To add columns we use **data. columns=[ “all columns”]**

***Note: There was a small space infront of the question mark, so make sure to include that if you're using the same dataset.***

for variable in **numericals**:

if not DATA [ DATA [ variable ] == ' ? '] .empty:

Print ( f ' { variable } contains missing values ( ?)')

We'll first take care of the missing categorical values. One option is to replace the missing values with the most frequent/mode, which we'll do below. However, options for dealing with missing ***categorical*** variables include:

* Remove observations with missing values if we are dealing with a large dataset and the number of records containing missing values are few.
* Remove the variable/column if it is not significant.
* Develop a model to predict missing values. KNN for example.
* Replace missing values with the most frequent in that column.

DATA ['workclass'] **.** fillna ( DATA ['workclass']**.**mode()[0], inplace**=True**)

DATA ['occupation'] **.** fillna ( DATA [ 'occupation ' ] **.** mode()[0], inplace**=True**)

DATA ['native\_country'] **.** fillna ( DATA [ ' native\_country ' ] **.** mode( )[0], inplace**=True**)

En·code -> convert into a coded form.

Since our categories don't really have any type of order to preserve, we'll use **one-hot encoding / get dummies**.

The only difference from **KNN** lab from **NAIVE BAYES** isis that here we're using RobustScaler

* Using the mean cross-validation, we can conclude that we expect the model to be around 0.8236% accurate on average.