TASK 1:

# **Web scrapping using python and storing in csv file**

import requests

from bs4 import BeautifulSoup

import csv

# Define the URL to scrape

url = 'http://quotes.toscrape.com'

# Send an HTTP GET request to the URL

response = requests.get(url)

# Check if the request was successful (status code 200)

if response.status\_code == 200:

# Parse the HTML content of the page using BeautifulSoup

soup = BeautifulSoup(response.text, 'html.parser')

# Create a list to store the scraped data

data = []

# Find and extract the quotes and their authors

for quote in soup.find\_all('div', class\_='quote'):

text = quote.find('span', class\_='text').text

author = quote.find('small', class\_='author').text

data.append({'Quote': text, 'Author': author})

# Define the CSV file name

csv\_file = 'scraped\_quotes.csv'

# Write the data to a CSV file

with open(csv\_file, 'w', newline='') as file:

fieldnames = ['Quote', 'Author']

writer = csv.DictWriter(file, fieldnames=fieldnames)

# Write the header row

writer.writeheader()

# Write the data rows

for item in data:

writer.writerow(item)

print(f'Data scraped successfully and saved to {csv\_file}')

else:

print(f'Failed to retrieve the web page. Status code: {response.status\_code}')

TASK 2:

# **Text based game (guessing the numbers in certain no of chances)**

import random

# Set the range and number of attempts

lower\_limit = 1

u pper\_limit = 100

max\_attempts = 10

# Generate a random number for the player to guess

secret\_number = random.randint(lower\_limit, upper\_limit)

print(f"Welcome to the Number Guessing Game! You have {max\_attempts} attempts to guess the number between {lower\_limit} and {upper\_limit}.")

# Main game loop

attempts = 0

while attempts < max\_attempts:

try:

# Ask the player for their guess

guess = int(input("Enter your guess: "))

# Check if the guess is correct

if guess == secret\_number:

print(f"Congratulations! You guessed the number {secret\_number} correctly in {attempts + 1} attempts.")

break

elif guess < secret\_number:

print("Try a higher number.")

else:

print("Try a lower number.")

attempts += 1

# Check if the player has run out of attempts

if attempts == max\_attempts:

print(f"Sorry, you've run out of attempts. The secret number was {secret\_number}.")

break

except ValueError:

print("Invalid input. Please enter a valid number.")