In [ ]:
```
import pandas as pd
import yfinance as yf
import matplotlib.pyplot as plt
import numpy as np
import datetime as dt
import tensorflow as tf
import math

from IPython.display import clear_output, display
from sklearn import preprocessing
```

In [ ]:
```
#Download the dataframe and saving it
raw = yf.download('IYW', period = 'max')
df = pd.DataFrame(raw.copy())
last_date = df.iloc[-1].name
clear_output()
```

In [ ]:
```
class myData:
    import numpy as np

    def __init__(self, array, days, scaler):
        self.array = array
        self.days = days
        self.scaler = scaler

        self.y_original = self.scaler.inverse_transform(self.array)

        self.x, self.y = [], []

        for i in range(days,len(array)):
            self.x.append(array[i - days:i])
            self.y.append(array[i])

        self.x, self.y = np.reshape(numpy_arr := np.array(self.x), (numpy
```

In [ ]:
```
scaler = preprocessing.MinMaxScaler(feature_range = (0,1))
scaled_data = scaler.fit_transform(df.filter(['Close']).values)
train_len = math.floor(len(scaled_data) * 0.90)
len(scaled_data), train_len
```

Out[ ]: (5816, 5234)

In [ ]:
```
mTrain: myData = myData(scaled_data[0:train_len], 30, scaler = scaler)
mTest: myData = myData(scaled_data[train_len:], 30, scaler = scaler)
```

In [ ]:
```
# x_train = np.reshape(x_train, (x_train.shape[0], x_train.shape[1], 1))
model = tf.keras.Sequential()
model.add(tf.keras.layers.LSTM(50, return_sequences=True, input_shape=(mT
model.add(tf.keras.layers.LSTM(50, return_sequences=False))
model.add(tf.keras.layers.Dense(25))
model.add(tf.keras.layers.Dense(1))

model.compile(optimizer='adam', loss='mean_squared_error', metrics = ['ac
```

In [ ]:
```
history = model.fit(mTrain.x, mTrain.y, batch_size = 32, epochs = 30, val
```

```
Epoch 1/30
131/131 [==============================] - 3s 15ms/step - loss: 2.9301e-04
- accuracy: 2.4021e-04 - mse: 2.9301e-04 - val_loss: 2.4234e-04 - val_accu
racy: 0.0000e+00 - val_mse: 2.4234e-04
Epoch 2/30
131/131 [==============================] - 1s 11ms/step - loss: 2.9585e-05
- accuracy: 2.4021e-04 - mse: 2.9585e-05 - val_loss: 2.6185e-04 - val_accu
racy: 0.0000e+00 - val_mse: 2.6185e-04
Epoch 3/30
131/131 [==============================] - 1s 11ms/step - loss: 2.7129e-05
- accuracy: 2.4021e-04 - mse: 2.7129e-05 - val_loss: 5.7200e-04 - val_accu
racy: 0.0000e+00 - val_mse: 5.7200e-04
Epoch 4/30
131/131 [==============================] - 1s 11ms/step - loss: 2.9185e-05
- accuracy: 2.4021e-04 - mse: 2.9185e-05 - val_loss: 2.4427e-04 - val_accu
racy: 0.0000e+00 - val_mse: 2.4427e-04
Epoch 5/30
131/131 [==============================] - 1s 11ms/step - loss: 2.3634e-05
- accuracy: 2.4021e-04 - mse: 2.3634e-05 - val_loss: 2.3723e-04 - val_accu
racy: 0.0000e+00 - val_mse: 2.3723e-04
Epoch 6/30
131/131 [==============================] - 1s 11ms/step - loss: 2.4855e-05
- accuracy: 2.4021e-04 - mse: 2.4855e-05 - val_loss: 2.1997e-04 - val_accu
racy: 0.0000e+00 - val_mse: 2.1997e-04
Epoch 7/30
131/131 [==============================] - 1s 11ms/step - loss: 2.3403e-05
- accuracy: 2.4021e-04 - mse: 2.3403e-05 - val_loss: 6.9756e-04 - val_accu
racy: 0.0000e+00 - val_mse: 6.9756e-04
Epoch 8/30
131/131 [==============================] - 1s 11ms/step - loss: 2.2930e-05
- accuracy: 2.4021e-04 - mse: 2.2930e-05 - val_loss: 2.0133e-04 - val_accu
racy: 0.0000e+00 - val_mse: 2.0133e-04
Epoch 9/30
131/131 [==============================] - 1s 11ms/step - loss: 1.9210e-05
- accuracy: 2.4021e-04 - mse: 1.9210e-05 - val_loss: 1.3243e-04 - val_accu
racy: 0.0000e+00 - val_mse: 1.3243e-04
Epoch 10/30
131/131 [==============================] - 1s 11ms/step - loss: 1.8663e-05
- accuracy: 2.4021e-04 - mse: 1.8663e-05 - val_loss: 1.5040e-04 - val_accu
racy: 0.0000e+00 - val_mse: 1.5040e-04
Epoch 11/30
131/131 [==============================] - 2s 12ms/step - loss: 1.6583e-05
- accuracy: 2.4021e-04 - mse: 1.6583e-05 - val_loss: 2.1518e-04 - val_accu
racy: 0.0000e+00 - val_mse: 2.1518e-04
Epoch 12/30
131/131 [==============================] - 2s 12ms/step - loss: 1.8016e-05
- accuracy: 2.4021e-04 - mse: 1.8016e-05 - val_loss: 1.4123e-04 - val_accu
racy: 0.0000e+00 - val_mse: 1.4123e-04
Epoch 13/30
131/131 [==============================] - 2s 12ms/step - loss: 1.7370e-05
- accuracy: 2.4021e-04 - mse: 1.7370e-05 - val_loss: 1.4842e-04 - val_accu
racy: 0.0000e+00 - val_mse: 1.4842e-04
Epoch 14/30
131/131 [==============================] - 2s 12ms/step - loss: 1.6642e-05
- accuracy: 2.4021e-04 - mse: 1.6642e-05 - val_loss: 3.0526e-04 - val_accu
racy: 0.0000e+00 - val_mse: 3.0526e-04
Epoch 15/30
131/131 [==============================] - 2s 12ms/step - loss: 1.5352e-05
- accuracy: 2.4021e-04 - mse: 1.5352e-05 - val_loss: 1.8137e-04 - val_accu
racy: 0.0000e+00 - val_mse: 1.8137e-04
```

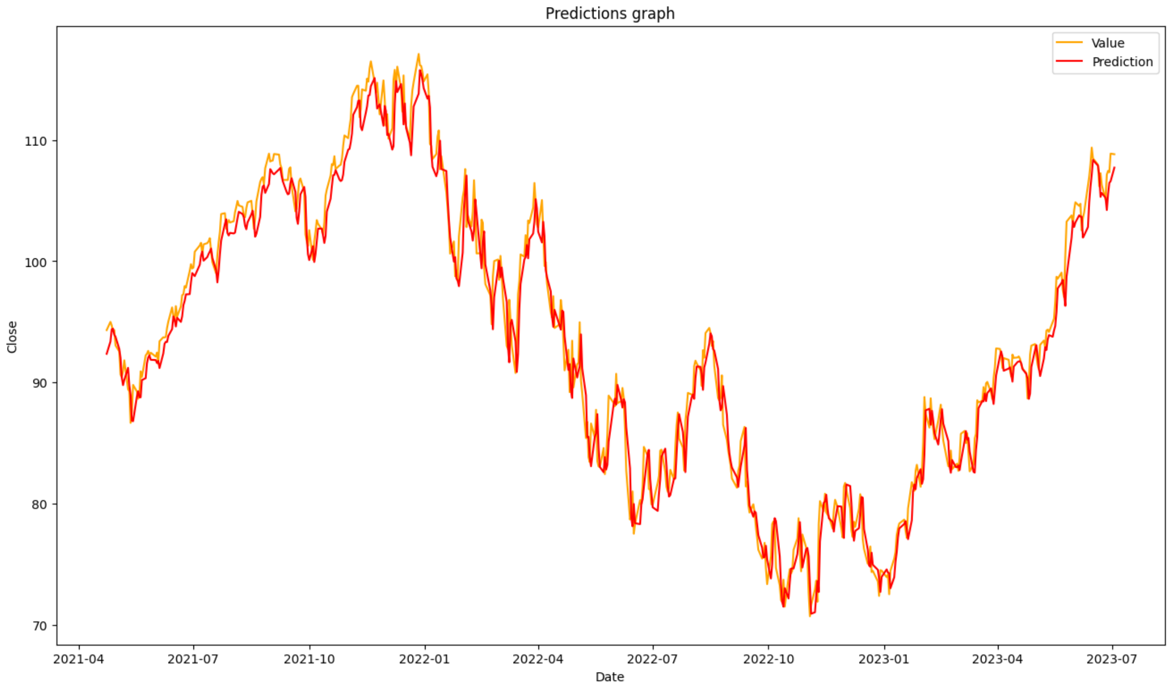
```
Epoch 16/30
131/131 [==============================] - 2s 12ms/step - loss: 1.5745e-05
- accuracy: 2.4021e-04 - mse: 1.5745e-05 - val_loss: 9.5655e-05 - val_accu
racy: 0.0000e+00 - val_mse: 9.5655e-05
Epoch 17/30
131/131 [==============================] - 2s 12ms/step - loss: 1.5081e-05
- accuracy: 2.4021e-04 - mse: 1.5081e-05 - val_loss: 9.3701e-05 - val_accu
racy: 0.0000e+00 - val_mse: 9.3701e-05
Epoch 18/30
131/131 [==============================] - 1s 11ms/step - loss: 1.5799e-05
- accuracy: 2.4021e-04 - mse: 1.5799e-05 - val_loss: 1.2254e-04 - val_accu
racy: 0.0000e+00 - val_mse: 1.2254e-04
Epoch 19/30
131/131 [==============================] - 1s 11ms/step - loss: 1.4814e-05
- accuracy: 2.4021e-04 - mse: 1.4814e-05 - val_loss: 1.3409e-04 - val_accu
racy: 0.0000e+00 - val_mse: 1.3409e-04
Epoch 20/30
131/131 [==============================] - 2s 12ms/step - loss: 1.2182e-05
- accuracy: 2.4021e-04 - mse: 1.2182e-05 - val_loss: 8.9607e-05 - val_accu
racy: 0.0000e+00 - val_mse: 8.9607e-05
Epoch 21/30
131/131 [==============================] - 1s 11ms/step - loss: 1.3183e-05
- accuracy: 2.4021e-04 - mse: 1.3183e-05 - val_loss: 1.1069e-04 - val_accu
racy: 0.0000e+00 - val_mse: 1.1069e-04
Epoch 22/30
131/131 [==============================] - 1s 11ms/step - loss: 1.4387e-05
- accuracy: 2.4021e-04 - mse: 1.4387e-05 - val_loss: 1.8219e-04 - val_accu
racy: 0.0000e+00 - val_mse: 1.8219e-04
Epoch 23/30
131/131 [==============================] - 1s 11ms/step - loss: 1.1482e-05
- accuracy: 2.4021e-04 - mse: 1.1482e-05 - val_loss: 7.8860e-05 - val_accu
racy: 0.0000e+00 - val_mse: 7.8860e-05
Epoch 24/30
131/131 [==============================] - 2s 12ms/step - loss: 1.0707e-05
- accuracy: 2.4021e-04 - mse: 1.0707e-05 - val_loss: 7.3658e-05 - val_accu
racy: 0.0000e+00 - val_mse: 7.3658e-05
Epoch 25/30
131/131 [==============================] - 1s 11ms/step - loss: 1.0522e-05
- accuracy: 2.4021e-04 - mse: 1.0522e-05 - val_loss: 7.9155e-05 - val_accu
racy: 0.0000e+00 - val_mse: 7.9155e-05
Epoch 26/30
131/131 [==============================] - 1s 11ms/step - loss: 1.0474e-05
- accuracy: 2.4021e-04 - mse: 1.0474e-05 - val_loss: 7.2950e-05 - val_accu
racy: 0.0000e+00 - val_mse: 7.2950e-05
Epoch 27/30
131/131 [==============================] - 1s 11ms/step - loss: 1.1920e-05
- accuracy: 2.4021e-04 - mse: 1.1920e-05 - val_loss: 1.8439e-04 - val_accu
racy: 0.0000e+00 - val_mse: 1.8439e-04
Epoch 28/30
131/131 [==============================] - 1s 11ms/step - loss: 8.9735e-06
- accuracy: 2.4021e-04 - mse: 8.9735e-06 - val_loss: 9.8290e-05 - val_accu
racy: 0.0000e+00 - val_mse: 9.8290e-05
Epoch 29/30
131/131 [==============================] - 2s 12ms/step - loss: 9.8147e-06
- accuracy: 2.4021e-04 - mse: 9.8147e-06 - val_loss: 9.1590e-05 - val_accu
racy: 0.0000e+00 - val_mse: 9.1590e-05
Epoch 30/30
131/131 [==============================] - 1s 11ms/step - loss: 1.0407e-05
- accuracy: 2.4021e-04 - mse: 1.0407e-05 - val_loss: 6.8024e-05 - val_accu
racy: 0.0000e+00 - val_mse: 6.8024e-05
```

In [ ]:
```
predictions = model.predict(mTest.x)
predictions = scaler.inverse_transform(predictions)
```

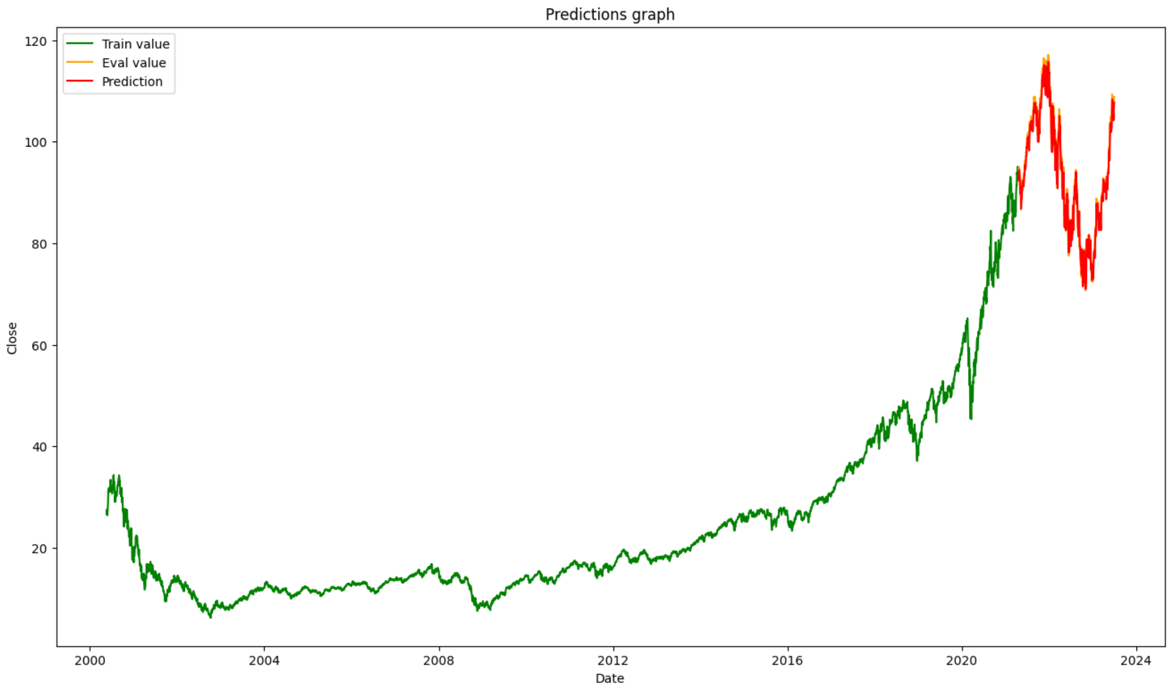
```
18/18 [==============================] - 0s 3ms/step
```

In [ ]:
```
train_chart = df.filter(['Close'])[:-predictions.size]
test_chart = df.filter(['Close'])[-predictions.size:]
test_chart['Predictions'] = predictions
```

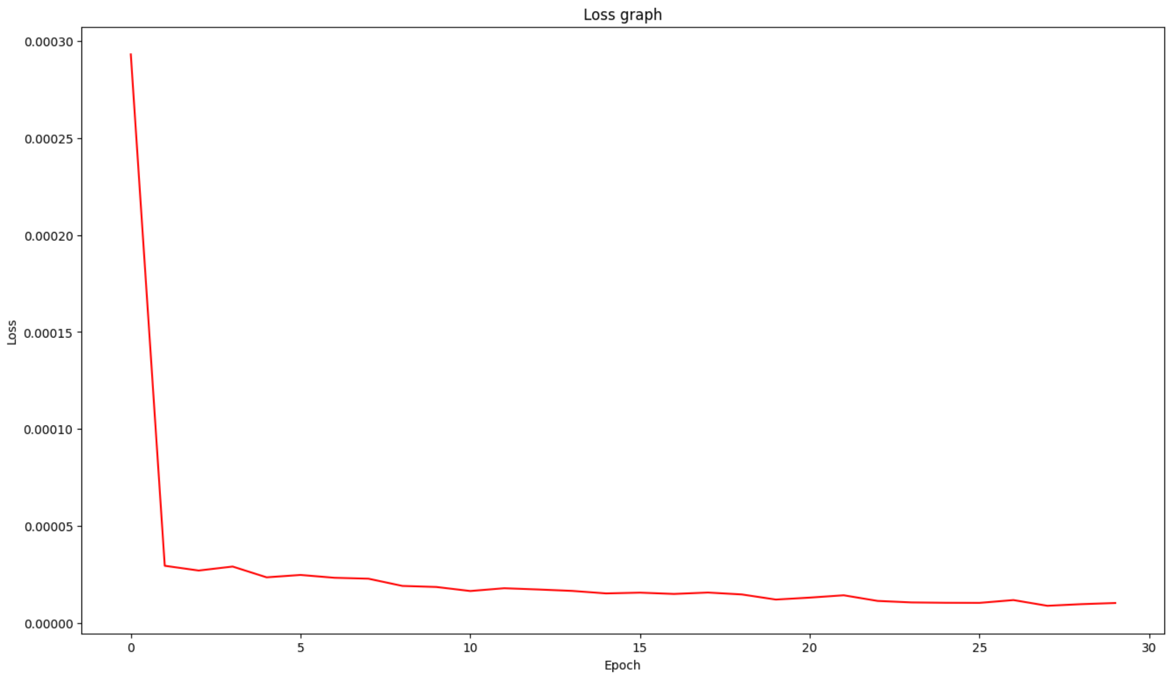
In [ ]:
```
plt.figure(figsize=(16,9))
plt.title('Predictions graph')
plt.xlabel('Date')
plt.ylabel('Close')
plt.plot(test_chart['Close'], c = 'orange')
plt.plot(test_chart['Predictions'], c = 'red')
plt.legend(['Value', 'Prediction'])
plt.show()
```



In [ ]:
```
plt.figure(figsize=(16,9))
plt.title('Predictions graph')
plt.xlabel('Date')
plt.ylabel('Close')
plt.plot(train_chart['Close'], c = 'green')
plt.plot(test_chart['Close'], c = 'orange')
plt.plot(test_chart['Predictions'], c = 'red')
plt.legend(['Train value', 'Eval value', 'Prediction'])
x = plt.show()
```



In [ ]:
```
plt.figure(figsize=(16,9))
plt.title('Loss graph')
plt.xlabel('Epoch')
plt.ylabel('Loss')
plt.plot(history.history['mse'], c = 'red')
plt.show()
```



Predicting next 365 days

In [ ]:
```
#MODEL 2:
m2Train: myData = myData(scaled_data[:], 30, scaler = scaler)
model2 = tf.keras.Sequential()
model2.add(tf.keras.layers.LSTM(64, return_sequences=True, input_shape=(m
model2.add(tf.keras.layers.LSTM(64, return_sequences=False))
model2.add(tf.keras.layers.Dense(32, activation='relu'))
model2.add(tf.keras.layers.Dense(1))
```

```
model2.compile(optimizer='adam', loss='mean_squared_error', metrics = ['a
model2.fit(m2Train.x, m2Train.y, batch_size = 32, epochs = 30, validation
```

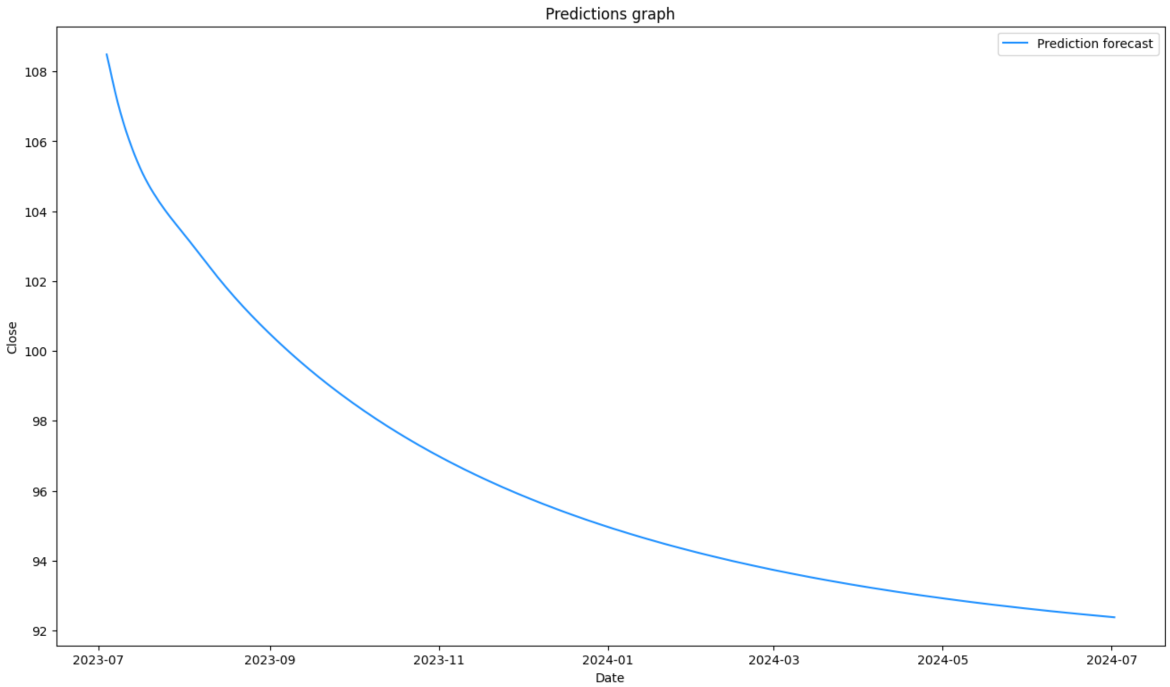
```
Epoch 1/30
172/172 [==============================] - 4s 16ms/step - loss: 0.0016 - a
ccuracy: 3.6390e-04 - mse: 0.0016 - val_loss: 0.0012 - val_accuracy: 0.000
0e+00 - val_mse: 0.0012
Epoch 2/30
172/172 [==============================] - 2s 13ms/step - loss: 1.6212e-04
- accuracy: 3.6390e-04 - mse: 1.6212e-04 - val_loss: 0.0011 - val_accurac
y: 0.0000e+00 - val_mse: 0.0011
Epoch 3/30
172/172 [==============================] - 2s 13ms/step - loss: 1.5354e-04
- accuracy: 3.6390e-04 - mse: 1.5354e-04 - val_loss: 0.0010 - val_accurac
y: 0.0000e+00 - val_mse: 0.0010
Epoch 4/30
172/172 [==============================] - 2s 14ms/step - loss: 1.4888e-04
- accuracy: 3.6390e-04 - mse: 1.4888e-04 - val_loss: 0.0012 - val_accurac
y: 0.0000e+00 - val_mse: 0.0012
Epoch 5/30
172/172 [==============================] - 2s 13ms/step - loss: 1.6743e-04
- accuracy: 3.6390e-04 - mse: 1.6743e-04 - val_loss: 0.0013 - val_accurac
y: 0.0000e+00 - val_mse: 0.0013
Epoch 6/30
172/172 [==============================] - 2s 13ms/step - loss: 1.2302e-04
- accuracy: 3.6390e-04 - mse: 1.2302e-04 - val_loss: 0.0011 - val_accurac
y: 0.0000e+00 - val_mse: 0.0011
Epoch 7/30
172/172 [==============================] - 2s 13ms/step - loss: 1.1508e-04
- accuracy: 3.6390e-04 - mse: 1.1508e-04 - val_loss: 8.0010e-04 - val_accu
racy: 0.0000e+00 - val_mse: 8.0010e-04
Epoch 8/30
172/172 [==============================] - 2s 13ms/step - loss: 1.3856e-04
- accuracy: 3.6390e-04 - mse: 1.3856e-04 - val_loss: 9.9308e-04 - val_accu
racy: 0.0000e+00 - val_mse: 9.9308e-04
Epoch 9/30
172/172 [==============================] - 2s 13ms/step - loss: 9.8097e-05
- accuracy: 3.6390e-04 - mse: 9.8097e-05 - val_loss: 5.7387e-04 - val_accu
racy: 0.0000e+00 - val_mse: 5.7387e-04
Epoch 10/30
172/172 [==============================] - 2s 13ms/step - loss: 9.2835e-05
- accuracy: 3.6390e-04 - mse: 9.2835e-05 - val_loss: 5.8161e-04 - val_accu
racy: 0.0000e+00 - val_mse: 5.8161e-04
Epoch 11/30
172/172 [==============================] - 2s 13ms/step - loss: 9.5503e-05
- accuracy: 3.6390e-04 - mse: 9.5503e-05 - val_loss: 5.2097e-04 - val_accu
racy: 0.0000e+00 - val_mse: 5.2097e-04
Epoch 12/30
172/172 [==============================] - 2s 13ms/step - loss: 8.3379e-05
- accuracy: 3.6390e-04 - mse: 8.3379e-05 - val_loss: 5.0081e-04 - val_accu
racy: 0.0000e+00 - val_mse: 5.0081e-04
Epoch 13/30
172/172 [==============================] - 2s 14ms/step - loss: 7.4986e-05
- accuracy: 3.6390e-04 - mse: 7.4986e-05 - val_loss: 4.7534e-04 - val_accu
racy: 0.0000e+00 - val_mse: 4.7534e-04
Epoch 14/30
172/172 [==============================] - 2s 13ms/step - loss: 8.9596e-05
- accuracy: 3.6390e-04 - mse: 8.9596e-05 - val_loss: 4.4930e-04 - val_accu
racy: 0.0000e+00 - val_mse: 4.4930e-04
Epoch 15/30
172/172 [==============================] - 2s 14ms/step - loss: 7.0248e-05
- accuracy: 3.6390e-04 - mse: 7.0248e-05 - val_loss: 4.5952e-04 - val_accu
racy: 0.0000e+00 - val_mse: 4.5952e-04
```

```
Epoch 16/30
172/172 [==============================] - 2s 13ms/step - loss: 7.7815e-05
- accuracy: 3.6390e-04 - mse: 7.7815e-05 - val_loss: 4.2379e-04 - val_accu
racy: 0.0000e+00 - val_mse: 4.2379e-04
Epoch 17/30
172/172 [==============================] - 2s 13ms/step - loss: 7.4541e-05
- accuracy: 3.6390e-04 - mse: 7.4541e-05 - val_loss: 3.8646e-04 - val_accu
racy: 0.0000e+00 - val_mse: 3.8646e-04
Epoch 18/30
172/172 [==============================] - 2s 13ms/step - loss: 6.2012e-05
- accuracy: 3.6390e-04 - mse: 6.2012e-05 - val_loss: 3.7798e-04 - val_accu
racy: 0.0000e+00 - val_mse: 3.7798e-04
Epoch 19/30
172/172 [==============================] - 2s 14ms/step - loss: 7.1943e-05
- accuracy: 3.6390e-04 - mse: 7.1943e-05 - val_loss: 4.2134e-04 - val_accu
racy: 0.0000e+00 - val_mse: 4.2134e-04
Epoch 20/30
172/172 [==============================] - 2s 13ms/step - loss: 5.0882e-05
- accuracy: 3.6390e-04 - mse: 5.0882e-05 - val_loss: 3.2875e-04 - val_accu
racy: 0.0000e+00 - val_mse: 3.2875e-04
Epoch 21/30
172/172 [==============================] - 2s 13ms/step - loss: 5.3817e-05
- accuracy: 3.6390e-04 - mse: 5.3817e-05 - val_loss: 4.4011e-04 - val_accu
racy: 0.0000e+00 - val_mse: 4.4011e-04
Epoch 22/30
172/172 [==============================] - 2s 13ms/step - loss: 6.0539e-05
- accuracy: 3.6390e-04 - mse: 6.0539e-05 - val_loss: 3.5416e-04 - val_accu
racy: 0.0000e+00 - val_mse: 3.5416e-04
Epoch 23/30
172/172 [==============================] - 2s 13ms/step - loss: 6.3887e-05
- accuracy: 3.6390e-04 - mse: 6.3887e-05 - val_loss: 4.4618e-04 - val_accu
racy: 0.0000e+00 - val_mse: 4.4618e-04
Epoch 24/30
172/172 [==============================] - 2s 13ms/step - loss: 6.8581e-05
- accuracy: 3.6390e-04 - mse: 6.8581e-05 - val_loss: 3.9532e-04 - val_accu
racy: 0.0000e+00 - val_mse: 3.9532e-04
Epoch 25/30
172/172 [==============================] - 2s 14ms/step - loss: 5.0205e-05
- accuracy: 3.6390e-04 - mse: 5.0205e-05 - val_loss: 3.3119e-04 - val_accu
racy: 0.0000e+00 - val_mse: 3.3119e-04
Epoch 26/30
172/172 [==============================] - 2s 13ms/step - loss: 4.9257e-05
- accuracy: 3.6390e-04 - mse: 4.9257e-05 - val_loss: 2.4746e-04 - val_accu
racy: 0.0000e+00 - val_mse: 2.4746e-04
Epoch 27/30
172/172 [==============================] - 2s 14ms/step - loss: 5.3369e-05
- accuracy: 3.6390e-04 - mse: 5.3369e-05 - val_loss: 3.7328e-04 - val_accu
racy: 0.0000e+00 - val_mse: 3.7328e-04
Epoch 28/30
172/172 [==============================] - 2s 13ms/step - loss: 5.9304e-05
- accuracy: 3.6390e-04 - mse: 5.9304e-05 - val_loss: 0.0015 - val_accurac
y: 0.0000e+00 - val_mse: 0.0015
Epoch 29/30
172/172 [==============================] - 2s 13ms/step - loss: 5.5551e-05
- accuracy: 3.6390e-04 - mse: 5.5551e-05 - val_loss: 2.6577e-04 - val_accu
racy: 0.0000e+00 - val_mse: 2.6577e-04
Epoch 30/30
172/172 [==============================] - 2s 13ms/step - loss: 4.0755e-05
- accuracy: 3.6390e-04 - mse: 4.0755e-05 - val_loss: 2.2722e-04 - val_accu
racy: 0.0000e+00 - val_mse: 2.2722e-04
```

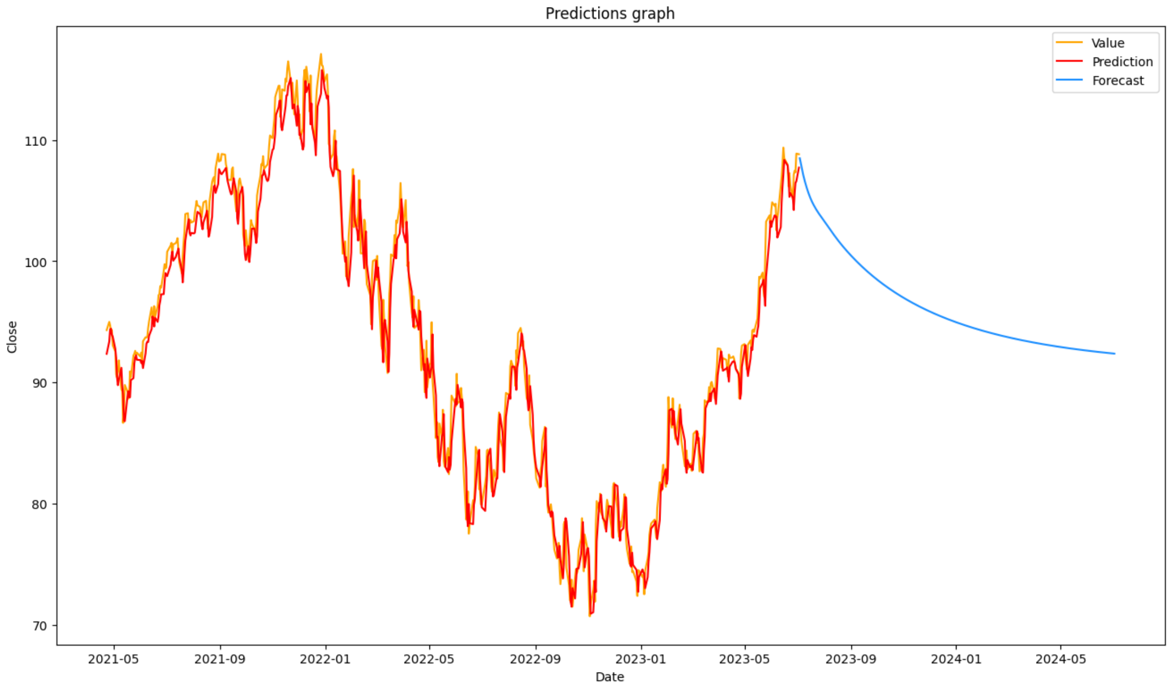
Out[ ]: <keras.src.callbacks.History at 0x29df0a210>

In [ ]:
```
d_time = dt.timedelta(days = 1)
start_prediction_time = last_date + d_time

one_year_future_d = []
one_year_future_val = []
last_30 = []
last_30 = df.filter(['Close']).values[-30:]
last_30 = scaler.transform(last_30)

for i in range(0,365):
    one_year_future_d.append(start_prediction_time + d_time * i)

for i in range(0,365):
    last_30 = last_30[-30:]
    last_30 = np.reshape(last_30,(1,-1,1))
    future_predict = model2.predict(last_30)
    last_30 = np.reshape(last_30, -1)
    last_30 = [i for i in last_30]
    last_30.append(future_predict[0][0])
    one_year_future_val.append(future_predict[0][0])

one_year_future_val = np.reshape(one_year_future_val,(-1,1))
one_year_future_val = scaler.inverse_transform(one_year_future_val)
one_year_future_val = [x[0] for x in one_year_future_val]
future_df = pd.DataFrame({'Date': one_year_future_d, 'Prediction': one_ye
clear_output()
```

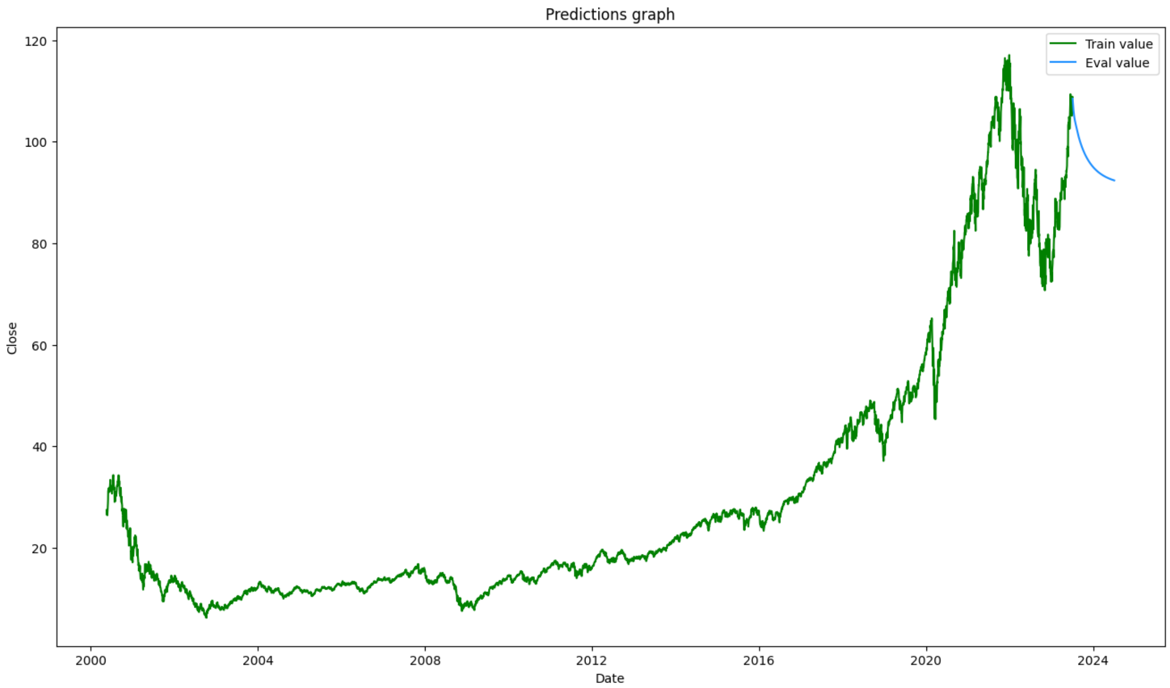
In [ ]:
```
plt.figure(figsize=(16,9))
plt.title('Predictions graph')
plt.xlabel('Date')
plt.ylabel('Close')
plt.plot(future_df['Date'], future_df['Prediction'], c = 'dodgerblue')
plt.legend(['Prediction forecast'])
plt.show()
```



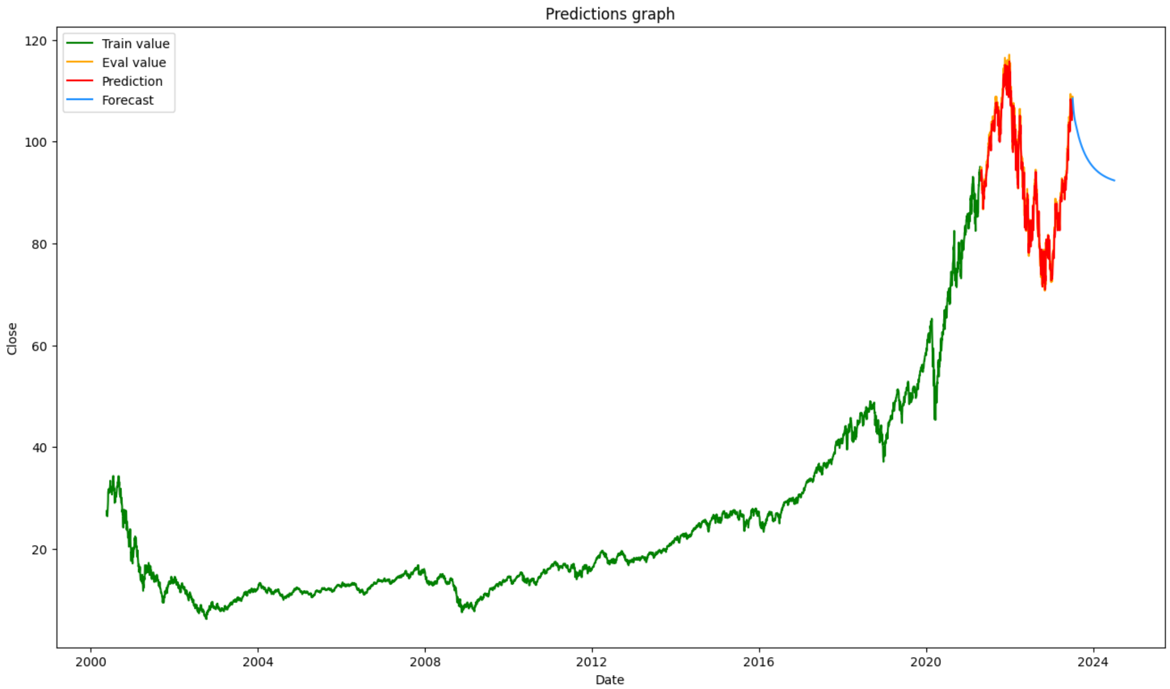
In [ ]:
```
plt.figure(figsize=(16,9))
plt.title('Predictions graph')
plt.xlabel('Date')
plt.ylabel('Close')
plt.plot(test_chart['Close'], c = 'orange')
plt.plot(test_chart['Predictions'], c = 'red')
plt.plot(future_df['Date'], future_df['Prediction'], c = 'dodgerblue')
plt.legend(['Value', 'Prediction', 'Forecast'])
plt.show()
```



In [ ]:
```
plt.figure(figsize=(16,9))
plt.title('Predictions graph')
plt.xlabel('Date')
plt.ylabel('Close')
plt.plot(df['Close'], c = 'green')
plt.plot(future_df['Date'], future_df['Prediction'], c = 'dodgerblue')
plt.legend(['Train value', 'Eval value', 'Prediction'])
plt.show()
```



In [ ]:
```
plt.figure(figsize=(16,9))
plt.title('Predictions graph')
plt.xlabel('Date')
plt.ylabel('Close')
plt.plot(train_chart['Close'], c = 'green')
plt.plot(test_chart['Close'], c = 'orange')
plt.plot(test_chart['Predictions'], c = 'red')
plt.plot(future_df['Date'], future_df['Prediction'], c = 'dodgerblue')

plt.legend(['Train value', 'Eval value', 'Prediction', 'Forecast'])
x = plt.show()
```



In [ ]:
```
future_df.index = future_df.index+1
future_df.to_csv('2023.07.01-2024.06.29-predictions.csv')
model.save('history-predictor')
model2.save('future-predictor')
```

```
INFO:tensorflow:Assets written to: history-predictor/assets
```

```
INFO:tensorflow:Assets written to: history-predictor/assets
INFO:tensorflow:Assets written to: future-predictor/assets
```

```
INFO:tensorflow:Assets written to: future-predictor/assets
```

In [ ]:

```python
future_df.head(30)
```

Out[ ]:

	Date	Prediction
1	2023-07-04	108.487152
2	2023-07-05	108.145027
3	2023-07-06	107.779419
4	2023-07-07	107.430679
5	2023-07-08	107.110229
6	2023-07-09	106.817047
7	2023-07-10	106.546799
8	2023-07-11	106.295761
9	2023-07-12	106.061295
10	2023-07-13	105.840591
11	2023-07-14	105.632332
12	2023-07-15	105.436447
13	2023-07-16	105.252686
14	2023-07-17	105.082420
15	2023-07-18	104.924225
16	2023-07-19	104.777313
17	2023-07-20	104.639366
18	2023-07-21	104.509247
19	2023-07-22	104.385345
20	2023-07-23	104.265984
21	2023-07-24	104.151154
22	2023-07-25	104.040298
23	2023-07-26	103.933769
24	2023-07-27	103.829872
25	2023-07-28	103.728722
26	2023-07-29	103.630180
27	2023-07-30	103.532143
28	2023-07-31	103.434212
29	2023-08-01	103.336067
30	2023-08-02	103.236496