# Code:

import pandas as pd

import numpy as np

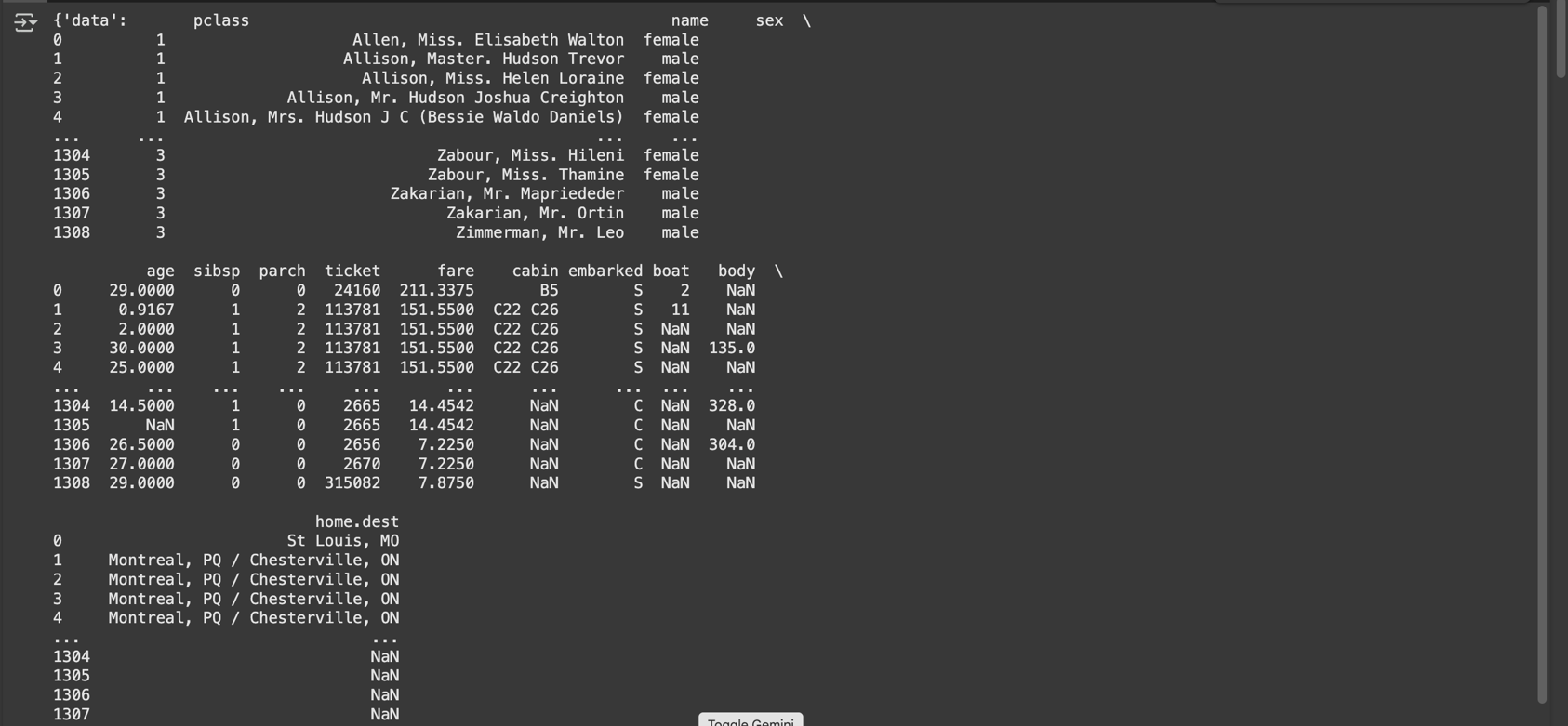
from sklearn.datasets import fetch\_openml

from sklearn.preprocessing import LabelEncoder,OneHotEncoder

#Load the dataset

data=fetch\_openml('titanic',version=1,as\_frame=True)

print(data)



data['feature\_names']

data\_f=data.frame.copy()

#Dropping the empty values

data=data\_f[['age','sex','fare','embarked','pclass','survived']].dropna()

set(list(data['embarked']))

#Lable Encoding

le=LabelEncoder()

data['embarked\_le']=le.fit\_transform(data['embarked'])

data.columns

data['embarked\_le']

#One-Hot Encoder

ohe=OneHotEncoder()

# df\_ohe=pd.get\_dummies(data['sex'])

df\_ohe=pd.get\_dummies(data,columns=['sex'])

df\_ohe

#Loadnthe dataset of diabetes

from sklearn.datasets import load\_diabetes

data=load\_diabetes(as\_frame=True)

data=data.frame

print(data)

data.describe()

import seaborn as sns

import matplotlib.pyplot as plt

#Get coorelation in-between the features

corr=data.corr()

plt.figure(figsize=(10,8))

sns.heatmap(corr,annot=True)

plt.show()

#Feature Importance

from sklearn.tree import DecisionTreeRegressor

from sklearn.model\_selection import train\_test\_split

data.columns

X\_train,X\_test,y\_train,y\_test=train\_test\_split(data.drop('target',axis=1),data['target'],test\_size=0.2,random\_state=42)

tree=DecisionTreeRegressor(max\_depth=3)

tree.fit(X\_train,y\_train)

importance=pd.Series(tree.feature\_importances\_,index=X\_train.columns)

importance