

```
In [ ]: ##### Creating new columns based on string data within columns.
```

```
In [4]: import pandas as pd
```

```
In [27]: e = pd.read_csv('ebola.csv')
e.head()
```

```
Out[27]:
```

	Indicator	Country	Date	value
0	Cumulative number of confirmed, probable and s...	Guinea	2015-03-10	3285.0
1	Cumulative number of confirmed Ebola cases	Guinea	2015-03-10	2871.0
2	Cumulative number of probable Ebola cases	Guinea	2015-03-10	392.0
3	Cumulative number of suspected Ebola cases	Guinea	2015-03-10	22.0
4	Cumulative number of confirmed, probable and s...	Guinea	2015-03-10	2170.0

```
In [19]: e.dtypes
```

```
Out[19]: Indicator      object
Country      object
Date         object
value       float64
dtype: object
```

## Improve the readability of indicators

```
In [28]: import re
```

```
In [29]: def covertIndicators(ind):
    result = 'Unknown : Unknown'
    lst_ind = [
        { 'regx': r'Case fatality rate \((CFR\) of (.*) Ebola cases', 'abbv': 'CFR'},
        { 'regx': r'Cumulative number of (.*) Ebola deaths', 'abbv': 'Cumulative Deaths'},
        { 'regx': r'Cumulative number of (.*) Ebola cases', 'abbv': 'Cumulative Cases'},
        { 'regx': r'Proportion of (.*) Ebola deaths that are from the last 21 days', 'abbv': 'Proportion of Ebola Deaths (Last 21 Days)'},
        { 'regx': r'Proportion of (.*) Ebola cases that are from the last 21 days', 'abbv': 'Proportion of Ebola Cases (Last 21 Days)'},
        { 'regx': r'Proportion of (.*) Ebola cases that are from the last 7 days', 'abbv': 'Proportion of Ebola Cases (Last 7 Days)'},
        { 'regx': r'Number of (.*) Ebola deaths in the last 21 days', 'abbv': 'Number of Ebola Deaths (Last 21 Days)'},
        { 'regx': r'Number of (.*) Ebola cases in the last 21 days', 'abbv': 'Number of Ebola Cases (Last 21 Days)'},
        { 'regx': r'Number of (.*) Ebola cases in the last 7 days', 'abbv': 'Number of Ebola Cases (Last 7 Days)'},
    ]
    for i in lst_ind:
        match = re.search(i['regx'], ind)
        if match:
            tmp = str(match.group(1))
            result = i['abbv'] + ' : ' + (tmp if len(tmp) < 20 else 'All')
            break
    return result
```

```
In [31]: e['ind'] = e['Indicator'].apply(covertIndicators)
```

```
In [32]: e['ind_class'] = e['ind'].apply(lambda x: x.split(' : ')[0])
e['ind_sub'] = e['ind'].apply(lambda x: x.split(' : ')[1])
```

```
In [33]: e.drop(['Indicator', 'ind'], axis=1, inplace=True)
e.head()
```

Out[33]:

	Country	Date	value	ind_class	ind_sub
0	Guinea	2015-03-10	3285.0	Cumulative Cases	All
1	Guinea	2015-03-10	2871.0	Cumulative Cases	confirmed
2	Guinea	2015-03-10	392.0	Cumulative Cases	probable
3	Guinea	2015-03-10	22.0	Cumulative Cases	suspected
4	Guinea	2015-03-10	2170.0	Cumulative Deaths	All

```
In [34]: e['ind_class'].unique()
```

```
Out[34]: array(['Cumulative Cases', 'Cumulative Deaths', 'Number Cases 21',
      'Number Cases 07', 'Proportion Cases 07', 'Proportion Cases 21',
      'CFR', 'Number Deaths 21', 'Proportion Deaths 21'], dtype=object)
```

```
In [ ]:
```

```
In [ ]:
```