Bahria University

Karachi Campus

COURSE: CSL-452

DATA MINING LAB

TERM: SPRING 2024, CLASS: BSE- 6(B)

SUBMITTED BY:

**AHSAN SAJJAD**

**79309**

**(Name) (Enrollment No.)**

SUBMITTED TO:

Engr. HAMZA/Engr. HINA SHAKIR

SIGNED REMARKS: SCORE:

INDEX

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SNO | DATE | LAB NO | LAB BJECTIVE | REMARKS/SIGN |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Bahria University,

Karachi Campus

A logo with text on it

Description automatically generated

LAB EXPERIMENT NO.

01

LIST OF TASKS

|  |  |
| --- | --- |
| TASK NO | OBJECTIVE |
| 01 | You work at a public library that wants to digitize its book catalog and membership management. Your task is to create a Library Management System using Google Colab to allow librarians to add/edit book records and manage member information. |
| 02 | You work for an e-commerce company and have been given a dataset with information on customer orders over the past year. Load the data into Pandas, analyze it using methods like .info(), .describe(), Which products have the highest/lowest sales? Which customer segments spend the most? |
| 03 | You are a data analyst at a real estate company. You have been given a dataset of housing sale prices in different regions over the past 5 years. Load the data into Pandas and preprocess it by handling missing values and formatting columns. |
| 04 | You are a data analyst working for an automobile company. You have been provided with the Vega dataset which contains details on different vehicle models like price, engine size, horsepower, dimensions etc. |
|  |  |
|  |  |

Submitted On:

18-02-2024

(Date: DD/MM/YYYY)

**Task No. 01**:You work at a public library that wants to digitize its book catalog and membership management. Your task is to create a Library Management System using Google Colab to allow librarians to add/edit book records and manage member information.

Create a Colab notebook with a simple frontend interface using Colab forms, input boxes, dropdowns, etc.

Allow librarians to add, edit and delete book records through the interface, with details like title, author, genre, publishing year, etc.

**Solution:**

import pandas as pd

class Book:

def \_\_init\_\_(self, title, author, genre, publishing\_year):

self.title = title

self.author = author

self.genre = genre

self.publishing\_year = publishing\_year

class Member:

def \_\_init\_\_(self, name, email, contact\_number):

self.name = name

self.email = email

self.contact\_number = contact\_number

self.membership\_status = "Active"

class LibraryManagementSystem:

def \_\_init\_\_(self):

self.books\_df = pd.DataFrame(columns=["Title", "Author", "Genre", "Publishing Year"])

self.members\_df = pd.DataFrame(columns=["Name", "Email", "Contact Number", "Membership Status"])

if option == "1":

title = input("Enter title of the book: ")

author = input("Enter author of the book: ")

genre = input("Enter genre of the book: ")

publishing\_year = input("Enter publishing year of the book: ")

book = Book(title, author, genre, publishing\_year)

library\_system.add\_book(book)

print("Book added successfully!")

elif option == "2":

name = input("Enter name of the member: ")

email = input("Enter email of the member: ")

contact\_number = input("Enter contact number of the member: ")

member = Member(name, email, contact\_number)

library\_system.add\_member(member)

print("Member added successfully!")

elif option == "3":

search\_title = input("Enter title to search: ")

search\_result = library\_system.search\_book(search\_title)

print("Search Results:")

print(search\_result)

elif option == "4":

search\_name = input("Enter name to search: ")

search\_result = library\_system.search\_member(search\_name)

print("Search Results:")

print(search\_result)

elif option == "5":

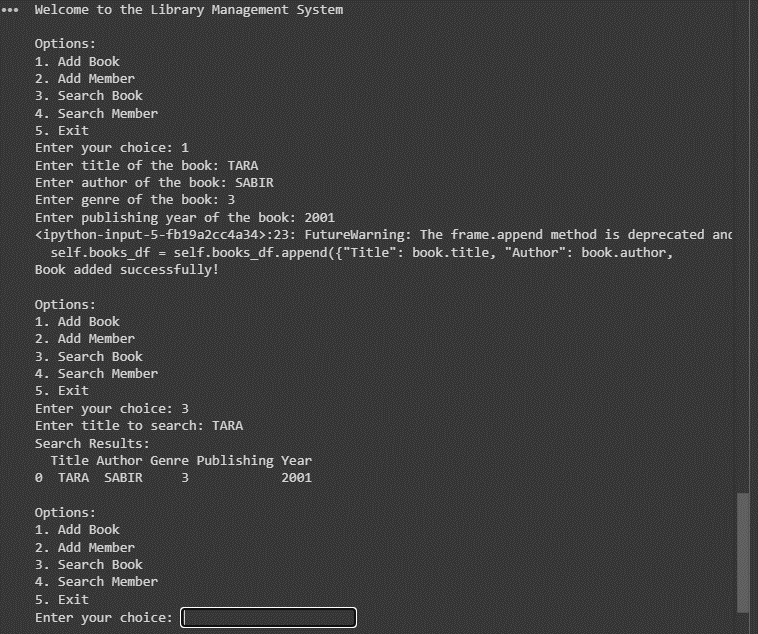
print("Exiting...")

break

else:

print("Invalid option. Please try again.")

**Output:**



**Task No. 02**: You work for an e-commerce company and have been given a dataset with information on customer orders over the past year. Load the data into Pandas, analyze it using methods like .info(), .describe(), Which products have the highest/lowest sales? Which customer segments spend the most?

**Solution:**

import pandas as pd

# Load the data from a CSV file

df = pd.read\_csv("/content/housing.csv")

missing\_values = df.isnull().sum()

# Print a summary of missing values in each column

print(missing\_values[missing\_values > 0])

# Drop columns with a significant amount of missing data

# Round numeric columns to two decimal places

for col in df.select\_dtypes(include=["int64", "float64"]):

df[col] = df[col].round(2)

# Display the DataFrame

print(df.head())

else:

print("The 'SalePrice' column does not exist in the DataFrame after dropping columns with missing values.")

A screenshot of a computer screen

Description automatically generated**Output:**

**Task No. 03**: You are a data analyst at a real estate company. You have been given a dataset of housing sale prices in different regions over the past 5 years. Load the data into Pandas and preprocess it by handling missing values and formatting columns.

**Solution:**

import pandas as pd

# Load the dataset into a Pandas DataFrame

df = pd.read\_csv("/content/housing.csv")

# Explore the dataset

print("Dataset Info:")

print(df.info())

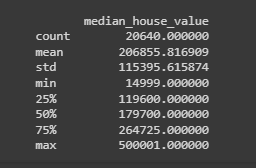
print("\nSummary Statistics:")

print(df.describe())

**Output:**

A screenshot of a computer

Description automatically generated



**Task No. 04**: You are a data analyst working for an automobile company. You have been provided with the Vega dataset which contains details on different vehicle models like price, engine size, horsepower, dimensions etc.

**Solution:**

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

import altair as alt

# Load the Vega dataset into a Pandas DataFrame

df = pd.DataFrame({data})

# Display the first few rows of the DataFrame

print(df.head())

# Scatterplot of engine size vs. price using Matplotlib

plt.figure(figsize=(10, 6))

plt.scatter(df['yearOfRegistration'], df['price'], alpha=0.5)

# Grouping by vehicle type and analyzing statistics

vehicle\_type\_stats = df.groupby('vehicleType').agg({'price': ['mean', 'median', 'std']}).reset\_index()

# Display statistics for each vehicle type

print("\nStatistics by Vehicle Type:")

print(vehicle\_type\_stats)

# Visualization using Altair for grouping by vehicle type

alt.Chart(df).mark\_bar().encode(

x='vehicleType',

y='mean(price)',

color='vehicleType',

tooltip=['vehicleType', 'mean(price)']

**A graph with numbers and lines

Description automatically generated**).properties(

title='Mean Price by Vehicle Type'

).interactive()

**Output:**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a graph

Description automatically generated**

**A graph with blue lines and a line

Description automatically generated**