# Label Encoder

Label encoding is a technique used in machine learning and data analysis to convert categorical variables into numerical format. It is particularly useful when working with algorithms that require numerical input, as most machine learning models can only operate on numerical data.

Let's see how to implement label encoding in Python using the scikit-learn library and also understand the challenges with label encoding:

```
#importing the libraries
import pandas as pd
import numpy as np

#reading the dataset
df=pd.read_csv("Salary.csv")
```

```
# Import label encoder
from sklearn import preprocessing
# label_encoder object knows how to understand word labels.
label_encoder = preprocessing.LabelEncoder()
# Encode labels in column 'Country'.
data['Country']= label_encoder.fit_transform(data['Country'])
print(data.head())
```

| Country | Age | Salary |
|---|---|---|
| 0 | 44 | 72000 |
| 2 | 34 | 65000 |
| 1 | 46 | 98000 |
| 2 | 35 | 45000 |
| 1 | 23 | 34000 |

As you can see here, label encoding uses alphabetical ordering. Hence, India has been encoded with 0, the US with 2, and Japan with 1.

## Challenges with Label Encoding:

In the above scenario, the Country names do not have an order or rank. But, when label encoding is performed, the country names are ranked based on the alphabets. Due to this, there is a very high probability that the model captures the relationship between countries such as India < Japan < the US.