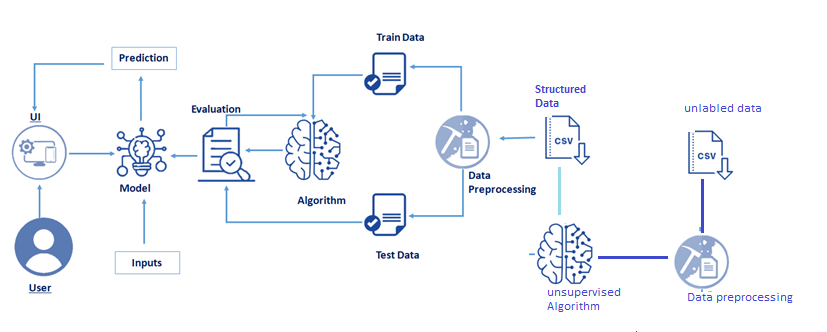
**CUSTOMER SEGMENTATION**

**Project Description:**

In today’s highly competitive world .The primal aim of any business is to grab the potential customers who can generate profits to the organisations, Increasing in the number of organisations in the market , companies wants to gain competitive advantage over other organisations. To gain the competitive advantage companies needs to concentrate on gaining the potential customers, which is a primal task for management to identify and filter out the potential customers from the set of customers, the task has been simplified with the help of Machine Learning models, which are able to classify the customers into various segments based on different the attributes they possess. Intervention of Data science and AI helping the Businesses to Build such a models which are able to analyize the customers and their products for better decision making to improvise the business process, to formulate better strategies and to improve the revenue.

**Technical Architecture:**

****

**Pre requisites:**

**To complete this project, you must require following software’s , concepts and packages**

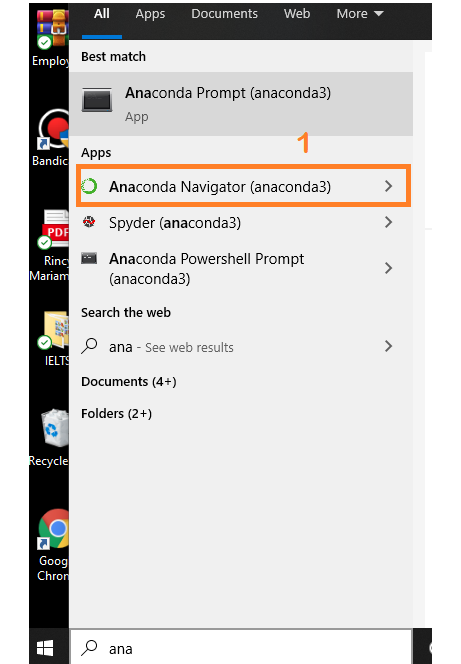
* **Anaconda navigator:**
  + Refer to the link below to download anaconda navigator
  + **Link :** [**https://www.youtube.com/watch?v=5mDYijMfSzs**](https://www.youtube.com/watch?v=5mDYijMfSzs)
* **Prior knowledge:**
  + **Customer segmentation use case:** [**https://www.youtube.com/watch?v=zPJtDohab-g**](https://www.youtube.com/watch?v=zPJtDohab-g)
  + **Customer segmentation model using machine learning** [**https://www.youtube.com/watch?v=Liff\_GA74EI**](https://www.youtube.com/watch?v=Liff_GA74EI)
  + Supervised Learning**:** [**https://www.javatpoint.com/supervised-machine-learning**](https://www.javatpoint.com/supervised-machine-learning)
  + Unsupervised Learning: [**https://www.javatpoint.com/unsupervised-machine-learning**](https://www.javatpoint.com/unsupervised-machine-learning)
* **Python packages:**

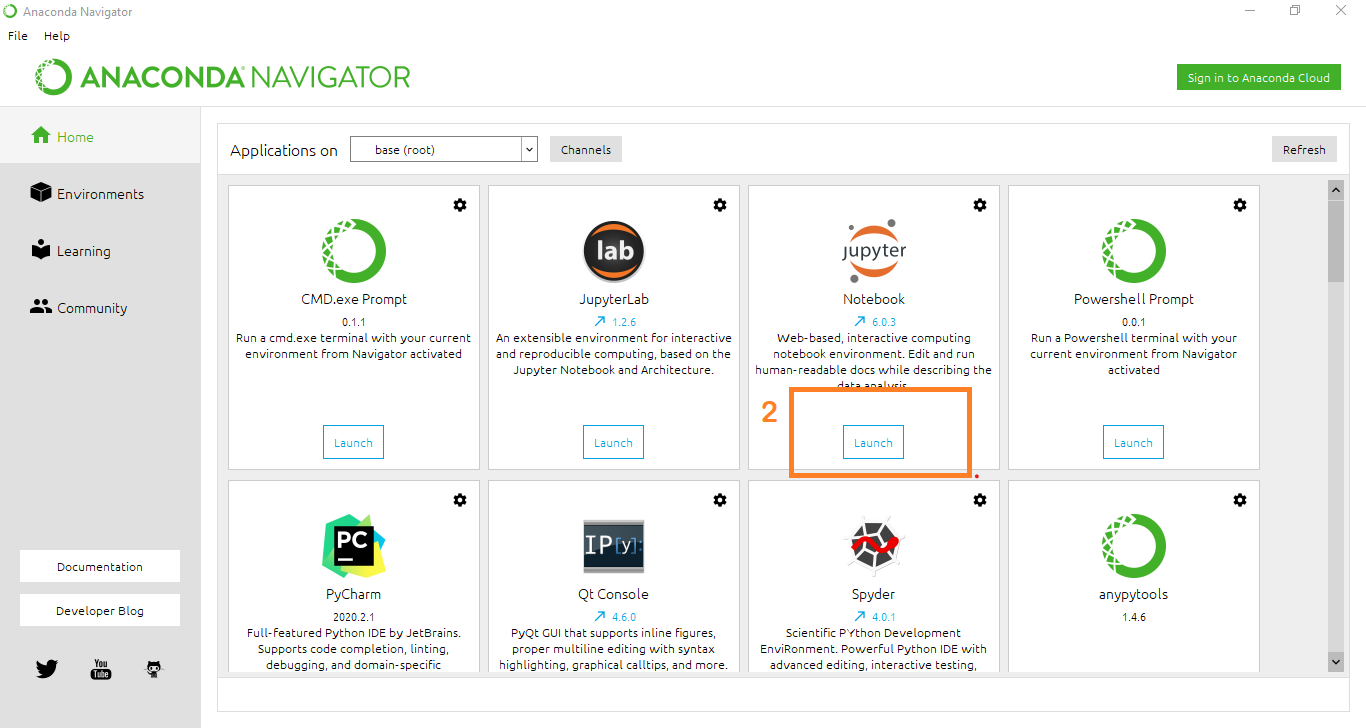
Open anaconda prompt as administrator.

* Type “pip install numpy” and click enter.
* Type “pip install pandas” and click enter.
* Type “pip install matplotlib” and click enter.
* Type “pip install scikit-learn” and click enter.
* Type “pip install Flask” and click enter.

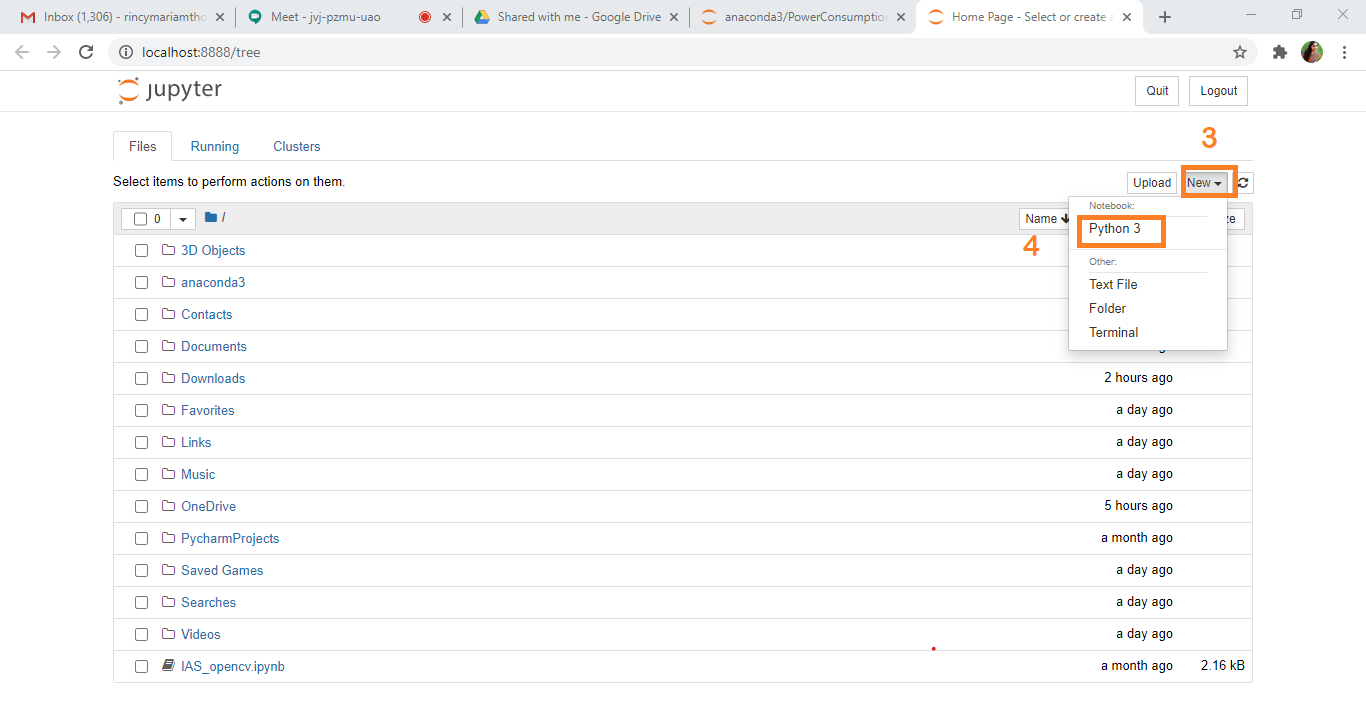
The above steps allow you to install the packages in the anaconda environment

* **Launch Jupyter**
  + Search for Anaconda Navigator and open Launch Jupyter notebook.





* Then you will be able to see that the jupyter notebook runs on local host:8888.
* To Create a new file Go to New 🡪Python3.The file in jupyter notebook is saved with .ipynb extension.



* + Flask Basics : <https://www.youtube.com/watch?v=lj4I_CvBnt0>

**Project Objectives:**

By the end of this project:

* This projects enables the learner to understand the business use case of how and why to segment the customers.
* You’ll able to understand the unsupervised learning methods such as H-clustering and k-means clustering
* You’ll be able to understand the problem to classify if it is a regression or a classification kind of problem.
* You will be able to know how to pre-process / clean the data using different data pre-processing techniques.
* You will able to analyse or get insights of data through visualization.
* Applying different algorithms according to dataset and based on visualization.
* You will able to know how to find accuracy of the model.
* You will be able to know how to build a web application using Flask framework.

**Project Flow:**

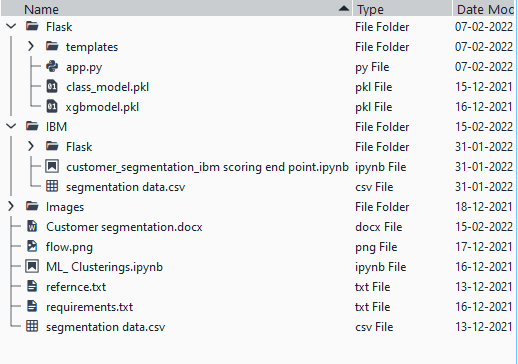
* User interacts with the UI (User Interface) to enter the input values
* Entered input values are analysed by the model which is integrated
* Once model analyses the input the prediction is showcased on the UI

To accomplish this, we have to complete all the activities and tasks listed below

* Data Collection.
* Collect the dataset or Create the dataset
* Data Pre-processing.
  + Import the Libraries.
  + Importing the dataset.
  + Checking for Null Values.
  + Data Visualization.
  + Taking care of Missing Data.
  + Feature Scaling.
* Unsupervised Model Building
  + Import the model building Libraries
  + Initializing the model
  + Fit and predict the clusters
  + Add the classes to the main data set and save the dataset
  + Splitting x and y
  + Splitting train and test data
* Supervised Model Building
  + Import the model building Libraries
  + Initializing the model
  + Model Training
  + Evaluating the Model
  + Save the Model
* Application Building
  + Create an HTML file
  + Build a Python Code

**Project Structure:**

Create a Project folder which contains files as shown below

A python file called app.py for server side scripting.

* We need the model which is saved and the saved model in this content is **xgbmodel.pkl**
* Templates folder which contains index.HTML file, chance.HTML file, noChance.HTML file.
* Scale.pkl for scaling,encoder.pkl file for encoding the categorical data,imputer.pkl file for filling out the missing values

**Milestone 1: Data Collection:**

ML depends heavily on data, without data, it is impossible for an “AI” to learn. It is the most crucial aspect that makes algorithm training possible. In Machine Learning projects, we need a training **data set.**It is the actual **data set** used to train the model for performing various actions.

**Activity1: Download the dataset**

You can collect datasets from different open sources like kaggle.com, data.gov, UCI machine learning repository etc.

Please refer to the link given below to download the data set and to know about the dataset

<https://docs.google.com/spreadsheets/d/1NnUMX3sjJgRRerkJTAXemlfdyo2GiUhgE_m4w-fAhvs/edit#gid=1219451115>

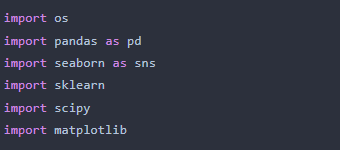
**Milestone 2: Data Pre-processing**

Data Pre-processing includes the following main tasks

* + Import the Libraries.
  + Importing the dataset.
  + Checking for Null Values.
  + Data Visualization.
  + Feature Scaling.

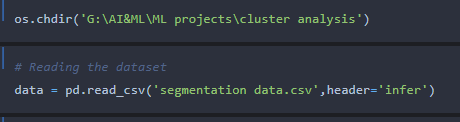
**Activity 1: Import Necessary Libraries**

* + It is important to import all the necessary libraries such as pandas, numpy, matplotlib.
  + **Numpy**- It is an open-source numerical Python library. It contains a multi-dimensional array and matrix data structures. It can be used to perform mathematical operations on arrays such as trigonometric, statistical, and algebraic routines.
  + **Pandas**- It is a fast, powerful, flexible and easy to use open source data analysis and manipulation tool, built on top of the Python programming language.
  + **Seaborn**- Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics.
  + **Matplotlib**- Visualisation with python. It is a comprehensive library for creating static,animated, and interactive visualizations in Python
  + **Sklearn** – which contains all the modules required for model building
  + **Scipy** – which contains all the modules required for scientific and computing functions



**Activity 2: Importing the Dataset**

* You might have your data in .csv files, .excel files
* Let’s load a .csv data file into pandas using **read\_csv() function.**We will need to locate the directory of the CSV file at first (it’s more efficient to keep the dataset in the same directory as your program).
* If your dataset is in some other location ,Then

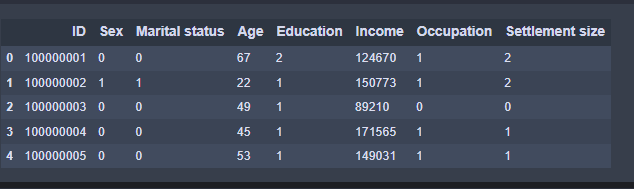
**Data=pd.read\_csv(r”File\_location/datasetname.csv”) **

**Note:**r stands for "raw" and will cause backslashes in the string to be interpreted as actual backslashes rather than special characters.

* If the dataset in same directory of your program, you can directly read it, without giving raw as r.
* Our Dataset segmentation.csv contains following Columns
* ID - Unique id of the customer
* Sex – Gender of the customer
* Marital status – whether the person is married or not
* Age = Age of the person
* Education – Education of the person
* Income – income of the person
* Occupation – indicates the profession of a person,employeed or unemployed or business
* Settlement size –Represents the no.of persons in a family

**Activity 3: Analyse the data**

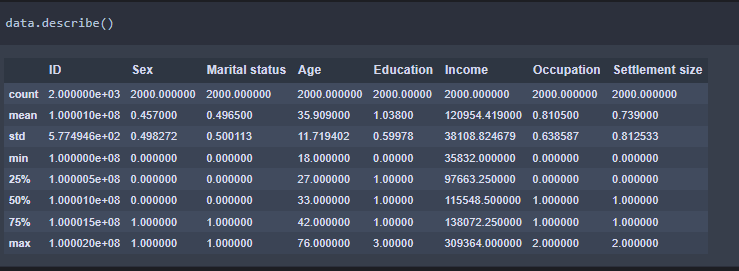
* head() method is used to return top n (5 by default) rows of a DataFrame or series.



* describe() method computes a summary of statistics like count, mean, standard deviation, min, max and quartile values.

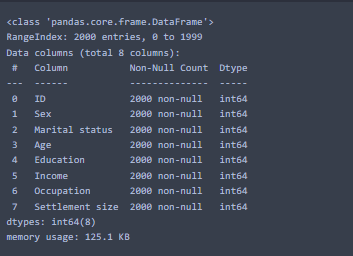


The output is as shown below



From the data we infer that there are only decimal values and no categorical values

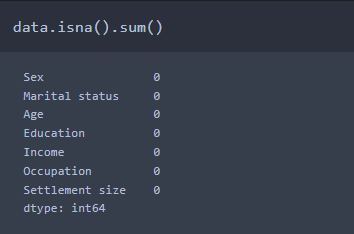
* info() gives information about the data



**Activity 4: Handling Missing Values**

1. After loading it is important to check the complete information of data as it can indication many of the hidden information such as null values in a column or a row

2.Check whether any null values are there or not. if it is present then following can be done,



**Activity 5: Data Visualisation**

* Data visualization is where a given data set is presented in a graphical format. It helps the detection of patterns, trends and correlations that might go undetected in text-based data.
* Understanding your data and the relationship present within it is just as important as any algorithm used to train your machine learning model. In fact, even the most sophisticated machine learning models will perform poorly on data that wasn’t visualized and understood properly.
* To visualize the dataset we need libraries called Matplotlib and Seaborn.
* The Matplotlib library is a Python 2D plotting library which allows you to generate plots, scatter plots, histograms, bar charts etc.

Let’s visualize our data using Matplotlib and searborn library.

Before diving into the code, let's look at some of the basic properties we will be using when plotting.

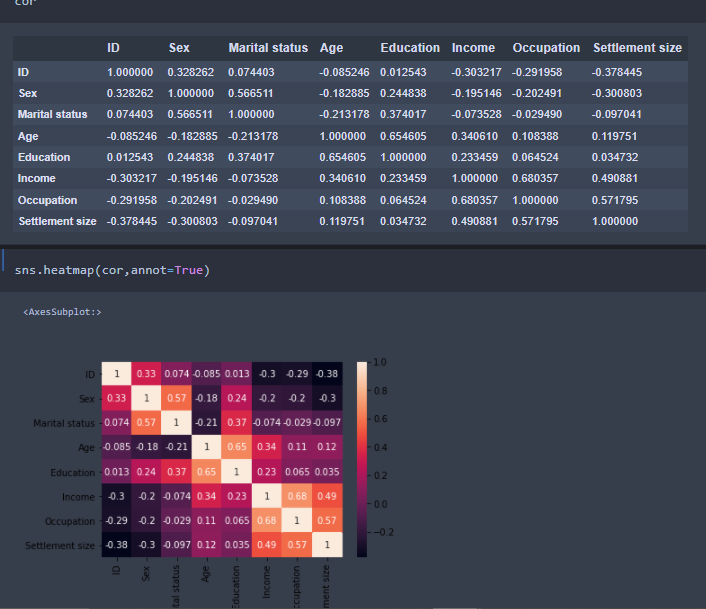
**xlabel:** Set the label for the x-axis.

**ylabel:** Set the label for the y-axis.

**title:** Set a title for the axes.

**Legend:** Place a legend on the axes.

1. data.corr() gives the correlation between the columns



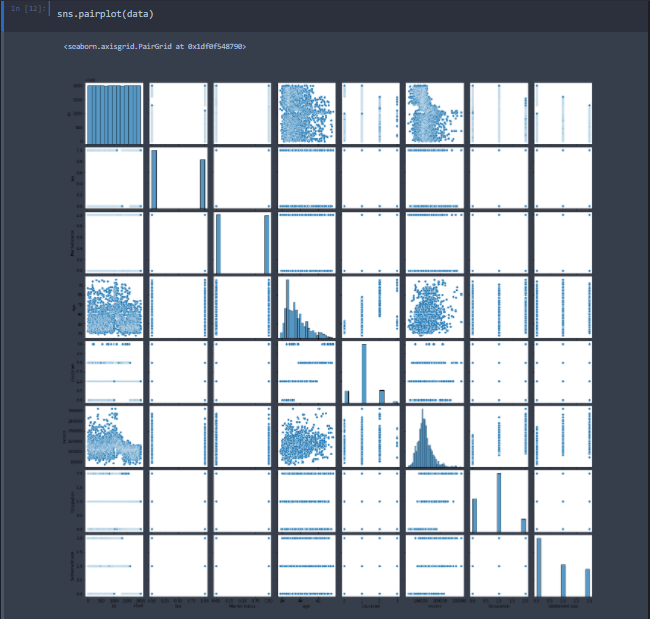
* Correlation strength varies based on colour, lighter the colour between two variables, more the strength between the variables, darker the colour displays the weaker correlation
* We can see the correlation scale values on left side of the above image

2.Pair Plot: Plot pairwise relationships in a dataset.

* By default, this function will create a grid of Axes such that each numeric variable in data will by shared across the y-axes across a single row and the x-axes across a single column. The diagonal plots are treated differently: a univariate distribution plot is drawn to show the marginal distribution of the data in each column.
* We implement this using the below code

**Code:- sns.pairplot(data)**

The output is as shown below

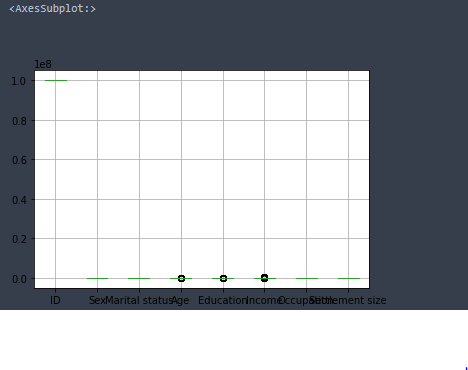


Pair plot usually gives pair wise relationships of the columns in the dataset

From the above pair plot we infer that

1.from the above plot we can draw inferences such as linearity and strength between the variables

2.how features are correlated(positive, neutral and negative)

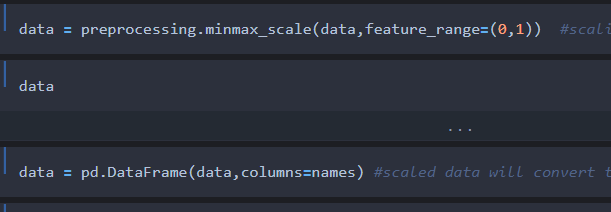
3.Box Plot: jupyter has a built-in function to create boxplot called boxplot(). A boxplot plot is a type of plot that shows the spread of data in all the quartiles 

From the above box plot we infer how the data points are spread and the existence of the outliers

**Activity 6: Feature Scaling**

There is huge disparity between the x values so let us use feature scaling.

Feature scaling is a method used to normalize the range of independent variables or features of data.



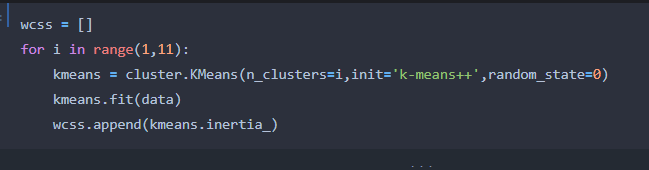
* After scaling the data will be converted into array form
* Loading the feature names before scaling and converting them back to data frame after standard scaling is applied

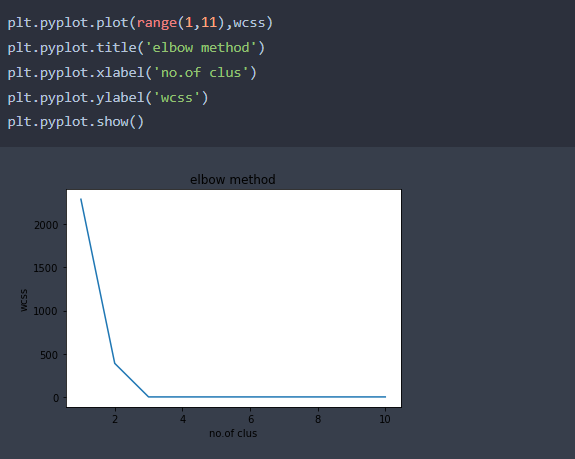
**Milestone 3: Unsupervised Model Building:**

* + Import the model building Libraries
  + Initializing the model
  + Fit and predict the clusters
  + Add the classes to the main data set and save the dataset
  + Splitting x and y
  + Splitting train and test data

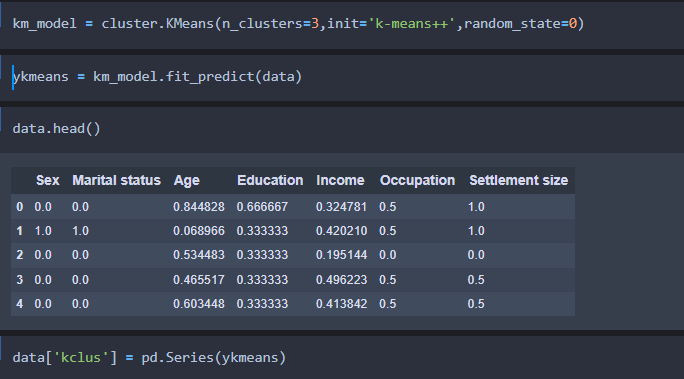
**Activity 1: Importing and initializing the model**

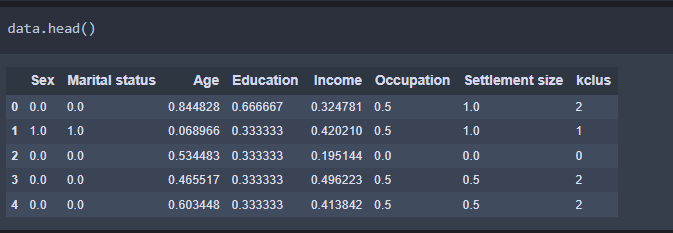
* **From sklearn.clusters import Kmeans**
* **from scipy import spatial**
* for selecting no.of clusters it is essential to plot a elbow curve,from that we can identify how many no.of clusters can be taken

****



From the above graph it can be inferred that curve has 3 bends (I.e, 0-2 ,2-3 and 3-10,so making it as 3 clusters.



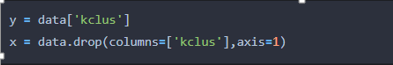
****

**Activity 6: Splitting the Dataset into Dependent and Independent variable**

* In machine learning, the concept of dependent variable (y) and independent variables(x) is important to understand. Here, Dependent variable is nothing but output in dataset and independent variable is all inputs in the dataset.
* With this in mind, we need to split our dataset into the matrix of independent variables and the vector or dependent variable. Mathematically, Vector is defined as a matrix that has just one column.

To read the columns, we will use **iloc** of pandas (used to fix the indexes for selection) which takes two parameters — [row selection, column selection].

Let’s split our dataset into independent and dependent variables.

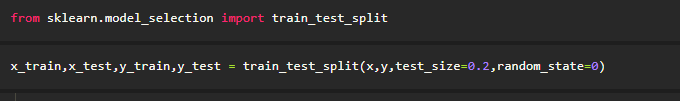


**Activity 8: Splitting the data into Train and Test**

* When you are working on a model and you want to train it, you obviously have a dataset. But after training, we have to test the model on some test dataset. For this, you will a dataset which is different from the training set you used earlier. But it might not always be possible to have so much data during the development phase. In such cases, the solution is to split the dataset into two sets, one for training and the other for testing.
* **Train Dataset**: Used to fit the machine learning model.
* **Test Dataset**: Used to evaluate the fit machine learning model.
* In general you can allocate 80% of the dataset to training set and the remaining 20% to test set.We will create 4 sets— X\_train (training part of the matrix of features), X\_test (test part of the matrix of features), Y\_train (training part of the dependent variables associated with the X train sets, and therefore also the same indices), Y\_test (test part of the dependent variables associated with the X test sets, and therefore also the same indices.
* There are a few other parameters that we need to understand before we use the class:
* **test\_size** — this parameter decides the size of the data that has to be split as the test dataset. This is given as a fraction. For example, if you pass 0.5 as the value, the dataset will be split 50% as the test dataset
* **train\_size** — you have to specify this parameter only if you’re not specifying the test\_size. This is the same as test\_size, but instead you tell the class what percent of the dataset you want to split as the training set.
* **random\_state** — here you pass an integer, which will act as the seed for the random number generator during the split. Or, you can also pass an instance of the Random\_state class, which will become the number generator. If you don’t pass anything, the Random\_state instance used by np.random will be used instead.
* Now split our dataset into train set and test using train\_test\_split class from scikit learn library.

**from sklearn import model\_selection**

**x\_train,x\_test,y\_train,y\_test=model\_selection.train\_test\_split(x,y,test\_size=0.2,random\_state =0)**



**Milestone 4: Supervised Model Building:**

Model building includes the following main tasks

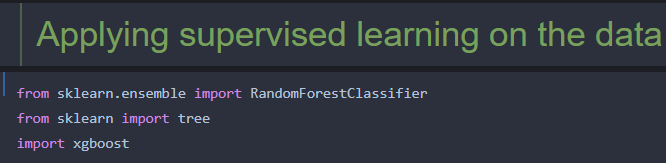
* + Import the model building Libraries
  + Initializing the model
  + Training and testing the model
  + Evaluation of Model
  + Save the Model

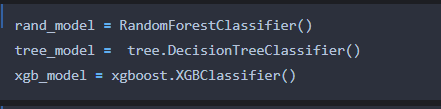
**Activity 1: Training and Testing the Model**

* Once after splitting the data into train and test, the data should be fed to an algorithm to build a model.
* There are several Machine learning algorithms to be used depending on the data you are going to process such as images, sound, text, and numerical values. The algorithms that you can choose according to the objective that you might have it may be Classification algorithms are classification algorithms.
  1. Decision Tree classifier
  2. Random Forest classifier
  3. xgboost

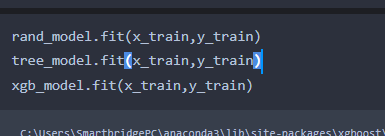
**Steps in Building the model:-**

* **Initialize the model**



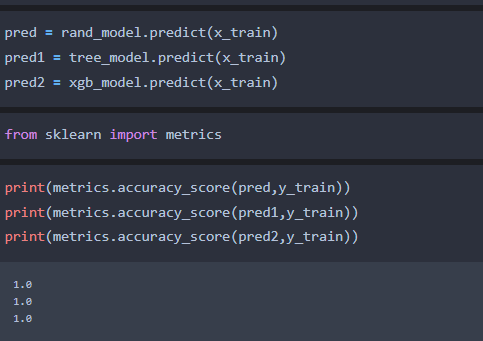


* **fit the initialized models with x\_train and y \_train data, it means that we are training the models using train data**

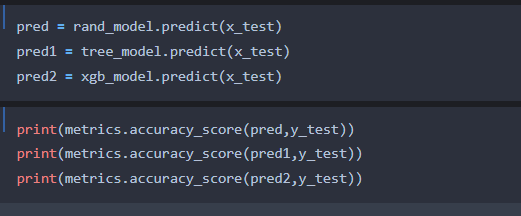


**Activity 2: Model Evaluation**

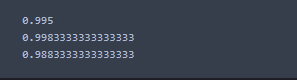
* **Accuracy testing using the train data**



* **Predict the y\_test values and calculate the accuracy**



**After predicting we will find the accuracy value of each model**

****

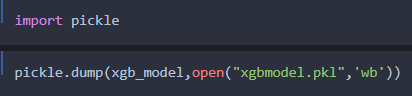
**From the above metrics we can conclude that model xgboost gives the best accuracy, other models falls under the category of over-fitting ,when measure with train data, so omitting other models and considering the xgboost model for deployment**

**Activity 3: Save the Model**

After building the model we have to save the model.

**Pickle** in **Python** is primarily **used** in serializing and deserializing a **Python** object structure. In other words, it's the process of converting a **Python** object into a byte stream to store it in a file/database, maintain program state across sessions, or transport data over the network. wb indicates write method and rd indicates read method.

This is done by the below code

****

**Milestone 4 : Application Building**

In this section, we will be building a web application that is integrated to the model we built. A UI is provided for the uses where he has to enter the values for predictions. The enter values are given to the saved model and prediction is showcased on the UI.

This section has the following tasks

* Building HTML Pages
* Building server side script

**Activity 1: Build HTML Code**

* + In this HTML page, we will create the front end part of the web page. In this page we will accept input from the user and Predict the values.

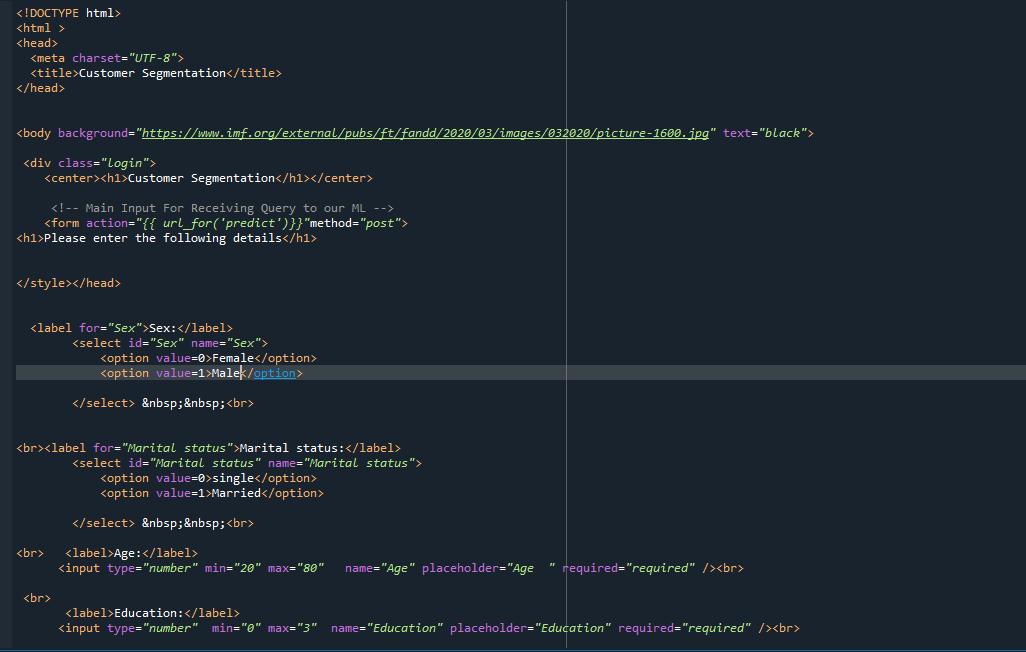
For more information regarding HTML

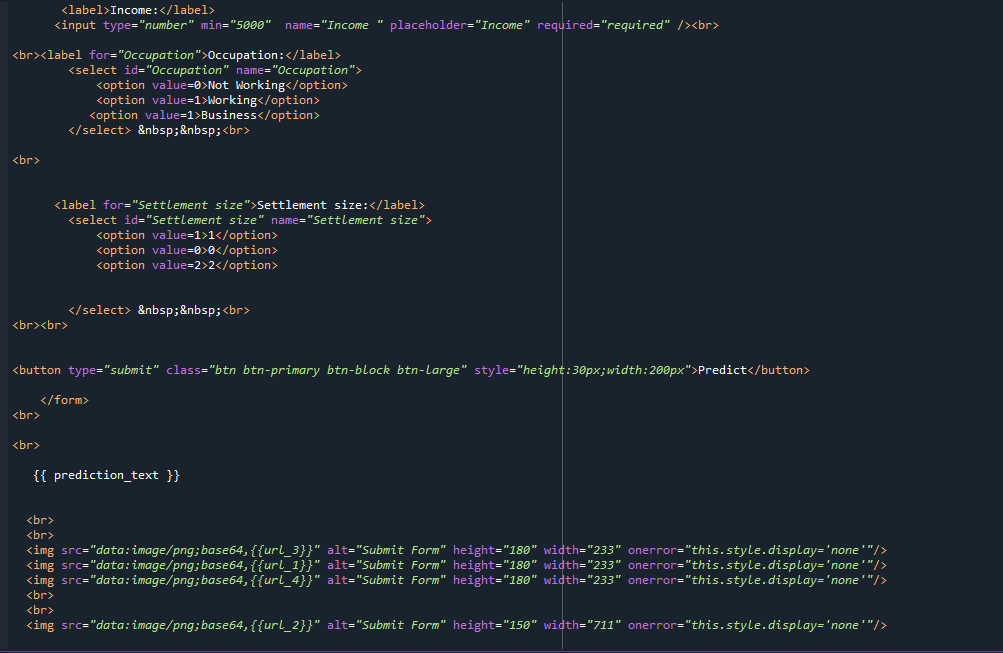
[**https://www.w3schools.com/html/**](https://www.w3schools.com/html/)

In our project we have HTML files ,they are

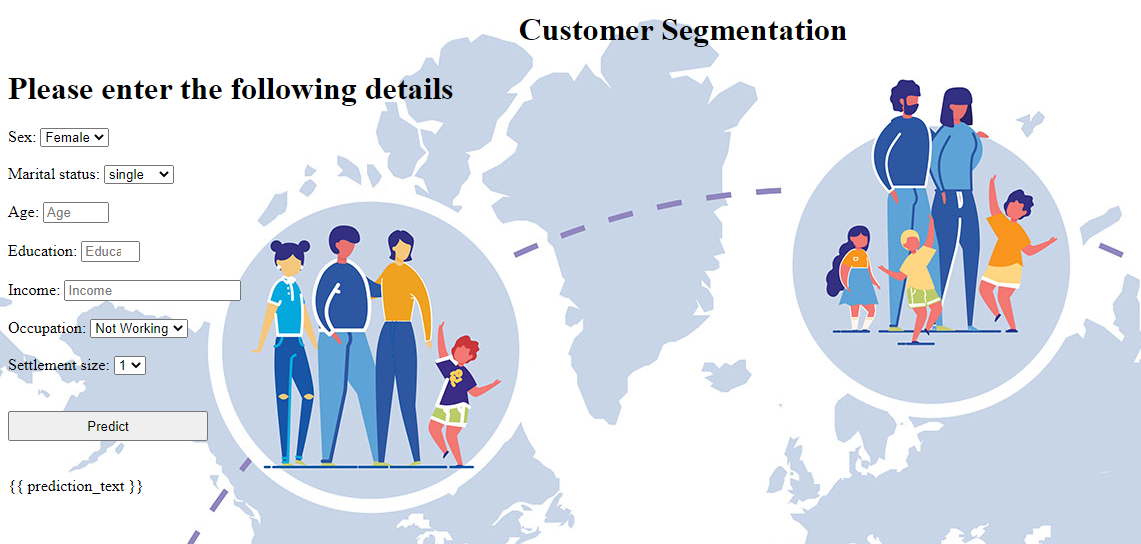
1.index.html

**index.html**

****

****

**The html page looks like**

****

It will display all the input parameters and the prediction text will display the output value of the data given by the user.

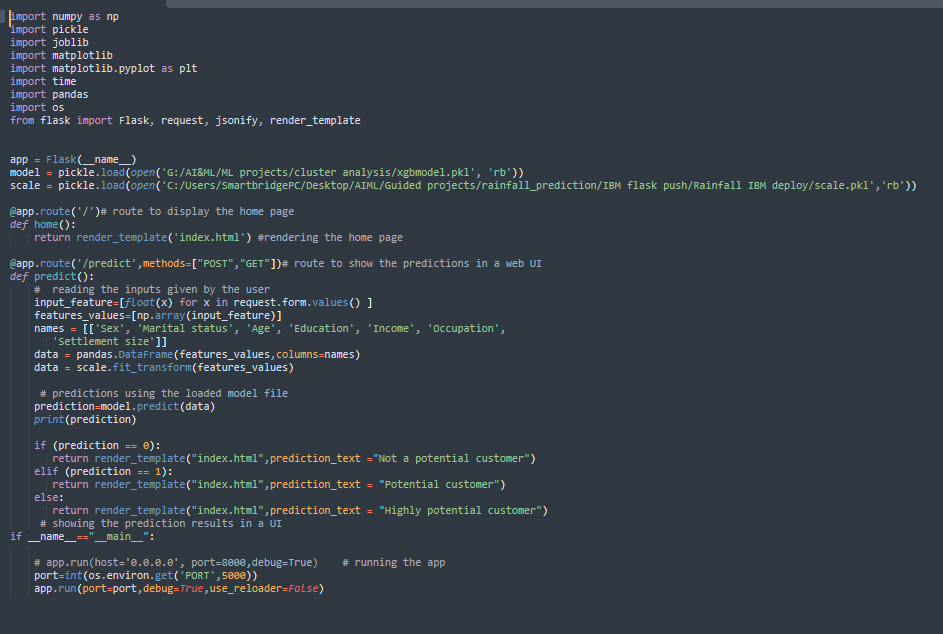
**Activity 2: Main Python Script**

Let us build app.py flask file which is a web framework written in python for server-side scripting. Let’s see step by step procedure for building the backend application.

In order to develop web api with respect to our model, we basically use Flask framework which is written in python.

* Importing the necessary libraries for building a flask application and to integrate model and html pages
* Initializing the flask app
* Calling the pkl models and saving into a variable
* Routing and rending to the html page
* Calling the inputs from the html page and saving into the variable
* Creating the data labels
* Forming the data frame with labels and the data
* Scaling the data
* Predicting the values ,by passing the data into the model
* Rendering the results on to the html pages based on the output
* If the output is class-0,it means a page which displays non potential customer will be rendered, if the output is 1 , a page with potential customer will be displayed and the output is 2 a page with highly potential customer will be rendered.

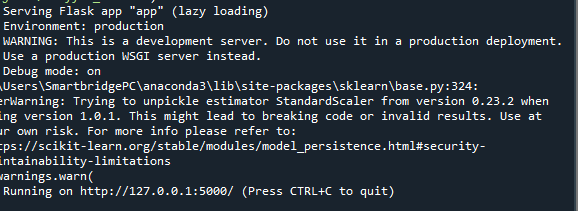
The value of \_\_name\_\_ is set to \_\_main\_\_ when module run as main program other wise it is set to name of the module

****

**Activity 3: Run the App**

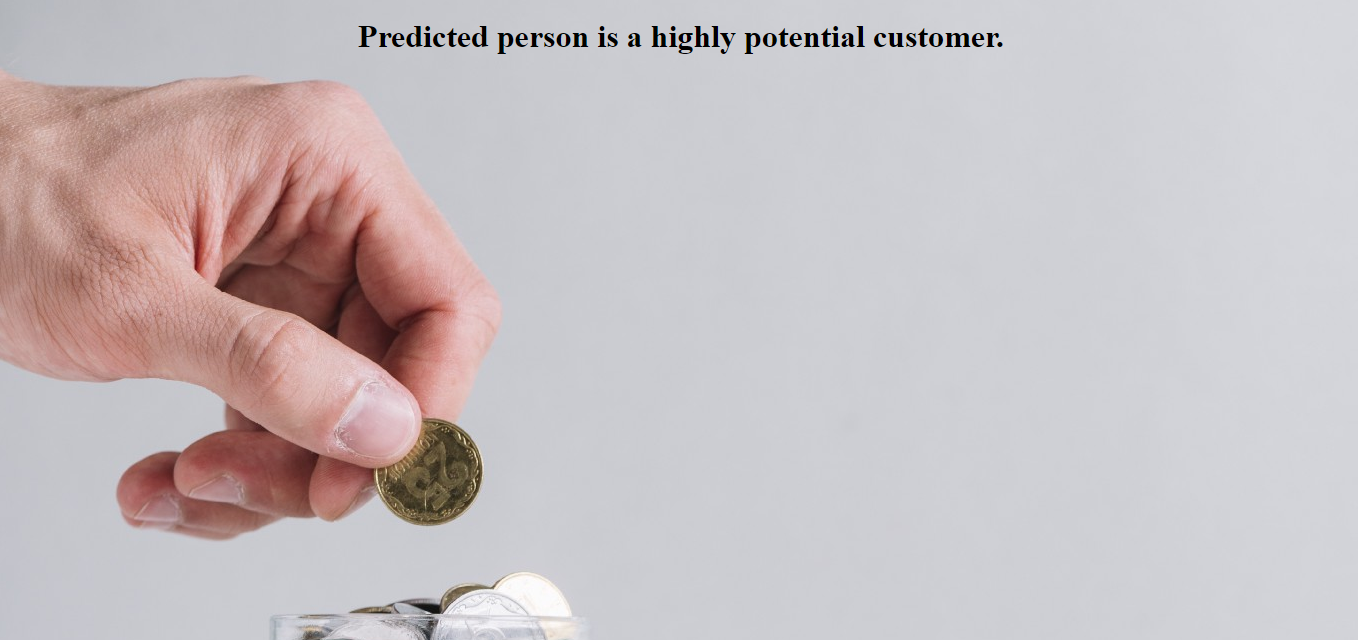
* + Open anaconda prompt from the start menu
  + Navigate to the folder where your python script is.
  + Now type “python app.py” command

Navigate to the localhost where you can view your web page,Then it will run on local host:5000



**Activity 4:**

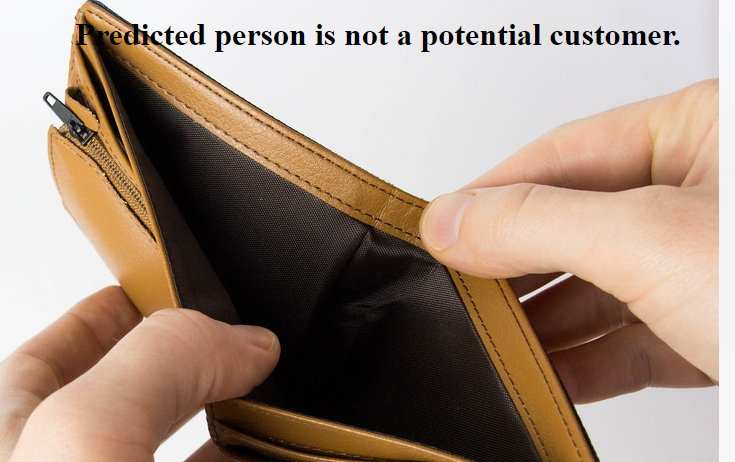
* Copy the http link and paste it in google link tab,it will display the form page
* Enter the values as per the form and click on predict buttion
* It will redirect to the page based on prediction output
* If the prediction belongs to class-2,it means that customer is highly potential



* If the prediction belongs to class-1,it means that customer is potential



* If the prediction belongs to class-0 ,it means the customer is a not potential



**MOHAMMED ALIUDDIN HYDER**

**Roll No: 19UK1A05J9**

**B-TECH CSE**