Label encoding and oridian Encoding

In [1]:
```
import pandas as pd
from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import OrdinalEncoder
```

In [2]:
```
df= pd.read_csv(r"train.csv")
df.head()
```

Out[2]:

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	PoolArea	PoolQC	Fence	Mis
0	1	60	RL	65.0	8450	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	
1	2	20	RL	80.0	9600	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	
2	3	60	RL	68.0	11250	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	
3	4	70	RL	60.0	9550	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	
4	5	60	RL	84.0	14260	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	

5 rows × 81 columns

In [3]:
```
df2=df[["KitchenQual","BldgType"]]
```

In [4]:
```
df2.head()
```

Out[4]:

	KitchenQual	BldgType
0	Gd	1Fam
1	TA	1Fam
2	Gd	1Fam
3	Gd	1Fam
4	Gd	1Fam

In [5]:
```
le=LabelEncoder()
```

In [6]:
```
le.fit_transform(df2["BldgType"])
```

Out[6]:
```
array([0, 0, 0, ..., 0, 0, 0])
```

In [7]:
```
df2["BldgType_L-ec"]=le.fit_transform(df2["BldgType"])
```

```
<ipython-input-7-7b26e7082f99>:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html
turning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returni
a-view-versus-a-copy)
  df2["BldgType_L-ec"]=le.fit_transform(df2["BldgType"])
```

In [8]:
```
df2.head(30)
```

Out[8]:

	KitchenQual	BldgType	BldgType_L-ec
0	Gd	1Fam	0
1	TA	1Fam	0
2	Gd	1Fam	0
3	Gd	1Fam	0
4	Gd	1Fam	0
5	TA	1Fam	0
6	Gd	1Fam	0
7	TA	1Fam	0
8	TA	1Fam	0
9	TA	2fmCon	1
10	TA	1Fam	0
11	Ex	1Fam	0
12	TA	1Fam	0
13	Gd	1Fam	0
14	TA	1Fam	0
15	TA	1Fam	0
16	TA	1Fam	0
17	TA	Duplex	2
18	Gd	1Fam	0
19	TA	1Fam	0
20	Gd	1Fam	0

	KitchenQual	BldgType	BldgType_L-ec
21	Gd	1Fam	0
22	Gd	1Fam	0
23	TA	TwnhsE	4
24	Gd	1Fam	0
25	Gd	1Fam	0
26	Gd	1Fam	0
27	Gd	1Fam	0
28	TA	1Fam	0
29	Fa	1Fam	0

In [9]:
```
df2["BldgType"].value_counts()
```

Out[9]:
```
1Fam      1220
TwnhsE     114
Duplex      52
Twnhs       43
2fmCon      31
Name: BldgType, dtype: int64
```

In [10]:
```
df2["KitchenQual"].value_counts()
```

Out[10]:
```
TA    735
Gd    586
Ex    100
Fa     39
Name: KitchenQual, dtype: int64
```

oridian Encoding

```
""" KitchenQual : kitchen Quality

    Ex= excellent

    Gd=  Good

    TA= typical / Average

    Fa= fair

    """
```

In [11]:
```
order_label={"EX":4,'Gd':3,'TA':2,'Fa':1}
```

In [12]:
```
df2["KitchenQual_org_enc"]=df2["KitchenQual"].map(order_label)
```

```
<ipython-input-12-2942fd9b899c>:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html
turning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returni
a-view-versus-a-copy)
  df2["KitchenQual_org_enc"]=df2["KitchenQual"].map(order_label)
```

In [13]:
```
df2
```

Out[13]:

	KitchenQual	BldgType	BldgType_L-ec	KitchenQual_org_enc
0	Gd	1Fam	0	3.0
1	TA	1Fam	0	2.0
2	Gd	1Fam	0	3.0
3	Gd	1Fam	0	3.0
4	Gd	1Fam	0	3.0
...	...	...	...	...
1455	TA	1Fam	0	2.0
1456	TA	1Fam	0	2.0
1457	Gd	1Fam	0	3.0
1458	Gd	1Fam	0	3.0
1459	TA	1Fam	0	2.0

1460 rows × 4 columns