

Setup

```
import pandas as pd
import numpy as np
df = pd.read_csv('Life Expectancy Data.csv')
df
```

	Country	Year	Status	Life expectancy	Adult Mortality	infant deaths	Alcohol	percentage expenditure	Hepatitis B	Measles	...	Polio	Total expenditure	Dipht
0	Afghanistan	2015	Developing	65.0	263.0	62	0.01	71.279624	65.0	1154	...	6.0	8.16	
1	Afghanistan	2014	Developing	59.9	271.0	64	0.01	73.523582	62.0	492	...	58.0	8.18	
2	Afghanistan	2013	Developing	59.9	268.0	66	0.01	73.219243	64.0	430	...	62.0	8.13	
3	Afghanistan	2012	Developing	59.5	272.0	69	0.01	78.184215	67.0	2787	...	67.0	8.52	
4	Afghanistan	2011	Developing	59.2	275.0	71	0.01	7.097109	68.0	3013	...	68.0	7.87	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
2933	Zimbabwe	2004	Developing	44.3	723.0	27	4.36	0.000000	68.0	31	...	67.0	7.13	
2934	Zimbabwe	2003	Developing	44.5	715.0	26	4.06	0.000000	7.0	998	...	7.0	6.52	
2935	Zimbabwe	2002	Developing	44.8	73.0	25	4.43	0.000000	73.0	304	...	73.0	6.53	
2936	Zimbabwe	2001	Developing	45.3	686.0	25	1.72	0.000000	76.0	529	...	76.0	6.16	
2937	Zimbabwe	2000	Developing	46.0	665.0	24	1.68	0.000000	79.0	1483	...	78.0	7.10	

2938 rows x 22 columns

```
df = pd.read_csv(
    'Life Expectancy Data.csv', index_col='Year', parse_dates=True
)

df
```

	Country	Status	Life expectancy	Adult Mortality	infant deaths	Alcohol	percentage expenditure	Hepatitis B	Measles	BMI	...	Polio	Total expenditure	Diph
Year														
2015-01-01	Afghanistan	Developing	65.0	263.0	62	0.01	71.279624	65.0	1154	19.1	...	6.0	8.16	
2014-01-01	Afghanistan	Developing	59.9	271.0	64	0.01	73.523582	62.0	492	18.6	...	58.0	8.18	
2013-01-01	Afghanistan	Developing	59.9	268.0	66	0.01	73.219243	64.0	430	18.1	...	62.0	8.13	
2012-01-01	Afghanistan	Developing	59.5	272.0	69	0.01	78.184215	67.0	2787	17.6	...	67.0	8.52	
2011-01-01	Afghanistan	Developing	59.2	275.0	71	0.01	7.097109	68.0	3013	17.2	...	68.0	7.87	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
2004-01-01	Zimbabwe	Developing	44.3	723.0	27	4.36	0.000000	68.0	31	27.1	...	67.0	7.13	
2003-01-01	Zimbabwe	Developing	44.5	715.0	26	4.06	0.000000	7.0	998	26.7	...	7.0	6.52	
2002-01-01	Zimbabwe	Developing	44.8	73.0	25	4.43	0.000000	73.0	304	26.3	...	73.0	6.53	
2001-01-01	Zimbabwe	Developing	45.3	686.0	25	1.72	0.000000	76.0	529	25.9	...	76.0	6.16	
2000-01-01	Zimbabwe	Developing	46.0	665.0	24	1.68	0.000000	79.0	1483	25.5	...	78.0	7.10	

2938 rows x 21 columns

```
led_df = df.copy()
```

```
df.dtypes
```

Country	object
Year	int64
Status	object
Life expectancy	float64
Adult Mortality	float64
infant deaths	int64
Alcohol	float64
percentage expenditure	float64
Hepatitis B	float64
Measles	int64
BMI	float64
under-five deaths	int64
Polio	float64
Total expenditure	float64
Diphtheria	float64
HIV/AIDS	float64
GDP	float64
Population	float64
thinness 1-19 years	float64
thinness 5-9 years	float64
Income composition of resources	float64
Schooling	float64
dtype: object	

```
df.isnull().sum()
```

Country	0
Year	0
Status	0
Life expectancy	10
Adult Mortality	10
infant deaths	0
Alcohol	194
percentage expenditure	0
Hepatitis B	553
Measles	0
BMI	34

```

under-five deaths      0
Polio                  19
Total expenditure      226
Diphtheria            19
HIV/AIDS              0
GDP                   448
Population             652
thinness 1-19 years    34
thinness 5-9 years     34
Income composition of resources 167
Schooling             163
dtype: int64

```

```
df['Year'].unique()
```

```

array([2015, 2014, 2013, 2012, 2011, 2010, 2009, 2008, 2007, 2006, 2005,
       2004, 2003, 2002, 2001, 2000])

```

```
def check_duplicates(df):
```

```

    if df[df.duplicated()].shape[0] != 0:
        print(df[df.duplicated()].shape[0])
    else:

```

```
        print("No existing duplicates")
```

```
check_duplicates(df)
```

```

    No existing duplicates

```

```
df.rename(columns={'Life expectancy': 'Life_expectancy',
```

```
               'Adult Mortality': 'Adult_mortality',
```

```
               'Hepatitis B': 'Hepatitis_B',
```

```
               'Total expenditure': 'Total_expenditure',
```

```
               'thinness 1-19 years': 'thinness_1_to_19_years',
```

```
               'thinness_5-9 years': 'thinness_5_to_9_years',
```

```
               'Income composition of resources': 'Income_composition_of_resources'},inplace=True) # rename column
```

```
df.isnull().sum()
```

```

Country      0
Year         0
Status       0
Life_expectancy 10
Adult_mortality 10
infant deaths 0
Alcohol      194
percentage expenditure 0
Hepatitis_B  553
Measles      0
BMI          34
under-five deaths 0
Polio        19
Total_expenditure 226
Diphtheria   19
HIV/AIDS     0
GDP          448
Population   652
thinness_1_to_19_years 34
thinness_5_to_9_years 34
Income_composition_of_resources 167
Schooling    163
dtype: int64

```

```
df.reindex(
```

```
    bitcoin.index, method='ffill'
```

```
).head(10).assign(
```

```
    day_of_week=lambda x: x.index.day_name()
```

```
)
```

```

-----
NameError                                Traceback (most recent call last)
<ipython-input-25-fe097a0550dc> in <cell line: 1>()
      1 df.reindex(
----> 2     bitcoin.index, method='ffill'
      3 ).head(10).assign(
      4     day_of_week=lambda x: x.index.day_name()
      5 )

NameError: name 'bitcoin' is not defined

```

```

df = df.reindex().assign(
    Life_expectancy=lambda x: x.Life_expectancy.fillna(method='ffill'),
    Adult_mortality=lambda x: x.Adult_mortality.fillna(method='ffill'),
    Alcohol=lambda x: x.Alcohol.fillna(method='ffill'),
    Hepatitis_B=lambda x: x.Hepatitis_B.fillna(method='ffill'),
    BMI=lambda x: x.BMI.fillna(method='ffill'),
    Polio=lambda x: x.Polio.fillna(method='ffill'),
    Total_expenditure=lambda x: x.Total_expenditure.fillna(method='ffill'),
    Diphtheria=lambda x: x.Diphtheria.fillna(method='ffill'),
    GDP=lambda x: x.GDP.fillna(method='ffill'),
    Population=lambda x: x.Population.fillna(method='ffill'),
    thinness_1_to_19_years=lambda x: x.thinness_1_to_19_years.fillna(method='ffill'),
    thinness_5_to_9_years=lambda x: x.thinness_5_to_9_years.fillna(method='ffill'),
    Income_composition_of_resources=lambda x: x.Income_composition_of_resources.fillna(method='ffill'),
    Schooling=lambda x: x.Schooling.fillna(method='ffill')
)

```

```
df.isnull().sum()
```

```

Country      0
Year          0
Status        0
Life_expectancy  0
Adult_mortality  0
infant deaths  0
Alcohol       0
percentage expenditure  0
Hepatitis_B   0
Measles        0
BMI            0
under-five deaths  0
Polio          0
Total_expenditure  0
Diphtheria     0
HIV/AIDS       0
GDP            0
Population     0
thinness_1_to_19_years  0
thinness_5_to_9_years  0
Income_composition_of_resources  0
Schooling      0
dtype: int64

```

```

def check_value_counts(column):
    print(df.value_counts(column))

```

```
check_value_counts(df['Life_expectancy'])
```

```

Life_expectancy
73.0    45
75.0    33
78.0    31
73.6    28
73.9    25
..
46.3     1
45.4     1
44.0     1
39.0     1
36.3     1
Name: count, Length: 362, dtype: int64

```

```
df.dtypes
```

```

Country          object
Year             int64
Status           object
Life_expectancy  float64
Adult_mortality  float64
infant deaths    int64
Alcohol          float64
percentage expenditure  float64
Hepatitis_B      float64
Measles          int64
BMI              float64
under-five deaths int64
Polio            float64
Total_expenditure float64
Diphtheria       float64
HIV/AIDS        float64
GDP              float64
Population       float64
thinness_1_to_19_years float64
thinness_5_to_9_years float64
Income_composition_of_resources float64
Schooling        float64
dtype: object

```

```

Country_type = list(df['Country'].unique())
Status_type = list(df['Status'].unique())

```

```

df['Country'] = df.apply(lambda x: Country_type.index(x['Country']) + 1, axis=1)
df['Status'] = df.apply(lambda x: Status_type.index(x['Status']) + 1, axis=1)

```

```
df.dtypes
```

```

Country          int64
Year             int64
Status           int64
Life_expectancy  float64
Adult_mortality  float64
infant deaths    int64
Alcohol          float64
percentage expenditure  float64
Hepatitis_B      float64
Measles          int64
BMI              float64
under-five deaths int64
Polio            float64
Total_expenditure float64
Diphtheria       float64
HIV/AIDS        float64
GDP              float64
Population       float64
thinness_1_to_19_years float64
thinness_5_to_9_years float64
Income_composition_of_resources float64
Schooling        float64
dtype: object

```

```
df['Year'].unique()
```

```

array([2015, 2014, 2013, 2012, 2011, 2010, 2009, 2008, 2007, 2006, 2005,
       2004, 2003, 2002, 2001, 2000])

```