1 SIMPLE LIBRARY MANAGEMENT SYSTEM

* I HAVE TAKEN BOOK AS CLASS THREE PROPERTIES AND I USE CONSTRUCTOR FOR THREE PROPERTIES

class Book:

def \_\_init\_\_(self, title, author, isbn):

self.title = title

self.author = author

self.isbn = isbn

def \_\_str\_\_(self): REPRESENTATION OF BOOK

return f"Title: {self.title}, Author: {self.author}, ISBN: {self.isbn}"

class Library: MANAGE THE BOOKS USING LIST FUNCTION

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

self.books = []

def add\_book(self, book: Book):ADDING BOOKS TO LIBRARY FUNCTION

self.books.append(book)

print(f"Book '{book.title}' by {book.author} added to the library.")

def remove\_book(self, isbn: str): USED TO REMOVE FROM LIBRARY

for book in self.books:

if book.isbn == isbn:

self.books.remove(book)

print(f"Book with ISBN {isbn} removed from the library.")

return

print(f"No book found with ISBN {isbn}.")

def display\_books(self):

if not self.books:

print("No books available in the library.")

else:

print("Books in the library:")

for book in self.books:

print(book)

2 Pandas: DataFrame Manipulation

import pandas as pd

data = {

'Product': ['A', 'B', 'A', 'C', 'B', 'C'],

'Sales': [100, 200, 150, 300, 250, 400],

'Region': ['North', 'South', 'North', 'East', 'South', 'East']

}

df = pd.DataFrame(data)

total\_sales\_per\_product = df.groupby('Product')['Sales'].sum()

print("Total Sales Per Product:)

highest\_sales\_per\_region = df.groupby('Region')['Sales'].max()

print("Highest Sales Per Region:")

df['Discounted Price'] = df['Sales'] \* 0.9

print("DataFrame with Discounted Price:")

3 REVERSE WORDS

def reverse\_words(s: str) -> str:

words = s.split()

reversed\_words = words[::-1] SLICING

reversed\_string = ' '.join(reversed\_words)

return reversed\_string

4 MISSING NUMBER

def find\_missing\_number(arr): n = len(arr) + 1

expected\_sum = n \* (n + 1) // 2

actual\_sum = sum(arr)

return (‘missing number’)

5 MULTHREADING SQUARE