```python
In [1]: import numpy as np
        import pandas as pd
        import matplotlib.pyplot as plt
        from sklearn.preprocessing import StandardScaler
        from sklearn.model_selection import train_test_split
        from sklearn.metrics import accuracy_score,classification_report,confusion_matrix
        from sklearn.neural_network import MLPClassifier
        from matplotlib.colors import ListedColormap
```

```python
In [5]: a=pd.read_csv(r"C:\Users\Admin\Downloads\suv dataset - suv dataset.csv")
        a
```

Out[5]:

	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
...	...	...	...	...	...
395	15691863	Female	46	41000	1
396	15706071	Male	51	23000	1
397	15654296	Female	50	20000	1
398	15755018	Male	36	33000	0
399	15594041	Female	49	36000	1

400 rows × 5 columns

```python
In [6]: a.head()
```

Out[6]:

	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

```python
In [7]: a.tail()
```

Out[7]:

	User ID	Gender	Age	EstimatedSalary	Purchased
395	15691863	Female	46	41000	1
396	15706071	Male	51	23000	1
397	15654296	Female	50	20000	1
398	15755018	Male	36	33000	0
399	15594041	Female	49	36000	1

In [8]:
```python
a.shape
```

Out[8]: (400, 5)

In [11]:
```python
X=a[['Age','EstimatedSalary']]
Y=a['Purchased']
X_train,X_test,Y_train,Y_test=train_test_split(X,Y,test_size=0.2,random_state=42)
model=MLPClassifier(hidden_layer_sizes=(10,10),activation='relu',
solver='adam',max_iter=1000,random_state=42)
sc=StandardScaler()
X_train=sc.fit_transform(X_train)
X_test=sc.transform(X_test)
model.fit(X_train,Y_train)
y_pred=model.predict(X_test)
```

In [12]:
```python
accuracy=accuracy_score(Y_test,y_pred)
accuracy
```

Out[12]: 0.9375

In [13]:
```python
cr=classification_report(Y_test,y_pred)
print('\n Classification Report \n',cr)
```

```
Classification Report
              precision    recall  f1-score   support

           0       0.98      0.92      0.95        52
           1       0.87      0.96      0.92        28

    accuracy                           0.94        80
   macro avg       0.93      0.94      0.93        80
weighted avg       0.94      0.94      0.94        80
```
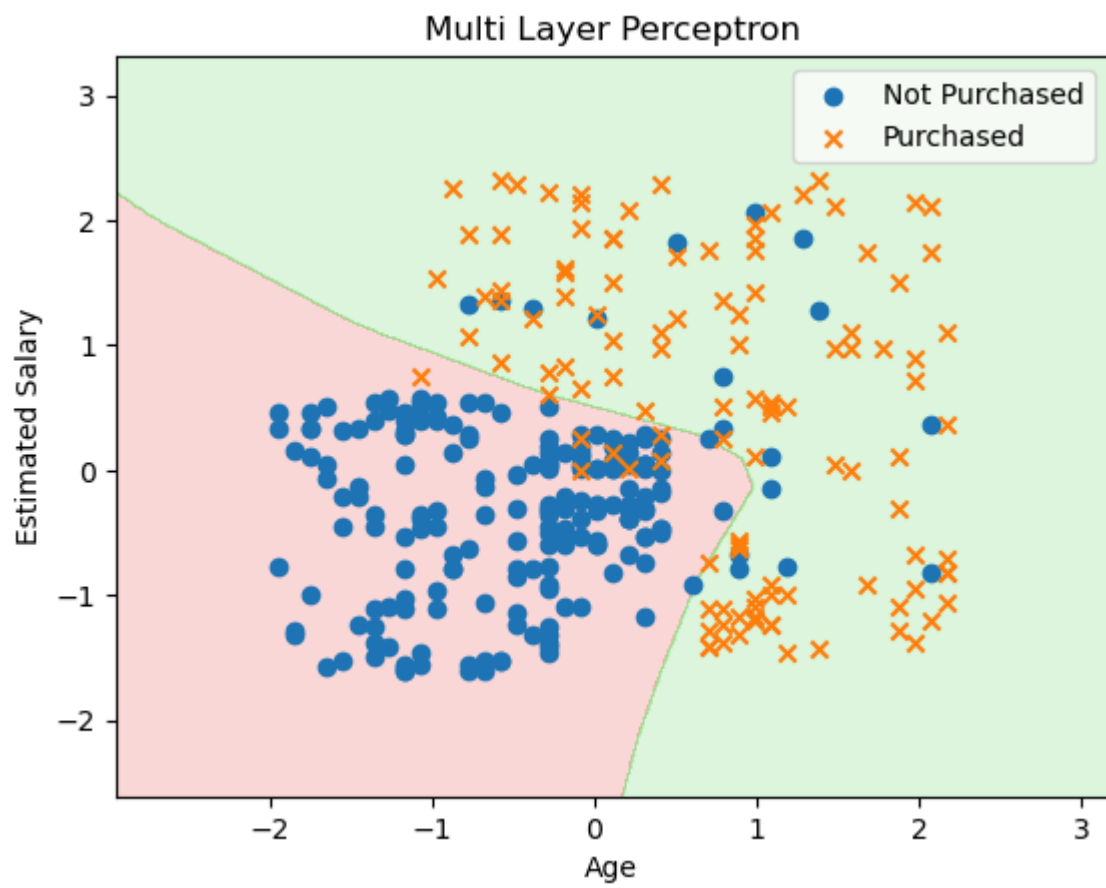
In [22]:
```python
X_set,Y_set=X_train,Y_train

X1,X2=np.meshgrid(np.arange(start=X_set[:,0].min()-1,stop=X_set[:,0].max()+1,step=0.01),
np.arange(start=X_set[:,1].min()-1,stop=X_set[:,1].max()+1,step=0.01))

plt.contourf(X1,X2,model.predict(np.array([X1.ravel(),X2.ravel()]).T).reshape(X1.shape),
alpha=0.3,cmap=ListedColormap(('lightCoral','lightgreen')))

plt.scatter(X_set[Y_set==0,0],X_set[Y_set==0,1],label='Not Purchased',marker='o')
plt.scatter(X_set[Y_set==1,0],X_set[Y_set==1,1],label='Purchased',marker='x')
plt.title('Multi Layer Perceptron')
plt.xlabel('Age')
plt.ylabel('Estimated Salary')
plt.legend()
plt.show()
```

Multi Layer Perceptron

In [ ]:

In [ ]: