## Summary:

## Study Task: Replication of McDonald's Case Study in Python.

The goal is to replicate and analyze parts of the McDonald's case study, focusing on data cleaning, analysis, and visualization. Each individual team member will independently complete the task to ensure a hands-on understanding of data analysis workflows.

## Requirements

1. **Dataset**: Use a publicly available dataset or create a mock dataset representing McDonald's data (e.g., menu items, nutrition facts, sales, customer feedback.
2. **Tools and Libraries**:Python, Pandas, NumPy, Matplotlib, Seaborn, and Scikit-learn (if machine learning is included).
3. **Task Guidelines**:Individuals must code their solutions.Clear documentation is required for better understanding.

#### 1. Data Preparation

- Load the dataset into Python using Pandas.
- Clean the dataset (handle missing values, remove duplicates, and ensure correct data types).
- Perform basic exploratory data analysis (EDA) to summarize the dataset (mean, median, unique values).

#### 2. Data Analysis

- **Nutritional Insights**:
    - Identify the most and least calorie-dense items.
    - Analyze the correlation between calories, protein, fat, and carbohydrates.
- **Price Analysis**:
    - Identify the average price per item by region.
    - Create a visualization of the price distribution.
- **Customer Feedback**:
    - Analyze average customer ratings by region or menu item.
    - Investigate the relationship between price and customer satisfaction.

#### 3. Visualization

- Use Matplotlib and Seaborn to create:
    - A heatmap showing the correlation between nutritional attributes.
    - A bar chart of top 10 highest-rated items.
    - A box plot showing price distribution by region.

**4. Optional (Advanced Tasks)**

- **Clustering**: Use k-means clustering to group menu items based on nutritional content.
- **Prediction**: Build a simple regression model to predict calories based on other nutritional values.

**5. Reporting**

- Summarize findings in a Jupyter Notebook with clear code explanations and insights.

---

**Step 1: Deciding (not) to Segment:**McDonald's can take the position that it caters to the entire market and that there is no need to understand systematic differences across market segments Alternatively, McDonald's can take the position that, despite their market power,there is value in investigating systematic heterogeneity among consumers andharvest these differences using a differentiated marketing strategy.

**>>My Point of View Understand:** In this Mcdonalds Understanding the problems clearly of the core issue or challenge the case study addresses . Break it down into small componentsI it will use analysing easily, Use data evidence,Consider multiple perspectives, focus on practical solutions like(considering resources and constraints the company faces.

## Step2:Specifying the ideal target segment

>>**Mypoint of understanding: 1.** We Can target regularly eating Food persons.2.Which food mostly likes 3. Which street peoples coming to eat regularly.4.we want to get some basic demographic data ( Like age , Address, Country , Phone number ,Mail….. Etc.5. Which seasons mostly likes this food and which food mostly eating.6.Target frequent customers as they are more likely to respond positively to campaigns.7.Use initial findings to draft potential target segment profiles.

## Step3: Collecting data

**>>My point of understanding:** 1. Understanding the data structure. The data set includes Perceptual attributes like Example Yummy Convenient, spicy ,fattening ,greasy, fast cheap, tasty,expensive and healthy. And demographic data  contains Like Age and Gender Use clustering techniques to group respondents based on their perceptions:k-Means Clustering: Groups individuals with similar attributes.Hierarchical Clustering: Useful for visualizing segment relationships.

## Step4: Collecting Data:

**>>My point of Understanding:**This step involves examining the dataset to understand its structure, identify patterns, and detect anomalies. Exploratory Data Analysis (EDA) helps you uncover insights, validate assumptions, and prepare the data for segmentation. We have mainly

Objectives Like(Understanding , Data structures, Identify patterns and trends , Detect anomalies, Summarise key statistics.

Example:Understand how customer preferences vary by age or region.Identify menu items with extreme calorie or price ranges.

These are the commands used to import libraries by manipulating and visualization  like bar charts , pie charts, plot charts, line charts, donut charts etc…..

(Import  pandas as pd, import seaborn as sns, import matplotlib.pyplot as plt, Import Numpy as np).

```
# Summary statistics

print(df.describe())

# Check for missing values

print(df.isnull().sum())

# Visualize correlations

sns.heatmap(df.corr(), annot=True, cmap="coolwarm")

plt.show()

# Explore a specific variable (e.g., Age distribution)

sns.distplot(df['Age'], bins=10, kde=True)

plt.title('Age Distribution')

plt.show().
```

## Step5:Extracting Segments:

## >> My point of view understanding:

This step involves dividing the dataset into distinct groups (segments) based on shared characteristics. These segments are created using clustering or other segmentation techniques to uncover meaningful patterns in the data.

**Group Similar Data Points**:Identify clusters of customers, products, or behaviors that share common attributes. And also mainly behaviour and based on profession  to analyse.

**Choose a Segmentation Method**:Use clustering algorithms like k-means, Gaussian Mixture Models, or hierarchical clustering.

**Optimize Segmentation**:Determine the ideal number of segments for meaningful results.

Here we have to find out these code .

```
from sklearn.cluster import KMeans


# Select features for clustering

features = df[['Calories', 'Protein', 'Fat', 'Carbohydrates']]


# Apply k-means (Groups data into k clusters based on distance to the nearest cluster center.)

kmeans = KMeans(n_clusters=3, random_state=42)

df['Segment'] = kmeans.fit_predict(features)


# View segments

print(df.groupby('Segment').mean())
```

**Using Mixtures of Distributions:**Uses Gaussian Mixture Models (GMM) for clustering based on probabilistic distributions.

```
from sklearn.mixture import GaussianMixture

# Apply GMM
gmm = GaussianMixture(n_components=3, random_state=42)
df['Segment'] = gmm.fit_predict(features)
```

In this By the end of this step, your data will be segmented into distinct groups, each representing customers or items with shared characteristics. These segments will be ready for profiling and deeper analysis in the next steps.

## Step 6: Profiling Segments:

## >>My point Of View Understanding:

The first step in this direction is to create a segment profile plot. The segment profile plot makes it easy to see key characteristics of each market segment and  This step involves analyzing the characteristics of each segment to understand what makes them unique. Profiling helps identify the key attributes that differentiate one segment from another, providing insights for targeted

strategies. We have an Objectives Like (summarize, Segment characteristics, Identify Key differences, prepare For targeting.)

We have an examples ::

Segment 1: Young adults (18–25) who prefer high-calorie, affordable menu items.

Segment 2: Health-conscious individuals who prioritize low-calorie options.

Segment 3: Families who purchase value meals.

```
# Group data by segment and calculate mean values

segment_profiles = df.groupby('Segment').mean()



# Print segment profiles

print(segment_profiles)



# Visualize differences across segments

import seaborn as sns

import matplotlib.pyplot as plt



# Example: Visualizing calorie preferences by segment

sns.boxplot(x='Segment', y='Calories', data=df)

plt.title('Calorie Preferences by Segment')

plt.show()
```

## Step 7: Describing Segments:

## >>My point of view Understanding:

In this step, you summarize the key insights and characteristics of each segment in a clear and actionable way. Descriptions should highlight the unique traits of each segment to communicate findings effectively to stakeholders.We have an Objectives Like(Simplify Insights,Highlight Differences,Provide Business Context.)

**Example Scenario**

For McDonald's:

- **Segment 1**: "Budget-Conscious Young Adults"
    - Customers aged 18–25, prefer affordable, high-calorie menu items.
- **Segment 2**: "Health-Conscious Professionals"
    - Aged 30–45, choose low-calorie options and salads.
- **Segment 3**: "Family Meal Buyers"
    - Families with kids, purchase combo meals and value packs.

```
# Summarize descriptive statistics for each segment

summary = df.groupby('Segment').agg({

    'Age': ['mean', 'median'],

    'Calories': ['mean', 'std'],

    'Price': ['mean'],

    'Region': lambda x: x.mode()[0]

})



# Print segment descriptions

print(summary)



# Example: Visualizing segment distributions

import seaborn as sns

import matplotlib.pyplot as plt



sns.barplot(x='Segment', y='Revenue', data=df)

plt.title('Revenue by Segment')

plt.show()
```

## Step 8: Selecting (the) Target Segment(s):

**>>My point of view Understanding:**This step involves identifying the most promising segment(s) to focus on based on business objectives, resources, and potential ROI. The chosen segment(s) will become the primary focus for marketing strategies, product development, or other initiatives.  The mainly used these tye of code for selecting target segments.

## Objectives:

**Align with Business Goals**:

- Choose segments that best support the organization's objectives (e.g., profitability, growth, customer retention).

**Assess Feasibility**:

- Ensure the segment is accessible, measurable, and actionable.

**Prioritize ROI**:

- Focus on segments with the highest potential for return on investment or strategic value.

Target Segment: "Budget-Conscious Young Adults" (Segment 1).

- Rationale: This group drives frequent purchases and responds well to promotions, making it cost-effective to target.

```
# Example: Evaluating potential target segments

    segment_evaluation = df.groupby('Segment').agg({

       'Revenue': 'mean',

       'Customer_Rating': 'mean',

       'Marketing_Cost': 'mean'

    })

print(segment_evaluation)

# Choosing the target segment based on high revenue and low marketing costtarget_segment = df[df['Segment'] == 1]

# Summary of the selected target segment

    print(target_segment.describe())
```

## Step 9: Customizing the Marketing Mix:

## >> My point of view Understanding:

This step involves tailoring the marketing strategies (the "marketing mix") to the chosen target segment(s).Customizing these elements for the selected segment ensures the offerings resonate with their specific needs and preferences.

In This customizing the marketing mix Have objective like(Product, price , place , promotion).

**Product**: Introduce healthier menu options like salads and fruit bowls for health-conscious professionals.

**Price**: Offer bundle discounts for families.

**Place**: Focus on urban locations for young adults who prefer convenience.

**Promotion**: Run targeted social media ads promoting low-calorie meals to health-conscious individuals.

## Implementation By Python

```
# Example: Analyzing the effect of promotional discounts

import seaborn as sns

import matplotlib.pyplot as plt

# Plotting revenue by segment and promotion type

sns.barplot(x='Segment', y='Revenue', hue='Promotion_Type', data=df)

plt.title('Revenue by Segment and Promotion')

plt.show()

# Example: Tailoring discounts based on segment behavior

discounted_segment = df[df['Segment'] == 1]

discounted_segment['Discounted_Price'] = discounted_segment['Price'] * 0.9
```

## Step 10: Evaluation and Monitoring:

## >> Understanding My point Of view

This step involves assessing the performance of the implemented strategies and continuously monitoring the results to ensure that the targeted segments respond as expected. It's crucial to evaluate the effectiveness of the marketing strategies and make adjustments as needed to optimize outcomes.Continuously track the performance of marketing strategies and segment responses. Adjust strategies based on performance data to ensure ongoing effectiveness and optimize results.

**Performance Metrics**: Monitor the sales of healthy menu items among health-conscious customers, and track promotional responses from budget-conscious segments.

**Adjustments**: If a particular promotion isn't performing well, tweak the offer or delivery method to# Example: Evaluate performance of the marketing strategy (e.g., tracking revenue)

```
performance = df.groupby('Segment').agg({

    'Revenue': 'sum',

    'Customer_Rating': 'mean'

})

# Visualize the performance over time

sns.lineplot(x='Month', y='Revenue', data=df)

plt.title('Monthly Revenue Performance')

plt.show()

# Adjust strategy based on performance (example: change discount for underperforming segments)

df['Adjusted_Price'] = df.apply(lambda row: row['Price'] * 0.85 if row['Segment'] == 1 else row['Price'], axis=1)
```

better appeal to the target segment.

## Outcome:

By the end of this step, you will have a continuous feedback loop in place to evaluate the effectiveness of your segmentation strategy and marketing efforts. Regular monitoring ensures that the strategies remain relevant and effective, allowing for timely adjustments to optimize results.

By systematically preparing, analyzing, and interpreting this dataset, you can derive actionable insights and create meaningful market segments for McDonald’s.