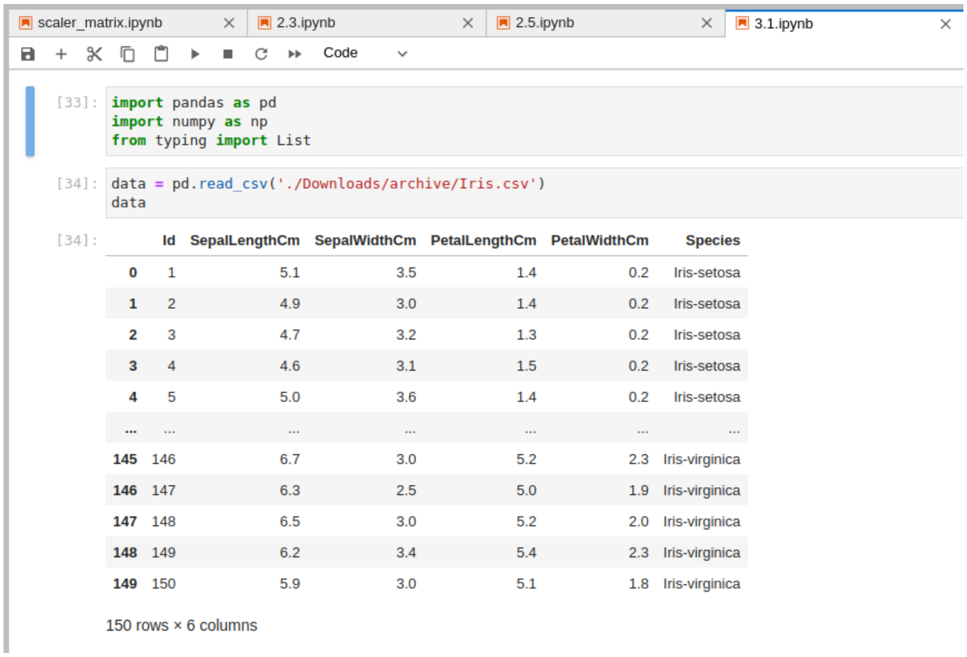
Talha Khan (2303.009.KHI.DEG)

Muhammad Moiz Khan (2303.022.KHI.DEG)

ASSIGNMNET 2.5



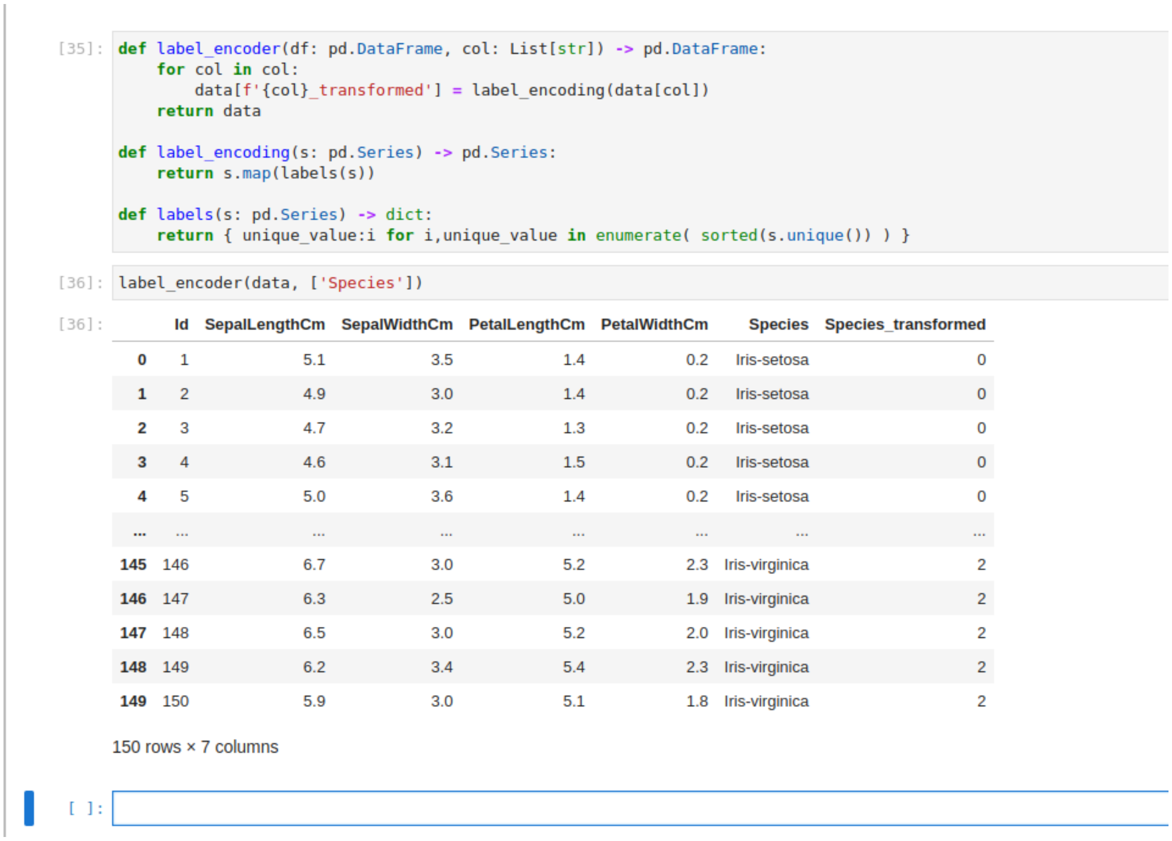
```
[33]: import pandas as pd
      import numpy as np
      from typing import List
```

```
[34]: data = pd.read_csv('./Downloads/archive/Iris.csv')
      data
```

```
[34]:
        Id  SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm         Species
  0      1            5.1           3.5            1.4           0.2     Iris-setosa
  1      2            4.9           3.0            1.4           0.2     Iris-setosa
  2      3            4.7           3.2            1.3           0.2     Iris-setosa
  3      4            4.6           3.1            1.5           0.2     Iris-setosa
  4      5            5.0           3.6            1.4           0.2     Iris-setosa
...    ...            ...           ...            ...           ...             ...
145    146            6.7           3.0            5.2           2.3  Iris-virginica
146    147            6.3           2.5            5.0           1.9  Iris-virginica
147    148            6.5           3.0            5.2           2.0  Iris-virginica
148    149            6.2           3.4            5.4           2.3  Iris-virginica
149    150            5.9           3.0            5.1           1.8  Iris-virginica

150 rows × 6 columns
```

- The pd.read_csv function is a pandas function that is used to read a comma-separated values (CSV) file and return a pandas DataFrame. The function takes the path of the CSV file as input.



[35]:
```
def label_encoder(df: pd.DataFrame, col: List[str]) -> pd.DataFrame:
    for col in col:
        data[f'{col}_transformed'] = label_encoding(data[col])
    return data

def label_encoding(s: pd.Series) -> pd.Series:
    return s.map(labels(s))

def labels(s: pd.Series) -> dict:
    return { unique_value:i for i,unique_value in enumerate( sorted(s.unique()) ) }
```

[36]:
```
label_encoder(data, ['Species'])
```

[36]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species	Species_transformed
0	1	5.1	3.5	1.4	0.2	Iris-setosa	0
1	2	4.9	3.0	1.4	0.2	Iris-setosa	0
2	3	4.7	3.2	1.3	0.2	Iris-setosa	0
3	4	4.6	3.1	1.5	0.2	Iris-setosa	0
4	5	5.0	3.6	1.4	0.2	Iris-setosa	0
...	...	...	...	...	...	...	...
145	146	6.7	3.0	5.2	2.3	Iris-virginica	2
146	147	6.3	2.5	5.0	1.9	Iris-virginica	2
147	148	6.5	3.0	5.2	2.0	Iris-virginica	2
148	149	6.2	3.4	5.4	2.3	Iris-virginica	2
149	150	5.9	3.0	5.1	1.8	Iris-virginica	2

150 rows × 7 columns

[ ]:

- ➢ The label_encoder function takes a pandas DataFrame and a list of column names as input. It returns the DataFrame with new columns for each specified column with '_transformed' appended to their names. These new columns contain the label encoded values for the original columns.
- ➢ The label_encoding function takes a pandas Series as input and returns a pandas Series of label encoded values using the labels function.
- ➢ The labels function takes a pandas Series as input and returns a dictionary with the unique values as keys and their corresponding integer label as values. The integer label is assigned based on the order of the unique values after sorting them in ascending order.
- ➢ Overall, these functions are useful for encoding categorical variables with label encoding technique. However, it's important to note that label encoding may not always be the best encoding technique for categorical variables, as it assumes an ordinal relationship between the categories. In cases where there is no such relationship, one-hot encoding or other techniques may be more appropriate.