Encoding

- 문자형 변수를 숫자형 변수로 인코딩

```
import warnings
warnings.filterwarnings('ignore')
```

I. 실습 데이터

1) seaborn 'mpg' Data Set

```
import seaborn as sns

DF = sns.load_dataset('mpg')
```

```
DF.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 398 entries, 0 to 397
Data columns (total 9 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   mpg           398 non-null    float64
 1   cylinders     398 non-null    int64  
 2   displacement  398 non-null    float64
 3   horsepower    392 non-null    float64
 4   weight        398 non-null    int64  
 5   acceleration  398 non-null    float64
 6   model_year    398 non-null    int64  
 7   origin        398 non-null    object 
 8   name          398 non-null    object 
dtypes: float64(4), int64(3), object(2)
memory usage: 28.1+ KB
```

- 문자형 데이터 : 'origin'

```
DF.head()
```

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	name
0	18.0	8	307.0	130.0	3504	12.0	70	usa	chevrolet chevelle malibu
1	15.0	8	350.0	165.0	3693	11.5	70	usa	buick skylark 320
2	18.0	8	318.0	150.0	3436	11.0	70	usa	plymouth satellite

```
type(DF.origin[0])
```

```
str
```

- 명목형 : 이름확인 및 빈도분석

```
DF.origin.value_counts()
```

```
usa       249
japan      79
europe     70
Name: origin, dtype: int64
```

- 'origin' Data

```
X = DF[['origin']]
```

```
X[111:115]
```

	origin
111	japan
112	usa
113	usa
114	europe

2) With LabelEncoder

- 정수(Integer) 인코딩

```
from sklearn.preprocessing import LabelEncoder

encoder1 = LabelEncoder()
LE = encoder1.fit_transform(X)
```

- 정수 인코딩 결과

```
LE[111:115]
```

```
array([1, 2, 2, 0])
```

3) With OneHotEncoder

- 원-핫(One-Hot) 인코딩

```
from sklearn.preprocessing import OneHotEncoder

encoder2 = OneHotEncoder()
OHE = encoder2.fit_transform(X)
```

- Array 변환 필요

```
print(OHE[111:115])
```

```
  (0, 1)        1.0
  (1, 2)        1.0
  (2, 2)        1.0
  (3, 0)        1.0
```

```
OHE.toarray()[111:115]
```

```
array([[0., 1., 0.],
       [0., 0., 1.],
       [0., 0., 1.],
       [1., 0., 0.]])
```

#

#

#

The End

#

#

#