**PROJECT - CUSTOMER SEGMENTATION**

**TEAM MEMBER**

**210621104012 – DHARUNRAJ**

**Phase-2 Innovation**

# 1.Introduction

The main goal of this project is to use advanced techniques in data science to group customers based on how they behave, what they like, and information about them like their age or location. This grouping strategy is really important because it helps businesses create special marketing plans for each group of customers. This can make customers happier and more likely to keep coming back to the business. The project has several important steps like getting the data, cleaning it up, making new useful information from the data, using different ways to group the customers, showing the data in pictures, and understanding what the groups mean.

## 2.Data Collection

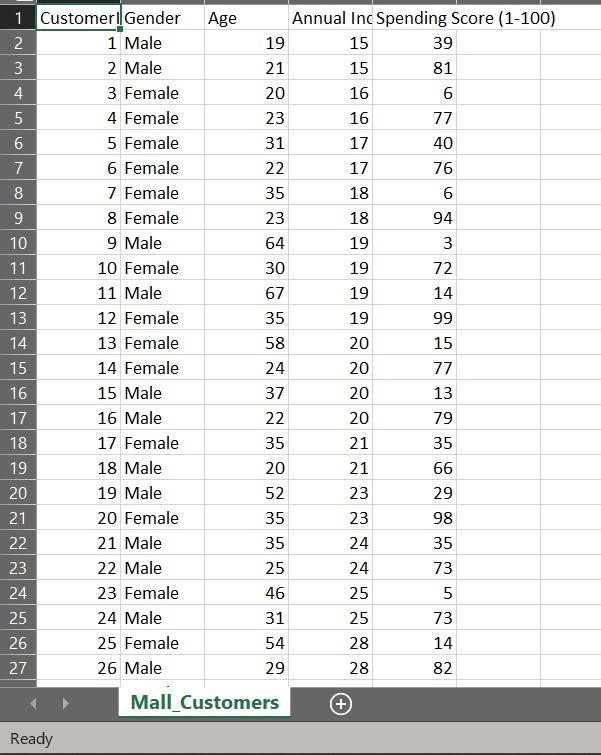
The data set is collected from Kaggle which is a leading collaborative data science platform.

Dataset Link :<https://www.kaggle.com/datasets/akram24/mall-customers>

All the data are stored in the Mall\_Customers.csv (Comma Separated Values) format, which is used to store the data efficiently.

Using pandas, we can use the .csv format for Data Processing and Manipulation

## 3.Columns being used



## The columns that are present are

1. CustomerID
2. Gender
3. Age
4. Annual Income
5. Spending Score

CustomerID only contains increasing numerical values so we will not be using it.

We will use all the other columns except CustomerID for Exploratory Data Analysis (EDA)

For grouping, we will be using Annual Income and Spending Score column.

## 4. Libraries used

**NumPy**: NumPy is a powerful library in Python for numerical and mathematical operations. It can be a valuable tool when working with data for customer segmentation tasks. Customer segmentation involves dividing a customer base into groups based on certain characteristics or behaviours, and NumPy can help you manipulate and analyse the data needed for this task

**Pandas:** Pandas, a powerful Python library, plays a crucial role in customer segmentation by facilitating data loading, exploration, cleaning, transformation, and aggregation. It supports data import from various sources, including CSV and databases, enabling the efficient manipulation of customer data. With Pandas, you can explore, clean, and preprocess data by detecting missing values and duplicates, convert data types, and create new features. Filtering and aggregating customers based on specific criteria become straightforward with Pandas.

**Matplotlib and Seaborn**: Matplotlib and Seaborn, both powerful Python libraries for data visualization, are essential tools in customer segmentation projects. Matplotlib offers a wide range of customizable plotting functions, allowing you to create insightful visualizations of segmented customer data. You can use it to plot histograms, scatterplots, and bar charts to explore customer characteristics and behavior.

**Scikit-learn:** Using Scikit-learn's K-means clustering algorithm for customer segmentation is a common and powerful technique in data analysis and marketing. Customer segmentation helps businesses understand their customer base better, tailor their marketing strategies, and improve customer experiences.

StandardScaler is a preprocessing technique commonly used in customer segmentation and clustering to ensure that the features used for clustering have a mean of 0 and a standard deviation of 1. Scaling the features is important because clustering algorithms like K-means are sensitive to the scale of the input data

**Installation:** All of the above libraries can be installed in Anaconda Environment using the anaconda prompt or locally in the system using pip

## 5. How to Train and Test

## K-means clustering is an unsupervised learning algorithm, which means it doesn't require training and testing in the same way as supervised machine learning models. However, you can evaluate the quality of the clusters indirectly using various techniques. Here's how you can approach it:

## Data Preparation:

## Split your dataset into two parts: one for clustering and one for evaluation (if needed).

## The majority of your data will be used for clustering (e.g., 80%), and a smaller portion for evaluation (e.g., 20%).

## Clustering:

## Perform K-means clustering on the training data using the steps mentioned in the previous responses.

## 6. Evaluating Performance

While K-means doesn't have a direct evaluation metric, you can assess the quality of your clusters indirectly:

**Inertia:** Inertia measures how far each point in a cluster is from the cluster's centroid. Lower inertia is generally better. You can calculate it using kmeans.inertia\_ in Scikit-learn.

**Silhouette Score:** The silhouette score measures how similar each data point in one cluster is to the data points in the neighboring clusters. A higher silhouette score is better.

**Visual Inspection:** Visualize the clusters to see if they make sense. You can use scatterplots, histograms, or othervisualization techniques to gain insights into the clustering**.**