Import Libraries

In [7]:
```
!pip install tensorflow
```

```
Requirement already satisfied: tensorflow in c:\users\pramo\anaconda3\lib\sit
e-packages (2.9.1)
Requirement already satisfied: numpy>=1.20 in c:\users\pramo\anaconda3\lib\si
te-packages (from tensorflow) (1.21.5)
Requirement already satisfied: google-pasta>=0.1.1 in c:\users\pramo\anaconda
3\lib\site-packages (from tensorflow) (0.2.0)
Requirement already satisfied: keras<2.10.0,>=2.9.0rc0 in c:\users\pramo\anac
onda3\lib\site-packages (from tensorflow) (2.9.0)
Requirement already satisfied: flatbuffers<2,>=1.12 in c:\users\pramo\anacond
a3\lib\site-packages (from tensorflow) (1.12)
Requirement already satisfied: keras-preprocessing>=1.1.1 in c:\users\pramo\a
naconda3\lib\site-packages (from tensorflow) (1.1.2)
Requirement already satisfied: libclang>=13.0.0 in c:\users\pramo\anaconda3\l
ib\site-packages (from tensorflow) (14.0.6)
Requirement already satisfied: tensorflow-io-gcs-filesystem>=0.23.1 in c:\use
rs\pramo\anaconda3\lib\site-packages (from tensorflow) (0.26.0)
Requirement already satisfied: typing-extensions>=3.6.6 in c:\users\pramo\ana
conda3\lib\site-packages (from tensorflow) (4.1.1)
Requirement already satisfied: tensorflow-estimator<2.10.0,>=2.9.0rc0 in c:\u
sers\pramo\anaconda3\lib\site-packages (from tensorflow) (2.9.0)
Requirement already satisfied: h5py>=2.9.0 in c:\users\pramo\anaconda3\lib\si
te-packages (from tensorflow) (3.6.0)
Requirement already satisfied: astunparse>=1.6.0 in c:\users\pramo\anaconda3\
lib\site-packages (from tensorflow) (1.6.3)
Requirement already satisfied: grpcio<2.0,>=1.24.3 in c:\users\pramo\anaconda
3\lib\site-packages (from tensorflow) (1.42.0)
Requirement already satisfied: absl-py>=1.0.0 in c:\users\pramo\anaconda3\li
b\site-packages (from tensorflow) (1.2.0)
Requirement already satisfied: termcolor>=1.1.0 in c:\users\pramo\anaconda3\l
ib\site-packages (from tensorflow) (1.1.0)
Requirement already satisfied: tensorboard<2.10,>=2.9 in c:\users\pramo\anaco
nda3\lib\site-packages (from tensorflow) (2.9.1)
Requirement already satisfied: protobuf<3.20,>=3.9.2 in c:\users\pramo\anacon
da3\lib\site-packages (from tensorflow) (3.19.1)
Requirement already satisfied: gast<=0.4.0,>=0.2.1 in c:\users\pramo\anaconda
3\lib\site-packages (from tensorflow) (0.4.0)
Requirement already satisfied: setuptools in c:\users\pramo\anaconda3\lib\sit
e-packages (from tensorflow) (61.2.0)
Requirement already satisfied: packaging in c:\users\pramo\anaconda3\lib\site
-packages (from tensorflow) (21.3)
Requirement already satisfied: wrapt>=1.11.0 in c:\users\pramo\anaconda3\lib\
site-packages (from tensorflow) (1.12.1)
Requirement already satisfied: six>=1.12.0 in c:\users\pramo\anaconda3\lib\si
te-packages (from tensorflow) (1.16.0)
Requirement already satisfied: opt-einsum>=2.3.2 in c:\users\pramo\anaconda3\
lib\site-packages (from tensorflow) (3.3.0)
Requirement already satisfied: wheel<1.0,>=0.23.0 in c:\users\pramo\anaconda
3\lib\site-packages (from astunparse>=1.6.0->tensorflow) (0.37.1)
Requirement already satisfied: requests<3,>=2.21.0 in c:\users\pramo\anaconda
3\lib\site-packages (from tensorboard<2.10,>=2.9->tensorflow) (2.27.1)
Requirement already satisfied: tensorboard-plugin-wit>=1.6.0 in c:\users\pram
o\anaconda3\lib\site-packages (from tensorboard<2.10,>=2.9->tensorflow) (1.8.
1)
Requirement already satisfied: google-auth<3,>=1.6.3 in c:\users\pramo\anacon
```

```
da3\lib\site-packages (from tensorboard<2.10,>=2.9->tensorflow) (1.33.0)
Requirement already satisfied: werkzeug>=1.0.1 in c:\users\pramo\anaconda3\li
b\site-packages (from tensorboard<2.10,>=2.9->tensorflow) (2.0.3)
Requirement already satisfied: tensorboard-data-server<0.7.0,>=0.6.0 in c:\us
ers\pramo\anaconda3\lib\site-packages (from tensorboard<2.10,>=2.9->tensorflo
w) (0.6.1)
Requirement already satisfied: google-auth-oauthlib<0.5,>=0.4.1 in c:\users\p
ramo\anaconda3\lib\site-packages (from tensorboard<2.10,>=2.9->tensorflow)
(0.4.6)
Requirement already satisfied: markdown>=2.6.8 in c:\users\pramo\anaconda3\li
b\site-packages (from tensorboard<2.10,>=2.9->tensorflow) (3.3.4)
Requirement already satisfied: cachetools<5.0,>=2.0.0 in c:\users\pramo\anaco
nda3\lib\site-packages (from google-auth<3,>=1.6.3->tensorboard<2.10,>=2.9->t
ensorflow) (4.2.2)
Requirement already satisfied: rsa<5,>=3.1.4 in c:\users\pramo\anaconda3\lib\
site-packages (from google-auth<3,>=1.6.3->tensorboard<2.10,>=2.9->tensorflo
w) (4.7.2)
Requirement already satisfied: pyasn1-modules>=0.2.1 in c:\users\pramo\anacon
da3\lib\site-packages (from google-auth<3,>=1.6.3->tensorboard<2.10,>=2.9->te
nsorflow) (0.2.8)
Requirement already satisfied: requests-oauthlib>=0.7.0 in c:\users\pramo\ana
conda3\lib\site-packages (from google-auth-oauthlib<0.5,>=0.4.1->tensorboard
<2.10,>=2.9->tensorflow) (1.3.1)
Requirement already satisfied: pyasn1<0.5.0,>=0.4.6 in c:\users\pramo\anacond
a3\lib\site-packages (from pyasn1-modules>=0.2.1->google-auth<3,>=1.6.3->tens
orboard<2.10,>=2.9->tensorflow) (0.4.8)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\pramo\anacon
da3\lib\site-packages (from requests<3,>=2.21.0->tensorboard<2.10,>=2.9->tens
orflow) (1.26.9)
Requirement already satisfied: idna<4,>=2.5 in c:\users\pramo\anaconda3\lib\s
ite-packages (from requests<3,>=2.21.0->tensorboard<2.10,>=2.9->tensorflow)
(3.3)
Requirement already satisfied: charset-normalizer~=2.0.0 in c:\users\pramo\an
aconda3\lib\site-packages (from requests<3,>=2.21.0->tensorboard<2.10,>=2.9->
tensorflow) (2.0.4)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\pramo\anaconda
3\lib\site-packages (from requests<3,>=2.21.0->tensorboard<2.10,>=2.9->tensor
flow) (2021.10.8)
Requirement already satisfied: oauthlib>=3.0.0 in c:\users\pramo\anaconda3\li
b\site-packages (from requests-oauthlib>=0.7.0->google-auth-oauthlib<0.5,>=0.
4.1->tensorboard<2.10,>=2.9->tensorflow) (3.2.0)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in c:\users\pramo\ana
conda3\lib\site-packages (from packaging->tensorflow) (3.0.4)
```

In [8]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import math
from sklearn.preprocessing import MinMaxScaler
from keras.models import Sequential
from keras.layers import Dense,LSTM
plt.style.use('fivethirtyeight')
```

In [9]:
```
df=pd.read_excel('1613615-Stock_Price_data_set.xlsx',index_col='Date')
df.head()
```

Out[9]:

Date	Open	High	Low	Close	Adj Close	Volume
2018-02-05	262.000000	267.899994	250.029999	254.259995	254.259995	11896100.0
2018-02-06	247.699997	266.700012	245.000000	265.720001	265.720001	12595800.0
2018-02-07	266.579987	272.450012	264.329987	264.559998	264.559998	8981500.0
2018-02-08	267.079987	267.619995	250.000000	250.100006	250.100006	9306700.0
2018-02-09	253.850006	255.800003	236.110001	249.470001	249.470001	16906900.0

In [10]:
```
#Get number of rows and columns in data set
df.shape
```

Out[10]: (1009, 6)

In [11]:
```
#visualization of the closing histry
plt.figure(figsize=(16,8))
plt.title('Close price history')
plt.plot(df['Close'])
plt.xlabel('date',fontsize=18)
plt.ylabel('close price ')
plt.show()
```

In [12]:
```
#create a new dataframe with only the close column
data=df.filter(['Close'])
#convert the dataframe into numpy arry
dataset=data.values
#get the number of rows to train the model
training_data_len=math.ceil(len(dataset)*0.8)
training_data_len
training_data_len
```

Out[12]:
```
808
```

In [13]:
```
#scale the data
scaler=MinMaxScaler(feature_range=(0,1))
scaled_data=scaler.fit_transform(dataset)
scaled_data
```

Out[13]:
```
array([[0.04451626],
       [0.06954849],
       [0.06701469],
       ...,
       [0.4272515 ],
       [0.37509011],
       [0.38507243]])
```

In [14]:
```
#create the training data set
#create the scaled training data set
train_data=scaled_data[0:training_data_len,:]
#split the data into x_train and y_train data sets
x_train=[]
y_train=[]

for i in range (60,len(train_data)):
    x_train.append(train_data[i-60:i,0])
    y_train.append(train_data[i,0])
    if i<=61:
        print(x_train)
        print(y_train)
        print()
```

```
[array([0.04451626, 0.06954849, 0.06701469, 0.03542955, 0.03405342,
       0.05257641, 0.05327534, 0.0701601 , 0.10133021, 0.09750767,
       0.09757319, 0.10301218, 0.09667768, 0.11369343, 0.13167034,
       0.12391599, 0.12559796, 0.12343551, 0.14672022, 0.1771914 ,
       0.19951508, 0.19064677, 0.18156003, 0.2131015 , 0.19095254,
       0.17911361, 0.19149862, 0.19049385, 0.18472731, 0.17387127,
       0.18265218, 0.18042421, 0.15906164, 0.14647998, 0.18887749,
       0.1459339 , 0.11334393, 0.13426968, 0.10137394, 0.10875693,
       0.12026823, 0.13125532, 0.12007165, 0.12243068, 0.14021101,
       0.15244317, 0.16463161, 0.16987394, 0.16142066, 0.22319301,
       0.21982915, 0.21585376, 0.20508505, 0.18525152, 0.15976057,
       0.15700838, 0.17496343, 0.17011425, 0.17164323, 0.17347804])]
[0.17360909661393864]

[array([0.04451626, 0.06954849, 0.06701469, 0.03542955, 0.03405342,
       0.05257641, 0.05327534, 0.0701601 , 0.10133021, 0.09750767,
       0.09757319, 0.10301218, 0.09667768, 0.11369343, 0.13167034,
       0.12391599, 0.12559796, 0.12343551, 0.14672022, 0.1771914 ,
       0.19951508, 0.19064677, 0.18156003, 0.2131015 , 0.19095254,
       0.17911361, 0.19149862, 0.19049385, 0.18472731, 0.17387127,
       0.18265218, 0.18042421, 0.15906164, 0.14647998, 0.18887749,
       0.1459339 , 0.11334393, 0.13426968, 0.10137394, 0.10875693,
       0.12026823, 0.13125532, 0.12007165, 0.12243068, 0.14021101,
       0.15244317, 0.16463161, 0.16987394, 0.16142066, 0.22319301,
       0.21982915, 0.21585376, 0.20508505, 0.18525152, 0.15976057,
       0.15700838, 0.17496343, 0.17011425, 0.17164323, 0.17347804]), array
([0.06954849, 0.06701469, 0.03542955, 0.03405342, 0.05257641,
       0.05327534, 0.0701601 , 0.10133021, 0.09750767, 0.09757319,
       0.10301218, 0.09667768, 0.11369343, 0.13167034, 0.12391599,
       0.12559796, 0.12343551, 0.14672022, 0.1771914 , 0.19951508,
       0.19064677, 0.18156003, 0.2131015 , 0.19095254, 0.17911361,
       0.19149862, 0.19049385, 0.18472731, 0.17387127, 0.18265218,
       0.18042421, 0.15906164, 0.14647998, 0.18887749, 0.1459339 ,
       0.11334393, 0.13426968, 0.10137394, 0.10875693, 0.12026823,
       0.13125532, 0.12007165, 0.12243068, 0.14021101, 0.15244317,
       0.16463161, 0.16987394, 0.16142066, 0.22319301, 0.21982915,
       0.21585376, 0.20508505, 0.18525152, 0.15976057, 0.15700838,
       0.17496343, 0.17011425, 0.17164323, 0.17347804, 0.1736091 ])]
[0.17360909661393864, 0.16996133223364263]
```

In [15]:
```
#convert the x_train and y_train to numpy arrays
x_train,y_train=np.array(x_train),np.array(y_train)
```

In [16]:
```
#reshape the data
x_train=np.reshape(x_train,(x_train.shape[0],x_train.shape[1],1))
x_train.shape
```

Out[16]:
```
(748, 60, 1)
```

In [17]:
```
#build the LSTM model
model=Sequential()
model.add(LSTM(50,return_sequences=True,input_shape=(x_train.shape[1],1)))
model.add(LSTM(50,return_sequences=False))
model.add(Dense(25))
model.add(Dense(1))
```

In [18]:
```
#compile the model
model.compile(optimizer='adam',loss='mean_squared_error')
```

In [19]:
```
#train the model
model.fit(x_train,y_train,batch_size=1,epochs=1)
```

```
748/748 [==============================] - 35s 37ms/step - loss: 0.0039
```

Out[19]:
```
<keras.callbacks.History at 0x1b3de5088e0>
```

In [20]:
```
#create the testiong data set
#create the new arry containing scaled values from index
test_data=scaled_data[training_data_len-60:,:]
#create the data set x_test and y_test
x_test=[]
y_test=dataset[training_data_len:, :]
for i in range (60,len(test_data)):
    x_test.append(test_data[i-60:i,0])
```

In [21]:
```
#convert data into numpy
x_test=np.array(x_test)
```

In [22]:
```
#reshape the data
x_test=np.reshape(x_test,(x_test.shape[0],x_test.shape[1],1))
```

In [23]:
```
#get the model predicted values
predictions=model.predict(x_test)
predictions=scaler.inverse_transform(predictions)
predictions
```

```
7/7 [==============================] - 2s 29ms/step
```

Out[23]:
```
array([[544.38666],
       [539.58795],
       [534.2168 ],
       [529.5256 ],
       [525.17456],
       [521.51184],
       [518.76306],
       [517.1312 ],
       [515.79596],
       [514.2001 ],
       [511.98685],
       [510.00726],
       [508.71066],
       [506.29517],
       [504.3575 ],
       [501.89258],
       [499.57373],
       [498.15106]
```

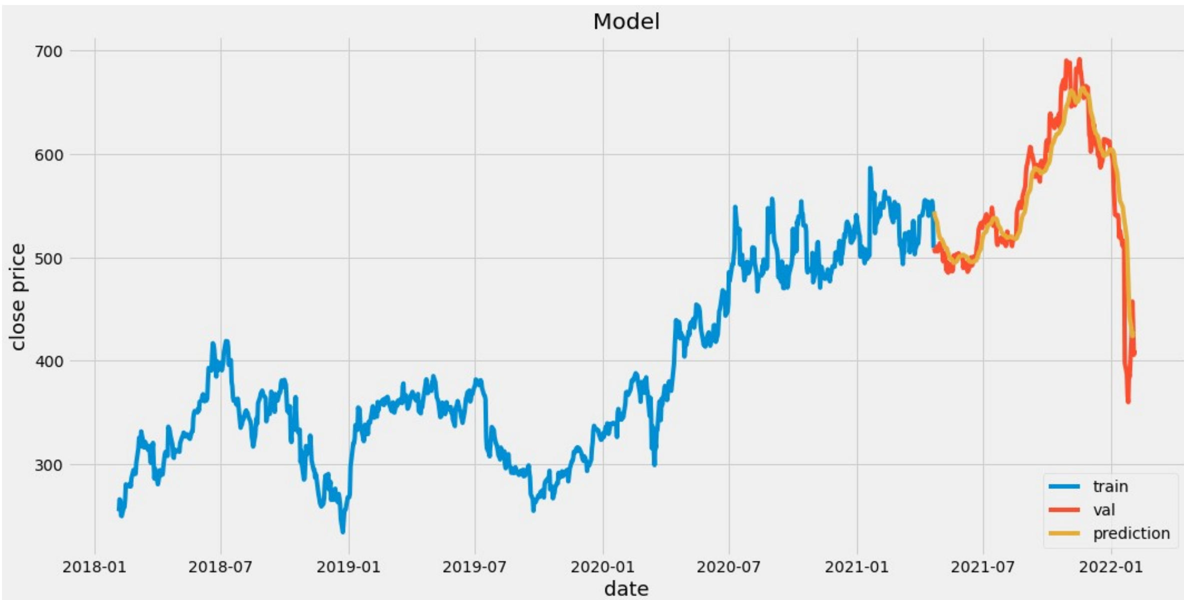
In [24]:
```
# get the root mean squared error(RMSE)
rmse=np.sqrt(np.mean(predictions-y_test)**2)
rmse
```

Out[24]:
```
1.412755384202038
```

In [25]:
```
#plot the data
train=data[:training_data_len]
valid=data[training_data_len:]
valid['predictions']=predictions
#visualization of data
plt.figure(figsize=(16,8))
plt.title('Model')
plt.xlabel('date',fontsize=18)
plt.ylabel('close price',fontsize=18)
plt.plot(train['Close'])
plt.plot(valid[['Close','predictions']])
plt.legend(['train','val','prediction'],loc='lower right')
plt.show()
```

```
C:\Users\pramo\AppData\Local\Temp\ipykernel_7940\2617299793.py:4: SettingWith
CopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/s
table/user_guide/indexing.html#returning-a-view-versus-a-copy (https://panda
s.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-ver
sus-a-copy)
  valid['predictions']=predictions
```



In [26]:
```
# show the valid and actual prices
valid
```

Out[26]:

	Close	predictions
Date		
2021-04-22	508.779999	544.386658
2021-04-23	505.549988	539.587952
2021-04-26	510.299988	534.216797
2021-04-27	505.549988	529.525574
2021-04-28	506.519989	525.174561
...	...	...
2022-01-31	427.140015	429.565735
2022-02-01	457.130005	423.688477
2022-02-02	429.480011	424.394684
2022-02-03	405.600006	425.844879
2022-02-04	410.170013	425.393127

201 rows × 2 columns

In [ ]: