

SOURCE CODE

HTML CODE FOR INDEX PAGE

```
<!doctype html>

<html lang="en">

<head>

<title> Crude Oil Price Prediction</title>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

<link href="https://fonts.googleapis.com/css?family=Lato:300,400,700&display=swap"
rel="stylesheet">

<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/font-awesome/4.7.0/css/font-
awesome.min.css">

<link rel="stylesheet" href="static/css/style.css">

<style>

body {

background-image: url("static/images/crudeoilbg.jpg");

background-repeat: no-repeat;

background-position: 0% 0%;

background-size: 100% 100%;

}

</style>

</head>
```

```
<body class="img js-fullheight" >

<section class="ftco-section">

<div class="container">

<div class="row justify-content-center">

<div class="col-md-6 text-center mb-5">

<h2 class="heading-section">Crude Oil Prediction</h2>

</div>

</div>

<div class="row justify-content-center">

<p>
```

Demand for oil is inelastic, therefore the rise in price is good news for producers because they will see an increase in

their revenue. Oil importers, however, will experience increased costs of purchasing oil.

Because oil is the largest traded commodity, the effects are guided significant. A rising oil price can even shift

economic political power from oil importers to oil exporters. The crude oil price movements are subject to diverse

influencing factors

</p>

</br></br>

<div class="col-md-6 col-lg-4">

<div class="login-wrap p-0">

<div class="form-group">

Predict Crude Oil Price

</div>

</div>

</div>

</br></br>

</div>

</div>

</section>

</body>

</html>

HTML CODE FOR PREDICTION

```
<!doctype html>

<html lang="en">

<head>

<title> Crude Oil Price Prediction</title>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

<link href="https://fonts.googleapis.com/css?family=Lato:300,400,700&display=swap"
rel="stylesheet">

<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/font-awesome/4.7.0/css/font-
awesome.min.css">

<link rel="stylesheet" href="static/css/style.css">

<style>

body {

    background-image: url("static/images/oil.jpg");

    background-repeat: no-repeat;

    background-position: 0% 0%;

    background-size: 100% 100%;

}

</style>

</head>

<body class="img js-fullheight" >
```

```
<section class="ftco-section">

<div class="container">

<div class="row justify-content-center">

<div class="col-md-6 text-center mb-5">

<h2 class="heading-section">Crude Oil Prediction</h2>

</div>

</div>

<div class="row justify-content-center">

<div class="col-md-6 col-lg-4">

<div class="login-wrap p-0">

<h3 class="mb-4 text-center">Enter the crude oil price for first 10 days</h3>

<form method="post" action="predicted" class="signin-form">

<div class="form-group">

<input type="text" class="form-control" id="year" name="year" required>

</div>

<div class="form-group">

<button type="submit" class="form-control btn btn-primary submit px-3">Submit</button>

</div>

</form>

</br></br>
```

```
<div class="social d-flex text-center">

<p class="w-100 text-center">{{ showcase }} </p>

</div>

</div>

</div>

</div>

</div>

</section>

</body>

</html>
```

FLASK CODE

```
import numpy as np

from flask import Flask,render_template,request

from tensorflow.python.keras.models import load_model

# This is a sample Python script.


# Press Shift+F10 to execute it or replace it with your code.

# Press Double Shift to search everywhere for classes, files, tool windows, actions, and settings.

app = Flask(__name__,template_folder='templates', static_url_path='/static')

model = load_model('predict.h5',)

@app.route('/')

def home():

    return render_template("index.html")

@app.route('/predict')


def home2():

    return render_template("prediction.html")

@app.route('/predicted',methods =['POST'])

def login():

    x_input=str(request.form['year'])
```

```
x_input=x_input.split(',')  
  
print(x_input)  
  
for i in range (0, len(x_input)):  
  
    x_input[i] = float(x_input[i])  
  
    print(x_input)  
  
x_input=np.array(x_input).reshape(1,-1)  
  
temp_input=list(x_input)  
  
temp_input=temp_input[0].tolist()  
  
lst_output=[]  
  
n_steps=10  
  
i=0  
  
while(i<1):  
  
    if(len(temp_input)>10):  
  
        x_input=np.array(temp_input[1:])  
  
        print("{} day input {}".format(i,x_input))  
  
        x_input=x_input.reshape(1,-1)  
  
        x_input=x_input.reshape((1,n_steps,1))
```



```
yhat = model.predict(x_input, verbose=0)

temp_input.extend(yhat[0].tolist())

temp_input=temp_input[1:]

lst_output.extend(yhat.tolist())

i=i+1

else:

x_input = x_input.reshape((1,n_steps,1))

yhat = model.predict(x_input, verbose=0)

print(yhat[0])

temp_input.extend(yhat[0].tolist())

print(len(temp_input))

lst_output.extend(yhat.tolist())

i=i+1

print(lst_output)

return render_template("prediction.html",showcase = "the next day predicted value is
:"+str(lst_output))

# Press the green button in the gutter to run the script.

if __name__ == '__main__':

app.run(debug = True,port=5000)
```





