```
1 import pandas as pd
2 import numpy as np
3 import sklearn
4 from sklearn.neighbors import KNeighborsClassifier
5 from sklearn.preprocessing import StandardScaler
6 from sklearn.model_selection import train_test_split
7 from sklearn.metrics import accuracy_score, confusion_matrix
```

```
1 df1=pd.read_csv("/content/User_Data.csv")
2 df2=df1[['Purchased']]
3 df1.drop(labels=['Purchased','User ID','Gender'],
4         axis=1, inplace=True)
```
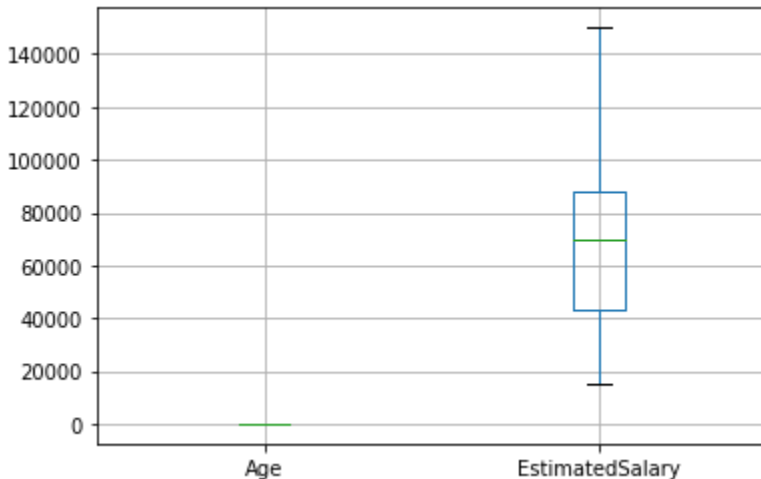
```
1 df1.describe()
```

	Age	EstimatedSalary
count	400.000000	400.000000
mean	37.655000	69742.500000
std	10.482877	34096.960282
min	18.000000	15000.000000
25%	29.750000	43000.000000
50%	37.000000	70000.000000
75%	46.000000	88000.000000
max	60.000000	150000.000000

```
1 df1.corr()
```

```
1 df1.corr()
```

	Age	EstimatedSalary
Age	1.000000	0.155238
EstimatedSalary	0.155238	1.000000

```
1 df1.boxplot()
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f611423b910>
```



```
1 ss=StandardScaler()
2 df1=ss.fit_transform(df1)
```

```
1 x_train, x_test, y_train, y_test=train_test_split(df1,df2,
2                                                   test_size=0.2,
3                                                   random_state=32)
```

```
1 knn=KNeighborsClassifier(n_neighbors=5)
2 knn.fit(x_train, y_train)
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:2:

KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='min
                     metric_params=None, n_jobs=None, n_neighbor
                     weights='uniform')
```

```
1 y_pred=knn.predict(x_test)
```

```
1 accuracy_score(y_test, y_pred)*100
```

```
91.25
```

```
1 confusion_matrix(y_test, y_pred)
```

```
array([[44,  5],
       [ 2, 29]])
```