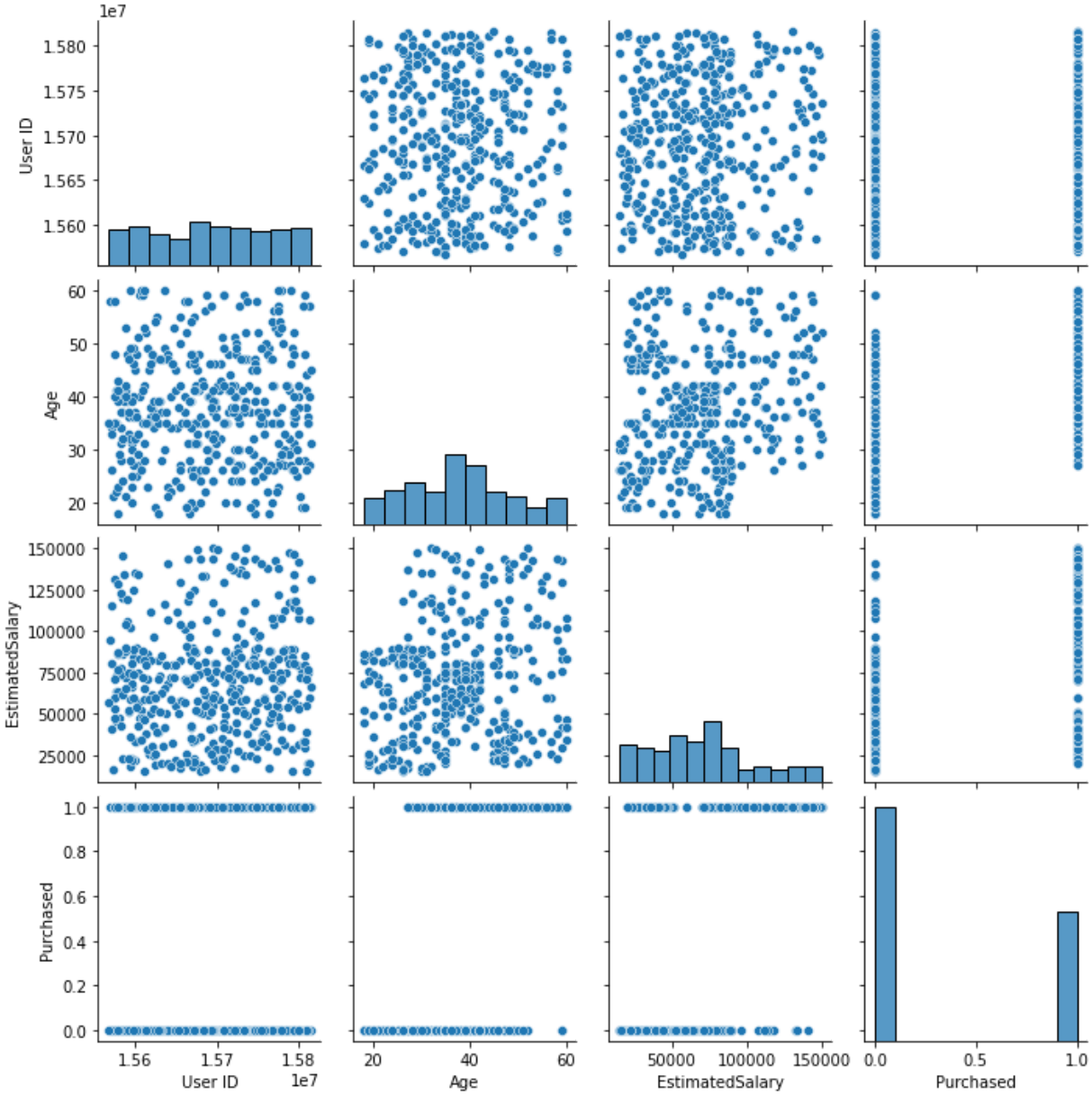
In [12]:
```
#Write the code to show the result(below)
```

Out[12]: Text(0.5, 0, 'Age')



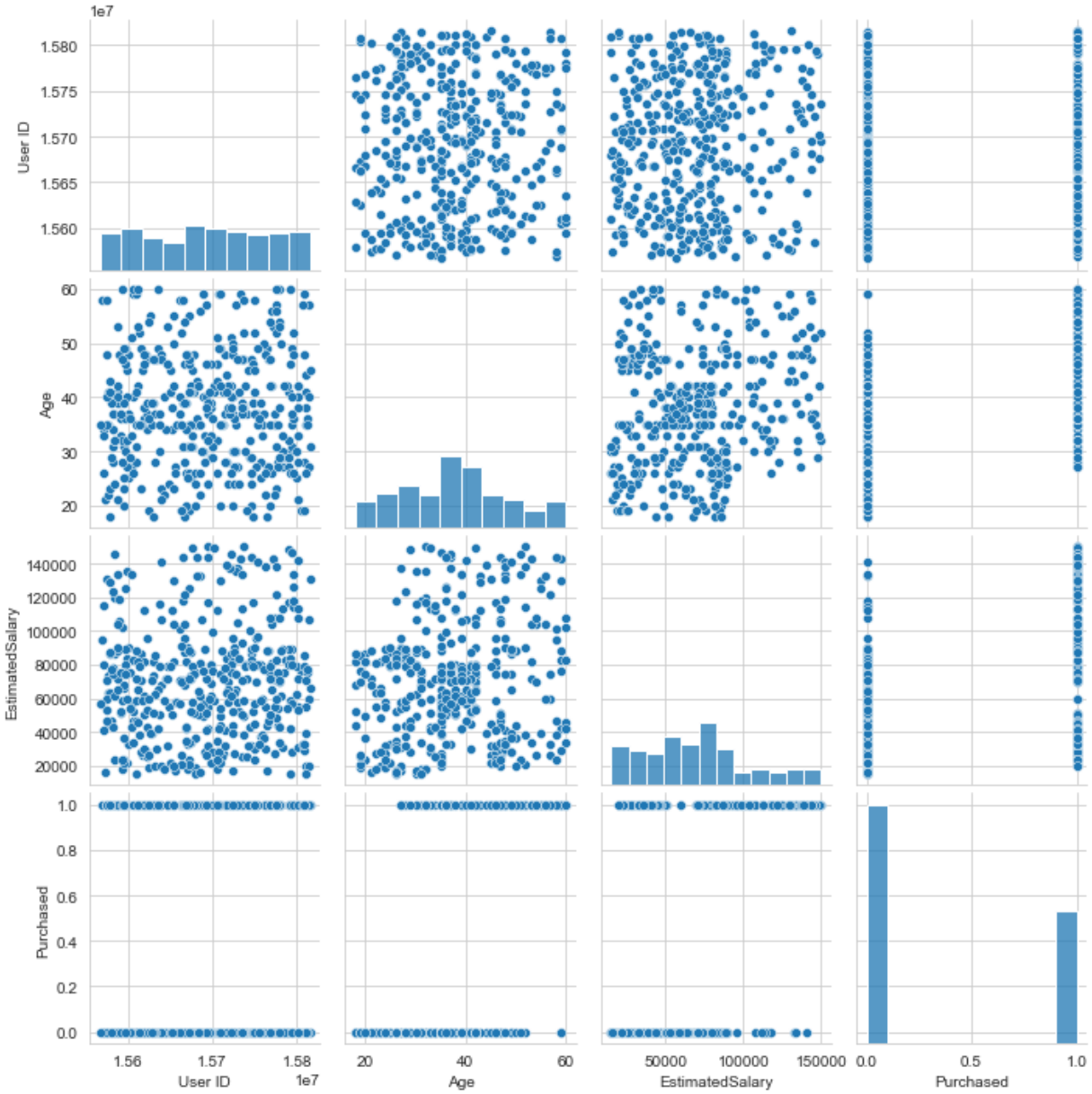
In [9]:
```
sns.pairplot(data)
```

Out[9]:
```
<seaborn.axisgrid.PairGrid at 0x1f7d5447790>
```



In [14]:
```
#Write the code to show the result(below)
```

Out[14]: <seaborn.axisgrid.PairGrid at 0x20141e07608>



Logistic Regression

Now it's time to do a train test split, and train our model. Check for categorical and Numerical values and buid the model based on your data type.

In [10]:
```
Gender = pd.get_dummies(data['Gender'],drop_first=True)
data.drop(['Gender'],axis=1,inplace=True)
data = pd.concat([data, Gender],axis=1)
data.head()
```

Out[10]:

	User ID	Age	EstimatedSalary	Purchased	Male
0	15624510	19	19000	0	1
1	15810944	35	20000	0	1
2	15668575	26	43000	0	0
3	15603246	27	57000	0	0
4	15804002	19	76000	0	1

In [8]:
```
from sklearn.model_selection import train_test_split
```

In [16]:
```
X_train, X_test, y_train, y_test = train_test_split(data.drop('Purchased',axis=1)
                                                    data['Purchased'], test_size
                                                    random_state=101)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_40744/4083481156.py in <module>
----> 1 X_train, X_test, y_train, y_test = train_test_split(data.drop('Purchase
d',axis=1),
      2                                                     data['Purchased'],
test_size = 0.30,
      3                                                     random_state=101)

NameError: name 'data' is not defined
```

In [10]:
```
from sklearn.linear_model import LogisticRegression
```

Predictions and Evaluations

In [21]:
```
logmodel = LogisticRegression()
logmodel.fit(X_train,y_train)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_64980/2572600564.py in <module>
      1 logmodel = LogisticRegression()
----> 2 logmodel.fit(X_train,y_train)

NameError: name 'X_train' is not defined
```

In [ ]:
```
print(y_train)
```

In [15]:
```
predictions = logmodel.predict(X_test)
print(predictions)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_40744/3812305382.py in <module>
----> 1 predictions = logmodel.predict(X_test)
      2 print(predictions)

NameError: name 'logmodel' is not defined
```

Classification Report

In [18]:
```
from sklearn.metrics import classification_report
```

In [19]:
```
print(classification_report(y_test,predictions))
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_64980/213323377.py in <module>
----> 1 print(classification_report(y_test,predictions))

NameError: name 'y_test' is not defined
```