|  |
| --- |
| **Project Presentation** |



|  |
| --- |
| **Source Code** |

#FIRST I IMPORTED ALL REQUIRED PACKAGAES AND LIBRARIES  
import sys  
import os  
import time  
import random  
import hyperlink  
import randfacts  
import requests  
from bs4 import BeautifulSoup  
import datetime  
import string  
from random import \*  
import re  
from nltk.corpus import wordnet  
import webbrowser  
  
#THEN I STARTED TO TAKE USERINPUT BY ASKING GENERAL QUESTIONS  
  
name = input("WIZ - Hello , what is your name? ")  
time.sleep(1)  
print("WIZ - HELLO " + name + ", MY NAME IS WIZ AND I AM HERE TO HELP YOU !!")  
use = input("WIZ - HOW ARE YOU?? ")  
list1 = [ "fine","great","good","amazing","i'm good","i am good","i am great","i am fine","i'm fine","and","awesome","good.","Good","Fine","Amazing"]  
list2 = ["bye","bie","goodbye","see you","see yaa","byee"]  
list3 = ["hey","hello","hi","heyaa"]  
  
#IF THE USER INPUT IS IN LISTS THEN I GAVE SPECIFY PRINT COMMAND FOR GENERAL GREETINGS  
  
if use.casefold() in list1:  
 print("WIZ - NICE!! , HOW CAN I HELP YOU? ")  
else:  
 print("WIZ - SORRY TO HEAR THAT , HOW CAN I HELP YOU?")  
  
print("IF YOU WANT TO KNOW WHAT I CAN DO JUST (TYPE HELP)")  
  
#I USED WHILE LOOP WITH OS LIBRARY SO THAT AFTER EVERY RESPONSE THERE IS CLEAR SCREEN  
#THEN I USED BASIC IF ELSE STATEMENTS TO CODE MY CHATBOT FUNCTIONS  
  
while True:  
 userinput = input(name + "- ")  
 os.system("cls")  
 if "help" in userinput.lower().split():  
 print("WIZ - I CAN TELL YOU WEATHER OF ANY CITY IN WORLD(TYPE WEATHER) \n I CAN TELL YOU LATEST NEWS (TYPE NEWS) \n I CAN SPLIT WORDS FOR SENTENCES TO TRY THIS (TYPE SPLIT) \n I CAN COUNTDOWN FOR YOU (TYPE COUNTDOWN) \n I CAN TELL YOU FAMOUS TOURIST ATTRACTIONS TO VISIT IN SOME POPULAR CITIES OF UNITED KINDOM (TYPE PLAN) \n I CAN CONVERT ANY TEXT INTO ANIMATED TEXT (TYPE ANIMATE) \n I CAN GENERATE RANDOM PASSWORDS (TYPE PASS) \n I CAN TELL YOU A RANDOM FACT (TYPE FACT)")  
 elif "plan" in userinput.lower().split():  
 print("WIZ - TYPE A NAME OF ANY MAIN CITY IN UNITED KINGDOM TO GET BEST TOURIST ATTRACTIONS THAT CITY")  
 print("YOU CAN TYPE ANY OF THE CITIES NAMES \n COVENTRY,BIRMINGHAM,LONDON,EDINBURGH,GLASGOW,DUBLIN,BRISTOL,NEWCASTLE UPON TYNE,LEEDS,MANCHESTER,LIVERPOOL")  
 print("TYPE (BACK) TO GO BACK TO MAIN PAGE")  
  
  
 #I USED CASEFOLD FUNCTION SO THAT EVEN IF USER WRITES THE INPUT IN ANY CASE ,BOT RECOGNIZES THE WORD  
 #I USED REQUESTS AND BBEAUTIFULSOUP WHICH WAS VERY HELPFUL TO ME TO SCRAPE TOP TOURIST ATTRACTIONS OF MAIN CITIES OF UK  
 #ALL OF THE DATA IS TAKEN FROM (https://www.planetware.com/)  
 #I GOT HELP ON THIS FROM GEEKSFORGEEKS.ORG  
  
  
 elif userinput.casefold() == "coventry":  
 import requests  
 from bs4 import BeautifulSoup  
  
 url = 'https://www.planetware.com/england/top-rated-things-to-do-in-coventry-eng-1-47.htm'  
 response = requests.get(url)  
 soup = BeautifulSoup(response.text, 'html.parser')  
 maindata = soup.find('body').find\_all('h2')  
 unwanted = ['More on England']  
  
 # I USED A FOR LOOP SO THAT IT PRINTS THE DATA FROM WEBPAGE AND REMOVE THE UNWANTED DATA  
 #I CHECKED ALL UNWANTED DATA BY RUNNING EACH IN DIFFERENT TERMINAL  
 for x in list(dict.fromkeys(maindata)):  
 if x.text.strip() not in unwanted:  
 print(x.text.strip())  
 print("TYPE (BACK) TO GO BACK TO MAIN PAGE")  
 elif userinput.casefold() == "birmingham":  
 url = 'https://www.planetware.com/tourist-attractions-/birmingham-eng-wm-brum.htm'  
 response = requests.get(url)  
 soup = BeautifulSoup(response.text, 'html.parser')  
 maindata = soup.find('body').find\_all('h2')  
 unwanted = ['More on England', 'Where to Stay in Birmingham for Sightseeing',  
 'More Related Articles on PlanetWare.com']  
 for x in list(dict.fromkeys(maindata)):  
 if x.text.strip() not in unwanted:  
 print(x.text.strip())  
 print("TYPE (BACK) TO GO BACK TO MAIN PAGE")  
  
  
  
 elif userinput.casefold() == "london":  
 url = 'https://www.planetware.com/tourist-attractions-/london-eng-l-lon.htm'  
 response = requests.get(url)  
 soup = BeautifulSoup(response.text, 'html.parser')  
 maindata = soup.find('body').find\_all('h2')  
 unwanted = ['Where to Stay in London for Sightseeing',  
 'Tips and Tours: How to Make the Most of Your Visit to London',  
 'More Related Articles on PlanetWare.com',  
 'More on England']  
 for x in list(dict.fromkeys(maindata)):  
 if x.text.strip() not in unwanted:  
 print(x.text.strip())  
 print("TYPE (BACK) TO GO BACK TO MAIN PAGE")  
  
  
  
 elif userinput.casefold() == "manchester":  
 url = 'https://www.planetware.com/tourist-attractions-/manchester-eng-m-man.htm'  
 response = requests.get(url)  
 soup = BeautifulSoup(response.text, 'html.parser')  
 maindata = soup.find('body').find\_all('h2')  
 unwanted = ['Where to Stay in Manchester for Sightseeing', 'More on England']  
 for x in list(dict.fromkeys(maindata)):  
 if x.text.strip() not in unwanted:  
 print(x.text.strip())  
 print("TYPE (BACK) TO GO BACK TO MAIN PAGE")  
  
  
  
 elif userinput.casefold() == "leeds":  
 url = 'https://www.planetware.com/tourist-attractions-/leeds-eng-wy-lee.htm'  
 response = requests.get(url)  
 soup = BeautifulSoup(response.text, 'html.parser')  
 maindata = soup.find('body').find\_all('h2')  
 unwanted = ['Where to Stay in Leeds for Sightseeing', 'More on England']  
 for x in list(dict.fromkeys(maindata)):  
 if x.text.strip() not in unwanted:  
 print(x.text.strip())  
 print("TYPE (BACK) TO GO BACK TO MAIN PAGE")  
  
  
  
 elif userinput.casefold() == "liverpool":  
 url = 'https://www.planetware.com/tourist-attractions-/liverpool-eng-mrs-liv.htm'  
 response = requests.get(url)  
 soup = BeautifulSoup(response.text, 'html.parser')  
 maindata= soup.find('body').find\_all('h2')  
 unwanted = ['Where to Stay in Liverpool for Sightseeing', 'More on England']  
 for x in list(dict.fromkeys(maindata)):  
 if x.text.strip() not in unwanted:  
 print(x.text.strip())  
 print("TYPE (BACK) TO GO BACK TO MAIN PAGE")  
  
  
  
 elif userinput.casefold() == "glasgow":  
 url = 'https://www.planetware.com/tourist-attractions-/glasgow-sco-stra-glas.htm'  
 response = requests.get(url)  
 soup = BeautifulSoup(response.text, 'html.parser')  
 maindata = soup.find('body').find\_all('h2')  
 unwanted = ['Where to Stay in Glasgow for Sightseeing', 'More on Scotland',  
 'More Must-See Attractions near Glasgow']  
 for x in list(dict.fromkeys(maindata)):  
 if x.text.strip() not in unwanted:  
 print(x.text.strip())  
 print("TYPE (BACK) TO GO BACK TO MAIN PAGE")  
  
  
  
 elif userinput.casefold() == "edinburgh":  
 url = 'https://www.planetware.com/tourist-attractions-/edinburgh-things-to-do-sco-loth-edin.htm'  
 response = requests.get(url)  
 soup = BeautifulSoup(response.text, 'html.parser')  
 maindata = soup.find('body').find\_all('h2')  
 unwanted = ['Where to Stay in Edinburgh for Sightseeing', 'More on Scotland',  
 'More Must-See Attractions near Glasgow',  
 'Tips and Tours: How to Make the Most of Your Visit to Edinburgh',  
 'Frequently Asked Questions',  
 'How do you get from Edinburgh Airport to the city center?'  
 , 'What are the best shopping areas in Edinburgh?'  
 , 'What are the must-visit destinations near Edinburgh?']  
 for x in list(dict.fromkeys(maindata)):  
 if x.text.strip() not in unwanted:  
 print(x.text.strip())  
 print("TYPE (BACK) TO GO BACK TO MAIN PAGE")  
  
  
  
 elif userinput.casefold() == "dublin":  
 url = 'https://www.planetware.com/tourist-attractions-/dublin-irl-db-dub.htm'  
 response = requests.get(url)  
 soup = BeautifulSoup(response.text, 'html.parser')  
 maindata = soup.find('body').find\_all('h2')  
 unwanted = ['Where to Stay in Edinburgh for Sightseeing', 'More on Ireland',  
 'More Must-See Attractions near Glasgow',  
 'Tips and Tours: How to Make the Most of Your Visit to Dublin',  
 'Frequently Asked Questions', "Editor's Tips", 'More Related Articles on PlanetWare.com',  
 'Where to Stay in Dublin for Sightseeing']  
 for x in list(dict.fromkeys(maindata)):  
 if x.text.strip() not in unwanted:  
 print(x.text.strip())  
 print("TYPE (BACK) TO GO BACK TO MAIN PAGE")  
  
  
  
 elif userinput.casefold() == "bristol":  
  
 url = 'https://www.planetware.com/tourist-attractions-/bristol-eng-av-bristol.htm'  
 response = requests.get(url)  
 soup = BeautifulSoup(response.text, 'html.parser')  
 maindata = soup.find('body').find\_all('h2')  
 unwanted = ['Where to Stay in Edinburgh for Sightseeing', 'More on England',  
 'More Must-See Attractions near Glasgow',  
 'Tips and Tours: How to Make the Most of Your Visit to Dublin',  
 'Frequently Asked Questions', "Editor's Tips", 'More Related Articles on PlanetWare.com',  
 'Where to Stay in Bristol for Sightseeing']  
 for x in list(dict.fromkeys(maindata)):  
 if x.text.strip() not in unwanted:  
 print(x.text.strip())  
 print("TYPE (BACK) TO GO BACK TO MAIN PAGE")  
  
  
  
 elif userinput.casefold() == "newcastle upon tyne":  
  
 url = 'https://www.planetware.com/tourist-attractions-/newcastle-upon-tyne-eng-tw-nut.htm'  
 response = requests.get(url)  
 soup = BeautifulSoup(response.text, 'html.parser')  
 maindata = soup.find('body').find\_all('h2')  
 unwanted = ['Where to Stay in Edinburgh for Sightseeing', 'More on England',  
 'More Must-See Attractions near Glasgow',  
 'Tips and Tours: How to Make the Most of Your Visit to Dublin',  
 'Frequently Asked Questions', "Editor's Tips", 'More Related Articles on PlanetWare.com',  
 'Where to Stay in Newcastle upon Tyne for Sightseeing']  
 for x in list(dict.fromkeys(maindata)):  
 if x.text.strip() not in unwanted:  
 print(x.text.strip())  
 print("TYPE (BACK) TO GO BACK TO MAIN PAGE")  
  
#IN THE CASE OF NEWS ALMOST SAME CODE IS USED AND SPILT FUNCTION IS USED  
#ALL THE DATA IS TAKEN FROM BBC.COM/NEWS  
  
 elif "news" in userinput.lower().split() :  
 url = 'https://www.bbc.com/news'  
 response = requests.get(url)  
  
 soup = BeautifulSoup(response.text, 'html.parser')  
 maindata = soup.find('body').find\_all('h3')  
 unwanted = ['BBC World News TV', 'BBC World Service Radio',  
 'News daily newsletter', 'Mobile app', 'Get in touch', 'BBC News Channel',  
 'BBC Radio 5 Live']  
  
 for x in list(dict.fromkeys(maindata)):  
 if x.text.strip() not in unwanted:  
 print(x.text.strip())  
 print("TYPE (BACK) TO GO BACK TO MAIN PAGE")  
  
#FOR WEATHER I TRIED TO USED API . IT IS A FREE API PROVIDED BY YAHOO AND I GOT TO KNOW ABOUT THIS FROM (RAPIDAPI.COM)  
  
 elif "weather" in userinput.lower().split():  
 inputof = input("WIZ - TYPE THE NAME OF CITY TO SHOW ITS WEATHER")  
 import requests  
  
 url = "https://yahoo-weather5.p.rapidapi.com/weather"  
  
 querystring = {"location": inputof, "format": "json", "u": "f"}  
  
 headers = {  
 "X-RapidAPI-Host": "yahoo-weather5.p.rapidapi.com",  
 "X-RapidAPI-Key": "9fd654d284msh3607ec1ec8113fap1aa355jsnd430c9cd2f34"  
 }  
  
 response = requests.request("GET", url, headers=headers, params=querystring)  
  
 print(response.text)  
 print( "(TYPE BACK ) TO GO BACK TO MAIN PAGE")  
#BASIC SPILIT COMMAND USED TO HELP SOMEONE SPLIT THEIR WORDS FROM PARAGRAPH OR LINE.  
 elif "split" in userinput.lower().split():  
 input4 = input("WIZ - WRITE YOUR SENTENCE OR PARAGRAPH AND I WILL SPLIT WORDS FROM IT")  
 words = input4.split()  
 print(words)  
 print(" (TYPE BACK ) TO GO BACK TO MAIN PAGE")  
#I USED FOR LOOP FOR COUNTDOWN SO USER CAN GET COUNTDOWN FOR ANY SECONDS  
 elif "countdown" in userinput.lower().split():  
 input5 = int(input("how many seconds timer you want?"))  
 for i in range(input5):  
 print(str(input5 - i) + "seconds remaining")  
 time.sleep(1)  
 print("WIZ - TIME IS UP")  
 print(" (TYPE BACK ) TO GO BACK TO MAIN PAGE")  
  
# USED CONTINUE HERE SO THAT IF USER WANTS TO GO TO BACK PAGE  
 elif "back" in userinput.lower().casefold():  
 print("WIZ - IF YOU WANT TO KNOW WHAT I CAN DO JUST (TYPE HELP)")  
 continue  
#IF NAME IS IN USER INPUT IT WILL GIVE RESPONSE BY TELLING ITS NAME  
  
#I REALLY WANTED TO ADD ONN MANY THINGS TO THE GREETING PART AND I ALSO SAW MANY TUTORIALS BUT DUE TO POOR CODING KNOWLEDGE IT WAS DIFFICULT  
 elif "name" in userinput.lower().split(" "):  
 print("WIZ - MY NAME IS WIZ AND I AM HERE TO HELP.")  
  
  
#USED MATPLOTLIB AND PYPLOT WHICH I LEARNED FROM 4005CEM DATA VISUALISATION IN PYTHON WHICH HELPED ME TO PERFORM THIS CODE  
#I GOT TO KNOW ABOUT THIS CODE FROM https://www.tutorialspoint.com/plot-animated-text-on-the-plot-in-matplotlib  
 elif "animate" in userinput.lower().split():  
 from matplotlib import pyplot as plt, animation  
  
 userinput = input("WIZ - TYPE TEXT THAT YOU WANT TO CONVERT TO ANIMATED TEXT")  
 plt.rcParams["figure.figsize"] = [8, 4]  
 plt.rcParams["figure.autolayout"] = True  
 fig, ax = plt.subplots()  
 ax.set(xlim=(-1, 1), ylim=(-1, 1))  
 string = userinput  
 label = ax.text(0, 0, string[0], ha='center', va='center', fontsize=20, color="Red")  
  
  
 def animate(i):  
 label.set\_text(string[:i + 1])  
  
  
 anim = animation.FuncAnimation(  
 fig, animate, interval=200, frames=len(string))  
 ax.axis('off')  
 plt.show()  
 print("TYPE (BACK) TO GO BACK TO MAIN PAGE")  
# I USED DATETIME PACKAGE SO THAT IF USER WANTS TO KNOW DATE AND TIME  
 elif "date" in userinput.lower().split():  
 now = datetime.datetime.now()  
 print ("Current date and time:")  
 print("WIZ - " +now.strftime("%y-%m-%d %H:%M:%S"))  
#USED BOOLEAN SO THAT IF HOW AND YOU IS IN USER INPUT AT SAME TIME IT SHOULD REPLY  
 elif "how" and "you" in userinput.lower().split():  
 print("WIZ - I AM GOOD AND THANKS FOR ASKING")  
#USED STRING AND RANDOM PACKAGE TO DO A CODE THAT CAN GIVE YOU RANDOM PASS WORD  
 elif "pass" in userinput.lower().split():  
  
 characters = string.ascii\_letters + string.punctuation + string.digits  
 password = "".join(choice(characters) for x in range(randint(8,16)))  
 print("WIZ - " + password)  
  
#BASIC RESPONSES TO USER INPUT DONE WITH THE HELP OF IF ELSE STATEMENT AND USED BREAK SO WHEN USER TYPES BREAK THE PROCESS ENDS  
 elif userinput.casefold() in list2 :  
 print("WIZ - BYEE!, TAKE CARE \n ")  
 break  
 elif userinput.lower() in list3:  
 print("WIZ - HELLO!")  
 elif "fact" in userinput.lower().split():  
 print("WIZ - "+randfacts.get\_fact())  
  
  
#IF USER INPUTS SOMEETHING THAT BOT DOESNT UNDERSTAND IT WILL REPLY BY SORRY I CANT UNDERSTAND  
 else:  
 print("WIZ - I AM SORRY , I CAN'T UNDERSTAND WHAT YOU ARE TRYING TO SAY \n PLEASE SEE THE INSTRUCTIONS ABOVE \n (TYPE BACK ) TO GO BACK TO MAIN PAGE ")