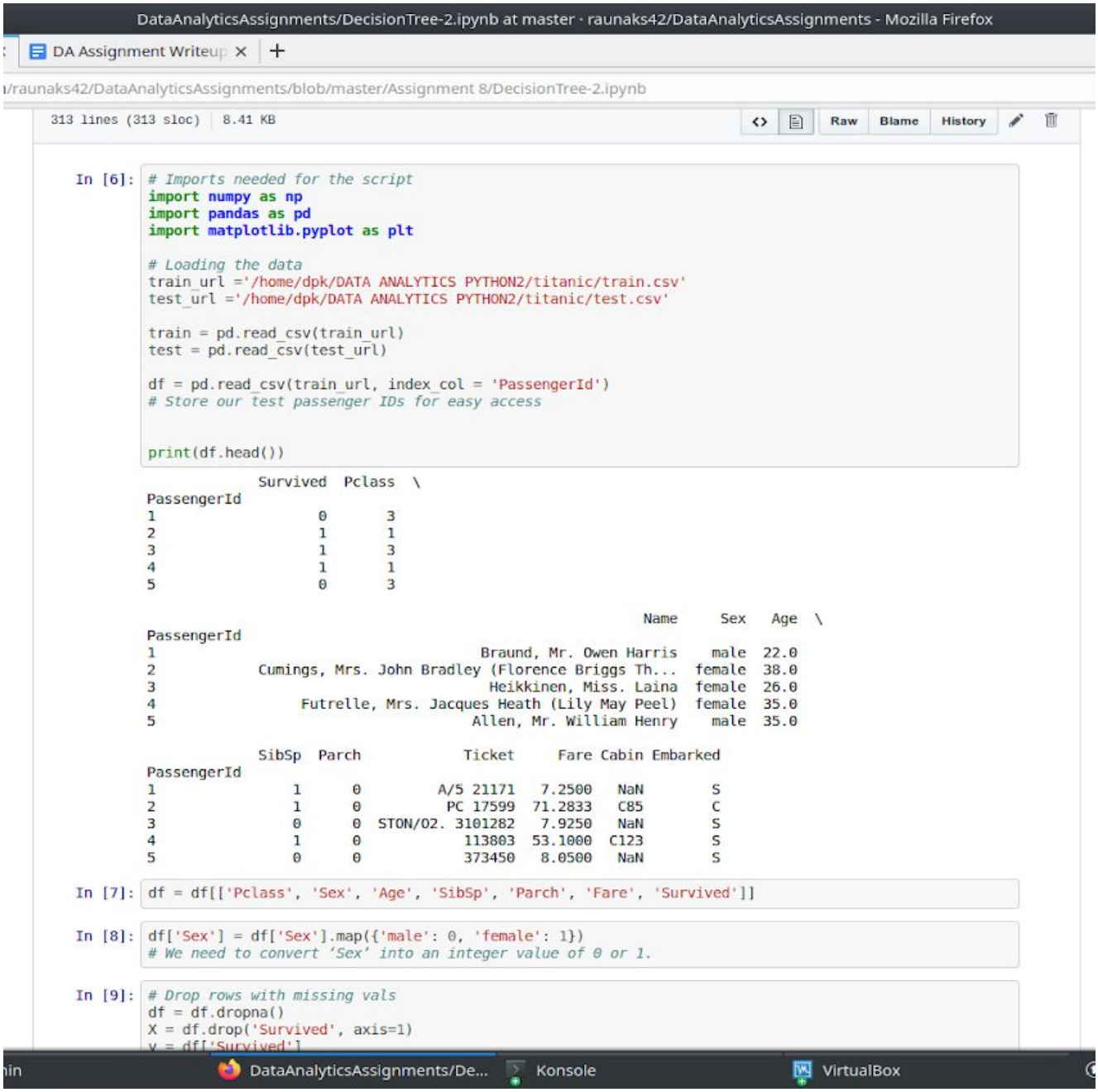
# DATA ANALYTICS

Deepika Karanji- PES1201700103
Dheemanth GR- PES1201700229
Raunak Sengupta- PES1201700072
Navneet Raju- PES1201701545

## ASSIGNMENT 4 - Decision Tree Implementation

### Problem statement - Use Decision Tree to Classify if whether the person survived or died in Titanic.



In [9]:
```
# Drop rows with missing vals
df = df.dropna()
X = df.drop('Survived', axis=1)
y = df['Survived']
```

In [11]:
```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = \
train_test_split(X, y, random_state=1)
```

In [12]:
```
from sklearn import tree

model = tree.DecisionTreeClassifier()
```

In [13]:
```
model
```

Out[13]:
```
DecisionTreeClassifier(class_weight=None, criterion='gini', max_depth=None,
                       max_features=None, max_leaf_nodes=None,
                       min_impurity_decrease=0.0, min_impurity_split=None,
                       min_samples_leaf=1, min_samples_split=2,
                       min_weight_fraction_leaf=0.0, presort=False,
                       random_state=None, splitter='best')
```

In [14]:
```
model.fit(X_train, y_train)
```

Out[14]:
```
DecisionTreeClassifier(class_weight=None, criterion='gini', max_depth=None,
                       max_features=None, max_leaf_nodes=None,
                       min_impurity_decrease=0.0, min_impurity_split=None,
                       min_samples_leaf=1, min_samples_split=2,
                       min_weight_fraction_leaf=0.0, presort=False,
                       random_state=None, splitter='best')
```

In [15]:
```
# Then we score the predicted output from model on
# our test data against our ground truth test data.
y_predict = model.predict(X_test)

from sklearn.metrics import accuracy_score

accuracy_score(y_test, y_predict)
```

Out[15]:
```
0.8212290502793296
```

In [16]:
```
from sklearn.metrics import confusion_matrix

pd.DataFrame(
    confusion_matrix(y_test, y_predict),
    columns=['Predicted Not Survival', 'Predicted Survival'],
    index=['True Not Survival', 'True Survival']
)
```

Out[16]:

| | Predicted Not Survival | Predicted Survival |
|---|---|---|
| **True Not Survival** | 96 | 16 |
| **True Survival** | 16 | 51 |

In [ ]:
```
tree.export_graphviz(model.tree_, out_file='tree.dot', feature_names=X.columns)
```