Valery Louis
Submission date: 01/21/2023
Submitted to: Data Glacier internship

Steps of Deployment

1. model.py

```python
import pandas as pd
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
import pickle

df = pd.read_csv("titanic.csv")

print(df.head())

# independent and dependent variable
X = df[["Pclass", "SibSp", "Parch", "Fare"]]
y = df["Survived"]

# split data set into train and test
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=50)

# Feature scaling
sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)

# Instantiate the model
classifier = RandomForestClassifier()

# Fitting the model with training data
classifier.fit(X_train, y_train)

# make pickle file of model
pickle.dump(classifier, open("model.pkl", "wb"))
```

2. app.py

```python
import numpy as np
from flask import Flask, request, render_template
import pickle

app = Flask(__name__)
app.config['TEMPLATES_AUTO_RELOAD'] = False

model = pickle.load(open("model.pkl", "rb"))


@app.route('/')
def home():
    return render_template("index.html")


@app.route("/predict", methods=["POST"])
def predict():
    float_features = [float(x) for x in request.form.values()]
    features = [np.array(float_features)]
    prediction = model.predict(features)
    return render_template("index.html", prediction_text="Survived: {}".format(prediction))


if __name__ == "__main__":
    app.run(debug=True)
```

3. index.html

```html
<!DOCTYPE html>
<html>
<head>
    <title>Titanic Survival Prediction</title>
</head>
<body>
    <h1>Titanic Survival Prediction</h1>
    <form action = "/predict" method = "POST">
        <label for="Pclass">Pclass:</label>
        <input type="text" id="Pclass" name="Pclass"><br><br>
        <label for="SibSp">SibSp:</label>
        <input type="text" id="SibSp" name="SibSp"><br><br>
        <label for="Parch">Parch:</label>
        <input type="text" id="Parch" name="Parch"><br><br>
        <label for="Fare">Fare:</label>
        <input type="text" id="Fare" name="Fare"><br><br>
        <input type="submit" value="Predict">
    </form>
    <br>
    <p>{{ prediction_text }}</p>
</body>
</html>
```