Earthquake Prediction Model using Python

AI_Phase3

Development Part-1

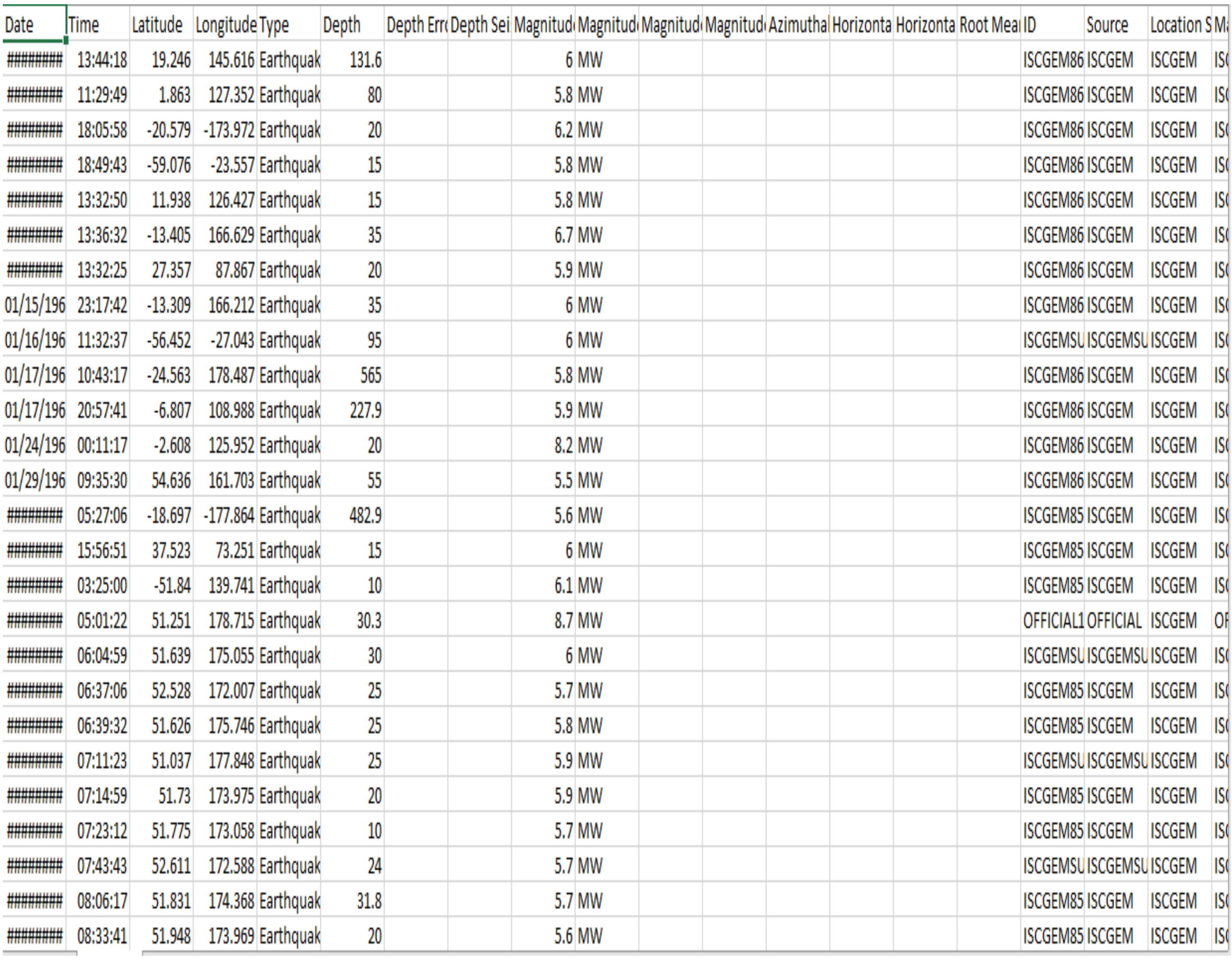
Introduction:

To begin building an earthquake prediction project using Python, you first need to load and preprocess the dataset. The dataset could contain information about earthquake occurrences, such as location, time, magnitude, and other relevant factors.

Loading the dataset involves importing the required Python libraries, such as pandas, numpy and reading the dataset file, which could be in formats like CSV or Excel.

By loading and preprocessing the dataset effectively, you create a solid foundation for further analysis and model development in earthquake prediction using Python.

Data Sheet:



Date	Time	Latitude	Longitude	Type	Depth	Depth Err	Depth Sei	Magnitud	Magnitud	Magnitud	Magnitud	Azimuthal	Horizonta	Horizonta	Root Mea	ID	Source	Location S	Ma
########	13:44:18	19.246	145.616	Earthquak	131.6			6	MW							ISCGEM86	ISCGEM	ISCGEM	IS
########	11:29:49	1.863	127.352	Earthquak	80			5.8	MW							ISCGEM86	ISCGEM	ISCGEM	IS
########	18:05:58	-20.579	-173.972	Earthquak	20			6.2	MW							ISCGEM86	ISCGEM	ISCGEM	IS
########	18:49:43	-59.076	-23.557	Earthquak	15			5.8	MW							ISCGEM86	ISCGEM	ISCGEM	IS
########	13:32:50	11.938	126.427	Earthquak	15			5.8	MW							ISCGEM86	ISCGEM	ISCGEM	IS
########	13:36:32	-13.405	166.629	Earthquak	35			6.7	MW							ISCGEM86	ISCGEM	ISCGEM	IS
########	13:32:25	27.357	87.867	Earthquak	20			5.9	MW							ISCGEM86	ISCGEM	ISCGEM	IS
01/15/196	23:17:42	-13.309	166.212	Earthquak	35			6	MW							ISCGEM86	ISCGEM	ISCGEM	IS
01/16/196	11:32:37	-56.452	-27.043	Earthquak	95			6	MW							ISCGEMSU	ISCGEMSU	ISCGEM	IS
01/17/196	10:43:17	-24.563	178.487	Earthquak	565			5.8	MW							ISCGEM86	ISCGEM	ISCGEM	IS
01/17/196	20:57:41	-6.807	108.988	Earthquak	227.9			5.9	MW							ISCGEM86	ISCGEM	ISCGEM	IS
01/24/196	00:11:17	-2.608	125.952	Earthquak	20			8.2	MW							ISCGEM86	ISCGEM	ISCGEM	IS
01/29/196	09:35:30	54.636	161.703	Earthquak	55			5.5	MW							ISCGEM86	ISCGEM	ISCGEM	IS
########	05:27:06	-18.697	-177.864	Earthquak	482.9			5.6	MW							ISCGEM85	ISCGEM	ISCGEM	IS
########	15:56:51	37.523	73.251	Earthquak	15			6	MW							ISCGEM85	ISCGEM	ISCGEM	IS
########	03:25:00	-51.84	139.741	Earthquak	10			6.1	MW							ISCGEM85	ISCGEM	ISCGEM	IS
########	05:01:22	51.251	178.715	Earthquak	30.3			8.7	MW							OFFICIAL1	OFFICIAL	ISCGEM	OF
########	06:04:59	51.639	175.055	Earthquak	30			6	MW							ISCGEMSU	ISCGEMSU	ISCGEM	IS
########	06:37:06	52.528	172.007	Earthquak	25			5.7	MW							ISCGEM85	ISCGEM	ISCGEM	IS
########	06:39:32	51.626	175.746	Earthquak	25			5.8	MW							ISCGEM85	ISCGEM	ISCGEM	IS
########	07:11:23	51.037	177.848	Earthquak	25			5.9	MW							ISCGEMSU	ISCGEMSU	ISCGEM	IS
########	07:14:59	51.73	173.975	Earthquak	20			5.9	MW							ISCGEM85	ISCGEM	ISCGEM	IS
########	07:23:12	51.775	173.058	Earthquak	10			5.7	MW							ISCGEM85	ISCGEM	ISCGEM	IS
########	07:43:43	52.611	172.588	Earthquak	24			5.7	MW							ISCGEMSU	ISCGEMSU	ISCGEM	IS
########	08:06:17	51.831	174.368	Earthquak	31.8			5.7	MW							ISCGEM85	ISCGEM	ISCGEM	IS
########	08:33:41	51.948	173.969	Earthquak	20			5.6	MW							ISCGEM85	ISCGEM	ISCGEM	IS

CODE:

By Importing the necessary libraries required for building the model and data analysis of the earthquakes.

Input:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

import os
print(os.listdir("path of the input"))
```

By reading the data from csv and also columns which are necessary for the model and the column which needs to be predicted.

Input:

```
data = pd.read_csv("Path of the input")
data.head()
```

Input:

```
data.columns
```

Pointing out the main features from earthquake data and create a object of that features, namely, Date, Time, Latitude, Longitude, Depth, Magnitude

Input:

```
data = data[['Date', 'Time', 'Latitude', 'Longitude', 'Depth', 'Magnitude']]
```

```
data.head()
```

Here, the data is random we need to scale according to inputs to the model. In this, we convert given Date and Time to Unix time which is in seconds and a numeral. This can be easily used as input for the network we built.

Input:

```
import datetime
import time

timestamp = []
for d, t in zip(data['Date'], data['Time']):
    try:
        ts = datetime.datetime.strptime(d+' '+t, '%m/%d/%Y %H:%M:%S')
        timestamp.append(time.mktime(ts.timetuple()))
    except ValueError:
        # print('ValueError')
        timestamp.append('ValueError')
```

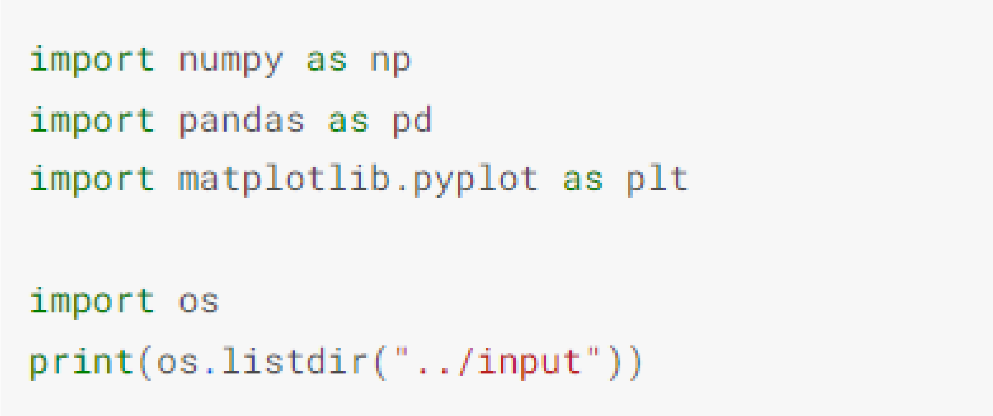
Input:

```
timeStamp = pd.Series(timestamp)
data['Timestamp'] = timeStamp.values
```

Input:

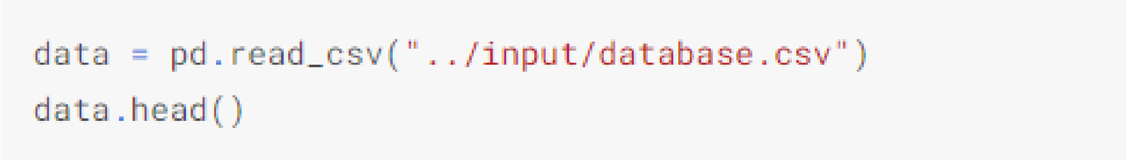
```
final_data = data.drop(['Date', 'Time'], axis=1)
final_data = final_data[final_data.Timestamp != '
ValueError']
final_data.head()
```

Input for importing necessary Libraries:



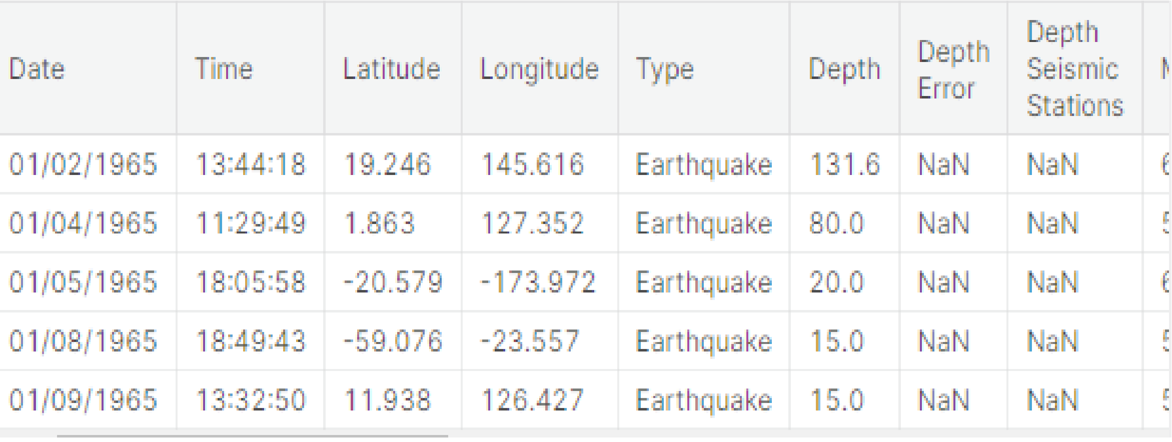
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

import os
print(os.listdir("../input"))
```

Reading data from csv file:



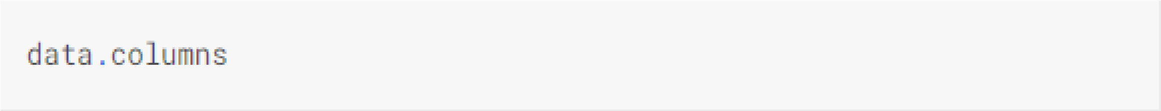
```
data = pd.read_csv("../input/database.csv")
data.head()
```

Output:

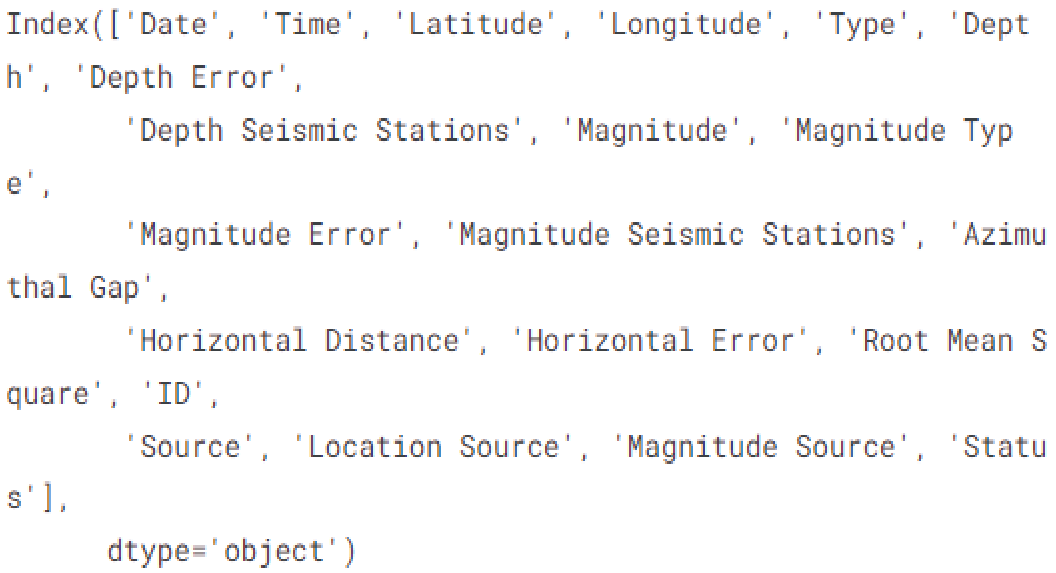


Date	Time	Latitude	Longitude	Type	Depth	Depth Error	Depth Seismic Stations	M
01/02/1965	13:44:18	19.246	145.616	Earthquake	131.6	NaN	NaN	6
01/04/1965	11:29:49	1.863	127.352	Earthquake	80.0	NaN	NaN	5
01/05/1965	18:05:58	-20.579	-173.972	Earthquake	20.0	NaN	NaN	6
01/08/1965	18:49:43	-59.076	-23.557	Earthquake	15.0	NaN	NaN	5
01/09/1965	13:32:50	11.938	126.427	Earthquake	15.0	NaN	NaN	5

Input and Output:

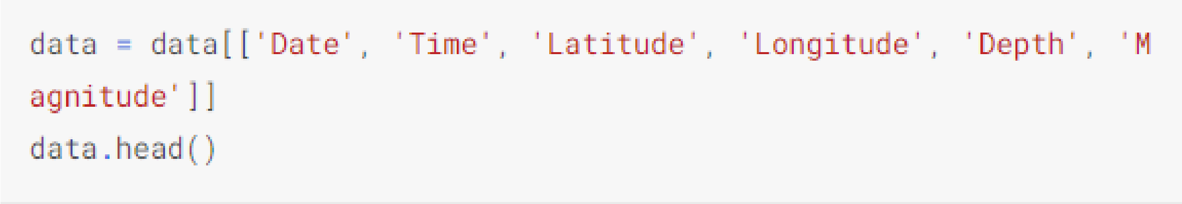


```
data.columns
```

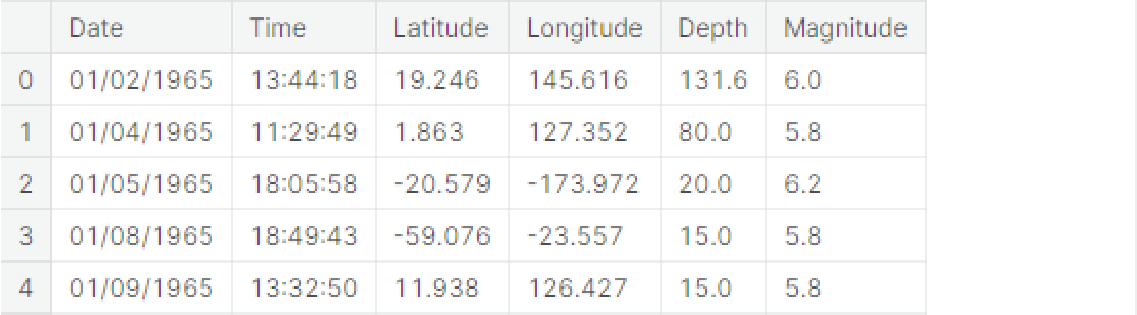


```
Index(['Date', 'Time', 'Latitude', 'Longitude', 'Type', 'Dept
h', 'Depth Error',
       'Depth Seismic Stations', 'Magnitude', 'Magnitude Typ
e',
       'Magnitude Error', 'Magnitude Seismic Stations', 'Azimu
thal Gap',
       'Horizontal Distance', 'Horizontal Error', 'Root Mean S
quare', 'ID',
       'Source', 'Location Source', 'Magnitude Source', 'Statu
s'],
      dtype='object')
```

Input and output for pointing main Features:

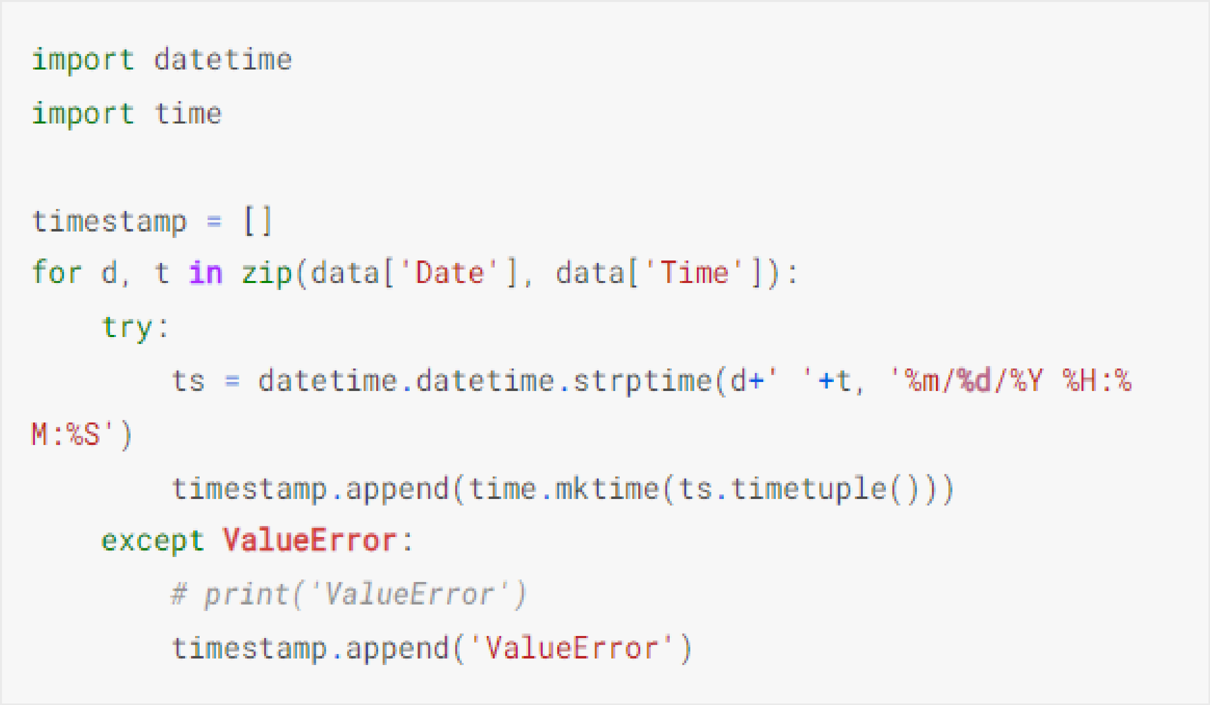


```
data = data[['Date', 'Time', 'Latitude', 'Longitude', 'Depth', 'M
agnitude']]
data.head()
```

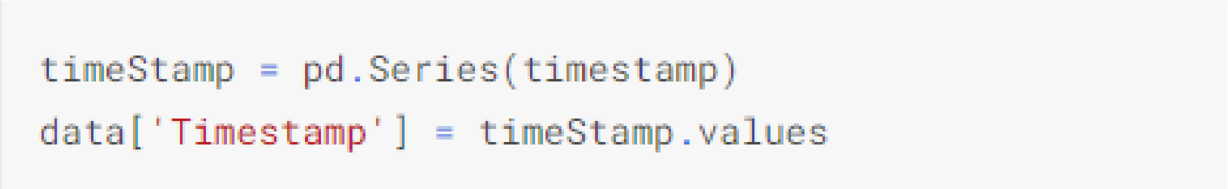


	Date	Time	Latitude	Longitude	Depth	Magnitude
0	01/02/1965	13:44:18	19.246	145.616	131.6	6.0
1	01/04/1965	11:29:49	1.863	127.352	80.0	5.8
2	01/05/1965	18:05:58	-20.579	-173.972	20.0	6.2
3	01/08/1965	18:49:43	-59.076	-23.557	15.0	5.8
4	01/09/1965	13:32:50	11.938	126.427	15.0	5.8

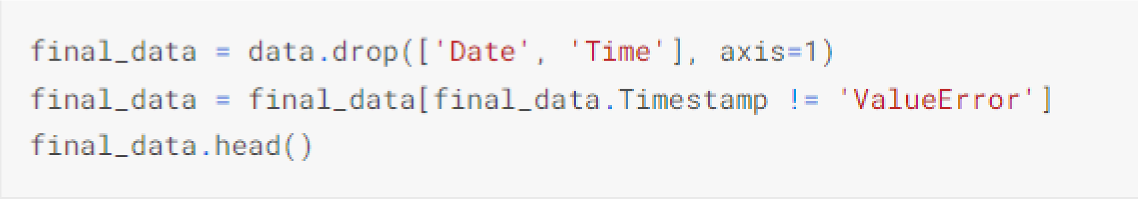
Inputs and output for given Date and Time to Unix time:



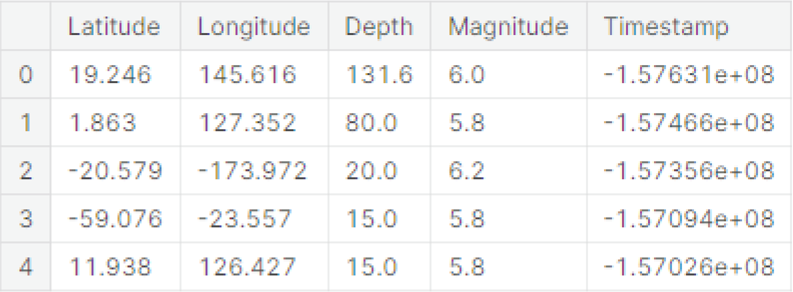
```
import datetime
import time

timestamp = []
for d, t in zip(data['Date'], data['Time']):
    try:
        ts = datetime.datetime.strptime(d+' '+t, '%m/%d/%Y %H:%
M:%S')
        timestamp.append(time.mktime(ts.timetuple()))
    except ValueError:
        # print('ValueError')
        timestamp.append('ValueError')
```



```
timeStamp = pd.Series(timestamp)
data['Timestamp'] = timeStamp.values
```



```
final_data = data.drop(['Date', 'Time'], axis=1)
final_data = final_data[final_data.Timestamp != 'ValueError']
final_data.head()
```



	Latitude	Longitude	Depth	Magnitude	Timestamp
0	19.246	145.616	131.6	6.0	-1.57631e+08
1	1.863	127.352	80.0	5.8	-1.57466e+08
2	-20.579	-173.972	20.0	6.2	-1.57356e+08
3	-59.076	-23.557	15.0	5.8	-1.57094e+08
4	11.938	126.427	15.0	5.8	-1.57026e+08

Conclusion:

Earthquake prediction using Python for projects can be achieved by effectively loading and preprocessing the dataset.

By employing Python's data manipulation and analysis libraries such as pandas and numpy, it is possible to handle large datasets with ease. Through loading the dataset into a structured format, the necessary preprocessing steps like data cleaning, normalization, and feature engineering can be efficiently performed. By leveraging the power of Python, researchers and engineers can make significant advancements in earthquake prediction, contributing to saving lives and reducing the impact of such natural disasters.