TD1. Data Visualization with Python

let's practice what we have learned so far and visualise some data !

in order to validate this checkpoint you are asked to accomplish the following tasks:

I. After importing the dataset using Pandas start by showing the head of the dataset then some general information about the data columns and values

II. As a next step apply what you have learned in the course to preprocess your data

III. Let's begin now the data visualization part :

1. Start by studying the distribution of the most important features based on your understanding of the dataset and the problem
2. Visualise the corrolation between Sex and Age in a plot of your choosing, the visualised plot should give us obvious deductions concerning the importance of age and Sex in the survival of the individuals.
3. Pick two other features and study their impact on the survival of the individuals
4. Have a look at this function:

```
def plot_correlation_map( df ):
    corr = df.corr()
    s , ax = plt.subplots( figsize =( 12 , 10 ) )
    cmap = sns.diverging_palette( 220 , 10 , as_cmap = True )
    s = sns.heatmap(
        corr,
        cmap = cmap,
        square=True,
        cbar_kws={ 'shrink' : .9 },
        ax=ax,
        annot = True,
        annot_kws = { 'fontsize' : 12 }
        )
```

5. Try running the function, analyse what does it do exactly and what are the utilities of it, write a pragraph describing your analysis
6. Use the groupby function combined with the mean() to view the relation between pclass and survived
7. As a final step try to drop the useless columns such as Names
   In fact, we dropped the column Names because we evaluated it as useless, well that's not the case, we can create a very useful feature from the names
8. your task now is to create a new column called Title which contain the appropriate Title for each individual (hint : extract the title from the column Names)
9. Visualise the correlation between Title and other features(e.g Sex, Fare, Age...)

10. As you should've seen, there is a lot of titles which makes the feature not very practical, let's try to group these titles into more useful ones
    hint: use the following dictionary

```
Title_Dictionary = {
                    "Capt":       "Officer",
                    "Col":        "Officer",
                    "Major":      "Officer",
                     "Dr":        "Officer",
                    "Rev":        "Officer”,
                    "Jonkheer":   "Royalty",
                    "Don":        "Royalty",
                    "Sir" :       "Royalty",
                   "Lady" :      "Royalty"
                  "the Countess": "Royalty",
                    "Dona":       "Royalty”,
                    "Mme":        "Miss",
                    "Mlle":       "Miss",
                    "Miss" :      "Miss",
                    "Ms":         "Mrs",
                    "Mr" :        "Mrs",
                    "Mrs" :       "Mrs
                    "Master" :    "Master"
                    }
```

let's visualise the same correlations using these new titles now.