In [ ]:

```
#Installing/Updating libraries
pip install numpy
pip install pandas
pip install pyplot
pip install seaborn
pip install sklearn
pip install keras
pip install tensorflow
```

Bitcoin forecast using LSTM

In [33]:

```
#Data manipulation libraries
import numpy as np
import pandas as pd
#Data visualisation libraries
from matplotlib import pyplot as plt
import seaborn as sns
```

Now, pulling Bitcoin data from Yahoo finance(Historical Data)

In [34]:

```
#only 370 instances
bitcoinPath2= "https://raw.githubusercontent.com/DamianWasTaken/Python/main/BTC-USD.csv"
#Used an user's Github BTC-USD upload instead of downloading and uploading it again from 1/10/2013 to 17/04/202
bitcoinPath = "https://raw.githubusercontent.com/Milisha-gupta/Status-Neo/main/BTC_USD_2013-10-01_2021-04-17-Co

btcData = pd.read_csv(bitcoinPath)

btcDataSorted = btcData.sort_values('Date')

#confirming date
btcDataSorted.head()
```

Out[34]:

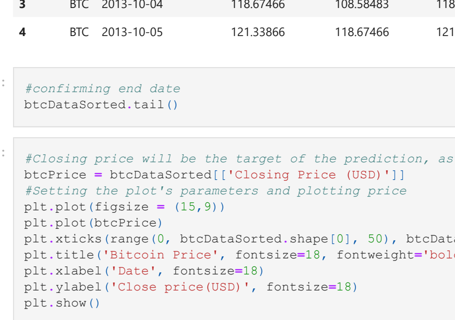
	Currency	Date	Closing Price (USD)	24h Open (USD)	24h High (USD)	24h Low (USD)
0	BTC	2013-10-01	123.65499	124.30466	124.75166	122.56349
1	BTC	2013-10-02	125.45500	123.65499	125.75850	123.63383
2	BTC	2013-10-03	108.58483	125.45500	125.66566	83.32833
3	BTC	2013-10-04	118.67466	108.58483	118.67500	107.05816
4	BTC	2013-10-05	121.33866	118.67466	121.93633	118.00566

In [ ]:

```
#confirming end date
btcDataSorted.tail()
```

In [35]:

```
#Closing price will be the target of the prediction, as such it needs to be separated from the rest of the data
btcPrice = btcDataSorted[['Closing Price (USD)']]
#Setting the plot's parameters and plotting price
plt.plot(figsize = (15,9))
plt.plot(btcPrice)
plt.xticks(range(0, btcDataSorted.shape[0], 50), btcDataSorted['Date'].loc[::50], rotation=45)
plt.title('Bitcoin Price', fontsize=18, fontweight='bold')
plt.xlabel('Date', fontsize=18)
plt.ylabel('Close price(USD)', fontsize=18)
plt.show()
```



In [6]:

```
#Looking at the instances and data type
btcPrice.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 2755 entries, 0 to 2754
Data columns (total 1 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   Closing Price (USD)  2755 non-null   float64
dtypes: float64(1)
memory usage: 43.0 KB
```

Preparing Data

In [36]:

```
#Normalising data, essentially putting all variables on the same scale
#so that certain disproportionate variable's values don't influence the weights too much
from sklearn.preprocessing import MinMaxScaler
minMax = MinMaxScaler() #default value for scaling is 0-10, scaling should be adjusted depending on the activat
normalisedData =  minMax.fit_transform(btcPrice.values)

normalisedData
```

Out[36]:

```
array([[2.38307842e-04],
       [2.66771807e-04],
       [0.00000000e+00],
       ...,
       [9.93354779e-01],
       [1.00000000e+00],
       [9.78161833e-01]])
```

Splitting data into training and testing, this needs to be done so as to evaluate the results obtained from testing the network that was trained on the training data set by feeding it data that it has never seen before, a much more effective and reliable way of splitting the data for this is bootstraping,

the % depends entirely on how much data you have, if it's hundreds of thousands to millions of instances, apply a ratio of about 2-5% Testing and 95-98% Training, if less around 20-80% should do the trick, obviously this is dependent on research question and data at hand

In [49]:

```
    def univariate_data(dataset, start_index, end_index, history_size, target_size):
        data = []
        labels = []

        start_index = start_index + history_size
        if end_index is None:
            end_index = len(dataset) - target_size

        for i in range(start_index, end_index):
            indices = range(i-history_size, i)

            data.append(np.reshape(dataset[indices],(history_size, 1)))
            labels.append(dataset[i + target_size])
        return np.array(data), np.array(labels)


    #These values represent days, essentailly we're using 5 days
    #of time searies events to learn paterns in the data and predict the furute_target
    past_history = 5
    future_target = 0

    #80-20% ratio split
    trainSplit = int(len(normalisedData)* 0.8)


    #Using univariate_data function to split the data into Train and Test
    x_Train, y_Train = univariate_data(normalisedData,
                                       0,
                                       trainSplit,
                                       past_history,
                                       future_target)

    x_Test, y_Test = univariate_data(normalisedData,
                                     trainSplit,
                                     None,
                                     past_history,
                                     future_target)
```

LSTM Model Building

This is where we can do adjustments that will have the major impacts in efficiency, by adjusting the Hyper parameters, whilst there are a few theories into hyper parameters selection, it's highly dependent on the Data and use case, as such deciding empirically usually brings good results

In [50]:

```
import keras
from keras.models import Sequential
from tensorflow.keras.optimizers import Adam
from keras.layers import Dense, LSTM, LeakyReLU, Dropout

#set the number or neurons, learning rate, function, optimiser function, loss function, batch and epochs, essen
num_units = 64
learning_Rate = 0.0001
activation_function = 'sigmoid'
adam = Adam(learning_rate=learning_Rate)
loss_function = 'mse'
batch_size = 5
num_epochs = 250

#Calling on sequential and putting everything together
model = Sequential()
model.add(LSTM(units = num_units, activation = activation_function, input_shape = (None, 1)))
model.add(LeakyReLU(alpha = 0.5))
model.add(Dropout(0.1))
model.add(Dense(units = 1))



model.compile(optimizer = adam, loss=loss_function)
```

In [16]:

```
model.summary()
```

```
Model: "sequential_4"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
lstm_4 (LSTM)                (None, 64)                16896
_________________________________________________________________
leaky_re_lu_1 (LeakyReLU)    (None, 64)                0
_________________________________________________________________
dropout_1 (Dropout)          (None, 64)                0
_________________________________________________________________
dense_1 (Dense)              (None, 1)                 65
=================================================================
Total params: 16,961
Trainable params: 16,961
Non-trainable params: 0
_________________________________________________________________
```

Training model

In [51]:

```
history = model.fit(
    x_Train,
    y_Train,
    validation_split = 0.1,
    batch_size = batch_size,
    epochs = num_epochs,
    shuffle=False
)
```

```
Epoch 1/250
396/396 [==============================] - 2s 3ms/step - loss: 0.0440 - val_loss: 0.0036
Epoch 2/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0208 - val_loss: 0.0064
Epoch 3/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0195 - val_loss: 0.0030
Epoch 4/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0175 - val_loss: 0.0050
Epoch 5/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0149 - val_loss: 0.0052
Epoch 6/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0136 - val_loss: 0.0035
Epoch 7/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0127 - val_loss: 0.0065
Epoch 8/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0117 - val_loss: 0.0061
Epoch 9/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0101 - val_loss: 0.0061
Epoch 10/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0095 - val_loss: 0.0053
Epoch 11/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0087 - val_loss: 0.0070
Epoch 12/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0071 - val_loss: 0.0046
Epoch 13/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0069 - val_loss: 0.0057
Epoch 14/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0061 - val_loss: 0.0059
Epoch 15/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0057 - val_loss: 0.0057
Epoch 16/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0050 - val_loss: 0.0059
Epoch 17/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0044 - val_loss: 0.0050
Epoch 18/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0043 - val_loss: 0.0057
Epoch 19/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0038 - val_loss: 0.0056
Epoch 20/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0034 - val_loss: 0.0052
Epoch 21/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0033 - val_loss: 0.0059
Epoch 22/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0031 - val_loss: 0.0053
Epoch 23/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0028 - val_loss: 0.0050
Epoch 24/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0025 - val_loss: 0.0058
Epoch 25/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0024 - val_loss: 0.0053
Epoch 26/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0023 - val_loss: 0.0052
Epoch 27/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0022 - val_loss: 0.0055
Epoch 28/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0019 - val_loss: 0.0059
Epoch 29/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0018 - val_loss: 0.0058
Epoch 30/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0017 - val_loss: 0.0053
Epoch 31/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0017 - val_loss: 0.0053
Epoch 32/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0015 - val_loss: 0.0057
Epoch 33/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0013 - val_loss: 0.0060
Epoch 34/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0014 - val_loss: 0.0067
Epoch 35/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0013 - val_loss: 0.0059
Epoch 36/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0012 - val_loss: 0.0059
Epoch 37/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0012 - val_loss: 0.0059
Epoch 38/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0011 - val_loss: 0.0059
Epoch 39/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0011 - val_loss: 0.0058
Epoch 40/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0011 - val_loss: 0.0056
Epoch 41/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0011 - val_loss: 0.0056
Epoch 42/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0010 - val_loss: 0.0056
Epoch 43/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0010 - val_loss: 0.0058
Epoch 44/250
396/396 [==============================] - 1s 2ms/step - loss: 0.0010 - val_loss: 0.0059
Epoch 45/250
396/396 [==============================] - 1s 2ms/step - loss: 9.9184e-04 - val_loss: 0.0056
Epoch 46/250
396/396 [==============================] - 1s 2ms/step - loss: 9.4758e-04 - val_loss: 0.0059
Epoch 47/250
396/396 [==============================] - 1s 3ms/step - loss: 9.6341e-04 - val_loss: 0.0057
Epoch 48/250
396/396 [==============================] - 1s 3ms/step - loss: 9.6363e-04 - val_loss: 0.0058
Epoch 49/250
396/396 [==============================] - 1s 3ms/step - loss: 9.1962e-04 - val_loss: 0.0055
Epoch 50/250
396/396 [==============================] - 1s 3ms/step - loss: 9.2602e-04 - val_loss: 0.0055
Epoch 51/250
396/396 [==============================] - 1s 3ms/step - loss: 9.1298e-04 - val_loss: 0.0056
Epoch 52/250
396/396 [==============================] - 1s 3ms/step - loss: 8.9952e-04 - val_loss: 0.0057
Epoch 53/250
396/396 [==============================] - 1s 3ms/step - loss: 8.9427e-04 - val_loss: 0.0059
Epoch 54/250
396/396 [==============================] - 1s 3ms/step - loss: 8.9517e-04 - val_loss: 0.0055
Epoch 55/250
396/396 [==============================] - 1s 3ms/step - loss: 9.2237e-04 - val_loss: 0.0055
Epoch 56/250
396/396 [==============================] - 1s 3ms/step - loss: 8.6985e-04 - val_loss: 0.0058
Epoch 57/250
396/396 [==============================] - 1s 3ms/step - loss: 8.8386e-04 - val_loss: 0.0055
Epoch 58/250
396/396 [==============================] - 1s 3ms/step - loss: 9.0334e-04 - val_loss: 0.0056
Epoch 59/250
396/396 [==============================] - 1s 3ms/step - loss: 8.8865e-04 - val_loss: 0.0058
Epoch 60/250
396/396 [==============================] - 1s 3ms/step - loss: 8.5589e-04 - val_loss: 0.0055
Epoch 61/250
396/396 [==============================] - 1s 3ms/step - loss: 8.8072e-04 - val_loss: 0.0058
Epoch 62/250
396/396 [==============================] - 1s 3ms/step - loss: 8.9077e-04 - val_loss: 0.0054
Epoch 63/250
396/396 [==============================] - 1s 3ms/step - loss: 8.6209e-04 - val_loss: 0.0056
Epoch 64/250
396/396 [==============================] - 1s 3ms/step - loss: 9.1208e-04 - val_loss: 0.0053
Epoch 65/250
396/396 [==============================] - 1s 3ms/step - loss: 8.8438e-04 - val_loss: 0.0055
Epoch 66/250
396/396 [==============================] - 1s 3ms/step - loss: 8.7617e-04 - val_loss: 0.0056
Epoch 67/250
396/396 [==============================] - ETA: 0s - loss: 8.7311e-0 - 1s 3ms/step - loss: 8.7205e-04 - val_los
s: 0.0055
Epoch 68/250
396/396 [==============================] - 1s 3ms/step - loss: 8.9616e-04 - val_loss: 0.0054
Epoch 69/250
396/396 [==============================] - 1s 3ms/step - loss: 8.8850e-04 - val_loss: 0.0056
Epoch 70/250
396/396 [==============================] - 1s 3ms/step - loss: 8.7989e-04 - val_loss: 0.0054
Epoch 71/250
396/396 [==============================] - 1s 3ms/step - loss: 8.8701e-04 - val_loss: 0.0052
Epoch 72/250
396/396 [==============================] - 1s 3ms/step - loss: 9.1396e-04 - val_loss: 0.0054
Epoch 73/250
396/396 [==============================] - 1s 3ms/step - loss: 9.0739e-04 - val_loss: 0.0052
Epoch 74/250
396/396 [==============================] - 1s 3ms/step - loss: 9.1432e-04 - val_loss: 0.0053
Epoch 75/250
396/396 [==============================] - 1s 3ms/step - loss: 9.1216e-04 - val_loss: 0.0052
Epoch 76/250
396/396 [==============================] - 1s 3ms/step - loss: 9.1733e-04 - val_loss: 0.0052
Epoch 77/250
396/396 [==============================] - 1s 3ms/step - loss: 9.4331e-04 - val_loss: 0.0051
Epoch 78/250
396/396 [==============================] - 1s 3ms/step - loss: 9.5240e-04 - val_loss: 0.0052
Epoch 79/250
396/396 [==============================] - 1s 3ms/step - loss: 9.3269e-04 - val_loss: 0.0049
Epoch 80/250
396/396 [==============================] - 1s 3ms/step - loss: 9.4067e-04 - val_loss: 0.0050
Epoch 81/250
396/396 [==============================] - 1s 3ms/step - loss: 9.4117e-04 - val_loss: 0.0049
Epoch 82/250
396/396 [==============================] - 1s 3ms/step - loss: 9.5418e-04 - val_loss: 0.0049
Epoch 83/250
396/396 [==============================] - 1s 3ms/step - loss: 9.5383e-04 - val_loss: 0.0048
Epoch 84/250
396/396 [==============================] - 1s 3ms/step - loss: 9.8769e-04 - val_loss: 0.0048
Epoch 85/250
396/396 [==============================] - 1s 3ms/step - loss: 9.9778e-04 - val_loss: 0.0047
Epoch 86/250
396/396 [==============================] - 1s 3ms/step - loss: 9.9636e-04 - val_loss: 0.0047
Epoch 87/250
396/396 [==============================] - 1s 3ms/step - loss: 9.9266e-04 - val_loss: 0.0047
Epoch 88/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0010 - val_loss: 0.0047
Epoch 89/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0010 - val_loss: 0.0045
Epoch 90/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0010 - val_loss: 0.0045
Epoch 91/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0010 - val_loss: 0.0044
Epoch 92/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0010 - val_loss: 0.0043
Epoch 93/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0011 - val_loss: 0.0042
Epoch 94/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0011 - val_loss: 0.0042
Epoch 95/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0011 - val_loss: 0.0041
Epoch 96/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0011 - val_loss: 0.0041
Epoch 97/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0011 - val_loss: 0.0040
Epoch 98/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0011 - val_loss: 0.0040
Epoch 99/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0011 - val_loss: 0.0041
Epoch 100/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0011 - val_loss: 0.0039
Epoch 101/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0012 - val_loss: 0.0039
Epoch 102/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0011 - val_loss: 0.0038
Epoch 103/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0012 - val_loss: 0.0038
Epoch 104/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0012 - val_loss: 0.0038
Epoch 105/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0012 - val_loss: 0.0036
Epoch 106/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0012 - val_loss: 0.0035
Epoch 107/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0012 - val_loss: 0.0036
Epoch 108/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0012 - val_loss: 0.0034
Epoch 109/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0012 - val_loss: 0.0035
Epoch 110/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0012 - val_loss: 0.0034
Epoch 111/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0034
Epoch 112/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0033
Epoch 113/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0033
Epoch 114/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0032
Epoch 115/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0032
Epoch 116/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0033
Epoch 117/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0031
Epoch 118/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0030
Epoch 119/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0031
Epoch 120/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0030
Epoch 121/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0029
Epoch 122/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0030
Epoch 123/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0029
Epoch 124/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0028
Epoch 125/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0030
Epoch 126/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0028
Epoch 127/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0028
Epoch 128/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0028
Epoch 129/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0027
Epoch 130/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0028
Epoch 131/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0028
Epoch 132/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0028
Epoch 133/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0027
Epoch 134/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0026
Epoch 135/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0027
Epoch 136/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0026
Epoch 137/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0026
Epoch 138/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0026
Epoch 139/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0026
Epoch 140/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0025
Epoch 141/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0025
Epoch 142/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0025
Epoch 143/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0025
Epoch 144/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0025
Epoch 145/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0024
Epoch 146/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0024
Epoch 147/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0024
Epoch 148/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0024
Epoch 149/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0024
Epoch 150/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0024
Epoch 151/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0024
Epoch 152/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0023
Epoch 153/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0024
Epoch 154/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0024
Epoch 155/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0024
Epoch 156/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0014 - val_loss: 0.0024
Epoch 157/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0023
Epoch 158/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0023
Epoch 159/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0023
Epoch 160/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0023
Epoch 161/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0023
Epoch 162/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0023
Epoch 163/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0023
Epoch 164/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0023
Epoch 165/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0023
Epoch 166/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0022
Epoch 167/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0022
Epoch 168/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0023
Epoch 169/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0022
Epoch 170/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0022
Epoch 171/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0012 - val_loss: 0.0022
Epoch 172/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0013 - val_loss: 0.0022
Epoch 173/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0012 - val_loss: 0.0022
Epoch 174/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0012 - val_loss: 0.0022
Epoch 175/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0012 - val_loss: 0.0022
Epoch 176/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0012 - val_loss: 0.0021
Epoch 177/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0012 - val_loss: 0.0021
Epoch 178/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0012 - val_loss: 0.0021
Epoch 179/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0011 - val_loss: 0.0021
Epoch 180/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0011 - val_loss: 0.0022
Epoch 181/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0011 - val_loss: 0.0021
Epoch 182/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0011 - val_loss: 0.0021
Epoch 183/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0011 - val_loss: 0.0021
Epoch 184/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0011 - val_loss: 0.0021
Epoch 185/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0010 - val_loss: 0.0020
Epoch 186/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0011 - val_loss: 0.0020
Epoch 187/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0010 - val_loss: 0.0020
Epoch 188/250
396/396 [==============================] - 1s 3ms/step - loss: 0.0010 - val_loss: 0.0020
Epoch 189/250
396/396 [==============================] - 1s 3ms/step - loss: 9.9696e-04 - val_loss: 0.0020
Epoch 190/250
396/396 [==============================] - 1s 3ms/step - loss: 9.7903e-04 - val_loss: 0.0020
Epoch 191/250
396/396 [==============================] - 1s 3ms/step - loss: 9.6402e-04 - val_loss: 0.0020
Epoch 192/250
396/396 [==============================] - 1s 3ms/step - loss: 9.6539e-04 - val_loss: 0.0020
Epoch 193/250
396/396 [==============================] - 1s 3ms/step - loss: 9.5777e-04 - val_loss: 0.0020
Epoch 194/250
396/396 [==============================] - 1s 3ms/step - loss: 9.2691e-04 - val_loss: 0.0019
Epoch 195/250
396/396 [==============================] - 1s 3ms/step - loss: 9.1344e-04 - val_loss: 0.0018
Epoch 196/250
396/396 [==============================] - 1s 3ms/step - loss: 9.3145e-04 - val_loss: 0.0020
Epoch 197/250
396/396 [==============================] - 1s 3ms/step - loss: 9.0111e-04 - val_loss: 0.0019
Epoch 198/250
396/396 [==============================] - 1s 3ms/step - loss: 8.8826e-04 - val_loss: 0.0020
Epoch 199/250
396/396 [==============================] - 1s 3ms/step - loss: 8.4433e-04 - val_loss: 0.0019
Epoch 200/250
396/396 [==============================] - 1s 3ms/step - loss: 8.6200e-04 - val_loss: 0.0019
Epoch 201/250
396/396 [==============================] - 1s 3ms/step - loss: 8.7014e-04 - val_loss: 0.0019
Epoch 202/250
396/396 [==============================] - 1s 3ms/step - loss: 8.2444e-04 - val_loss: 0.0018
Epoch 203/250
396/396 [==============================] - 1s 3ms/step - loss: 8.3823e-04 - val_loss: 0.0019
Epoch 204/250
396/396 [==============================] - 1s 3ms/step - loss: 7.7195e-04 - val_loss: 0.0018
Epoch 205/250
396/396 [==============================] - 1s 3ms/step - loss: 7.8947e-04 - val_loss: 0.0019
Epoch 206/250
396/396 [==============================] - 1s 3ms/step - loss: 7.4853e-04 - val_loss: 0.0019
Epoch 207/250
396/396 [==============================] - 1s 3ms/step - loss: 7.5348e-04 - val_loss: 0.0018
Epoch 208/250
396/396 [==============================] - 1s 3ms/step - loss: 7.1370e-04 - val_loss: 0.0018
Epoch 209/250
396/396 [==============================] - 1s 3ms/step - loss: 7.1323e-04 - val_loss: 0.0018
Epoch 210/250
396/396 [==============================] - 1s 3ms/step - loss: 7.1286e-04 - val_loss: 0.0018
Epoch 211/250
396/396 [==============================] - 1s 3ms/step - loss: 6.8632e-04 - val_loss: 0.0017
Epoch 212/250
396/396 [==============================] - 1s 3ms/step - loss: 6.9073e-04 - val_loss: 0.0018
Epoch 213/250
396/396 [==============================] - 1s 3ms/step - loss: 6.6780e-04 - val_loss: 0.0018
Epoch 214/250
396/396 [==============================] - 1s 3ms/step - loss: 6.4435e-04 - val_loss: 0.0018
Epoch 215/250
396/396 [==============================] - 1s 3ms/step - loss: 6.1563e-04 - val_loss: 0.0018
Epoch 216/250
396/396 [==============================] - 1s 3ms/step - loss: 6.4368e-04 - val_loss: 0.0018
Epoch 217/250
396/396 [==============================] - 1s 3ms/step - loss: 6.0787e-04 - val_loss: 0.0019
Epoch 218/250
396/396 [==============================] - 1s 3ms/step - loss: 6.0720e-04 - val_loss: 0.0017
Epoch 219/250
396/396 [==============================] - 1s 3ms/step - loss: 5.6835e-04 - val_loss: 0.0017
Epoch 220/250
396/396 [==============================] - 1s 3ms/step - loss: 5.6888e-04 - val_loss: 0.0017
Epoch 221/250
396/396 [==============================] - 1s 3ms/step - loss: 5.6732e-04 - val_loss: 0.0017
Epoch 222/250
396/396 [==============================] - 1s 3ms/step - loss: 5.5985e-04 - val_loss: 0.0017
Epoch 223/250
396/396 [==============================] - 1s 3ms/step - loss: 5.2535e-04 - val_loss: 0.0017
Epoch 224/250
396/396 [==============================] - 1s 3ms/step - loss: 5.2692e-04 - val_loss: 0.0016
Epoch 225/250
396/396 [==============================] - 1s 3ms/step - loss: 5.0164e-04 - val_loss: 0.0016
Epoch 226/250
396/396 [==============================] - 1s 3ms/step - loss: 5.1960e-04 - val_loss: 0.0016
Epoch 227/250
396/396 [==============================] - 1s 3ms/step - loss: 4.9410e-04 - val_loss: 0.0017
Epoch 228/250
396/396 [==============================] - 1s 3ms/step - loss: 5.0655e-04 - val_loss: 0.0017
Epoch 229/250
396/396 [==============================] - 1s 3ms/step - loss: 4.9181e-04 - val_loss: 0.0017
Epoch 230/250
396/396 [==============================] - 1s 3ms/step - loss: 4.8086e-04 - val_loss: 0.0017
Epoch 231/250
396/396 [==============================] - 1s 3ms/step - loss: 4.8052e-04 - val_loss: 0.0016
Epoch 232/250
396/396 [==============================] - 1s 3ms/step - loss: 4.6146e-04 - val_loss: 0.0016
Epoch 233/250
396/396 [==============================] - 1s 3ms/step - loss: 4.5167e-04 - val_loss: 0.0017
Epoch 234/250
396/396 [==============================] - 1s 3ms/step - loss: 4.4183e-04 - val_loss: 0.0016
Epoch 235/250
396/396 [==============================] - 1s 3ms/step - loss: 4.4451e-04 - val_loss: 0.0014
Epoch 236/250
396/396 [==============================] - 1s 3ms/step - loss: 4.3032e-04 - val_loss: 0.0016
Epoch 237/250
396/396 [==============================] - 1s 3ms/step - loss: 4.1048e-04 - val_loss: 0.0015
Epoch 238/250
396/396 [==============================] - 1s 3ms/step - loss: 4.1255e-04 - val_loss: 0.0015
Epoch 239/250
396/396 [==============================] - 1s 3ms/step - loss: 4.0867e-04 - val_loss: 0.0015
Epoch 240/250
396/396 [==============================] - 1s 3ms/step - loss: 4.0160e-04 - val_loss: 0.0015
Epoch 241/250
396/396 [==============================] - 1s 3ms/step - loss: 3.6823e-04 - val_loss: 0.0014
Epoch 242/250
396/396 [==============================] - 1s 3ms/step - loss: 3.8743e-04 - val_loss: 0.0014
Epoch 243/250
396/396 [==============================] - 1s 3ms/step - loss: 3.9054e-04 - val_loss: 0.0016
Epoch 244/250
396/396 [==============================] - 1s 3ms/step - loss: 3.5230e-04 - val_loss: 0.0014
Epoch 245/250
396/396 [==============================] - 1s 3ms/step - loss: 3.7384e-04 - val_loss: 0.0014
Epoch 246/250
396/396 [==============================] - 1s 3ms/step - loss: 3.4968e-04 - val_loss: 0.0014
Epoch 247/250
396/396 [==============================] - 1s 3ms/step - loss: 3.5277e-04 - val_loss: 0.0014
Epoch 248/250
396/396 [==============================] - 1s 3ms/step - loss: 3.4096e-04 - val_loss: 0.0014
Epoch 249/250
396/396 [==============================] - 1s 3ms/step - loss: 3.1931e-04 - val_loss: 0.0014
Epoch 250/250
396/396 [==============================] - 1s 3ms/step - loss: 3.2682e-04 - val_loss: 0.0013
```

In [53]:

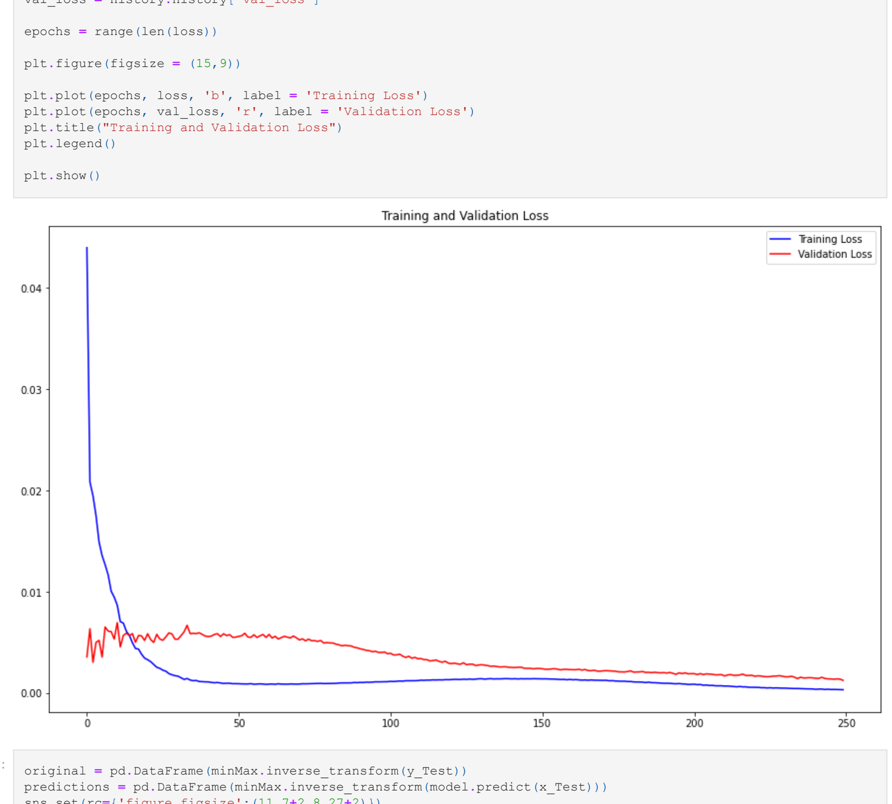
```
loss = history.history['loss']
val_loss = history.history['val_loss']

epochs = range(len(loss))

plt.figure(figsize = (15,9))

plt.plot(epochs, loss, 'b', label = 'Training Loss')
plt.plot(epochs, val_loss, 'r', label = 'Validation Loss')
plt.title("Training and Validation Loss")
plt.legend()

plt.show()
```



In [58]:

```
original = pd.DataFrame(minMax.inverse_transform(y_Test))
predictions = pd.DataFrame(minMax.inverse_transform(model.predict(x_Test)))
sns.set(rc={'figure.figsize':(11.7+2,8.27+2)})
ax = sns.lineplot(x=original.index, y=original[0], label='Test Data', color='royalblue')
ax = sns.lineplot(x=predictions.index, y=predictions[0], label='predictions', color='tomato')
ax.set_title('Bitcoins Price', size = 14, fontweight='bold')
ax.set_xlabel('Days', size=14)
ax.set_ylabel('Cost (USD)', size = 14)
ax.set_xticklabels('', size=10)
```

Out[58]:

```
[Text(-100.0, 0, ''),
 Text(0.0, 0, ''),
 Text(100.0, 0, ''),
 Text(200.0, 0, ''),
 Text(300.0, 0, ''),
 Text(400.0, 0, ''),
 Text(500.0, 0, ''),
 Text(600.0, 0, '')]
```