WEB Practical

Name :- Joginder Singh
class :- CSE-1
ROLL NO. :- 0161320077
Paper code :- ETCS-458

Aim :- How dataframe replace method can be used in mapping replace specific values with some other values.

Theory :- In Python, Pandas data frame is two dimensional in size-mutable, potentially heterogenous table data structure with labelled axes (rows & columns). Pandas data frame consists of 3 principal components ie the data, row, and column we can perform basic operations on rows/columns like selecting deleting & renaming.

Replace method replace the specific value with another specific value. It also searches the entire Dataframe & replace every case of the specified value.

Syntax :- df.replace (toreplace, value, inplace, limit, regex, method);

code :-

```
import pandas as pd;
new_df = df.replace({-9999: np.nan, 'no-event': sunny});

new_df = df.replace({'temperature': '[A-Za-z]',
                     'windspeed': '[a-z]'}, regex, True);
```

replacing list with another list

```
df = pd.DataFrame({'score': ['exceptional', 'average',
                             'good', 'poor',
                             'average', 'exceptional']});

df.replace(['poor', 'average', 'good', 'exceptional'],
           [1, 2, 3, 4]);

df.describe();
```

Replacing by using mapping

```
new_df = df.replace({
        -99999: np.nan,
        'no event': 'Sunny',
    })
new_df
```

	day	temperature	windspeed	event
0	1/1/2017	32.0	6.0	Rain
1	1/2/2017	NaN	7.0	Sunny
2	1/3/2017	28.0	NaN	Snow
3	1/4/2017	NaN	7.0	0
4	1/5/2017	32.0	NaN	Rain
5	1/6/2017	31.0	2.0	Sunny
6	1/6/2017	34.0	5.0	0

Regex

```
# when windspeed is 6 mph, 7 mph etc. & temperature is 32 F, 28 F etc.
new_df = df.replace({'temperature': '[A-Za-z]', 'windspeed': '[a-z]'},'', regex=True)
new_df
```

	day	temperature	windspeed	event
0	1/1/2017	32	6	Rain
1	1/2/2017	-99999	7	Sunny
2	1/3/2017	28	-99999	Snow
3	1/4/2017	-99999	7	0
4	1/5/2017	32	-99999	Rain
5	1/6/2017	31	2	Sunny
6	1/6/2017	34	5	0

Replacing list with another list

```
df = pd.DataFrame({
    'score': ['exceptional','average', 'good', 'poor', 'average', 'exceptional'],
    'student': ['rob', 'maya', 'parthiv', 'tom', 'julian', 'erica']
})
df
```

	score	student
0	exceptional	rob
1	average	maya
2	good	parthiv
3	poor	tom
4	average	julian
5	exceptional	erica

```
df.replace(['poor', 'average', 'good', 'exceptional'], [1,2,3,4])
```

	score	student
0	4	rob
1	2	maya
2	3	parthiv
3	1	tom
4	2	julian
5	4	erica