DELIVERABLE WEEK 9

Group Name: The Powerpuff Girls

Specialization: Data Science

Team Members:

1. **Name:** Memudu Alimatou Sadia Anike

   Email: anikesadia01@gmail.com

   Country: Nigeria

   College: University of Ilorin

   Specialization: Data science

2. **Name:** Chaithanya Shivakumar Ittamadu

   Email: sichaithanya889@gmail.com

   Country: Ireland

   College: Dublin Business School

   Specialization: Data science

3. **Name:** Lakshmi Chandana Vupputuri

   Email: vupputuri.chandana@gmail.com

   Country: Ireland

   Specialization: Data Science

Problem description

ABC Bank wants to sell it's term deposit product to customers and before launching the product they want to develop a model which help them in understanding whether a particular customer will buy their product or not (based on customer's past interaction with bank or other Financial Institution).

Here we are using different approaches to clean and transform the data in order to solve the above mentioned problem.

GitHub Repo link

Group repos: https://github.com/memudualimatou/Bank-DataScienceProject

Memudu sadia notebook: https://github.com/memudualimatou/Bank-DataScienceProject/blob/main/BankMarketing_%20(Memudu%20sadia).ipynb

Chandana notebook: https://github.com/memudualimatou/Bank-DataScienceProject/blob/main/Bank_Marketing_(chandana).ipynb

Chaitanya notebook: https://github.com/memudualimatou/Bank-DataScienceProject/blob/main/Bank_Marketing_(chaithanya).ipynb

Final Cleaning Notebook: https://github.com/memudualimatou/Bank-DataScienceProject/blob/main/Week9_DataCeaning.ipynb

Data Cleansing and Transformation

Results and Approaches:

- Remove the quotes in the values of the data
- No missing data
- No duplicated values
- Provide the appropriate column name to the data
- Provide the correct data type to each column
- All the unknown data has been deleted because they are considered as missing value
- Calculate the skewed value of each numerical value
- Check outliers in the data

Data Transformation

The Education values need to the transformed. It contains 7 different values and 3 of the same type (basic.9y, basic.6y and basic.4y) which can be categorized into basic what I accomplished below.

```
In [18]:
# mapping all types of basic education into one

print ("Before mapping: ",data['education'].value_counts() ,'\n''\n')


data['education'] = data['education'].map({'basic.9y':'basic','basic.6y':'basic','basic.4y':'basic', "university.degree": "u
                                           "high.school":"HighSchool","professional.course":"professional","illiterate":"illi


print ("after mapping: ", data['education'].value_counts())
```

```
Before mapping:  university.degree      10412
high.school             7699
professional.course     4321
basic.9y                4276
basic.4y                2380
basic.6y                1389
illiterate                11
Name: education, dtype: int64


after mapping:  university      10412
basic            8045
HighSchool       7699
professional     4321
illiterate         11
Name: education, dtype: int64
```

Outliers

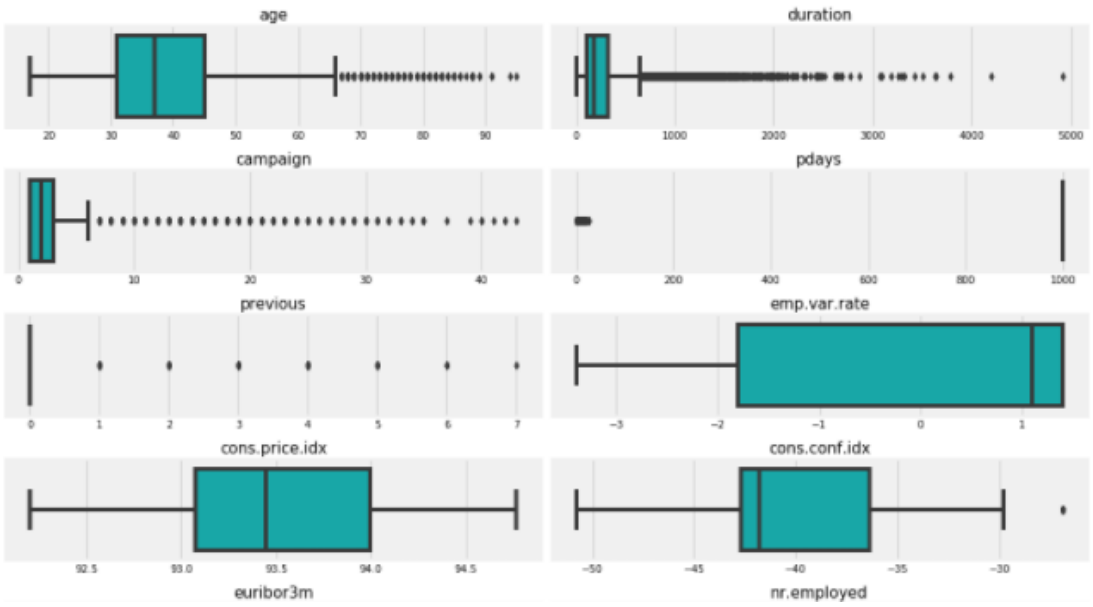
The image below shows all the data columns that has outliers, but we decided to keep them because we can see certain number of Outliers in 'age', 'duration', and ‘campaign’ etc.

But it’s important to note that since this is a sensitive Bank Dataset the above columns should be treated as 'Extreme values' which provides important insights and not 'Outliers

In [21]:

```
# Checking for outliers

fig, ax  = plt.subplots(5,2,figsize = (18,15))
count = 0
cols = data.select_dtypes(include = np.number)
cols = cols.columns
for i in range(5):
    for j in range(2):
        s = cols[count+j]
        sb.boxplot(data[s].values,ax = ax[i][j],color = 'c')
        ax[i][j].set_title(s,fontsize = 15)
        fig = plt.gcf()
        fig.set_size_inches(15,10)
        plt.tight_layout()
    count = count+j+1
```



Skewed

In [22]:

```
# Calculate the Skewed value of each variable

for i in num_cols:
    print(f"Skewness {i} : " + str(data[i].skew()))
```

```
Skewness age : 0.9802100594305216
Skewness duration : 3.3895760594757802
Skewness campaign : 4.896935174352032
Skewness pdays : -4.507904504882473
Skewness previous : 3.5946072353163
Skewness emp.var.rate : -0.5489200845738652
Skewness cons.price.idx : -0.1187247500320943
Skewness cons.conf.idx : 0.373946969377056
Skewness euribor3m : -0.525050282109819
Skewness nr.employed : -0.8937557149000335
```

As you can see the most skew varibale is the campaign followed by the duration solution np.log1p

The most skewed value are the duration and campaign, which means they don't have a normal distribution and are right-skewed. We can solve this issue using numpy.log1p