# Virtual Sid

***Siddiq M***

## TABLE OF CONTENTS

REQUIREMENTS

- LIBRARIES & THEIR

PURPOSES

- CODING
- OUTPUT
- BIBLIOGRAPHY

# HARDWARE REQUIREMENTS

Processor: Intel(R) Core(TM) i3-2120 CPU @ 3.30GHZ
RAM: 4.00 GB
Hard Disk: 64 GB
Monitor: 15 VGA COLOR
Mouse: Any
Keyboard: Any

# SOFTWARE REQUIREMENTS

Operating System: 64-bit Operating System,

x64-based processor

Software: IDLE

Coding Language: Python

Automation Software: Chromium (Latest released Version)

Module Support Software: PyAudio

Additional Package Management System: pipwin

# LIBRARIES & THEIR PURPOSES

- Os Module→ Provides functions to interact

with operation system

- Time Module→ Used to create time frameworks inside the program
- Math Module→ Used to perform mathematical functions
- Keyboard Module→ Used to operate keyboard
- Requests Module→ Used to send http requests to URLs
- Webbrowser Module→ Used to open URLs
- gtts Module→ Used to convert text to MP3 file
- Speech_Recognition Module→ Used to Voice input to text
- Datetime Module→ Used to show date, day, time, month, year, etc…
- bs4→ It is an XML and HTML parser and used for web scraping
- Selenium→ Used for web browser automation
- Collections Module→ It is a data-structure module
- playsound Module→ used to play MP3 files

# DESCRIPTION

Sid is a Virtual Assistant and a Voice Assistant named Glenda. They could I can update you with ⩥Weather ⩥Stocks ⩥Trending Hashtags ⩥News Headlines ⩥Trending Subjects ⩥Latest News ⩥News Coverage ⩥World News ⩥Sports News ⩥Business News ⩥Tech news ⩥Local News. They could get you to Google, Wikipedia, Facebook, Netflix, Instagram, YouTube, Amazon & So On… i.e. they can Get you anything from Any WebSite and any Application in the Device for you. They can Play any Song of your Choice, Can Answer doubts or Questions you ask! Can tell the News on a Specific Subject/Topic you ask! Can tell about anyone/anything you ask. They can Extract Whole HTML file of the Website/URL You feed, They can thoroughly Scan the Folder you feed and List you the Files in it. They can play Live News. They do the Mathematical Logics and Conversions and computer base conversions too. They can tell you Jokes too. They can ShutDown / LogOut / Restart the Device with your

command. They have Fitness Tips and Sid has a Personal Sketch to say…..
And Truth is, They can Literally do Anything!

# CODING

```
########=======_Virtual
Sid_=======##########
import os
import time
import math
import random
import keyboard
import requests
import webbrowser
from gtts import gTTS
import speech_recognition
from datetime import datetime
from bs4 import BeautifulSoup
from selenium import webdriver
from collections import Counter
from playsound import playsound
v='YES'#input("Do you want VoiceOver:").capatalize()
def process():
    while True:
        try:
            os.remove("C:\Python Files\Audiofiles\VoiceOver.
mp3")
        except:
            pass
        q_=input().casefold()
        if q_=='':
            try:
                _r_=speech_recognition.Recognizer()
                with speech_recognition.Microphone() as h:
                    _r_.adjust_for_ambient_noise(h)
                    _r_.energy_threshold=5000
```

```
                playsound(r"C:\Python Files\Audiofiles\On.mp3"
)
                a=_r_.listen(h)
            playsound(r"C:\Python Files\Audiofiles\Off.mp3")
            d_=_r_.recognize_google(a).casefold()
            if (("hey" in d_ or "hi" in d_ or "hay" in d_ or "ok" in
d_ or  "hai" in d_ or “hello” in d_) and ("siri" in d_ or "google"
in d_ or "cortana" in d_ or "bixby" in d_ or "alexa" in d_ or
“jarvis” in d_)) and ("about" not in d_ and "who" not in d_
and "what" not in d_ and "how" not in d_ and "where" not in
d_ and "when" not in d_ and "in" not in d_):
                r=["For clearification: I'm Sid", "That's Awkward!",
 "That's Awful", "I did'nt Expect that!", "You're
dissappointing Me!", "You're here with different Person in a
different Platform", "I thought you Recognise me!"]
                r_=random.randint(0,6)
                t_=r[r_]
                (t_)
                r1=["That is Awkward!", "That is Awful", "I did'nt
Expect that!", "You're dissappointing Me!", "You're here
with different Person in a different Platform", "I thought you
Recognise me!"]
                r1_=random.randint(0,5)
                t=r1[r1_]
                if v == "YES":
                   s=gTTS(text="hmm,{}".format(t),lang='en-uk')
                   s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                   playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                   os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            else:
               print(" ◄▼►")
```

```
                if len(d_)>100:
                    pass
                else:
                    q_=d_
                    time.sleep(1)
                print(d_)
            except:
                q_=input("").casefold()
        else:
            q_=q_
        try:
            if q_ is True:
                pass
            elif q_ == "":
                pass
            elif "hi " in q_ or "hay " in q_ or "hey" in q_ or "hai" in
q_ or "hello" in q_:
                __r__=["Hey Hi","Hi!","Hello","Hola"]
                i=random.randint(0,len(__r__)-1)
                print(__r__[i])
                if v == "YES":
                    s=gTTS(text=__r__[i],lang='en-uk')
                    s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            elif "play" in q_ and "how" not in q_:
                try:
                    l= q_.split()
                    try:
                        i=l.index("about")+1
                    except:
```

```
            i=l.index("play")+1
        try:
            j=l.index("in")
            w=' '.join(l[i:j])
        except:
            try:
                j=l.index("on")
                w=' '.join(l[i:j])
            except:
                try:
                    j=l.index("from")
                    w=' '.join(l[i:j])
                except:
                    w=' '.join(l[i:])
        print("Just Sit Back And Hold Tight!")
        print("We Are Heading To Youtube...")
        chrome_options = webdriver.ChromeOptions()
        d= webdriver.
Chrome(executable_path='C:\Program Files (x86)
\Google\Chrome\Application\chromedriver.exe',options =
chrome_options)
        d.get("https://www.youtube.com/")
        time.sleep(1)
        keyboard.press_and_release('win + Up')
        time.sleep(1)
        keyboard.press_and_release('win + Up')
        time.sleep(1)
        keyboard.press_and_release('win + Right')
        time.sleep(1)
        s_ = d.find_element_by_name('search_query')
        s_.send_keys(w)
        s_.submit()
        c= d.find_element_by_xpath('//*[@id="video-title"
]/yt-formatted-string')
```

```
                c.click()
                print("And.....Now Playing {} on Youtube".
format(w))
                if v == "YES":
                    s=gTTS(text="hmm! now playing, {} in Youtube"
.format(w),lang='en-uk')
                    s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except:
                try:
                    try:
                        l= q_.split()
                        try:
                            l.remove("sid")
                            w=' '.join(l)
                        except:
                            l.remove("glenda")
                            w=' '.join(l)
                    except:
                        w=q_
                    print("Just Sit Back And Hold Tight!")
                    webbrowser.open("https://google.com/
search?q={}".format(w))
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
                    keyboard.press_and_release('win + Right')
                    time.sleep(1)
```

```
                try:
                    if v == "YES":
                        s=gTTS(text="hmm These are results for {}
in internet".format(w),lang='en-uk')
                        s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                except:
                    pass
            except requests.exceptions.ConnectionError:
                print(" \t⚠Oops...Make Sure That You Are
Connected With Internet⚠")
                if v == "YES":
                    playsound(r"C:\Python
Files\Audiofiles\offline.mp3")
        elif ("search" or "look" in q_ or "find" in q_) and "wiki"
in q_ or "from wiki" in q_ or "in wiki" in q_ :
            try:
                l= q_.split()
                try:
                    i=l.index("some")+1
                except:
                    try:
                        i=l.index("for")+1
                    except:
                        try:
                            i=l.index("look")+1
                        except:
                            try:
                                i=l.index("search")+1
                            except:
```

```
                    try:
                        i=l.index("out")+1
                    except:
                        i=l.index("find")+1
            try:
                j=l.index("in")
                w=' '.join(l[i:j])
            except:
                try:
                    j=l.index("on")
                    w=' '.join(l[i:j])
                except:
                    try:
                        j=l.index("from")
                        w=' '.join(l[i:j])
                    except:
                        w=' '.join(l[i:])
            if w == "":
                w=' '.join(l[i:])
            print("Just Sit Back And Hold Tight!")
            print("We Are Heading To Wikipedia...")
            chrome_options = webdriver.ChromeOptions()
            d= webdriver.
Chrome(executable_path='C:\Program Files (x86)
\Google\Chrome\Application\chromedriver.exe',options =
chrome_options)
            d.get("https://www.wikipedia.org/")
            s_ = d.find_element_by_id('searchInput')
            s_.send_keys(w)
            s_.submit()
            if v == "YES":
                s=gTTS(text="hmm! these are the result found
about {} on Wikipedia".format(w),lang='en-uk')
                s.save(r"C:\Python Files\Audiofiles\VoiceOver.
```

```
mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except:
                print(" \t⚠Oops...Make Sure That You Are
Connected With Internet⚠")
                if v == "YES":
                    playsound(r"C:\Python Files\Audiofiles\offline.
mp3")
        elif "Stocks" in q_ or "share market" in q_:
            try:
                print("Just Sit Back And Hold Tight!")
                webbrowser.open("https://www.moneycontrol.
com/stocksmarketsindia/")
                if v == "YES":
                    s=gTTS(text="hmm these are live stock
updates",lang='en-uk')
                    s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except requests.exceptions.ConnectionError:
                print(" \t⚠Oops...Make Sure That You Are
Connected With Internet⚠")
                if v == "YES":
                    playsound(r"C:\Python Files\Audiofiles\offline.
mp3")
        elif "trending" in q_ and ("hash" in q_ or "#" in q_):
            if "world" in q_:
                r=requests.get("https://twitter-trends.iamrohit.
```

```
in/")
                s = BeautifulSoup(r.content,"html.parser")
                print("\t                Trending Hashtags")
                n=s.find(class_='panel-body')
                x=[]
                for __l__ in n.find_all('a'):
                   print("◉",__l__.text)
                   x.append(__l__.text)
                for t in x:
                   if len(t)>1:
                      if v == "YES":
                         s=gTTS(text="{}".format(t),lang='en-uk')
                         s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                         playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                         os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                print("That's It For NOW!")
             else:
                r=requests.get("https://twitter-trends.iamrohit.
in/india")
                s = BeautifulSoup(r.content,"html.parser")
                print("\t                Trending Hashtags")
                n=s.find(class_='panel-body')
                x=[]
                for __l__ in n.find_all('a'):
                   print("◉",__l__.text)
                   x.append(__l__.text)
                for t in x:
                   if len(t)>1:
                      if v == "YES":
                         s=gTTS(text="{}".format(t),lang='en-uk')
                         s.save(r"C:\Python
```

```
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                print("That's It For NOW!")
            elif "latest news" in q_ or "present news" in q_:
                try:
                    l=q_.split()
                    i=l.index("top")+1
                    j=l.index("latest")
                    u=' '.join(l[i:j])
                except:
                    u=""
                    pass
                r=requests.get("https://news.google.com/topics/
CAAqJggKIiBDQkFTRWdvSUwyMHZNRFZxYUdjU0FtVnVHZ
0pKVGlnQVAB?hl=en-IN&gl=IN&ceid=IN%3Aen")
                s = BeautifulSoup(r.text,"html.parser")
                if u == "" or not u.isdigit():
                    u=15
                    print("\t                Latest NEWS Headlines")
                else:
                    print("\t                Top {} Latest NEWS
Headlines".format(u))
                e = int(u)*3 + 50
                if "all" in q_ or "full" in q_:
                    e=""
                    u=""
                x=[]
                for __l__ in s.find_all('a')[52:e]:
                    print("◉",__l__.text)
                    x.append(__l__.text)
                print("That's It For NOW!")
```

```
            for t in x:
                if len(t)>42:
                    if v == "YES":
                        s=gTTS(text="{}".format(t),lang='en-uk')
                        s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            if v == "YES":
                s=gTTS(text="hmm Here we have the top {}
Latest news headlines".format(u),lang='en-uk')
                s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif "sport" in q_ and "news" in q_:
            try:
                l=q_.split()
                i=l.index("top")+1
                j=l.index("sports")
                u=' '.join(l[i:j])
            except:
                u=""
                pass
            r=requests.get("https://news.google.com/topics/
CAAqJggKIiBDQkFTRWdvSUwyMHZNRFp1ZEdvU0FtVnVHZ
0pKVGlnQVAB?hl=en-IN&gl=IN&ceid=IN%3Aen")
            s = BeautifulSoup(r.text,"html.parser")
            if u == "" or not u.isdigit:
                u=15
```

```
            print("\t                    Sports NEWS")
        else:
            print("\t                Top {} Sports NEWS".
format(u))
        e = int(u)*3 + 2
        if "all" in q_ or "full" in q_:
            e=677
            u=""
        n=s.find(id='yDmH0d')
        i=n.find(class_='fe4pJf')
        x=[]
        for __l__ in i.find_all('a')[2:e]:
            print("◉",__l__.text,"\n")
            x.append(__l__.text)
        print("That's It For NOW!")
        for t in x:
            if len(t)>42:
                if v == "YES":
                    s=gTTS(text="{}".format(t),lang='en-uk')
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        if v == "YES":
            s=gTTS(text="hmm Here we have the Top {}
Sports News".format(u),lang='en-uk')
            s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
            playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
            os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
```

```
        elif "tech news" in q_ or "technology news" in q_:
            try:
                l=q_.split()
                i=l.index("top")+1
                try:
                    j=l.index("technology")
                except:
                    j=l.index("tech")
                u=' '.join(l[i:j])
            except:
                u=""
                pass
            r=requests.get("https://news.google.com/topics/
CAAqJggKIiBDQkFTRWdvSUwyMHZNRGRqTVhZU0FtVnVHZ
0pKVGlnQVAB?hl=en-IN&gl=IN&ceid=IN%3Aen")
            s = BeautifulSoup(r.text,"html.parser")
            if u == "" or not u.isdigit:
                u=15
                print("\t                    Tech NEWS")
            else:
                print("\t                Top {} Tech NEWS".format(u)
)
            e = int(u)*3 + 2
            if "all" in q_ or "full" in q_:
                e=""
                u=""
            n=s.find(id='yDmH0d')
            i=n.find(class_='MNK4Vd')
            x=[]
            for __l__ in i.find_all('a')[2:e]:
                print("◉",__l__.text,"\n")
                x.append(__l__.text)
            print("That's It For NOW!")
            for t in x:
```

```
                if len(t)>42:
                    if v == "YES":
                        s=gTTS(text="{}".format(t),lang='en-uk')
                        s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            if v == "YES":
                s=gTTS(text="hmm Here we have the Top {} Tech
News".format(u),lang='en-uk')
                s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif "world news" in q_ or "global news" in q_:
            try:
                l=q_.split()
                i=l.index("top")+1
                try:
                    j=l.index("global")
                except:
                    j=l.index("world")
                u=' '.join(l[i:j])
            except:
                u=""
                pass

            if u == "" or not u.isdigit:
                u=15
                print("\t                    World NEWS")
```

```
            else:
                print("\t                Top {} World NEWS".
format(u))
            e = int(u)*3
            if "all" in q_ or "full" in q_:
                e=""
                u=""
            n=s.find(id='yDmH0d')
            i=n.find(class_='MNK4Vd')
            x=[]
            for __l__ in i.find_all('a')[:e]:
                print("◉",__l__.text,"\n")
                x.append(__l__.text)
            print("That's It For NOW!")
            for t in x:
                if len(t)>42:
                    if v == "YES":
                        s=gTTS(text="{}".format(t),lang='en-uk')
                        s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            print("That's It For NOW!")
            if v == "YES":
                s=gTTS(text="hmm Here we have the top {}
Global News".format(u),lang='en-uk')
                s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
```

```
        elif "business news" in q_ or "trade news" in q_ or "
economy news" in q_:
            try:
                l=q_.split()
                i=l.index("top")+1
                try:
                    j=l.index("business")
                except:
                    try:
                        j=l.index("trade")
                    except:
                        j=l.index("economy")
                u=' '.join(l[i:j])
            except:
                u=""
                pass
            r=requests.get("https://news.google.com/topics/
CAAqJggKIiBDQkFTRWdvSUwyMHZNRGx6TVdZU0FtVnVHZ
0pKVGlnQVAB?hl=en-IN&gl=IN&ceid=IN%3Aen")
            s = BeautifulSoup(r.text,"html.parser")
            if u == "" or not u.isdigit:
                u=15
                print("\t                    Business NEWS")
            else:
                print("\t                Top {} Business NEWS".
format(u))
            if "all" in q_ or "full" in q_:
                e=""
                u=""
            e = int(u)*3 + 2
            n=s.find(id='yDmH0d')
            i=n.find(class_='MNK4Vd')
            x=[]
            for __l__ in i.find_all('a')[2:e]:
```

```
                print("◉",__l__.text,"\n")
                x.append(__l__.text)
            print("That's It For NOW!")
            for t in x:
                if len(t)>42:
                    if v == "YES":
                        s=gTTS(text="{}".format(t),lang='en-uk')
                        s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            if v == "YES":
                s=gTTS(text="hmm Here we have the top {}
Economy News".format(u),lang='en-uk')
                s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif "health news" in q_:
            try:
                l=q_.split()
                i=l.index("top")+1
                j=l.index("health")
                u=' '.join(l[i:j])
            except:
                u=""
                pass
            r=requests.get("https://news.google.com/topics/
CAAqIQgKIhtDQkFTRGdvSUwyMHZNR3QwTlRFU0FtVnVLQ
UFQAQ?hl=en-IN&gl=IN&ceid=IN%3Aen")
```

```
            s = BeautifulSoup(r.text,"html.parser")
            if u == "" or not u.isdigit:
               u=15
               print("\t                    Health NEWS")
            else:
               print("\t                Top {} Health NEWS".
format(u))
            e = int(u)*3+2
            if "all" in q_ or "full" in q_:
               e=""
               u=""
            n=s.find(id='yDmH0d')
            i=n.find(class_='MNK4Vd')
            x=[]
            for __l__ in i.find_all('a')[2:e]:
               print("◉",__l__.text,"\n")
               x.append(__l__.text)
            print("That's It For NOW!")
            for t in x:
               if len(t)>42:
                  if v == "YES":
                     s=gTTS(text="{}".format(t),lang='en-uk')
                     s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                     playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                     os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            if v == "YES":
               s=gTTS(text="hmm Here we have the Top {}
Health News".format(u),lang='en-uk')
               s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
               playsound(r"C:\Python
```

```
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif "local news" in q_:
            try:
                l=q_.split()
                i=l.index("top")+1
                j=l.index("local")
                u=' '.join(l[i:j])
            except:
                u=""
                pass
            r=requests.get("https://news.google.com/topics/
CAAqHAgKIhZDQklTQ2pvSWJHOWpZV3hmZGpJb0FBUAE?h
l=en-IN&gl=IN&ceid=IN%3Aen")
            s = BeautifulSoup(r.text,"html.parser")
            if u == "" or not u.isdigit:
                u=15
                print("\t                    Local NEWS")
            else:
                print("\t                Top {} Local NEWS".format(u)
)
            e = int(u)*3+2
            if "all" in q_ or "full" in q_:
                e=""
                u=""
            n=s.find(id='yDmH0d')
            i=n.find(class_='MNK4Vd')
            x=[]
            for __l__ in i.find_all('a')[2:e]:
                print("◉",__l__.text,"\n")
                x.append(__l__.text)
            print("That's It For NOW!")
            for t in x:
```

```
                if len(t)>42:
                    if v == "YES":
                        s=gTTS(text="{}".format(t),lang='en-uk')
                        s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            if v == "YES":
                s=gTTS(text="hmm Here we have the Top {}
Local News".format(u),lang='en-uk')
                s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif "news coverage" in q_ or "news stories" in q_:
            try:
                r=requests.get("https://news.google.com/
topstories?hl=en-IN&gl=IN&ceid=IN:en")
                s = BeautifulSoup(r.text,"html.parser")
                print("\t                    NEWS Coverage")
                n=s.find(id='yDmH0d')
                i=n.find(class_='lBwEZb BL5WZb xP6mwf')
                for __l__ in i.extract():
                    g=''
                    g_=[]
                    _g_=[]
                    if len(__l__.text)>100:
                        for h in __l__.text:
                            if h.isupper():
                                g=g+' '+h
```

```
                        else:
                            g=g+h
                    for k in g.split():
                        if 'ampvideo_youtube' in k or
'bookmark_bordersharemore_vert' in k or
k=='coveragekeyboard_arrow_up':
                            pass
                        else:
                            _g_.append(k)
                    print("◉",' '.join(_g_),"\n")
            print("That's It For NOW!")
            if v == "YES":
                s=gTTS(text="hmm Here we have the Top
coverage stories",lang='en-uk')
                s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        except requests.exceptions.ConnectionError:
            print(" \t⚠Oops...Make Sure That You Are
Connected With Internet⚠")
            if v == "YES":
                playsound(r"C:\Python Files\Audiofiles\offline.
mp3")
    elif ("news" in q_ or "update" in q_) and ("trend" in q_
or "about" in q_ or "on" in q_ or "for" in q_ or "in" in q_):
        l=q_.split()
        try:
            i=l.index("about")+1
        except:
            try:
                i=l.index("on")+1
```

```
            except:
                try:
                    i=l.index("for")+1
                except:
                    g="yo"
            print("I'm on it...")
            if v == "YES":
                s_=gTTS(text="just wait... i'm looking for it",
lang='en-uk')
                s_.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            try:
                if g=="yo":
                    w=q_
            except:
                w=' '.join(l[i:])
            r=requests.get("https://news.google.com/
search?q={}&hl=en-US&gl=US&ceid=US%3Aen".format(w))
            s=BeautifulSoup(r.content,'html.parser')
            if "all" in q_ or "full" in q_:
                e=""
            else:
                e=15
            for _l_ in s.findAll('a')[53:e]:
                if len(_l_.text)>35:
                    if "play_arrow" in _l_.text or "Getty" in _l_.text:
                        pass
                    else:
                        print("\n ◉",_l_.text)
                        if v == "YES":
```

```
                    s_=gTTS(text="{}".format(_l_.text),lang='en-
uk')
                    s_.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                else:
                    pass
            print("Here's the Update")
            if v == "YES":
                s_=gTTS(text="here's the update",lang='en-uk')
                s_.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif "headlines" in q_ or "news" in q_:
            try:
                l=q_.split()
                i=l.index("top")+1
                try:
                    j=l.index("news")
                except:
                    j=l.index("headlines")
                u=' '.join(l[i:j])
            except:
                u=""
                pass
            r=requests.get("https://news.google.com/
topstories?hl=en-IN&gl=IN&ceid=IN:en")
            s = BeautifulSoup(r.text,"html.parser")
```

```
            if u == "" or not u.isdigit:
                u=15
                print("\t                    NEWS Headlines")
            else:
                print("\t                Top {} NEWS Headlines".
format(u))
            e = int(u)*3+4
            if "all" in q_ or "full" in q_:
                e=685
                u=""
            n=s.find(id='yDmH0d')
            i=n.find(class_='lBwEZb BL5WZb xP6mwf')
            x=[]
            for __l__ in i.find_all('a')[2:e]:
                print("◉",__l__.text,"\n")
                x.append(__l__.text)
            print("That's It For NOW!")
            for t in x:
                if len(t)>42:
                    if v == "YES":
                        s=gTTS(text="{}".format(t),lang='en-uk')
                        s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            if v == "YES":
                s=gTTS(text="hmm Here we have the Top {}
news headlines".format(u),lang='en-uk')
                s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
```

```
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            elif "how" in q_ and ("life" in q_ or "are you" in q_):
                __r__=["Just Smooth","Great","Cool","Not Bad","Fine"
]
                i=random.randint(0,len(__r__)-1)
                print(__r__[i])
                if v == "YES":
                    s=gTTS(text=__r__[i],lang='en-uk')
                    s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            elif "what can you do" in q_ or "what are you meant"
in q_:
                l=["⚜I can update you with ⩥Weather ⩥Stocks
⩥Trending Hashtags ⩥News Headlines ⩥Trending
Subjects ⩥Latest News ⩥News Coverage ⩥World News
⩥Sports News ⩥Business News ⩥Tech news ⩥Local News
\n","⚜I could get you to Google, Wikipedia, Facebook, Netflix,
 Instagram, YouTube, Amazon & So On... \n","⚜I can Get you
anything from Any WebSite and any Application in the
Device for you \n","⚜I could Play any Song of your Choice
\n","⚜I Can Answer doubts or Questions you ask! \n","⚜I
Could tell the News on a Specific Subject/Topic you ask! \n",
"⚜I can Extract Whole HTML file of the Website/Url You
feed me \n","⚜I can thoroughly Scan the Folder you feed
and List you the Files in it \n","⚜I could do all the
Mathematical Logics and Conversions \n","⚜I have a Voice
Assistant She's Glenda \n","⚜I can tell you Jokes too \n","⚜I
can ShutDown / LogOut / Restart the Device \n","⚜I have
Fitness Tips to say \n","⚜I do have a Personal Sketch about
```

```
Myself \n","⚜Truth is,I can Literally do Anything! \n"]
            for i in l:
                for j in i:
                    print(j, end="")
            l_="we can update you with Weather, Stocks,
Trending Hashtags, News Headlines, Trending Subjects,
Latest News, News Coverage, World News, Sports News,
Business News, Tech news, Local News. we could get you
Google, Wikipedia, Facebook, Instagram, Netflix, YouTube,
Amazon & So On...we can Get you anything from Any
WebSite and any application in the device For you. we could
Play any Song of your Choice. I Can Answer doubts or
Questions you ask! we Could tell the News on a Specific
Subject/Topic you ask! we can Extract Whole HTML file of
the Website/Url You ask me. we can thoroughly Scan the
Folder you feed and List you the Files in it. enabled with a
Voice Assistant and thats me Glenda. we can tell you Jokes
too. we can ShutDown / LogOut / Restart the Device. we
have Fitness Tips to say too. sid do have a Personal Sketch
about himself. Truth is,we can Literally do Anything! for
you"
            try:
                if v == "YES":
                    s=gTTS(text="{}".format(l_),lang='en-uk')
                    s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    s=gTTS(text="and Myself And Sid are here to
Assist you! ",lang='en-uk')
                    s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
```

```
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except requests.exceptions.ConnectionError:
                pass
        elif ("you" in q_ and "work" in q_ ) and "for" in q_:
            __r__=["Haha, Sounds Great","LOL","I don't think so",
"Do you think, I would?","Haha, Satire","Why should I?","
Maybe..."]
            i=random.randint(0,len(__r__)-1)
            print(__r__[i])
            if v == "YES":
                s=gTTS(text=__r__[i],lang='en-uk')
                s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif "you" in q_ and "do" in q_ and "have" in q_:
            __r__=["I wish I could","Maybe","Probabaly","No, not
at all","Haha...","Nope","Yep"]
            i=random.randint(0,len(__r__)-1)
            print(__r__[i])
            if v == "YES":
                s=gTTS(text=__r__[i],lang='en-uk')
                s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif "you" in q_ and "do" in q_ and "me" in q_:
```

```
            __r__=["I wish I could","Maybe","LOL","No, not at all",
"I don't Recognize you","Nope","Yep"]
            i=random.randint((0,len(__r__)-1))
            print(__r__[i])
            if v == "YES":
               s=gTTS(text=__r__[i],lang='en-uk')
               s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
               playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
               os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
         elif "who" in q_ and "is" in q_ and "your" in q_:
            __r__=["'I Don't Know!'","I couldn't say that","That's
Personal","you know, Iam Virtual","Hahaha... Nice Joke","idk"
]
            i=random.randint(0,len(__r__)-1)
            print(__r__[i])
            if v == "YES":
               s=gTTS(text=__r__[i],lang='en-uk')
               s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
               playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
               os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
         elif "can" in q_ and "you" in q_:
            __r__=["I wish I could!","Nope","Yeah Definitely","
you know, Iam Virtual","Hahaha... Nice Joke","That's
Impossible"]
            i=random.randint(0,len(__r__)-1)
            print(__r__[i])
            if v == "YES":
               s=gTTS(text=__r__[i],lang='en-uk')
```

```
            s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
            playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
            os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif "will" in q_ and "you" in q_:
            __r__=["Never","Nah","Definitely yes","you know,
Iam Virtual","Hahaha... Nice Joke","Maybe..","My Pleasure"]
            i=random.randint(0,len(__r__)-1)
            print(__r__[i])
            if v == "YES":
                s=gTTS(text=__r__[i],lang='en-uk')
                s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif "have" in q_ and "you" in q_:
            __r__=["Never","Nah","Definitely yes","I think, I
didn't","you know, it's yes for me!","Maybe...","That's
Personal"]
            i=random.randint(0,len(__r__)-1)
            print(__r__[i])
            if v == "YES":
                s=gTTS(text=__r__[i],lang='en-uk')
                s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif "are" in q_ and "you" in q_:
```

```
            __r__=["Why do you bother?", "Just Virtual","Should
i answer that?!","That's personal"]
            i=random.randint(0,len(__r__)-1)
            print(__r__[i])
            if v == "YES":
               s=gTTS(text=__r__[i],lang='en-uk')
               s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
               playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
               os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
         elif "i" in q_ and "love you" in q_:
            __r__=["I didn't expect that!", "There are different
types of love, but our's is Friendship!","Not available","Sorry,
 not interested","I love you Too","Aww me too"]
            i=random.randint(0,len(__r__)-1)
            print(__r__[i])
            if v == "YES":
               s=gTTS(text=__r__[i],lang='en-uk')
               s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
               playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
               os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
         elif "who is siddiq" in q_:
            print("He is the one Who Built me!")
            try:
               if v == "YES":
                  s=gTTS(text="hmm he He is the one who Built
Us!",lang='en-uk')
                  s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
```

```
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except requests.exceptions.ConnectionError:
                pass
        elif (("who" in q_ and "is" in q_) or ("who" in q_ and "
re" in q_)) and ("this" in q_ or " sid" in q_ or "glenda" in q_ or
("you" in q_ and "yout" not in q_) ):
            print("Iam Sid, A Virtual Assistant!")
            print("And Glenda is My Voice Assistant!")
            try:
                if v == "YES":
                    s=gTTS(text="hmm I'm Glenda, Sid's Voice
Assistant",lang='en-uk')
                    s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except requests.exceptions.ConnectionError:
                pass
        elif "who" in q_ and ("you" in q_ or " sid" in q_ or "
glenda" in q_):
            print("Siddiq, the one who Created & Developed
me!")
            try:
                if v == "YES":
                    s=gTTS(text="hmm he He is the one who
Created & Developed Us!",lang='en-uk')
                    s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                    playsound(r"C:\Python
```

```
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                except requests.exceptions.ConnectionError:
                    pass
            elif "jokes" in q_:
                j=["Question: What do you call a boomerang that
doesn't work? Answer: A stick.", 'Question: What do you call
four bullfighters in quicksand? Answer: Quattro sinko.', 'Ed:
The same bike tries to run me down every day. Fred: Sounds
like a vicious cycle...', 'Question: Why should you never date
tennis players? Answer: Love means nothing to them.',
'Question: How do you weigh a millennial? Answer: In
Instagrams.', 'Jenny: I can tell if someone is lying just by
looking at him. Penny: Really? Jenny: Yep. I can tell if he is
standing too.', 'Fred: Can you tell me about that new do-it-
yourself orthodontist? Ted: Brace yourself.', 'Question: What
happened to the guy who sued over his missing luggage?
Answer: He lost his case.', 'Therapist: I've concluded that
you are incapable of describing your feelings. Patient: I can't
say that I am surprised!', 'Marriage Counselor: So, what
brings you here today? Wife: He takes everything literally.  I
can't stand it. Husband: My truck.', 'Question: What weighs
more, a gallon of water or a gallon of butane? Answer: The
water. Butane is lighter fluid.', 'Mike: Someone stole the
wheels off of all the police cars! Spike: The cops are working
on it—tirelessly.', 'Question:  What do you call a guy who's
had too much to drink? Answer:  A cab.', 'Ann: I herd that
you are a hypochondriac. Stan: Well, my doctor says I'm not,
but I spent 3 days reading about it on the internet and I have
all...', 'Hal: How did you get hit on the head with a book? Sal:
I only have my shelf to blame.', 'Question: What kind of tree
has a hand? Answer: A palm tree.', 'Question: What kind of
shoes does a lazy person wear? Answer: Loafers.', 'Question:
```

Why should you save your pennies? Answer: It makes good cents.', 'Question: What kind of jokes are told on a farm? Answer: Corny ones.', 'Question: What has T in the beginning, T in the middle, and T at the end? Answer: A teapot.', 'Question: What do you call a penguin in the desert? Answer: Lost', 'Question: What is a tree's favorite soda? Answer: Root Beer.', 'A lawyer is driving a car down the street and instead of stopping at the stop sign, the lawyer slows down. A policeman sees this and pulls the car over...', " Question: Why doesn't McDonald's serve escargot? Answer: It's not fast food!", 'Question: Hear about the two guys who stole a calendar? Answer: They both got 6 months.', 'A guy gets pulled over by a cop. The cop asks, "You're speeding! Didn't you see the speed limit sign?" The man replied, "Yeah I saw the speed limit sign,...', 'The last year I entered a marathon. The race started and immediately I was the last of the runners. It was embarrassing. The guy who was in front of me,...', 'Question: What did the SNAIL say while riding on the turtles back? Answer: Wheeeeeeeee', 'I work in the front office of a housing complex that supports people living with mental illness. On one particularly hectic day, a tenant came in to pay her rent....', 'As the dentist labored over my teeth, he tried to make small talk. "What do you do?" he asked. "I'm a comedian," I answered. "Interesting." After a pause, he said,...', 'I admit it—I have a tendency to exaggerate, and I was afraid when I joined the Navy that my "creativity" might get me in trouble. But my fears were put...', 'Our base's Army Exchange Service carried a particular brand of underarm deodorant that I liked and bought for years. Then one day I couldn't find it. I asked an employee...', 'Fred Astaire and Ginger Rogers were dining in New York. Ginger was resplendent in a ball gown and pearls, and Fred also sported evening wear. But the meal was marred...', 'To the guy who stole my antidepressants: I hope you're happy

now.', 'Spotted in the legal notices section of the Maryland-based Daily Times: Michael Ray Dipirro petitioned the circuit court to change his name to Michael Ray Forbes. His reason for doing...', '“This is your great-grandma and great grandpa,” I told my grandson as I handed him a photo of my parents. “Do you think I look like them?” He shook his...', 'While shopping for a bathroom scale, I found one that tracks not only weight but also body fat, bone mass, and water percentage. I nixed that one in favor of...', 'The topic of conversation was nose jobs. My slightly confused young daughter asked, “Where does the doctor get the new noses to replace the old ones?” “They have a place...', 'In his late 80s, my father-in-law went to the DMV to renew his driver’s license. At one point during the road test, he approached a four-way stop, looked to his...', 'After my husband injured himself, I ran him over to the doctor’s office. There, the nurse dressed his wound and gave him instructions on how to care for it. She...', 'A man is at the funeral of an old friend. He tentatively approaches the deceased’s wife and asks whether he can say a word. The widow nods. The man clears...', 'I was instructing new recruits when an officer entered my classroom to observe and report on my teaching style. I thought I was on top of my game that day,...', 'Comedian Martha Raye was a great supporter of the military and made many trips to Vietnam to entertain the troops. She also liked her scotch. One day, I was told...', 'I was trapped in an elevator for 30 minutes before the doors finally opened. Relieved, I said to a fellow hostage, “There’s a first time for everything.” She grumbled back,...', 'After my wife accidentally swallowed my prostate medication, our daughter called a pharmacist to ask whether there was any cause for alarm. He replied, “Only if she starts hanging out...', 'My 35-year-old son and I had just finished our meal when I realized I’d left my wallet in my truck. As I headed out the

door, I told the waitress...', 'Starving after hours of driving nonstop, my husband and I pulled over at a truck stop. While he gassed up the car, I went into the restaurant and placed our...', 'As part of my Naval Reserve requirements at Emory University Dental School, I attended a talk about proper dental procedures following nuclear warfare. Evidently, one of my classmates found the...', 'When I was a Navy student pilot, I visited the home of a classmate. I met his wife and baby and was impressed that he had all his flight gear...', 'While taking a clinical history from an elderly patient, I asked, “How’s your love life?” “I don’t know,” he said. “I’ll ask my wife.” He got up, walked into the...', 'A coworker was telling us all about her trip to Las Vegas. “That sounds great. Where’d you stay?” asked a colleague. “I can’t remember,” she said. “But I think it...', 'Sometimes honesty isn’t the best policy.A patient showed up at our medical office and asked, “You’re Mary, aren’t you?” I smiled. “No, sorry, I’m not.” “Are you sure? You look...', 'I just read that 4,153,237 people got married last year. Not to cause any trouble, but shouldn’t that be an even number?', 'My husband cooks for me like I’m a god—by placing burnt offerings before me every night.', 'Question: What happens when an artist has trouble finding inspiration? Answer: She draws a blank.', 'Something tells me I need to lose some weight. During a recent trip to visit my son and his family, I stopped off at a bakery to pick up dessert....', 'Over dinner, I could sense something was bothering my mother, so I asked if anything was wrong. “Yes,” she admitted. “What’s all this I hear on the news about banning...', 'Descartes walked into a bar and ordered a beer. “Want another?” asked the bartender. “I think not”, Descartes replied … then he disappeared.', 'Did you hear about the young actor who fell through the floorboards? He was just going through a stage.', 'Question: What did the left eye say to the right eye?

Answer: Between you and me, something smells.', 'Question: How does the solar system organize a party? Answer: They planet!', 'I went to a smoke shop to discover that it has been replaced by an apparel store. Clothes, but no cigar.', 'Question: What is the best way to cook a gator? Answer: In a crock-pot', "Question: What did the numerator say to the denominator when they broke up? Answer: I'm so over you!" , "I'm sick of following my dreams. I'm just going to ask where they're going and hook up with 'em later.", "I've reached the age where my prescription bill has caught up to my bar bill.", 'Did you hear how they caught the great produce bandit? He stopped to take a leek.', 'Question: What do you get when you combine an insomniac, an agnostic, and a dyslexic? Answer: Someone who lays awake at night wondering the true meaning of Dog.', 'I was working from home, interviewing a famous neurologist for an article, when my three-year-old announced she had to go potty and waddled into the bathroom. After some loud moans,...', 'My job as a facilities maintenance engineer required a wide range of skills. One day I might have to fix the furnace, while the next day could see me painting...', 'Our manager kept reminding us waitresses to encourage customers to order dessert. At the end of an especially exhausting day, I walked over to a couple who had just sat...', 'A man goes to a job interview and the interviewer begins with the question, “What do you think is your biggest weakness?” The man thinks for a moment, then says,...', 'A man goes to the doctor, concerned about his wife’s hearing. The doctor says, “Stand behind her and say something and tell me how close you are when she hears...']

```
j_=random.randint(0,73)
_j_=random.randint(0,73)
j__=random.randint(0,73)
print(j[j_])
```

```
            print(j[_j_])
            print(j[j__])
            if v == "YES":
                s=gTTS(text="hmm,{}".format(j[j_]+' '+j[_j_]+'
'+j[j__]),lang='en-uk')
                s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            print("I Guess That puts a Smile on your Face (ツ)")
        elif "joke" in q_:
            j=["Question: What do you call a boomerang that
doesn't work? Answer: A stick.", 'Question: What do you call
four bullfighters in quicksand? Answer: Quattro sinko.', 'Ed:
The same bike tries to run me down every day. Fred: Sounds
like a vicious cycle...', 'Question: Why should you never date
tennis players? Answer: Love means nothing to them.',
'Question: How do you weigh a millennial? Answer: In
Instagrams.', 'Jenny: I can tell if someone is lying just by
looking at him. Penny: Really? Jenny: Yep. I can tell if he is
standing too.', 'Fred: Can you tell me about that new do-it-
yourself orthodontist? Ted: Brace yourself.', 'Question: What
happened to the guy who sued over his missing luggage?
Answer: He lost his case.', 'Therapist: I've concluded that
you are incapable of describing your feelings. Patient: I can't
say that I am surprised!', 'Marriage Counselor: So, what
brings you here today? Wife: He takes everything literally.  I
can't stand it. Husband: My truck.', 'Question: What weighs
more, a gallon of water or a gallon of butane? Answer: The
water. Butane is lighter fluid.', 'Mike: Someone stole the
wheels off of all the police cars! Spike: The cops are working
on it—tirelessly.', 'Question:  What do you call a guy who's
```

had too much to drink? Answer:  A cab.', 'Ann: I herd that you are a hypochondriac. Stan: Well, my doctor says I’m not, but I spent 3 days reading about it on the internet and I have all...', 'Hal: How did you get hit on the head with a book? Sal: I only have my shelf to blame.', 'Question: What kind of tree has a hand? Answer: A palm tree.', 'Question: What kind of shoes does a lazy person wear? Answer: Loafers.', 'Question: Why should you save your pennies? Answer: It makes good cents.', 'Question: What kind of jokes are told on a farm? Answer: Corny ones.', 'Question: What has T in the beginning, T in the middle, and T at the end? Answer: A teapot.', 'Question: What do you call a penguin in the desert? Answer: Lost', 'Question: What is a tree’s favorite soda? Answer: Root Beer.', 'A lawyer is driving a car down the street and instead of stopping at the stop sign, the lawyer slows down. A policeman sees this and pulls the car over...', " Question: Why doesn't McDonald's serve escargot? Answer: It's not fast food!", 'Question: Hear about the two guys who stole a calendar? Answer: They both got 6 months.', 'A guy gets pulled over by a cop. The cop asks, “You’re speeding! Didn’t you see the speed limit sign?” The man replied, “Yeah I saw the speed limit sign,...', 'The last year I entered a marathon. The race started and immediately I was the last of the runners. It was embarrassing. The guy who was in front of me,...', 'Question: What did the SNAIL say while riding on the turtles back? Answer: Wheeeeeeeee', 'I work in the front office of a housing complex that supports people living with mental illness. On one particularly hectic day, a tenant came in to pay her rent....', 'As the dentist labored over my teeth, he tried to make small talk. “What do you do?” he asked. “I’m a comedian,” I answered. “Interesting.” After a pause, he said,...', 'I admit it—I have a tendency to exaggerate, and I was afraid when I joined the Navy that my “creativity” might get me in trouble. But my fears were put...',

'Our base's Army Exchange Service carried a particular brand of underarm deodorant that I liked and bought for years. Then one day I couldn't find it. I asked an employee...', 'Fred Astaire and Ginger Rogers were dining in New York. Ginger was resplendent in a ball gown and pearls, and Fred also sported evening wear. But the meal was marred...', 'To the guy who stole my antidepressants: I hope you're happy now.', 'Spotted in the legal notices section of the Maryland-based Daily Times: Michael Ray Dipirro petitioned the circuit court to change his name to Michael Ray Forbes. His reason for doing...', '"This is your great-grandma and great grandpa," I told my grandson as I handed him a photo of my parents. "Do you think I look like them?" He shook his...', 'While shopping for a bathroom scale, I found one that tracks not only weight but also body fat, bone mass, and water percentage. I nixed that one in favor of...', 'The topic of conversation was nose jobs. My slightly confused young daughter asked, "Where does the doctor get the new noses to replace the old ones?" "They have a place...', 'In his late 80s, my father-in-law went to the DMV to renew his driver's license. At one point during the road test, he approached a four-way stop, looked to his...', 'After my husband injured himself, I ran him over to the doctor's office. There, the nurse dressed his wound and gave him instructions on how to care for it. She...', 'A man is at the funeral of an old friend. He tentatively approaches the deceased's wife and asks whether he can say a word. The widow nods. The man clears...', 'I was instructing new recruits when an officer entered my classroom to observe and report on my teaching style. I thought I was on top of my game that day,...', 'Comedian Martha Raye was a great supporter of the military and made many trips to Vietnam to entertain the troops. She also liked her scotch. One day, I was told...', 'I was trapped in an elevator for 30 minutes before the doors

finally opened. Relieved, I said to a fellow hostage, “There’s a first time for everything.” She grumbled back,...', 'After my wife accidentally swallowed my prostate medication, our daughter called a pharmacist to ask whether there was any cause for alarm. He replied, “Only if she starts hanging out...', 'My 35-year-old son and I had just finished our meal when I realized I’d left my wallet in my truck. As I headed out the door, I told the waitress...', 'Starving after hours of driving nonstop, my husband and I pulled over at a truck stop. While he gassed up the car, I went into the restaurant and placed our...', 'As part of my Naval Reserve requirements at Emory University Dental School, I attended a talk about proper dental procedures following nuclear warfare. Evidently, one of my classmates found the...', 'When I was a Navy student pilot, I visited the home of a classmate. I met his wife and baby and was impressed that he had all his flight gear...', 'While taking a clinical history from an elderly patient, I asked, “How’s your love life?” “I don’t know,” he said. “I’ll ask my wife.” He got up, walked into the...', 'A coworker was telling us all about her trip to Las Vegas. “That sounds great. Where’d you stay?” asked a colleague. “I can’t remember,” she said. “But I think it...', 'Sometimes honesty isn’t the best policy.A patient showed up at our medical office and asked, “You’re Mary, aren’t you?” I smiled. “No, sorry, I’m not.” “Are you sure? You look...', 'I just read that 4,153,237 people got married last year. Not to cause any trouble, but shouldn’t that be an even number?', 'My husband cooks for me like I’m a god—by placing burnt offerings before me every night.', 'Question: What happens when an artist has trouble finding inspiration? Answer: She draws a blank.', 'Something tells me I need to lose some weight. During a recent trip to visit my son and his family, I stopped off at a bakery to pick up dessert....', 'Over dinner, I could sense something was bothering my mother, so I asked

if anything was wrong. “Yes,” she admitted. “What’s all this I hear on the news about banning...', 'Descartes walked into a bar and ordered a beer. “Want another?” asked the bartender. “I think not”, Descartes replied … then he disappeared.', 'Did you hear about the young actor who fell through the floorboards? He was just going through a stage.', 'Question: What did the left eye say to the right eye? Answer: Between you and me, something smells.', 'Question: How does the solar system organize a party? Answer: They planet!', 'I went to a smoke shop to discover that it has been replaced by an apparel store. Clothes, but no cigar.', 'Question: What is the best way to cook a gator? Answer: In a crock-pot', "Question: What did the numerator say to the denominator when they broke up? Answer: I'm so over you!" , "I'm sick of following my dreams. I'm just going to ask where they're going and hook up with 'em later.", "I've reached the age where my prescription bill has caught up to my bar bill.", 'Did you hear how they caught the great produce bandit? He stopped to take a leek.', 'Question: What do you get when you combine an insomniac, an agnostic, and a dyslexic? Answer: Someone who lays awake at night wondering the true meaning of Dog.', 'I was working from home, interviewing a famous neurologist for an article, when my three-year-old announced she had to go potty and waddled into the bathroom. After some loud moans,...', 'My job as a facilities maintenance engineer required a wide range of skills. One day I might have to fix the furnace, while the next day could see me painting...', 'Our manager kept reminding us waitresses to encourage customers to order dessert. At the end of an especially exhausting day, I walked over to a couple who had just sat...', 'A man goes to a job interview and the interviewer begins with the question, “What do you think is your biggest weakness?” The man thinks for a moment, then says,...', 'A man goes to the doctor,

```
concerned about his wife’s hearing. The doctor says, “Stand behind her and say something and tell me how close you are when she hears...']
            j_=random.randint(0,73)
            print(j[j_])
            if v == "YES":
                t=j[j_]
                if v == "YES":
                   s=gTTS(text="hmm,{}".format(t),lang='en-uk')
                   s.save(r"C:\Python Files\Audiofiles\VoiceOver.mp3")
                   playsound(r"C:\Python Files\Audiofiles\VoiceOver.mp3")
                   os.remove("C:\Python Files\Audiofiles\VoiceOver.mp3")
            print("I Guess That puts a Smile on your Face (ツ)")
        elif ("who are you" in q_ or "what are you" in q_ or " who is sid" in q_ or "describe you" in q_ or "about you" in q_ or "personal sketch" in q_ or "about sid" in q_ or (" description" in q_ and "you" in q_)) and "youtube" not in q_:
            s=" I'm Sid , A Virtual Assistant \n I was Built & Developed by Siddiq_Moideen \n I Built with a Female Voice Assistant She's Glenda \n I Acquire data via Internet, you know it's just 'Web Scraping' \n I'm also Built-in with few offline features as fed to me \n I was built during the Quarantined Days \n My File was first Commenced on Wednesday March 25 2020, 10:29:24 PM and Still being Developed! \n In the beginning in was in the Journey to be Developed as an Artifical Intelligence but eventually Iam been Destined to Virtual Assistant \n So now I'm here to Assist You!"
            for i in s:
               for j in i:
                  print(j, end="")
```

```
            s_=" Sid , A Virtual Assistant. he was Build &
Developed by Siddiq_Moideen. he is Built with a Female
Voice Assistant called Glenda and thats me! we Acquire data
via Internet, you know it's just 'Web Scraping'. he is also
Built-in with few offline features as fed to us. we were built
during the Quarantine Days. our File was first Commenced
on Wednesday March 25 2020, 10:29:24 PM and Still being
Developed! In the beginning he was expected to be build as
Artifical Intelligence but eventually Destined to a Virtual
Assistant. So now he's here to Assist You!"
            try:
               if v == "YES":
                  s=gTTS(text="{}".format(s_),lang='en-uk')
                  s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                  playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                  os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                  s=gTTS(text="hmmmm That is a Small brief
about Sid",lang='en-uk')
                  s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                  playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                  os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except:
               pass
         elif "fitness tip" in q_:
            print("\t                Always remember: No Pain! No
Gain! \n")
            f=" ⊗Fix a Workout Schedule Aleast 20mins a Day
\n ⊗Always Remember there're no Faster Results! \n
```

```
⊗Sleep is a Very Important Aspect \n ⊗Go for HIIT(High-Intensity Interval Training) workouts \n ⊗Don't Go for WeightTraining unless you are done with Body-WeightTraining \n ⊗Avoid Junk Foods \n ⊗Don't Go for Diet unless you are trying 6-pack ABS \n ⊗Never Over-Indulge Food \n ⊗Have Self-Control \n ⊗Cardio+BodyWeightTraining Give Spectacular Results!"
            print(f)
            try:
               if v == "YES":
                  s=gTTS(text="hmm Just try these out....Quiet Essential!",lang='en-uk')
                  s.save(r"C:\Python Files\Audiofiles\VoiceOver.mp3")
                  playsound(r"C:\Python Files\Audiofiles\VoiceOver.mp3")
                  os.remove("C:\Python Files\Audiofiles\VoiceOver.mp3")
            except:
               pass
         elif "do you know my" in q_ or "did you know my" in q_ or "do you know me" in q_ or "recognize" in q_ or " recognise" in q_ or "remember" in q_:
            __r__=["'I Don't think so","Sorry I don't","Nope"," Maybe","Sorry I don't remember"]
            i=random.randint(0,len(__r__)-1)
            print(__r__[i])
            if v == "YES":
               s=gTTS(text=__r__[i],lang='en-uk')
               s.save(r"C:\Python Files\Audiofiles\VoiceOver.mp3")
               playsound(r"C:\Python Files\Audiofiles\VoiceOver.mp3")
               os.remove("C:\Python
```

```
Files\Audiofiles\VoiceOver.mp3")
        elif "html file" in q_ or "extract html" in  q_ or "html
content" in q_:
            try:
               if v == "YES":
                  s=gTTS(text="Enter the URL:",lang='en-uk')
                  s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                  playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                  os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
               url=input("Enter the URL:")
               r=requests.get(url)
               s = BeautifulSoup(r.text,"html.parser")
               print(s.prettify())
               if v == "YES":
                  s=gTTS(text="here's the HTML file",lang='en-
uk')
                  s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                  playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                  os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except requests.exceptions.ConnectionError:
               print(" \t⚠Oops...Make Sure That You Are
Connected With Internet⚠")
               if v == "YES":
                  playsound(r"C:\Python Files\Audiofiles\offline.
mp3")
        elif "weather" in q_ or "climate" in q_ or "forecast" in
q_ or ("wind" in q_ and ("direction" in q_ or "speed" in q_))
or "visibility" in q_ or ("dew" in q_ and "point" in q_) or ("
```

```
barometer" in q_ and "reading" in q_) or ("atm" in q_ and "
pressure" in q_) or "humidity" in q_:
            try:
               if ("weather" in q_ or "climate" in q_ or "forecast"
in q_ or ("wind" in q_ and ("direction" in q_ or "speed" in q_))
 or "visibility" in q_ or ("dew" in q_ and "point" in q_) or ("
barometer" in q_ and "reading" in q_) or ("atmo" in q_ and "
pressure" in q_)) or "humidity" in q_ and ("in" in q_ or "on"
in q_ or "from" in q_):
                  r = requests.get("https://google.com/
search?q={} msn weather".format(q_))
                  s = BeautifulSoup(r.content, 'html.parser')
                  e=[]
                  for a_ in s.find_all('a', href=True):
                     e.append(a_['href'])
                  x=[]
                  for t in e:
                     if "/url?q=https://www.msn.com/en" in t:
                        x.append(e.index(t))
                  l=e[x[0]]
                  i=l.index('&')
                  q=l[7:i]
                  r=requests.get("{}".format(q))
               else:
                  r=requests.get("https://www.msn.com/en-in/
weather/today/")
               s = BeautifulSoup(r.text,"html.parser")
               print("\t                    Weather Forecast
Report ")
               w=s.find(class_='weather-info')
               t=s.find(class_='current-info')
               t_=(t.text).split()
               w_=(w.text).split()
               try:
```

```
                j=l.index('%')-8
                c=l[47:j]
                print("Location:",c,"\n")
            except:
                pass
            if "F" in t_[1]:
                _t_= (int(t_[0])-32)*(5/9)
                t__='1'
            else:
                _t_= (int(t_[0])*(9/5))+32
                t__='2'
            print("Temperature:",t_[0],t_[1],"/////",
round(_t_),t_[2])
            print('\n')
            print("Weather ↴",w.text)
            print("\n                ","\t That's the Current
Weather \t")
            l_=(w.text).split()
            if t__=='1':
                t__='fahrenheit'
            else:
                t__='celsius'
            if "weather" in q_:
                j=l_.index('Feels')
                v_=' '.join(l_[:j])
                v_='weather is ' + v_
            elif "climate" in q_:
                i=l_.index('Feels')
                j=l_.index('Wind')
                v_=' '.join(l_[i:j])
                v_='it ' + v_ + ' ' +t__
            elif "wind" in q_ and "direction" in q_:
                i=l_.index('Wind')+1
                v_=l_[i]
```

```
            v_='wind is passing via ' + v_ +" direction"
        elif "wind" in q_:
            i=l_.index('Wind')+1
            j=l_.index('Barometer')
            v_=' '.join(l_[i:j])
            v_='wind speed is ' + v_
        elif "barometer" in q_ or "pressure" in q_:
            i=l_.index('Barometer')+1
            j=l_.index('mb')
            v_=' '.join(l_[i:j])
            v_='the pressure is ' + v_ + ' milli bar'
        elif "visib" in q_:
            i=l_.index('Visibility')+1
            j=l_.index('Humidity')
            v_=' '.join(l_[i:j])
            v_='visibility is ' + v_
        elif "humid" in q_:
            i=l_.index('Humidity')+1
            j=l_.index('Dew')
            v_=' '.join(l_[i:j])
            v_='humidity is ' + v_
        elif "dew" in q_:
            i=l_.index('Point')
            v_=' '.join(l_[i:])
            v_='dew point is ' + v_ + ' ' +t__
        if v == "YES":
            if "in" in q_ or "on" in q_ or "from" in q_:
                try:
                    s=gTTS(text="hmm Right now {} in {}".
format(v_,c),lang='en-uk')
                except:
                    s=gTTS(text="hmm Right now {}".
format(v_),lang='en-uk')
            else:
```

```
                    s=gTTS(text="hmm Right now {}".format(v_),
lang='en-uk')
                    s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except requests.exceptions.ConnectionError:
                print(" \t⚠Oops...Make Sure That You Are
Connected With Internet⚠")
                if v == "YES":
                    playsound(r"C:\Python Files\Audiofiles\offline.
mp3")
        elif "time" in q_ and "timer" not in q_ and "in" not in q_
and ("now" in q_ or "current" in q_):
            t=datetime.now()
            t_h=int(t.strftime("%H"))
            if t_h >= 12:
                t__='PM'
                if t_h==12:
                    pass
                else:
                    t_h = t_h - 12
            else:
                t__='AM'
            t_m=t.strftime("%M")
            t_s=t.strftime("%S")
            print("Time Right Now:",t_h,":",t_m,":",t_s," ",t__)
            try:
                if v == "YES":
                    s=gTTS(text="hmm the time is {}:{}:{} {}".
format(t_h,t_m,t_s,t__),lang='en-uk')
                    s.save(r"C:\Python Files\Audiofiles\VoiceOver.
```

```
mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except requests.exceptions.ConnectionError:
                pass
        elif "day" in q_ and ("this" in q_ or "present" in q_ or "
today" in q_ or "now" in q_ or "current" in q_):
            try:
                r=requests.get("https://www.daysoftheyear.com/
")
                s=BeautifulSoup(r.content,"html.parser")
                n=s.find(class_='card__title heading')
                t=datetime.now()
                d=n.text
                d_=t.strftime("%A")
                print("Today is {}, To be accurate it's {}".format(d,
d_))
                if v == "YES":
                    s=gTTS(text="hmm Today is {}, To be accurate
it's {}".format(d,d_),lang='en-uk')
                    s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except:
                t=datetime.now()
                d_=t.strftime("%A")
                print("Today is {}".format(d_))
                if v == "YES":
                    s=gTTS(text="hmm Today is {}, To be accurate
```

```
it's {}".format(d,d_),lang='en-uk')
                    s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
          elif "date" in q_ and ("this" in q_ or "present" in q_ or "
today" in q_ or "now" in q_ or "current" in q_):
              d=datetime.now()
              d_=d.strftime("%d/%m/%Y")
              print("Date Today:",d_)
              try:
                 if v == "YES":
                    s=gTTS(text="hmm today's date is {}".
format(d_),lang='en-uk')
                    s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
              except requests.exceptions.ConnectionError:
                 pass
          elif "year" in q_ and ("this" in q_ or "present" in q_ or "
now" in q_ or "current" in q_):
              y=datetime.now()
              y_=y.strftime("%Y")
              print("Year:",y_)
              try:
                 if v == "YES":
                    s=gTTS(text="hmm this year is {}".format(y_),
lang='en-uk')
                    s.save(r"C:\Python Files\Audiofiles\VoiceOver.
```

```
mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                except requests.exceptions.ConnectionError:
                    pass
            elif "month" in q_ and ("this" in q_ or "present" in q_
or "now" in q_ or "current" in q_):
                m=datetime.now()
                m_=m.strftime("%B")
                print("This Month:",m_)
                try:
                    if v == "YES":
                        s=gTTS(text="hmm this month is {}".format(m_)
,lang='en-uk')
                        s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                        playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                except requests.exceptions.ConnectionError:
                    pass
            elif ("meaning" in q_ or "meant" in q_ or "mean" in q_)
and "what" in q_:
                try:
                    try:
                        if "sid" in q_ or "glenda" in q_:
                            l= q_.split()
                            try:
                                l.remove("sid")
                                w=' '.join(l)
                            except:
```

```
                    l.remove("glenda")
                    w=' '.join(l)
                else:
                    w=q_
                g = requests.get("https://google.com/
search?q={} in cambridge dictionary".format(w))
                _s_ = BeautifulSoup(g.content, 'html.parser')
                y=[]
                for a in _s_.find_all('a', href=True):
                    y.append(a['href'])
                z=[]
                for t in y:
                    if "/url?q=https://dictionary.cambridge.org/
dictionary/english/" in t:
                        z.append(y.index(t))
                l=y[z[0]]
                j=l.index('&')
                k=l[59:j]
                if k == "":
                    print("ERROR 404:Not Found!")
                else:
                    print("\t                \||~~~~||||",k.capitalize()
,"||||~~~~||/")
                    try:
                        r = requests.get("https://dictionary.
cambridge.org/dictionary/english/{}".format(k))
                        print(r)
                        s = BeautifulSoup(r.content, 'html.parser')
                        print(k)
                        i= s.find(class_="ddef_h")
                        e=[]
                        for b in i.findAll():
                            e.append(b.text)
                        x=[]
```

```
                    for h in e:
                        if ":" in h:
                            x.append(e.index(h))
                    p_=e[x[0]]
                    print("As Per Cambridge Dictionary:",p_)
                except:
                    pass
                try:
                    r_ = requests.get("https://www.
oxfordlearnersdictionaries.com/definition/english/{}".
format(k))
                    s_ = BeautifulSoup(r_.content, 'html.parser')
                    i_= s_.find(class_="def")
                    print("\nAs Per Oxford Dictionary:",(i_.
extract()).text)
                except:
                    pass
                try:
                    r__ = requests.get("https://www.
macmillandictionary.com/dictionary/british/{}".format(k))
                    s__ = BeautifulSoup(r__.content, 'html.
parser')
                    i__= s__.find(class_="DEFINITION")
                    print("\nAs Per Macmillan Dictionary:",(i__.
extract()).text)
                except:
                    pass
            try:
                try:
                    if len((i_.extract()).text)>len((i__.extract()).
text):
                        m=(i_.extract()).text
                    else:
                        m=(i__.extract()).text
```

```
                    except:
                        try:
                            m=(i_.extract()).text
                        except:
                            m=(i__.extract()).text
                    if v == "YES":
                        s=gTTS(text="hmm {} means that {}".
format(k,m),lang='en-uk')
                        s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                except:
                    pass
            except:
                webbrowser.open("https://google.com/
search?q={}".format(w))
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Right')
                time.sleep(1)
        except requests.exceptions.ConnectionError:
            print(" \t⚠Oops...Make Sure That You Are
Connected With Internet⚠ ")
            if v == "YES":
                playsound(r"C:\Python Files\Audiofiles\offline.
mp3")
    elif ("+" in q_ or "plus" in q_ or "add" in q_ or ("add" in
q_ and ("and" in q_ or "&" in q_)) and ("-" not in q_ and "/"
```

```
not in q_ and "*" not in q_ and "sub" not in q_ and "minus"
not in q_ and "multi" not in q_ and "into" not in q_ and "div"
not in q_ and "by" not in q_ and "x" not in q_)):
            try:
                l=q_.split()
                x_=[]
                y_=[]
                _x_=[]
                z='p'
                if "+" in q_:
                    for i in l:
                        if "+" in i:
                            k=l.index(i)
                        else:
                            pass
                    if len(l[k])>1:
                        k_=l[k]
                        if k_[0].isdigit() and k_[len(k_)-1].isdigit():
                            print("It is",eval(k_))
                            _v_="It is",eval(k_)
                        elif k_[0].isdigit() and not(k_[len(k_)-1].
isdigit()):
                            w=l[k+1:]
                            _k_=''.join(w)
                            for x in k_:
                                if "-" in x:
                                    z='n'
                                if x.isdigit():
                                    x_.append(x)
                            for x in _k_:
                                if "-" in x:
                                    z_='n'
                                if x.isdigit():
                                    y_.append(x)
```

```
                    n=eval(''.join(x_))
                    n_=eval(''.join(y_))
                    if z=='n':
                        n=-n
                    if z_=='n':
                        n_=-n_
                    print("It is",n+n_)
                    _v_="It is",n+n_
                elif not(k_[0].isdigit()) and k_[len(k_)-1].
isdigit():
                    w=l[:k]
                    _k_=''.join(w)
                    for x in k_:
                        if "-" in x:
                            z_='n'
                        if x.isdigit():
                            x_.append(x)
                    for x in _k_:
                        if "-" in x:
                            z='n'
                        if x.isdigit():
                            y_.append(x)
                    print(y_)
                    n_=eval(''.join(x_))
                    n=eval(''.join(y_))
                    if z=='n':
                        n=-n
                    if z_=='n':
                        n_=-n_
                    print("It is",n+n_)
                    _v_="It is",n+n_
                else:
                    w=l[:k]
                    w_=l[k:]
```

```
        e=''.join(w)
        e_=''.join(w_)
        for x in e:
            if "-" in x:
                z_='n'
            if x.isdigit():
                x_.append(x)
        for x in e_:
            if "-" in x:
                z='n'
            if x.isdigit():
                y_.append(x)
        n_=eval(''.join(x_))
        n=eval(''.join(y_))
        if z=='n':
            n=-n
        if z_=='n':
            n_=-n_
        print("It is",n+n_)
        _v_="It is",n+n_
    elif "plus" in q_:
        for i in l:
            if "plus" in i:
                k=l.index(i)
            else:
                pass
        if len(l[k])>4:
            k_=l[k]
            if k_[0].isdigit() and k_[len(k_)-1].isdigit():
                o=k_.index('p')
                w=k_[:o]
                w_=k_[o:]
                e=''.join(w)
                e_=''.join(w_)
```

```
                    for x in e:
                        if "-" in x:
                            z_='n'
                        if x.isdigit():
                            x_.append(x)
                    for x in e_:
                        if "-" in x:
                            z='n'
                        if x.isdigit():
                            y_.append(x)
                    n_=eval(''.join(x_))
                    n=eval(''.join(y_))
                    if z=='n':
                        n=-n
                    if z_=='n':
                        n_=-n_
                    print("It is",n+n_)
                    _v_="It is",n+n_
                elif k_[0].isdigit() and not(k_[len(k_)-1].
isdigit()):
                    w=l[k+1:]
                    _k_=''.join(w)
                    for x in k_:
                        if "-" in x:
                            z='n'
                        if x.isdigit():
                            x_.append(x)
                    for x in _k_:
                        if "-" in x:
                            z_='n'
                        if x.isdigit():
                            y_.append(x)
                    n=eval(''.join(x_))
                    n_=eval(''.join(y_))
```

```
                    if z=='n':
                        n=-n
                    if z_=='n':
                        n_=-n_
                    print("It is",n+n_)
                    _v_="It is",n+n_
                elif not(k_[0].isdigit()) and k_[len(k_)-1].
isdigit():
                    w=l[:k]
                    _k_=''.join(w)
                    for x in k_:
                        if "-" in x:
                            z_='n'
                        if x.isdigit():
                            x_.append(x)
                    for x in _k_:
                        if "-" in x:
                            z='n'
                        if x.isdigit():
                            y_.append(x)
                    n_=eval(''.join(x_))
                    n=eval(''.join(y_))
                    if z=='n':
                        n=-n
                    if z_=='n':
                        n_=-n_
                    print("It is",n+n_)
                    _v_="It is",n+n_
                else:
                    w=l[:k]
                    w_=l[k:]
                    e=''.join(w)
                    e_=''.join(w_)
                    for x in e:
```

```
            if "-" in x:
                z_='n'
            if x.isdigit():
                x_.append(x)
        for x in e_:
            if "-" in x:
                z='n'
            if x.isdigit():
                y_.append(x)
        n_=eval(''.join(x_))
        n=eval(''.join(y_))
        if z=='n':
            n=-n
        if z_=='n':
            n_=-n_
        print("It is",n+n_)
        _v_="It is",n+n_
elif "add" in q_ and "and" in q_:
    for i in l:
        if "and" in i:
            k=l.index(i)
        else:
            pass
    if len(l[k])>3:
        k_=l[k]
        if k_[0].isdigit() and k_[len(k_)-1].isdigit():
            o=k_.index('n')
            w=k_[:o]
            w_=k_[o:]
            e=''.join(w)
            e_=''.join(w_)
            for x in e:
                if "-" in x:
                    z_='n'
```

```
                    if x.isdigit():
                        x_.append(x)
                for x in e_:
                    if "-" in x:
                        z='n'
                    if x.isdigit():
                        y_.append(x)
                n_=eval(''.join(x_))
                n=eval(''.join(y_))
                if z=='n':
                    n=-n
                if z_=='n':
                    n_=-n_
                print("It is",n+n_)
                _v_="It is",n+n_
            elif k_[0].isdigit() and not(k_[len(k_)-1].
isdigit()):
                w=l[k+1:]
                _k_=''.join(w)
                for x in k_:
                    if "-" in x:
                        z='n'
                    if x.isdigit():
                        x_.append(x)
                for x in _k_:
                    if "-" in x:
                        z_='n'
                    if x.isdigit():
                        y_.append(x)
                n=eval(''.join(x_))
                n_=eval(''.join(y_))
                if z=='n':
                    n=-n
                if z_=='n':
```

```
                    n_=-n_
                print("It is",n+n_)
                _v_="It is",n+n_
            elif not(k_[0].isdigit()) and k_[len(k_)-1].
isdigit():
                w=l[:k]
                _k_=''.join(w)
                for x in k_:
                    if "-" in x:
                        z_='n'
                    if x.isdigit():
                        x_.append(x)
                for x in _k_:
                    if "-" in x:
                        z='n'
                    if x.isdigit():
                        y_.append(x)
                n_=eval(''.join(x_))
                n=eval(''.join(y_))
                if z=='n':
                    n=-n
                if z_=='n':
                    n_=-n_
                print("It is",n+n_)
                _v_="It is",n+n_
            else:
                w=l[:k]
                w_=l[k:]
                e=''.join(w)
                e_=''.join(w_)
                for x in e:
                    if "-" in x:
                        z_='n'
                    if x.isdigit():
```

```
            x_.append(x)
        for x in e_:
            if "-" in x:
                z='n'
            if x.isdigit():
                y_.append(x)
        n_=eval(''.join(x_))
        n=eval(''.join(y_))
        if z=='n':
            n=-n
        if z_=='n':
            n_=-n_
        print("It is",n+n_)
        _v_="It is",n+n_
elif "add" in q_ and "&" in q_:
    for i in l:
        if "&" in i:
            k=l.index(i)
        else:
            pass
    if len(l[k])>1:
        k_=l[k]
        if k_[0].isdigit() and k_[len(k_)-1].isdigit():
            o=k_.index('&')
            w=k_[:o]
            w_=k_[o:]
            e=''.join(w)
            e_=''.join(w_)
            for x in e:
                if "-" in x:
                    z_='n'
                if x.isdigit():
                    x_.append(x)
            for x in e_:
```

```
                    if "-" in x:
                        z='n'
                    if x.isdigit():
                        y_.append(x)
                n_=eval(''.join(x_))
                n=eval(''.join(y_))
                if z=='n':
                    n=-n
                if z_=='n':
                    n_=-n_
                print("It is",n+n_)
                _v_="It is",n+n_
            elif k_[0].isdigit() and not(k_[len(k_)-1].
isdigit()):
                w=l[k+1:]
                _k_=''.join(w)
                for x in k_:
                    if "-" in x:
                        z='n'
                    if x.isdigit():
                        x_.append(x)
                for x in _k_:
                    if "-" in x:
                        z_='n'
                    if x.isdigit():
                        y_.append(x)
                n=eval(''.join(x_))
                n_=eval(''.join(y_))
                if z=='n':
                    n=-n
                if z_=='n':
                    n_=-n_
                print("It is",n+n_)
                _v_="It is",n+n_
```

```
                elif not(k_[0].isdigit()) and k_[len(k_)-1].
isdigit():
                    w=l[:k]
                    _k_=''.join(w)
                    for x in k_:
                        if "-" in x:
                            z_='n'
                        if x.isdigit():
                            x_.append(x)
                    for x in _k_:
                        if "-" in x:
                            z='n'
                        if x.isdigit():
                            y_.append(x)
                    n_=eval(''.join(x_))
                    n=eval(''.join(y_))
                    if z=='n':
                        n=-n
                    if z_=='n':
                        n_=-n_
                    print("It is",n+n_)
                    _v_="It is",n+n_
                else:
                    w=l[:k]
                    w_=l[k:]
                    e=''.join(w)
                    e_=''.join(w_)
                    for x in e:
                        if "-" in x:
                            z_='n'
                        if x.isdigit():
                            x_.append(x)
                    for x in e_:
                        if "-" in x:
```

```
                        z='n'
                    if x.isdigit():
                        y_.append(x)
                n_=eval(''.join(x_))
                n=eval(''.join(y_))
                if z=='n':
                    n=-n
                if z_=='n':
                    n_=-n_
                print("It is",n+n_)
                _v_="It is",n+n_
            try:
                if v=="YES":
                    s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                except:
                    pass
            except:
                try:
                    try:
                        l= q_.split()
                        try:
                            l.remove("sid")
                            w=' '.join(l)
                        except:
                            l.remove("glenda")
                            w=' '.join(l)
                    except:
                        w=q_
```

```
                print("Just Sit Back And Hold Tight!")
                webbrowser.open("https://google.com/
search?q={}".format(w))
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Right')
                time.sleep(1)
            except requests.exceptions.ConnectionError:
                print(" \t⚠Oops...Make Sure That You Are
Connected With Internet⚠ ")
                if v == "YES":
                    playsound(r"C:\Python
Files\Audiofiles\offline.mp3")
        elif ("-" in q_ or "minus" in q_ or "sub" in q_ or ("sub"
in q_ and ("and" in q_ or "&" in q_)) and ("+" not in q_ and "/"
not in q_ and "*" not in q_ and "add" not in q_ and "plus" not
in q_ and "multi" not in q_ and "into" not in q_ and "div" not
in q_ and "by" not in q_ and "x" not in q_)):
            try:
                l=q_.split()
                x_=[]
                y_=[]
                _x_=[]
                z='p'
                if "-" in q_:
                    for i in l:
                        if "-" in i:
                            k=l.index(i)
                        else:
                            pass
                    if len(l[k])>1:
```

```
                k_=l[k]
                if k_[0].isdigit() and k_[len(k_)-1].isdigit():
                    print("It is",eval(k_))
                    _v_="It is",eval(k_)
                elif k_[0].isdigit() and not(k_[len(k_)-1].
isdigit()):
                    w=l[k+1:]
                    _k_=''.join(w)
                    for x in k_:
                        if "-" in x:
                            z='n'
                        if x.isdigit():
                            x_.append(x)
                    for x in _k_:
                        if "-" in x:
                            z_='n'
                        if x.isdigit():
                            y_.append(x)
                    n=eval(''.join(x_))
                    n_=eval(''.join(y_))
                    if z=='n':
                        n=-n
                    if z_=='n':
                        n_=-n_
                    print("It is",n-n_)
                    _v_="It is",n-n_
                elif not(k_[0].isdigit()) and k_[len(k_)-1].
isdigit():
                    w=l[:k]
                    _k_=''.join(w)
                    for x in k_:
                        if "-" in x:
                            z_='n'
                        if x.isdigit():
```

```
            x_.append(x)
        for x in _k_:
            if "-" in x:
                z='n'
            if x.isdigit():
                y_.append(x)
        n_=eval(''.join(x_))
        n=eval(''.join(y_))
        if z=='n':
            n=-n
        if z_=='n':
            n_=-n_
        print("It is",n-n_)
        _v_="It is",n-n_
else:
    w=l[:k]
    w_=l[k:]
    e=''.join(w)
    e_=''.join(w_)
    for x in e:
        if "-" in x:
            z_='n'
        if x.isdigit():
            x_.append(x)
    for x in e_:
        if "-" in x:
            z='n'
        if x.isdigit():
            y_.append(x)
    n_=eval(''.join(x_))
    n=eval(''.join(y_))
    if z=='n':
        n=-n
    if z_=='n':
```

```
            n_=-n_
        print("It is",n-n_)
        _v_="It is",n-n_
elif "minus" in q_:
    for i in l:
        if "minus" in i:
            k=l.index(i)
        else:
            pass
    if len(l[k])>5:
        k_=l[k]
        if k_[0].isdigit() and k_[len(k_)-1].isdigit():
            o=k_.index('m')
            w=k_[:o]
            w_=k_[o:]
            e=''.join(w)
            e_=''.join(w_)
            for x in e:
                if "-" in x:
                    z_='n'
                if x.isdigit():
                    x_.append(x)
            for x in e_:
                if "-" in x:
                    z='n'
                if x.isdigit():
                    y_.append(x)
            n_=eval(''.join(x_))
            n=eval(''.join(y_))
            if z=='n':
                n=-n
            if z_=='n':
                n_=-n_
            print("It is",n-n_)
```

```
                    _v_="It is",n-n_
                elif k_[0].isdigit() and not(k_[len(k_)-1].
isdigit()):
                    w=l[k+1:]
                    _k_=''.join(w)
                    for x in k_:
                        if "-" in x:
                            z='n'
                        if x.isdigit():
                            x_.append(x)
                    for x in _k_:
                        if "-" in x:
                            z_='n'
                        if x.isdigit():
                            y_.append(x)
                    n=eval(''.join(x_))
                    n_=eval(''.join(y_))
                    if z=='n':
                        n=-n
                    if z_=='n':
                        n_=-n_
                    print("It is",n-n_)
                    _v_="It is",n-n_
                elif not(k_[0].isdigit()) and k_[len(k_)-1].
isdigit():
                    w=l[:k]
                    _k_=''.join(w)
                    for x in k_:
                        if "-" in x:
                            z_='n'
                        if x.isdigit():
                            x_.append(x)
                    for x in _k_:
                        if "-" in x:
```

```
            z='n'
        if x.isdigit():
            y_.append(x)
    n_=eval(''.join(x_))
    n=eval(''.join(y_))
    if z=='n':
        n=-n
    if z_=='n':
        n_=-n_
    print("It is",n-n_)
    _v_="It is",n-n_
else:
  w=l[:k]
  w_=l[k:]
  e=''.join(w)
  e_=''.join(w_)
  for x in e:
    if "-" in x:
        z_='n'
    if x.isdigit():
        x_.append(x)
  for x in e_:
    if "-" in x:
        z='n'
    if x.isdigit():
        y_.append(x)
  n_=eval(''.join(x_))
  n=eval(''.join(y_))
  if z=='n':
    n=-n
  if z_=='n':
    n_=-n_
  print("It is",n-n_)
  _v_="It is",n-n_
```

```
            elif "sub" in q_ and "and" in q_:
                for i in l:
                    if "and" in i:
                        k=l.index(i)
                    else:
                        pass
                if len(l[k])>3:
                    k_=l[k]
                    if k_[0].isdigit() and k_[len(k_)-1].isdigit():
                        o=k_.index('n')
                        w=k_[:o]
                        w_=k_[o:]
                        e=''.join(w)
                        e_=''.join(w_)
                        for x in e:
                            if "-" in x:
                                z_='n'
                            if x.isdigit():
                                x_.append(x)
                        for x in e_:
                            if "-" in x:
                                z='n'
                            if x.isdigit():
                                y_.append(x)
                        n_=eval(''.join(x_))
                        n=eval(''.join(y_))
                        if z=='n':
                            n=-n
                        if z_=='n':
                            n_=-n_
                        print("It is",n_-n)
                        _v_="It is",n-n_
                    elif k_[0].isdigit() and not(k_[len(k_)-1].
isdigit()):
```

```
                    w=l[k+1:]
                    _k_=''.join(w)
                    for x in k_:
                        if "-" in x:
                            z='n'
                        if x.isdigit():
                            x_.append(x)
                    for x in _k_:
                        if "-" in x:
                            z_='n'
                        if x.isdigit():
                            y_.append(x)
                    n=eval(''.join(x_))
                    n_=eval(''.join(y_))
                    if z=='n':
                        n=-n
                    if z_=='n':
                        n_=-n_
                    print("It is",n-n_)
                    _v_="It is",n-n_
                elif not(k_[0].isdigit()) and k_[len(k_)-1].
isdigit():
                    w=l[:k]
                    _k_=''.join(w)
                    for x in k_:
                        if "-" in x:
                            z_='n'
                        if x.isdigit():
                            x_.append(x)
                    for x in _k_:
                        if "-" in x:
                            z='n'
                        if x.isdigit():
                            y_.append(x)
```

```
            n_=eval(''.join(x_))
            n=eval(''.join(y_))
            if z=='n':
                n=-n
            if z_=='n':
                n_=-n_
            print("It is",n-n_)
            _v_="It is",n-n_
        else:
            w=l[:k]
            w_=l[k:]
            e=''.join(w)
            e_=''.join(w_)
            for x in e:
                if "-" in x:
                    z_='n'
                if x.isdigit():
                    x_.append(x)
            for x in e_:
                if "-" in x:
                    z='n'
                if x.isdigit():
                    y_.append(x)
            n_=eval(''.join(x_))
            n=eval(''.join(y_))
            if z=='n':
                n=-n
            if z_=='n':
                n_=-n_
            print("It is",n-n_)
            _v_="It is",n-n_
    elif "sub" in q_ and "&" in q_:
        for i in l:
            if "&" in i:
```

```
                k=l.index(i)
            else:
                pass
        if len(l[k])>1:
            k_=l[k]
            if k_[0].isdigit() and k_[len(k_)-1].isdigit():
                o=k_.index('&')
                w=k_[:o]
                w_=k_[o:]
                e=''.join(w)
                e_=''.join(w_)
                for x in e:
                    if "-" in x:
                        z_='n'
                    if x.isdigit():
                        x_.append(x)
                for x in e_:
                    if "-" in x:
                        z='n'
                    if x.isdigit():
                        y_.append(x)
                n_=eval(''.join(x_))
                n=eval(''.join(y_))
                if z=='n':
                    n=-n
                if z_=='n':
                    n_=-n_
                print("It is",n_-n)
                _v_="It is",n-n_
            elif k_[0].isdigit() and not(k_[len(k_)-1].
isdigit()):
                w=l[k+1:]
                _k_=''.join(w)
                for x in k_:
```

```
                    if "-" in x:
                        z='n'
                    if x.isdigit():
                        x_.append(x)
                for x in _k_:
                    if "-" in x:
                        z_='n'
                    if x.isdigit():
                        y_.append(x)
                n=eval(''.join(x_))
                n_=eval(''.join(y_))
                if z=='n':
                    n=-n
                if z_=='n':
                    n_=-n_
                print("It is",n-n_)
                _v_="It is",n-n_
            elif not(k_[0].isdigit()) and k_[len(k_)-1].
isdigit():
                w=l[:k]
                _k_=''.join(w)
                for x in k_:
                    if "-" in x:
                        z_='n'
                    if x.isdigit():
                        x_.append(x)
                for x in _k_:
                    if "-" in x:
                        z='n'
                    if x.isdigit():
                        y_.append(x)
                n_=eval(''.join(x_))
                n=eval(''.join(y_))
                if z=='n':
```

```
                    n=-n
                if z_=='n':
                    n_=-n_
                print("It is",n-n_)
                _v_="It is",n-n_
            else:
                w=l[:k]
                w_=l[k:]
                e=''.join(w)
                e_=''.join(w_)
                for x in e:
                    if "-" in x:
                        z_='n'
                    if x.isdigit():
                        x_.append(x)
                for x in e_:
                    if "-" in x:
                        z='n'
                    if x.isdigit():
                        y_.append(x)
                n_=eval(''.join(x_))
                n=eval(''.join(y_))
                if z=='n':
                    n=-n
                if z_=='n':
                    n_=-n_
                print("It is",n-n_)
                _v_="It is",n-n_
        try:
            if v=="YES":
                s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                playsound(r"C:\Python
```

```
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                except:
                    pass
            except:
                try:
                    try:
                        l= q_.split()
                        try:
                            l.remove("sid")
                            w=' '.join(l)
                        except:
                            l.remove("glenda")
                            w=' '.join(l)
                    except:
                        w=q_
                    print("Just Sit Back And Hold Tight!")
                    webbrowser.open("https://google.com/
search?q={}".format(w))
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
                    keyboard.press_and_release('win + Right')
                    time.sleep(1)
                except requests.exceptions.ConnectionError:
                    print(" \t⚠Oops...Make Sure That You Are
Connected With Internet⚠ ")
                    if v == "YES":
                        playsound(r"C:\Python
Files\Audiofiles\offline.mp3")
        elif ("*" in q_ or "into" in q_ or "x" in q_ or "multi" in q_
```

```
or ("multi" in q_ and ("and" in q_ or "&" in q_))) and ("+" not
in q_ and "/" not in q_ and "-" not in q_ and "add" not in q_
and "plus" not in q_ and "sub" not in q_ and "minus" not in q_
and "div" not in q_ and "by" not in q_):
            try:
               l=q_.split()
               x_=[]
               y_=[]
               _x_=[]
               z='p'
               if "*" in q_:
                  for i in l:
                     if "*" in i:
                        k=l.index(i)
                     else:
                        pass
                  if len(l[k])>1:
                     k_=l[k]
                     if k_[0].isdigit() and k_[len(k_)-1].isdigit():
                        print("It is",eval(k_))
                        _v_="It is",eval(k)
                     elif k_[0].isdigit() and not(k_[len(k_)-1].
isdigit()):
                        w=l[k+1:]
                        _k_=''.join(w)
                        for x in k_:
                           if "-" in x:
                              z='n'
                           if x.isdigit():
                              x_.append(x)
                        for x in _k_:
                           if "-" in x:
                              z_='n'
                           if x.isdigit():
```

```
                    y_.append(x)
                n=eval(''.join(x_))
                n_=eval(''.join(y_))
                if z=='n':
                    n=-n
                if z_=='n':
                    n_=-n_
                print("It is",n*n_)
                _v_="It is",n*n_
            elif not(k_[0].isdigit()) and k_[len(k_)-1].
isdigit():
                w=l[:k]
                _k_=''.join(w)
                for x in k_:
                    if "-" in x:
                        z_='n'
                    if x.isdigit():
                        x_.append(x)
                for x in _k_:
                    if "-" in x:
                        z='n'
                    if x.isdigit():
                        y_.append(x)
                n_=eval(''.join(x_))
                n=eval(''.join(y_))
                if z=='n':
                    n=-n
                if z_=='n':
                    n_=-n_
                print("It is",n*n_)
                _v_="It is",n*n_
            else:
                w=l[:k]
                w_=l[k:]
```

```
        e=''.join(w)
        e_=''.join(w_)
        for x in e:
            if "-" in x:
                z_='n'
            if x.isdigit():
                x_.append(x)
        for x in e_:
            if "-" in x:
                z='n'
            if x.isdigit():
                y_.append(x)
        n_=eval(''.join(x_))
        n=eval(''.join(y_))
        if z=='n':
            n=-n
        if z_=='n':
            n_=-n_
        print("It is",n*n_)
        _v_="It is",n*n_
    elif "into" in q_:
        for i in l:
            if "into" in i:
                k=l.index(i)
            else:
                pass
        if len(l[k])>4:
            k_=l[k]
            if k_[0].isdigit() and k_[len(k_)-1].isdigit():
                o=k_.index('t')
                w=k_[:o]
                w_=k_[o:]
                e=''.join(w)
                e_=''.join(w_)
```

```
                    for x in e:
                        if "-" in x:
                            z_='n'
                        if x.isdigit():
                            x_.append(x)
                    for x in e_:
                        if "-" in x:
                            z='n'
                        if x.isdigit():
                            y_.append(x)
                    n_=eval(''.join(x_))
                    n=eval(''.join(y_))
                    if z=='n':
                        n=-n
                    if z_=='n':
                        n_=-n_
                    print("It is",n*n_)
                    _v_="It is",n*n_
                elif k_[0].isdigit() and not(k_[len(k_)-1].
isdigit()):
                    w=l[k+1:]
                    _k_=''.join(w)
                    for x in k_:
                        if "-" in x:
                            z='n'
                        if x.isdigit():
                            x_.append(x)
                    for x in _k_:
                        if "-" in x:
                            z_='n'
                        if x.isdigit():
                            y_.append(x)
                    n=eval(''.join(x_))
                    n_=eval(''.join(y_))
```

```
                    if z=='n':
                        n=-n
                    if z_=='n':
                        n_=-n_
                    print("It is",n*n_)
                    _v_="It is",n*n_
                elif not(k_[0].isdigit()) and k_[len(k_)-1].
isdigit():
                    w=l[:k]
                    _k_=''.join(w)
                    for x in k_:
                        if "-" in x:
                            z_='n'
                        if x.isdigit():
                            x_.append(x)
                    for x in _k_:
                        if "-" in x:
                            z='n'
                        if x.isdigit():
                            y_.append(x)
                    n_=eval(''.join(x_))
                    n=eval(''.join(y_))
                    if z=='n':
                        n=-n
                    if z_=='n':
                        n_=-n_
                    print("It is",n*n_)
                    _v_="It is",n*n_
                else:
                    w=l[:k]
                    w_=l[k:]
                    e=''.join(w)
                    e_=''.join(w_)
                    for x in e:
```

```
            if "-" in x:
                z_='n'
            if x.isdigit():
                x_.append(x)
        for x in e_:
            if "-" in x:
                z='n'
            if x.isdigit():
                y_.append(x)
        n_=eval(''.join(x_))
        n=eval(''.join(y_))
        if z=='n':
            n=-n
        if z_=='n':
            n_=-n_
        print("It is",n*n_)
        _v_="It is",n*n_
elif "multi" in q_ and "and" in q_:
    for i in l:
        if "and" in i:
            k=l.index(i)
        else:
            pass
    if len(l[k])>3:
        k_=l[k]
        if k_[0].isdigit() and k_[len(k_)-1].isdigit():
            o=k_.index('n')
            w=k_[:o]
            w_=k_[o:]
            e=''.join(w)
            e_=''.join(w_)
            for x in e:
                if "-" in x:
                    z_='n'
```

```
                    if x.isdigit():
                        x_.append(x)
                for x in e_:
                    if "-" in x:
                        z='n'
                    if x.isdigit():
                        y_.append(x)
                n_=eval(''.join(x_))
                n=eval(''.join(y_))
                if z=='n':
                    n=-n
                if z_=='n':
                    n_=-n_
                print("It is",n_*n)
                _v_="It is",n*n_
            elif k_[0].isdigit() and not(k_[len(k_)-1].
isdigit()):
                w=l[k+1:]
                _k_=''.join(w)
                for x in k_:
                    if "-" in x:
                        z='n'
                    if x.isdigit():
                        x_.append(x)
                for x in _k_:
                    if "-" in x:
                        z_='n'
                    if x.isdigit():
                        y_.append(x)
                n=eval(''.join(x_))
                n_=eval(''.join(y_))
                if z=='n':
                    n=-n
                if z_=='n':
```

```
                    n_=-n_
                print("It is",n*n_)
                _v_="It is",n*n_
            elif not(k_[0].isdigit()) and k_[len(k_)-1].
isdigit():
                w=l[:k]
                _k_=''.join(w)
                for x in k_:
                    if "-" in x:
                        z_='n'
                    if x.isdigit():
                        x_.append(x)
                for x in _k_:
                    if "-" in x:
                        z='n'
                    if x.isdigit():
                        y_.append(x)
                n_=eval(''.join(x_))
                n=eval(''.join(y_))
                if z=='n':
                    n=-n
                if z_=='n':
                    n_=-n_
                print("It is",n*n_)
                _v_="It is",n*n_
            else:
                w=l[:k]
                w_=l[k:]
                e=''.join(w)
                e_=''.join(w_)
                for x in e:
                    if "-" in x:
                        z_='n'
                    if x.isdigit():
```

```
            x_.append(x)
        for x in e_:
            if "-" in x:
                z='n'
            if x.isdigit():
                y_.append(x)
        n_=eval(''.join(x_))
        n=eval(''.join(y_))
        if z=='n':
            n=-n
        if z_=='n':
            n_=-n_
        print("It is",n*n_)
        _v_="It is",n*n_
elif "multi" in q_ and "&" in q_:
    for i in l:
        if "&" in i:
            k=l.index(i)
        else:
            pass
    if len(l[k])>1:
        k_=l[k]
        if k_[0].isdigit() and k_[len(k_)-1].isdigit():
            o=k_.index('&')
            w=k_[:o]
            w_=k_[o:]
            e=''.join(w)
            e_=''.join(w_)
            for x in e:
                if "-" in x:
                    z_='n'
                if x.isdigit():
                    x_.append(x)
            for x in e_:
```

```
                    if "-" in x:
                        z='n'
                    if x.isdigit():
                        y_.append(x)
                n_=eval(''.join(x_))
                n=eval(''.join(y_))
                if z=='n':
                    n=-n
                if z_=='n':
                    n_=-n_
                print("It is",n_*n)
                _v_="It is",n*n_
            elif k_[0].isdigit() and not(k_[len(k_)-1].
isdigit()):
                w=l[k+1:]
                _k_=''.join(w)
                for x in k_:
                    if "-" in x:
                        z='n'
                    if x.isdigit():
                        x_.append(x)
                for x in _k_:
                    if "-" in x:
                        z_='n'
                    if x.isdigit():
                        y_.append(x)
                n=eval(''.join(x_))
                n_=eval(''.join(y_))
                if z=='n':
                    n=-n
                if z_=='n':
                    n_=-n_
                print("It is",n*n_)
                _v_="It is",n*n_
```

```
                elif not(k_[0].isdigit()) and k_[len(k_)-1].
isdigit():
                    w=l[:k]
                    _k_=''.join(w)
                    for x in k_:
                        if "-" in x:
                            z_='n'
                        if x.isdigit():
                            x_.append(x)
                    for x in _k_:
                        if "-" in x:
                            z='n'
                        if x.isdigit():
                            y_.append(x)
                    n_=eval(''.join(x_))
                    n=eval(''.join(y_))
                    if z=='n':
                        n=-n
                    if z_=='n':
                        n_=-n_
                    print("It is",n*n_)
                    _v_="It is",n*n_
                else:
                    w=l[:k]
                    w_=l[k:]
                    e=''.join(w)
                    e_=''.join(w_)
                    for x in e:
                        if "-" in x:
                            z_='n'
                        if x.isdigit():
                            x_.append(x)
                    for x in e_:
                        if "-" in x:
```

```
                z='n'
            if x.isdigit():
                y_.append(x)
        n_=eval(''.join(x_))
        n=eval(''.join(y_))
        if z=='n':
            n=-n
        if z_=='n':
            n_=-n_
        print("It is",n*n_)
        _v_="It is",n*n_
elif "x" in q_:
    for i in l:
        if "x" in i:
            k=l.index(i)
        else:
            pass
    if len(l[k])>1:
        k_=l[k]
        if k_[0].isdigit() and k_[len(k_)-1].isdigit():
            o=k_.index('x')
            w=k_[:o]
            w_=k_[o:]
            e=''.join(w)
            e_=''.join(w_)
            for x in e:
                if "-" in x:
                    z_='n'
                if x.isdigit():
                    x_.append(x)
            for x in e_:
                if "-" in x:
                    z='n'
                if x.isdigit():
```

```
                    y_.append(x)
                n_=eval(''.join(x_))
                n=eval(''.join(y_))
                if z=='n':
                    n=-n
                if z_=='n':
                    n_=-n_
                print("It is",n_*n)
                _v_="It is",n*n_
            elif k_[0].isdigit() and not(k_[len(k_)-1].
isdigit()):
                w=l[k+1:]
                _k_=''.join(w)
                for x in k_:
                    if "-" in x:
                        z='n'
                    if x.isdigit():
                        x_.append(x)
                for x in _k_:
                    if "-" in x:
                        z_='n'
                    if x.isdigit():
                        y_.append(x)
                n=eval(''.join(x_))
                n_=eval(''.join(y_))
                if z=='n':
                    n=-n
                if z_=='n':
                    n_=-n_
                print("It is",n*n_)
                _v_="It is",n*n_
            elif not(k_[0].isdigit()) and k_[len(k_)-1].
isdigit():
                w=l[:k]
```

```
        _k_=''.join(w)
        for x in k_:
            if "-" in x:
                z_='n'
            if x.isdigit():
                x_.append(x)
        for x in _k_:
            if "-" in x:
                z='n'
            if x.isdigit():
                y_.append(x)
        n_=eval(''.join(x_))
        n=eval(''.join(y_))
        if z=='n':
            n=-n
        if z_=='n':
            n_=-n_
        print("It is",n*n_)
        _v_="It is",n*n_
else:
    w=l[:k]
    w_=l[k:]
    e=''.join(w)
    e_=''.join(w_)
    for x in e:
        if "-" in x:
            z_='n'
        if x.isdigit():
            x_.append(x)
    for x in e_:
        if "-" in x:
            z='n'
        if x.isdigit():
            y_.append(x)
```

```
                    n_=eval(''.join(x_))
                    n=eval(''.join(y_))
                    if z=='n':
                        n=-n
                    if z_=='n':
                        n_=-n_
                    print("It is",n*n_)
                    _v_="It is",n*n_
                try:
                    if v=="YES":
                        s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                        s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                except:
                    pass
            except:
                try:
                    try:
                        l= q_.split()
                        try:
                            l.remove("sid")
                            w=' '.join(l)
                        except:
                            l.remove("glenda")
                            w=' '.join(l)
                    except:
                        w=q_
                    print("Just Sit Back And Hold Tight!")
                    webbrowser.open("https://google.com/
search?q={}".format(w))
```

```
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Right')
                time.sleep(1)
            except requests.exceptions.ConnectionError:
                print(" \t⚠Oops...Make Sure That You Are Connected With Internet⚠ ")
                if v == "YES":
                    playsound(r"C:\Python Files\Audiofiles\offline.mp3")
        elif ("/" in q_ or "by" in q_ or "div" in q_ or ("div" in q_ and ("and" in q_ or "&" in q_)) and ("+" not in q_ and "-" not in q_ and "*" not in q_ and "add" not in q_ and "plus" not in q_ and "multi" not in q_ and "into" not in q_ and "sub" not in q_ and "minus" not in q_ and "x" not in q_)):
            try:
                l=q_.split()
                x_=[]
                y_=[]
                _x_=[]
                z='p'
                if "/" in q_:
                    for i in l:
                        if "/" in i:
                            k=l.index(i)
                        else:
                            pass
                    if len(l[k])>1:
                        k_=l[k]
                        if k_[0].isdigit() and k_[len(k_)-1].isdigit():
                            print("It is",eval(k_))
```

```
                    _v_="It is",eval(k_)
                elif k_[0].isdigit() and not(k_[len(k_)-1].
isdigit()):
                    w=l[k+1:]
                    _k_=''.join(w)
                    for x in k_:
                        if "-" in x:
                            z='n'
                        if x.isdigit():
                            x_.append(x)
                    for x in _k_:
                        if "-" in x:
                            z_='n'
                        if x.isdigit():
                            y_.append(x)
                    n=eval(''.join(x_))
                    n_=eval(''.join(y_))
                    if z=='n':
                        n=-n
                    if z_=='n':
                        n_=-n_
                    print("It is",n/n_)
                    _v_="It is",n/n_
                elif not(k_[0].isdigit()) and k_[len(k_)-1].
isdigit():
                    w=l[:k]
                    _k_=''.join(w)
                    for x in k_:
                        if "-" in x:
                            z_='n'
                        if x.isdigit():
                            x_.append(x)
                    for x in _k_:
                        if "-" in x:
```

```
            z='n'
        if x.isdigit():
            y_.append(x)
    n_=eval(''.join(x_))
    n=eval(''.join(y_))
    if z=='n':
        n=-n
    if z_=='n':
        n_=-n_
    print("It is",n/n_)
    _v_="It is",n/n_
else:
  w=l[:k]
  w_=l[k:]
  e=''.join(w)
  e_=''.join(w_)
  for x in e:
    if "-" in x:
        z_='n'
    if x.isdigit():
        x_.append(x)
  for x in e_:
    if "-" in x:
        z='n'
    if x.isdigit():
        y_.append(x)
  n_=eval(''.join(x_))
  n=eval(''.join(y_))
  if z=='n':
    n=-n
  if z_=='n':
    n_=-n_
  print("It is",n/n_)
  _v_="It is",n/n_
```

```
                elif "by" in q_:
                    for i in l:
                        if "by" in i:
                            k=l.index(i)
                        else:
                            pass
                    if len(l[k])>2:
                        k_=l[k]
                        if k_[0].isdigit() and k_[len(k_)-1].isdigit():
                            o=k_.index('y')
                            w=k_[:o]
                            w_=k_[o:]
                            e=''.join(w)
                            e_=''.join(w_)
                            for x in e:
                                if "-" in x:
                                    z_='n'
                                if x.isdigit():
                                    x_.append(x)
                            for x in e_:
                                if "-" in x:
                                    z='n'
                                if x.isdigit():
                                    y_.append(x)
                            n_=eval(''.join(x_))
                            n=eval(''.join(y_))
                            if z=='n':
                                n=-n
                            if z_=='n':
                                n_=-n_
                            print("It is",n/n_)
                            _v_="It is",n/n_
                        elif k_[0].isdigit() and not(k_[len(k_)-1].
isdigit()):
```

```
                    w=l[k+1:]
                    _k_=''.join(w)
                    for x in k_:
                        if "-" in x:
                            z='n'
                        if x.isdigit():
                            x_.append(x)
                    for x in _k_:
                        if "-" in x:
                            z_='n'
                        if x.isdigit():
                            y_.append(x)
                    n=eval(''.join(x_))
                    n_=eval(''.join(y_))
                    if z=='n':
                        n=-n
                    if z_=='n':
                        n_=-n_
                    print("It is",n/n_)
                    _v_="It is",n/n_
                elif not(k_[0].isdigit()) and k_[len(k_)-1].
isdigit():
                    w=l[:k]
                    _k_=''.join(w)
                    for x in k_:
                        if "-" in x:
                            z_='n'
                        if x.isdigit():
                            x_.append(x)
                    for x in _k_:
                        if "-" in x:
                            z='n'
                        if x.isdigit():
                            y_.append(x)
```

```
            n_=eval(''.join(x_))
            n=eval(''.join(y_))
            if z=='n':
                n=-n
            if z_=='n':
                n_=-n_
            print("It is",n/n_)
            _v_="It is",n/n_
        else:
            w=l[:k]
            w_=l[k:]
            e=''.join(w)
            e_=''.join(w_)
            for x in e:
                if "-" in x:
                    z_='n'
                if x.isdigit():
                    x_.append(x)
            for x in e_:
                if "-" in x:
                    z='n'
                if x.isdigit():
                    y_.append(x)
            n_=eval(''.join(x_))
            n=eval(''.join(y_))
            if z=='n':
                n=-n
            if z_=='n':
                n_=-n_
            print("It is",n/n_)
            _v_="It is",n/n_
    elif "div" in q_ and "and" in q_:
        for i in l:
            if "and" in i:
```

```
                k=l.index(i)
            else:
                pass
        if len(l[k])>3:
            k_=l[k]
            if k_[0].isdigit() and k_[len(k_)-1].isdigit():
                o=k_.index('n')
                w=k_[:o]
                w_=k_[o:]
                e=''.join(w)
                e_=''.join(w_)
                for x in e:
                    if "-" in x:
                        z_='n'
                    if x.isdigit():
                        x_.append(x)
                for x in e_:
                    if "-" in x:
                        z='n'
                    if x.isdigit():
                        y_.append(x)
                n_=eval(''.join(x_))
                n=eval(''.join(y_))
                if z=='n':
                    n=-n
                if z_=='n':
                    n_=-n_
                print("It is",n_/n)
                _v_="It is",n/n_
            elif k_[0].isdigit() and not(k_[len(k_)-1].
isdigit()):
                w=l[k+1:]
                _k_=''.join(w)
                for x in k_:
```

```
                    if "-" in x:
                        z='n'
                    if x.isdigit():
                        x_.append(x)
                for x in _k_:
                    if "-" in x:
                        z_='n'
                    if x.isdigit():
                        y_.append(x)
                n=eval(''.join(x_))
                n_=eval(''.join(y_))
                if z=='n':
                    n=-n
                if z_=='n':
                    n_=-n_
                print("It is",n/n_)
                _v_="It is",n/n_
            elif not(k_[0].isdigit()) and k_[len(k_)-1].
isdigit():
                w=l[:k]
                _k_=''.join(w)
                for x in k_:
                    if "-" in x:
                        z_='n'
                    if x.isdigit():
                        x_.append(x)
                for x in _k_:
                    if "-" in x:
                        z='n'
                    if x.isdigit():
                        y_.append(x)
                n_=eval(''.join(x_))
                n=eval(''.join(y_))
                if z=='n':
```

```
            n=-n
        if z_=='n':
            n_=-n_
        print("It is",n/n_)
        _v_="It is",n/n_
    else:
        w=l[:k]
        w_=l[k:]
        e=''.join(w)
        e_=''.join(w_)
        for x in e:
            if "-" in x:
                z_='n'
            if x.isdigit():
                x_.append(x)
        for x in e_:
            if "-" in x:
                z='n'
            if x.isdigit():
                y_.append(x)
        n_=eval(''.join(x_))
        n=eval(''.join(y_))
        if z=='n':
            n=-n
        if z_=='n':
            n_=-n_
        print("It is",n/n_)
        _v_="It is",n/n_
elif "div" in q_ and "&" in q_:
    for i in l:
        if "&" in i:
            k=l.index(i)
        else:
            pass
```

```
                if len(l[k])>1:
                    k_=l[k]
                    if k_[0].isdigit() and k_[len(k_)-1].isdigit():
                        o=k_.index('&')
                        w=k_[:o]
                        w_=k_[o:]
                        e=''.join(w)
                        e_=''.join(w_)
                        for x in e:
                            if "-" in x:
                                z_='n'
                            if x.isdigit():
                                x_.append(x)
                        for x in e_:
                            if "-" in x:
                                z='n'
                            if x.isdigit():
                                y_.append(x)
                        n_=eval(''.join(x_))
                        n=eval(''.join(y_))
                        if z=='n':
                            n=-n
                        if z_=='n':
                            n_=-n_
                        print("It is",n_/n)
                        _v_="It is",n/n_
                    elif k_[0].isdigit() and not(k_[len(k_)-1].
isdigit()):
                        w=l[k+1:]
                        _k_=''.join(w)
                        for x in k_:
                            if "-" in x:
                                z='n'
                            if x.isdigit():
```

```
                x_.append(x)
            for x in _k_:
                if "-" in x:
                    z_='n'
                if x.isdigit():
                    y_.append(x)
            n=eval(''.join(x_))
            n_=eval(''.join(y_))
            if z=='n':
                n=-n
            if z_=='n':
                n_=-n_
            print("It is",n/n_)
            _v_="It is",n/n_
        elif not(k_[0].isdigit()) and k_[len(k_)-1].
isdigit():
            w=l[:k]
            _k_=''.join(w)
            for x in k_:
                if "-" in x:
                    z_='n'
                if x.isdigit():
                    x_.append(x)
            for x in _k_:
                if "-" in x:
                    z='n'
                if x.isdigit():
                    y_.append(x)
            n_=eval(''.join(x_))
            n=eval(''.join(y_))
            if z=='n':
                n=-n
            if z_=='n':
                n_=-n_
```

```
                    print("It is",n/n_)
                    _v_="It is",n/n_
                else:
                    w=l[:k]
                    w_=l[k:]
                    e=''.join(w)
                    e_=''.join(w_)
                    for x in e:
                        if "-" in x:
                            z_='n'
                        if x.isdigit():
                            x_.append(x)
                    for x in e_:
                        if "-" in x:
                            z='n'
                        if x.isdigit():
                            y_.append(x)
                    n_=eval(''.join(x_))
                    n=eval(''.join(y_))
                    if z=='n':
                        n=-n
                    if z_=='n':
                        n_=-n_
                    print("It is",n/n_)
                    _v_="It is",n/n_
            try:
                if v=="YES":
                    s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
```

```
                except:
                    pass
            except:
                try:
                    try:
                        l= q_.split()
                        try:
                            l.remove("sid")
                            w=' '.join(l)
                        except:
                            l.remove("glenda")
                            w=' '.join(l)
                    except:
                        w=q_
                    print("Just Sit Back And Hold Tight!")
                    webbrowser.open("https://google.com/
search?q={}".format(w))
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
                    keyboard.press_and_release('win + Right')
                    time.sleep(1)
                except requests.exceptions.ConnectionError:
                    print(" \t⚠Oops...Make Sure That You Are
Connected With Internet⚠ ")
                    if v == "YES":
                        playsound(r"C:\Python
Files\Audiofiles\offline.mp3")
        elif (("what" in q_ or "value" in q_) and "factorial" in
q_) and ("+" not in q_ and "-" not in q_ and "/" not in q_ and "
*" not in q_ and "plus" not in q_ and "add" not in q_ and "sub"
not in q_ and "minus" not in q_ and "multi" not in q_ and "
```

```
into" not in q_ and "div" not in q_ and "by" not in q_ and "x" not in q_):
            l=q_.split()
            x_=[]
            y_=[]
            _x_=[]
            for x in q_:
                for y in x:
                    if y.isdigit():
                        x_.append(y)
            n=eval(''.join(x_))
            print("The Factorial of",n,"is",math.factorial(n))
        elif (("what" in q_ or "value" in q_) and ("log" in q_ or ("natural" in q_ and "log" in q_)) and "base" not in q_) and ("+" not in q_ and "-" not in q_ and "/" not in q_ and "*" not in q_ and "plus" not in q_ and "add" not in q_ and "sub" not in q_ and "minus" not in q_ and "multi" not in q_ and "into" not in q_ and "div" not in q_ and "by" not in q_ and "x" not in q_):
            l=q_.split()
            x_=[]
            y_=[]
            _x_=[]
            for x in q_:
                for y in x:
                    if y.isdigit():
                        x_.append(y)
            n=eval(''.join(x_))
            print("The Logarithm of",n,"is",math.log(n))
            _v_="The Logarithm of",n,"is",math.log(n)
            if v=="YES":
                s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                s.save(r"C:\Python Files\Audiofiles\VoiceOver.mp3")
                playsound(r"C:\Python
```

```
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif (("what" in q_ or "value" in q_) and ("log" in q_ and
"base" in q_)) and ("+" not in q_ and "-" not in q_ and "/" not
in q_ and "*" not in q_ and "plus" not in q_ and "add" not in
q_ and "sub" not in q_ and "minus" not in q_ and "multi" not
in q_ and "into" not in q_ and "div" not in q_ and "by" not in
q_ and "x" not in q_):
            l=q_.split()
            x_=[]
            y_=[]
            _x_=[]
            if "base e" in q_ or ("natural" in q_ and "base" in q_):
                for x in q_:
                    for y in x:
                        if y.isdigit():
                            _x_.append(y)
                n=eval(''.join(_x_))
                print("The Logarithm of",n,"is",math.log(n))
                if v=="YES":
                    s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                    s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            else:
                for i in l:
                    if "base" in i:
                        j=l.index(i)
                    elif "log" in i:
                        k=l.index(i)
```

```
            if k>j:
                w_=''.join(l[k:])
                _w_=''.join(l[:k])
                for x in w_:
                    for y in x:
                        if y.isdigit():
                            x_.append(y)
                for x in _w_:
                    for y in x:
                        if y.isdigit():
                            y_.append(y)
                n=eval(''.join(x_))
                n_=eval(''.join(y_))
                print("The Logarithm of",n,"to the Base",n_,"is",
math.log(n,n_))
            elif j>k:
                w_=''.join(l[:j])
                _w_=''.join(l[j:])
                for x in w_:
                    for y in x:
                        if y.isdigit():
                            x_.append(y)
                for x in _w_:
                    for y in x:
                        if y.isdigit():
                            y_.append(y)
                n=eval(''.join(x_))
                n_=eval(''.join(y_))
                print("The Logarithm of",n,"to the Base",n_,"is",
math.log(n,n_))
            _v_="The Logarithm of",n,"to the Base",n_,"is",
math.log(n,n_)
            if v=="YES":
                s=gTTS(text="hmm "+str(_v_),lang='en-uk')
```

```
                s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif (("what" in q_ or "value" in q_) and "root" in q_
and "square" in q_) and ("+" not in q_ and "-" not in q_ and "/
" not in q_ and "*" not in q_ and "plus" not in q_ and "add"
not in q_ and "sub" not in q_ and "minus" not in q_ and "
multi" not in q_ and "into" not in q_ and "div" not in q_ and "
by" not in q_ and "x" not in q_):
            l=q_.split()
            x_=[]
            y_=[]
            _x_=[]
            for x in q_:
                for y in x:
                    if y.isdigit():
                        x_.append(y)
            n=eval(''.join(x_))
            for i in l:
                if "square" in i:
                    j=l.index(i)
                elif "root" in i:
                    k=l.index(i)
            if j>k:
                print("The SquareRoot value is",math.sqrt(n))
                _v_="The SquareRoot value is",math.sqrt(n)
                if v=="YES":
                    s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                    s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                    playsound(r"C:\Python
```

```
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            else:
                print("The SquareRoot of",n,"is",math.sqrt(n))
                _v_="The SquareRoot of",n,"is",math.sqrt(n)
                if v=="YES":
                    s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                    s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif (("what" in q_ or "value" in q_) and "square" in q_
and ("area" not in q_ or "volume" not in q_ and "root" not in
q_ and "form" not in q_)) and ("+" not in q_ and "-" not in q_
and "/" not in q_ and "*" not in q_ and "plus" not in q_ and "
add" not in q_ and "sub" not in q_ and "minus" not in q_ and "
multi" not in q_ and "into" not in q_ and "div" not in q_ and "
by" not in q_ and "x" not in q_):
            l=q_.split()
            x_=[]
            y_=[]
            _x_=[]
            for x in q_:
                for y in x:
                    if y.isdigit():
                        x_.append(y)
            n=eval(''.join(x_))
            print("The Square of",n,"is",math.pow(n,2))
            _v_="The Square of",n,"is",math.pow(n,2)
            if v=="YES":
                s=gTTS(text="hmm "+str(_v_),lang='en-uk')
```

```
            s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
            playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
            os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif (("what" in q_ or "value" in q_) and "cube" in q_
and ("area" not in q_ or "volume" not in q_ and "root" not in
q_ and "form" not in q_)) and ("+" not in q_ and "-" not in q_
and "/" not in q_ and "*" not in q_ and "plus" not in q_ and "
add" not in q_ and "sub" not in q_ and "minus" not in q_ and "
multi" not in q_ and "into" not in q_ and "div" not in q_ and "
by" not in q_ and "x" not in q_):
            l=q_.split()
            x_=[]
            y_=[]
            _x_=[]
            for x in q_:
               for y in x:
                  if y.isdigit():
                     x_.append(y)
            n=eval(''.join(x_))
            print("The Cube of",n,"is",math.pow(n,3))
            _v_="The Cube of",n,"is",math.pow(n,3)
            if v=="YES":
               s=gTTS(text="hmm "+str(_v_),lang='en-uk')
               s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
               playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
               os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif (("what" in q_ or "value" in q_) and "power" in q_)
and ("+" not in q_ and "-" not in q_ and "/" not in q_ and "*"
```

```
not in q_ and "plus" not in q_ and "add" not in q_ and "sub"
not in q_ and "minus" not in q_ and "multi" not in q_ and "
into" not in q_ and "div" not in q_ and "by" not in q_ and "x"
not in q_):
            l=q_.split()
            x_=[]
            y_=[]
            _x_=[]
            w=l.index("power")
            w_=''.join(l[w:])
            _w_=''.join(l[:w])
            for x in _w_:
               for y in x:
                  if y.isdigit():
                     x_.append(y)
            for x in w_:
               for y in x:
                  if y.isdigit():
                     y_.append(y)
            n=eval(''.join(x_))
            n_=eval(''.join(y_))
            print("The Value of",n,"Power",n_,"is",math.pow(n,
n_))
            _v_="The Value of",n,"Power",n_,"is",math.pow(n,n_)
            if v=="YES":
               s=gTTS(text="hmm "+str(_v_),lang='en-uk')
               s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
               playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
               os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
         elif (("what" in q_ or "value" in q_) and "sin" in q_) and
("+" not in q_ and "-" not in q_ and "/" not in q_ and "*" not in
```

```
q_ and "plus" not in q_ and "add" not in q_ and "sub" not in
q_ and "minus" not in q_ and "multi" not in q_ and "into" not
in q_ and "div" not in q_ and "by" not in q_ and "x" not in q_):
            l=q_.split()
            x_=[]
            y_=[]
            _x_=[]
            for x in q_:
               for y in x:
                  if y.isdigit():
                     x_.append(y)
            n=eval(''.join(x_))
            if "rad" not in q_:
               n_=math.radians(n)
            else:
               n_=n
            print("The Value of Sine",n_,"radians is",math.
sin(n_))
            _v_="The Value of Sine",n_,"radians is",math.sin(n_)
            if v=="YES":
               s=gTTS(text="hmm "+str(_v_),lang='en-uk')
               s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
               playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
               os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
         elif (("what" in q_ or "value" in q_) and "cosec" in q_)
and ("+" not in q_ and "-" not in q_ and "/" not in q_ and "*"
not in q_ and "plus" not in q_ and "add" not in q_ and "sub"
not in q_ and "minus" not in q_ and "multi" not in q_ and "
into" not in q_ and "div" not in q_ and "by" not in q_ and "x"
not in q_):
            l=q_.split()
```

```
            x_=[]
            y_=[]
            _x_=[]
            for x in q_:
                for y in x:
                    if y.isdigit():
                        x_.append(y)
            n=eval(''.join(x_))
            if "rad" not in q_:
                n_=math.radians(n)
            else:
                n_=n
            print("The Value of Cosecant",n_,"radians is",1/
math.sin(n_))
            _v_="The Value of Cosecant",n_,"radians is",1/math.
sin(n_)
            if v=="YES":
                s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif (("what" in q_ or "value" in q_) and "cos" in q_)
and ("+" not in q_ and "-" not in q_ and "/" not in q_ and "*"
not in q_ and "plus" not in q_ and "add" not in q_ and "sub"
not in q_ and "minus" not in q_ and "multi" not in q_ and "
into" not in q_ and "div" not in q_ and "by" not in q_ and "x"
not in q_):
            l=q_.split()
            x_=[]
            y_=[]
            _x_=[]
```

```
            for x in q_:
                for y in x:
                    if y.isdigit():
                        x_.append(y)
            n=eval(''.join(x_))
            if "rad" not in q_:
                n_=math.radians(n)
            else:
                n_=n
            print("The Value of Cosine",n_,"radians is",math.
cos(n_))
            _v_="The Value of Cosine",n_,"radians is",math.
cos(n_)
            if v=="YES":
                s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif (("what" in q_ or "value" in q_) and "tan" in q_)
and ("+" not in q_ and "-" not in q_ and "/" not in q_ and "*"
not in q_ and "plus" not in q_ and "add" not in q_ and "sub"
not in q_ and "minus" not in q_ and "multi" not in q_ and "
into" not in q_ and "div" not in q_ and "by" not in q_ and "x"
not in q_):
            l=q_.split()
            x_=[]
            y_=[]
            _x_=[]
            for x in q_:
                for y in x:
                    if y.isdigit():
```

```
                x_.append(y)
        n=eval(''.join(x_))
        if "rad" not in q_:
            n_=math.radians(n)
        else:
            n_=n
        print("The Value of Tan",n_,"radians is",math.tan(n_)
)
        _v_="The Value of Tan",n_,"radians is",math.tan(n_)
        if v=="YES":
            s=gTTS(text="hmm "+str(_v_),lang='en-uk')
            s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
            playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
            os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
    elif (("what" in q_ or "value" in q_) and "sec" in q_) and
("+" not in q_ and "-" not in q_ and "/" not in q_ and "*" not in
q_ and "plus" not in q_ and "add" not in q_ and "sub" not in
q_ and "minus" not in q_ and "multi" not in q_ and "into" not
in q_ and "div" not in q_ and "by" not in q_ and "x" not in q_):
        l=q_.split()
        x_=[]
        y_=[]
        _x_=[]
        for x in q_:
            for y in x:
                if y.isdigit():
                    x_.append(y)
        n=eval(''.join(x_))
        if "rad" not in q_:
            n_=math.radians(n)
        else:
```

```
            n_=n
        print("The Value of Secant",n_,"radians is",1/math.
cos(n_))
        _v_="The Value of Secant",n_,"radians is",1/math.
cos(n_)
        if v=="YES":
            s=gTTS(text="hmm "+str(_v_),lang='en-uk')
            s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
            playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
            os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
    elif (("what" in q_ or "value" in q_) and "cot" in q_) and
("+" not in q_ and "-" not in q_ and "/" not in q_ and "*" not in
q_ and "plus" not in q_ and "add" not in q_ and "sub" not in
q_ and "minus" not in q_ and "multi" not in q_ and "into" not
in q_ and "div" not in q_ and "by" not in q_ and "x" not in q_):
        l=q_.split()
        x_=[]
        y_=[]
        _x_=[]
        for x in q_:
            for y in x:
                if y.isdigit():
                    x_.append(y)
        n=eval(''.join(x_))
        if "rad" not in q_:
            n_=math.radians(n)
        else:
            n_=n
        print("The Value of Cotan",n_,"radians is",1/math.
tan(n_))
        _v_="The Value of Cotan",n_,"radians is",1/math.
```

```
tan(n_)
            if v=="YES":
                s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            elif (("conver" in q_ or "what" in q_) and "to" in q_)
and ("celsius" in q_ or "farenheit" in q_ or "kelvin" in q_):
                try:
                    l=q_.split()
                    x_=[]
                    if ("faren" in q_ and "cel" in q_) and "kel" not in q_:
                        for x in q_:
                            for y in x:
                                if y.isdigit():
                                    x_.append(y)
                        n=float(''.join(x_))
                        for i in l:
                            if "faren" in i:
                                j=l.index(i)
                            elif "cel" in i:
                                k=l.index(i)
                        if k>j:
                            p=(n-32)*(5/9)
                            print(p,"°Celsius")
                            _v_=p,"°Celsius"
                        elif j>k:
                            p=((n)*(9/5))+32
                            print(p,"°Farenheit")
                            _v_=p,"°Farenheit"
                        if v=="YES":
```

```
                    s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                elif ("kel" in q_ and "cel" in q_) and "faren" not in
q_:
                    for x in q_:
                        for y in x:
                            if y.isdigit():
                                x_.append(y)
                    n=float(''.join(x_))
                    for i in l:
                        if "kel" in i:
                            j=l.index(i)
                        elif "cel" in i:
                            k=l.index(i)
                    if k>j:
                        p=n-273.15
                        print(p,"°Celsius")
                        _v_=p,"°Celsius"
                    elif j>k:
                        p=n+273.15
                        print(p,"Kelvin")
                        _v_=(p,"Kelvin")
                    if v=="YES":
                        s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                        s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        os.remove("C:\Python
```

```
Files\Audiofiles\VoiceOver.mp3")
            elif ("kel" in q_ and "faren" in q_) and "cel" not in
q_:
                for x in q_:
                    for y in x:
                        if y.isdigit():
                            x_.append(y)
                n=float(''.join(x_))
                for i in l:
                    if "faren" in i:
                        j=l.index(i)
                    elif "kel" in i:
                        k=l.index(i)
                if k>j:
                    p=(n-32)*(5/9)
                    p_=p+273.15
                    print(p_,"Kelvin")
                    _v_=p,"Kelvin"
                elif j>k:
                    p=n-273.15
                    p_=((p)*(9/5))+32
                    print(p_,"°Farenheit")
                    _v_=p,"°Farenheit"
                if v=="YES":
                    s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            else:
                try:
                    try:
```

```
                    l= q_.split()
                    try:
                        l.remove("sid")
                        w=' '.join(l)
                    except:
                        l.remove("glenda")
                        w=' '.join(l)
                except:
                    w=q_
                print("Just Sit Back And Hold Tight!")
                webbrowser.open("https://google.com/
search?q={}".format(w))
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Right')
                time.sleep(1)
            except requests.exceptions.ConnectionError:
                print(" \t⚠Oops...Make Sure That You Are
Connected With Internet⚠ ")
                if v == "YES":
                    playsound(r"C:\Python
Files\Audiofiles\offline.mp3")
        except:
            try:
                try:
                    l= q_.split()
                    try:
                        l.remove("sid")
                        w=' '.join(l)
                    except:
                        l.remove("glenda")
```

```
                    w=' '.join(l)
                except:
                    w=q_
                print("Just Sit Back And Hold Tight!")
                webbrowser.open("https://google.com/
search?q={}".format(w))
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Right')
                time.sleep(1)
            except requests.exceptions.ConnectionError:
                print(" \t⚠Oops...Make Sure That You Are
Connected With Internet⚠ ")
                if v == "YES":
                    playsound(r"C:\Python
Files\Audiofiles\offline.mp3")
        elif (("conver" in q_ or "what" in q_) and "to" in q_)
and ("binary" in q_ or "octal" in q_ or "hexa" in q_ or "
decimal" in q_):
            try:
                l=q_.split()
                x_=[]
                y_=[]
                if ("bin" in q_ and "oct" in q_) and ("dec" not in q_
and "hex" not in q_):
                    for x in q_:
                        for y in x:
                            if y.isdigit():
                                x_.append(y)
                    n=int(''.join(x_))
                    for i in l:
```

```
                    if "bin" in i:
                        j=l.index(i)
                    elif "oct" in i:
                        k=l.index(i)
                if k>j:
                    p=oct(int(str(n), 2))
                    print("Octal Value:",p[2:])
                    _v_="Octal Value:",p[2:]
                elif j>k:
                    p=bin(int(str(n), 8))
                    print("Binary Value:",p[2:])
                    _v_="Binary Value:",p[2:]
                if v=="YES":
                    s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            elif ("oct" in q_ and "hex" in q_) and (" dec" not in
q_ and "bin" not in q_):
                for x in q_:
                    for y in x:
                        y_.append(y)
                        if y.isdigit():
                            x_.append(y)
                h=(y_.index(x_[0]))+len(x_)
                if y_[h]=='a' or y_[h]=='b' or y_[h]=='c' or
y_[h]=='d' or y_[h]=='e' or y_[h]=='f':
                    x_.append(y_[h])
                n=''.join(x_)
                for i in l:
                    if "oct" in i:
```

```
                    j=l.index(i)
                elif "hex" in i:
                    k=l.index(i)
            if k>j:
                p=hex(int(str(n), 8))
                print("Hexadecimal Value:",p[2:])
                _v_="Hexadecimal Value:",p[2:]
            elif j>k:
                p=oct(int(str(n), 16))
                print("Octal Value:",p[2:])
                _v_="Octal Value:",p[2:]
            if v=="YES":
                s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif ("bin" in q_ and "hex" in q_) and (" dec" not in
q_ and "oct" not in q_):
            for x in q_:
                for y in x:
                    y_.append(y)
                    if y.isdigit():
                        x_.append(y)
            h=(y_.index(x_[0]))+len(x_)
            if y_[h]=='a' or y_[h]=='b' or y_[h]=='c' or
y_[h]=='d' or y_[h]=='e' or y_[h]=='f':
                x_.append(y_[h])
            n=''.join(x_)
            for i in l:
                if "bin" in i:
                    j=l.index(i)
```

```
                elif "hex" in i:
                    k=l.index(i)
            if k>j:
                p=hex(int(str(n), 2))
                print("Hexadecimal Value:",p[2:])
                _v_="Hexadecimal Value:",p[2:]
            elif j>k:
                p=bin(int(str(n), 16))
                print("Binary Value:",p[2:])
                _v_="Binary Value:",p[2:]
            if v=="YES":
                s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif (("bin" in q_ and "dec" in q_) or ("bin" in q_
and "dec" not in q_)) and ("oct" not in q_ and "hex" not in q_):
            for x in q_:
                for y in x:
                    if y.isdigit():
                        x_.append(y)
            n=int(''.join(x_))
            if ("bin" in q_ and "dec" in q_):
                for i in l:
                    if "bin" in i:
                        j=l.index(i)
                    elif "dec" in i:
                        k=l.index(i)
                if k>j:
                    p=int(str(n), 2)
                    print("Decimal Value:",p)
```

```
                    _v_="Decimal Value:",p
                elif j>k:
                    p=bin(int(n))
                    print("Binary Value:",p[2:])
                    _v_="Binary Value:",p[2:]
                if v=="YES":
                    s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            else:
                p=bin(int(n))
                print("Binary Value:",p[2:])
                _v_="Binary Value:",p[2:]
                if v=="YES":
                    s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif (("dec" in q_ and "hex" in q_) or ("dec" not in
q_ and "hex" in q_)) and ("bin" not in q_ and "oct" not in q_):
            for x in q_:
                for y in x:
                    y_.append(y)
                    if y.isdigit():
                        x_.append(y)
            h=(y_.index(x_[0]))+len(x_)
            if y_[h]=='a' or y_[h]=='b' or y_[h]=='c' or
```

```
y_[h]=='d' or y_[h]=='e' or y_[h]=='f':
                        x_.append(y_[h])
                    n=''.join(x_)
                    if "dec" in q_ and "hex" in q_:
                        for i in l:
                            if " dec" in i:
                                j=l.index(i)
                            elif "hex" in i:
                                k=l.index(i)
                        if k>j:
                            p=hex(int(n))
                            print("Hexadecimal Value:",p[2:])
                            _v_="Hexadecimal Value:",p[2:]
                        elif j>k:
                            p=int(str(n), 16)
                            print("Decimal Value:",p)
                            _v_="Decimal Value:",p
                        if v=="YES":
                            s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                            s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                            playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                            os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    else:
                        p=hex(int(n))
                        print("Hexadecimal Value:",p[2:])
                        _v_="Hexadecimal Value:",p[2:]
                        if v=="YES":
                            s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                            s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                            playsound(r"C:\Python
```

```
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            elif (("dec" in q_ and "oct" in q_) or ("oct" in q_
and "dec" not in q_)) and ("bin" not in q_ and "hex" not in q_)
:
                for x in q_:
                    for y in x:
                        if y.isdigit():
                            x_.append(y)
                n=int(''.join(x_))
                if "dec" in q_ and "oct" in q_:
                    for i in l:
                        if "dec" in i:
                            j=l.index(i)
                        elif "oct" in i:
                            k=l.index(i)
                    if k>j:
                        p=oct(int(n))
                        print("Octal Value:",p[2:])
                        _v_="Octal Value:",p[2:]
                    elif j>k:
                        p=int(str(n), 8)
                        print("Decimal Value:",p)
                        _v_="Decimal Value:",p
                    if v=="YES":
                        s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                        s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                else:
```

```
                    p=oct(int(n))
                    print("Octal Value:",p[2:])
                    _v_="Octal Value:",p[2:]
                    if v=="YES":
                        s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                        s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                elif ("oct" in q_ and "hex" in q_) and (" dec" not in
q_ and "bin" not in q_):
                    for x in q_:
                        for y in x:
                            y_.append(y)
                            if y.isdigit():
                                x_.append(y)
                    h=(y_.index(x_[0]))+len(x_)
                    if y_[h]=='a' or y_[h]=='b' or y_[h]=='c' or
y_[h]=='d' or y_[h]=='e' or y_[h]=='f':
                        x_.append(y_[h])
                    n=''.join(x_)
                    for i in l:
                        if "oct" in i:
                            j=l.index(i)
                        elif "hex" in i:
                            k=l.index(i)
                    if k>j:
                        p=hex(int(str(n), 8))
                        print("Hexadecimal Value:",p[2:])
                        _v_="Hexadecimal Value:",p[2:]
                    elif j>k:
                        p=oct(int(str(n), 16))
```

```
                    print("Octal Value:",p[2:])
                    _v_="Octal Value:",p[2:]
                if v=="YES":
                    s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            else:
                try:
                    try:
                        l= q_.split()
                        try:
                            l.remove("sid")
                            w=' '.join(l)
                        except:
                            l.remove("glenda")
                            w=' '.join(l)
                    except:
                        w=q_
                    print("Just Sit Back And Hold Tight!")
                    webbrowser.open("https://google.com/
search?q={}".format(w))
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
                    keyboard.press_and_release('win + Right')
                    time.sleep(1)
                except requests.exceptions.ConnectionError:
                    print(" \t⚠Oops...Make Sure That You Are
```

```
Connected With Internet⚠ ")
                    if v == "YES":
                        playsound(r"C:\Python
Files\Audiofiles\offline.mp3")
                except:
                    try:
                        try:
                            l= q_.split()
                            try:
                                l.remove("sid")
                                w=' '.join(l)
                            except:
                                l.remove("glenda")
                                w=' '.join(l)
                        except:
                            w=q_
                        print("Just Sit Back And Hold Tight!")
                        webbrowser.open("https://google.com/
search?q={}".format(w))
                        time.sleep(1)
                        keyboard.press_and_release('win + Up')
                        time.sleep(1)
                        keyboard.press_and_release('win + Up')
                        time.sleep(1)
                        keyboard.press_and_release('win + Right')
                        time.sleep(1)
                    except requests.exceptions.ConnectionError:
                        print(" \t⚠Oops...Make Sure That You Are
Connected With Internet⚠ ")
                        if v == "YES":
                            playsound(r"C:\Python
Files\Audiofiles\offline.mp3")
            elif (("conver" in q_ or "what" in q_) and "to" in q_)
and ("sec" in q_ or "min" in q_ or "h" in q_ or "day" in q_ or "
```

```
week" in q_ or "month" in q_ or "year" in q_):
            try:
                l=q_.split()
                x_=[]
                if ("min" in q_ and "sec" in q_) and (" h" not in q_
and "day" not in q_ and "week" not in q_ and "month" not in
q_ and "year" not in q_):
                    for x in q_:
                        for y in x:
                            if y.isdigit():
                                x_.append(y)
                    n=int(''.join(x_))
                    for i in l:
                        if "min" in i:
                            j=l.index(i)
                        elif "sec" in i:
                            k=l.index(i)
                    if k>j:
                        p=n*60
                        print(p,"Seconds")
                        _v_="It is",p,"Seconds"
                    elif j>k:
                        p=n/60
                        print(p,"Minutes")
                        _v_="It is",p,"Minutes"
                    if v=="YES":
                        s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                        s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                elif ("h" in q_ and "sec" in q_) and ("min" not in q_
```

```
and "day" not in q_ and "week" not in q_ and "month" not in
q_ and "year" not in q_):
                    for x in q_:
                        for y in x:
                            if y.isdigit():
                                x_.append(y)
                    n=int(''.join(x_))
                    for i in l:
                        if "h" in i:
                            j=l.index(i)
                        elif "sec" in i:
                            k=l.index(i)
                    if k>j:
                        p=n*60*60
                        print(p,"Seconds")
                        _v_="It is",p,"Seconds"
                    elif j>k:
                        p=n/(60*60)
                        if n%(60*60)!=0:
                            if p>1:
                                p_=math.ceil(p)-1
                                d=round((p-p_)*(60))
                            else:
                                p_=p
                                d=round((p_)*(60))
                            print(p_,"Hours &",d,"Minutes, But to be
Accurate",p,"Hours")
                                _v_="It is",p_,"Hours &",d,"Minutes, But to
be Accurate",p,"Hours"
                        else:
                            print(int(p),"Hours")
                            _v_="It is",int(p),"Hours"
                    if v=="YES":
                        s=gTTS(text="hmm "+str(_v_),lang='en-uk')
```

```
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                elif ("day" in q_ and "sec" in q_) and ("min" not in
q_ and " h" not in q_ and "week" not in q_ and "month" not in
q_ and "year" not in q_):
                    for x in q_:
                        for y in x:
                            if y.isdigit():
                                x_.append(y)
                    n=int(''.join(x_))
                    for i in l:
                        if "day" in i:
                            j=l.index(i)
                        elif "sec" in i:
                            k=l.index(i)
                    if k>j:
                        p=n*60*60*24
                        print(p,"Seconds")
                        _v_="It is",p,"Seconds"
                    elif j>k:
                        p=n/(60*60*24)
                        if n%(60*60*24)!=0:
                            if p>1:
                                p_=math.ceil(p)-1
                                d=round((p-p_)*(24))
                            else:
                                p_=p
                                d=round((p_)*(24))
                            print(p_,"Days &",d,"Hours, But to be
Accurate",p,"Days")
```

```
                    _v_="It is",p_,"Days &",d,"Hours, But to be
Accurate",p,"Days"
                else:
                    print(int(p),"Days")
                    _v_="It is",int(p),"Days"
            if v=="YES":
                s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            elif ("month" in q_ and "sec" in q_) and ("min" not
in q_ and " h" not in q_ and "day" not in q_ and "week" not in
q_ and "year" not in q_):
            for x in q_:
                for y in x:
                    if y.isdigit():
                        x_.append(y)
            n=int(''.join(x_))
            for i in l:
                if "month" in i:
                    j=l.index(i)
                elif "sec" in i:
                    k=l.index(i)
            if k>j:
                p=n*60*60*24*30
                print(p,"Seconds")
                _v_="It is",p,"Seconds"
            elif j>k:
                p=n/(60*60*24*30)
                print(p)
                if n%(60*60*24*30)!=0:
```

```
                    if p>1:
                        p_=math.ceil(p)-1
                        d=round((p-p_)*(30))
                    else:
                        p_=p
                        d=round(p_*(30))
                    print(d)
                    print(p_,"Months &",d,"Days, But to be
Accurate",p,"Months")
                    _v_="It is",p_,"Months &",d,"Days, But to be
Accurate",p,"Months"
                else:
                    print(int(p),"Months")
                    _v_="It is",int(p),"Months"
            if v=="YES":
                s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif ("year" in q_ and "sec" in q_) and ("min" not in
q_ and " h" not in q_ and "day" not in q_ and "week" not in q_
and "month" not in q_):
            for x in q_:
                for y in x:
                    if y.isdigit():
                        x_.append(y)
            n=int(''.join(x_))
            for i in l:
                if "year" in i:
                    j=l.index(i)
                elif "sec" in i:
```

```
                k=l.index(i)
            if k>j:
                p=n*60*60*24*30*365
                print(p,"Seconds")
                _v_="It is",p,"Seconds"
            elif j>k:
                p=n/(60*60*24*30*365)
                if n%(60*60*24*30*365)!=0:
                    if p>1:
                        p_=math.ceil(p)-1
                        d=round((p-p_)*(365))
                    else:
                        p_=p
                        d=round(p_*(365))
                    print(d)
                    print(p_,"Years &",d,"Months, But to be
Accurate",p,"Years")
                    _v_="It is",p_,"Years &",d,"Months, But to be
Accurate",p,"Years"
                else:
                    print(int(p),"Years")
                    _v_="It is",int(p),"Years"
            if v=="YES":
                s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif ("min" in q_ and "h" in q_) and ("sec" not in q_
and "day" not in q_ and "week" not in q_ and "month" not in
q_ and "year" not in q_):
            for x in q_:
```

```
                    for y in x:
                        if y.isdigit():
                            x_.append(y)
                    n=int(''.join(x_))
                    for i in l:
                        if " h" in i:
                            j=l.index(i)
                        elif "min" in i:
                            k=l.index(i)
                    if k>j:
                        p=n*60
                        print(p,"Minute")
                        _v_="It is",p,"Minute"
                    elif j>k:
                        p=n/60
                        print(p,"Hours")
                        _v_="It is",p,"Hours"
                    if v=="YES":
                        s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                        s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                elif ("day" in q_ and "min" in q_) and ("sec" not in
q_ and " h" not in q_ and "week" not in q_ and "month" not in
q_ and "year" not in q_):
                    for x in q_:
                        for y in x:
                            if y.isdigit():
                                x_.append(y)
                    n=int(''.join(x_))
                    for i in l:
```

```
                    if "day" in i:
                        j=l.index(i)
                    elif "min" in i:
                        k=l.index(i)
                if k>j:
                    p=n*60*24
                    print(p,"Minutes")
                    _v_="It is",p,"Minutes"
                elif j>k:
                    p=n/(60*24)
                    if n%(60*24)!=0:
                        if p>1:
                            p_=math.ceil(p)-1
                            d=round((p-p_)*(24))
                        else:
                            p_=p
                            d=round((p_)*(24))
                        print(p_,"Days &",d,"Hours, But to be
Accurate",p,"Days")
                        _v_="It is",p_,"Days &",d,"Hours, But to be
Accurate",p,"Days"
                    else:
                        print(int(p),"Days")
                        _v_="It is",int(p),"Days"
                if v=="YES":
                    s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            elif ("month" in q_ and "min" in q_) and ("sec" not
in q_ and " h" not in q_ and "day" not in q_ and "week" not in
```

```
q_ and "year" not in q_):
                    for x in q_:
                       for y in x:
                          if y.isdigit():
                             x_.append(y)
                    n=int(''.join(x_))
                    for i in l:
                       if "month" in i:
                          j=l.index(i)
                       elif "min" in i:
                          k=l.index(i)
                    if k>j:
                       p=n*60*24*30
                       print(p,"Minutes")
                       _v_="It is",p,"Minutes"
                    elif j>k:
                       p=n/(60*24*30)
                       print(p)
                       if n%(60*24*30)!=0:
                          if p>1:
                             p_=math.ceil(p)-1
                             d=round((p-p_)*(30))
                          else:
                             p_=p
                             d=round(p_*(30))
                          print(p_,"Months &",d,"Days, But to be
Accurate",p,"Months")
                          _v_="It is",p_,"Months &",d,"Days, But to be
Accurate",p,"Months"
                       else:
                          print(int(p),"Months")
                          _v_="It is",int(p),"Months"
                    if v=="YES":
                       s=gTTS(text="hmm "+str(_v_),lang='en-uk')
```

```
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                elif ("year" in q_ and "min" in q_) and ("sec" not in
q_ and " h" not in q_ and "day" not in q_ and "week" not in q_
and "month" not in q_):
                    for x in q_:
                        for y in x:
                            if y.isdigit():
                                x_.append(y)
                    n=int(''.join(x_))
                    for i in l:
                        if "year" in i:
                            j=l.index(i)
                        elif "min" in i:
                            k=l.index(i)
                    if k>j:
                        p=n*60*24*30*365
                        print(p,"Minute")
                        _v_="It is",int(p),"Months"
                    elif j>k:
                        p=n/(60*24*30*365)
                        if n%(60*24*30*365)!=0:
                            if p>1:
                                p_=math.ceil(p)-1
                                d=round((p-p_)*(365))
                            else:
                                p_=p
                                d=round(p_*(365))
                            print(d)
                            print(p_,"Years &",d,"Months, But to be
```

```
Accurate",p,"Years")
                    _v_="It is",p_,"Years &",d,"Months, But to be
Accurate",p,"Years"
                else:
                    print(int(p,"Years"))
                    _v_="It is",int(p,"Years")
            if v=="YES":
                s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            elif ("h" in q_ and "day" in q_) and ("sec" not in q_
and "min" not in q_ and "week" not in q_ and "month" not in
q_ and "year" not in q_):
                for x in q_:
                    for y in x:
                        if y.isdigit():
                            x_.append(y)
                n=int(''.join(x_))
                for i in l:
                    if "day" in i:
                        j=l.index(i)
                    elif " h" in i:
                        k=l.index(i)
                if k>j:
                    p=n*24
                    print(p,"Days")
                    _v_="It is",p,"Days"
                elif j>k:
                    p=n/24
                    print(p,"Hours")
```

```
                    _v_="It is",p,"Hours"
                if v=="YES":
                    s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            elif ("month" in q_ and "h" in q_) and ("sec" not in
q_ and "min" not in q_ and "day" not in q_ and "week" not in
q_ and "year" not in q_):
                for x in q_:
                    for y in x:
                        if y.isdigit():
                            x_.append(y)
                n=int(''.join(x_))
                for i in l:
                    if "month" in i:
                        j=l.index(i)
                    elif "hr" in i or "ho" in i:
                        k=l.index(i)
                if k>j:
                    p=n*24*30
                    print(p,"Hours")
                    _v_="It is",p,"Hours"
                elif j>k:
                    p=n/(24*30)
                    if n%(24*30)!=0:
                        if p>1:
                            p_=math.ceil(p)-1
                            d=round((p-p_)*(30))
                        else:
                            p_=p
```

```
                    d=round(p_*(30))
                print(p_,"Months &",d,"Days, But to be
Accurate",p,"Months")
                _v_="It is",p_,"Months &",d,"Days, But to be
Accurate",p,"Months"
            else:
                print(int(p),"Months")
                _v_="It is",int(p),"Months"
        if v=="YES":
            s=gTTS(text="hmm "+str(_v_),lang='en-uk')
            s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
            playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
            os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif ("year" in q_ and "day" in q_) and ("sec" not in
q_ and "h" not in q_ and "min" not in q_ and "week" not in q_
and "month" not in q_):
            for x in q_:
                for y in x:
                    if y.isdigit():
                        x_.append(y)
            n=int(''.join(x_))
            for i in l:
                if "year" in i:
                    j=l.index(i)
                elif "day" in i:
                    k=l.index(i)
            if k>j:
                p=n*24*30*365
                print(p,"Days")
                _v_="It is",
            elif j>k:
```

```
                    p=n/(24*30*365)
                    if n%(24*30*365)!=0:
                        if p>1:
                            p_=math.ceil(p)-1
                            d=round((p-p_)*(365))
                        else:
                            p_=p
                            d=round(p_*(365))
                        print(p_,"Years &",d,"Months, But to be
Accurate",p,"Years")
                            _v_="It is",p_,"Years &",d,"Months, But to be
Accurate",p,"Years"
                    else:
                        print(int(p),"Years")
                        _v_="It is",int(p),"Years"
                if v=="YES":
                    s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            elif ("year" in q_ and "h" in q_) and ("sec" not in q_
and "min" not in q_ and "day" not in q_ and "week" not in q_
and "month" not in q_):
                for x in q_:
                    for y in x:
                        if y.isdigit():
                            x_.append(y)
                n=int(''.join(x_))
                for i in l:
                    if "year" in i:
                        j=l.index(i)
```

```
                elif "h" in i:
                    k=l.index(i)
            if k>j:
                p=n*60*24*30*365
                print(p,"Hours")
                _v_="It is",p,"Hours"
            elif j>k:
                p=n/(60*24*30*365)
                if n%(60*24*30*365)!=0:
                    if p>1:
                        p_=math.ceil(p)-1
                        d=round((p-p_)*(365))
                    else:
                        p_=p
                        d=round(p_*(365))
                    print(d)
                    print(p_,"Years &",d,"Months, But to be
Accurate",p,"Years")
                    _v_="It is",p_,"Years &",d,"Months, But to be
Accurate",p,"Years"
                else:
                    print(int(p),"Years")
                    _v_="It is",int(p),"Years"
            if v=="YES":
                s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif ("sec" in q_ and "week" in q_) and ("min" not
in q_ and "h" not in q_ and "day" not in q_ and "month" not in
q_ and "year" not in q_):
```

```
                for x in q_:
                    for y in x:
                        if y.isdigit():
                            x_.append(y)
                n=int(''.join(x_))
                for i in l:
                    if "week" in i:
                        j=l.index(i)
                    elif "sec" in i:
                        k=l.index(i)
                if k>j:
                    p=n*7*24*60*60
                    print(p,"Seconds")
                    _v_="It is",p,"Seconds"
                elif j>k:
                    p=n/(7*24*60*60)
                    if n%(7*24*60*60)!=0:
                        if p>1:
                            p_=math.ceil(p)-1
                            d=round((p-p_)*7)
                        else:
                            p_=p
                            d=round((p_)*7)
                        print(p_,"Weeks &",d,"Days, But to be
Accurate",p,"Weeks")
                        _v_="It is",p_,"Weeks &",d,"Days, But to be
Accurate",p,"Weeks"
                    else:
                        print(int(p),"Weeks")
                        _v_="It is",int(p),"Weeks"
                if v=="YES":
                    s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
```

```
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                elif ("min" in q_ and "week" in q_) and ("sec" not
in q_ and "h" not in q_ and "day" not in q_ and "month" not in
q_ and "year" not in q_):
                    for x in q_:
                        for y in x:
                            if y.isdigit():
                                x_.append(y)
                    n=int(''.join(x_))
                    for i in l:
                        if "week" in i:
                            j=l.index(i)
                        elif "min" in i:
                            k=l.index(i)
                    if k>j:
                        p=n*7*24*60
                        print(p,"Minutes")
                        _v_="It is",p,"Minutes"
                    elif j>k:
                        p=n/(7*24*60)
                        if n%(7*24*60)!=0:
                            if p>1:
                                p_=math.ceil(p)-1
                                d=round((p-p_)*7)
                            else:
                                p_=p
                                d=round((p_)*7)
                            print(p_,"Weeks &",d,"Days, But to be
Accurate",p,"Weeks")
                            _v_="It is",p_,"Weeks &",d,"Days, But to be
Accurate",p,"Weeks"
```

```
                    else:
                        print(int(p),"Weeks")
                        _v_="It is",int(p),"Weeks"
                if v=="YES":
                    s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            elif ("h" in q_ and "week" in q_) and ("sec" not in
q_ and "min" not in q_ and "day" not in q_ and "month" not
in q_ and "year" not in q_):
                for x in q_:
                    for y in x:
                        if y.isdigit():
                            x_.append(y)
                n=int(''.join(x_))
                for i in l:
                    if "week" in i:
                        j=l.index(i)
                    elif "h" in i:
                        k=l.index(i)
                if k>j:
                    p=n*7*24
                    print(p,"Hours")
                    _v_="It is",p,"Hours"
                elif j>k:
                    p=n/(7*24)
                    if n%(7*24)!=0:
                        if p>1:
                            p_=math.ceil(p)-1
                            d=round((p-p_)*7)
```

```
                    else:
                        p_=p
                        d=round((p_)*7)
                    print(p_,"Weeks &",d,"Days, But to be
Accurate",p,"Weeks")
                    _v_="It is",p_,"Weeks &",d,"Days, But to be
Accurate",p,"Weeks"
                else:
                    print(int(p,"Weeks"))
                    _v_="It is",int(p,"Weeks")
            if v=="YES":
                s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
        elif ("day" in q_ and "week" in q_) and ("sec" not
in q_ and "min" not in q_ and " h" not in q_ and "month" not
in q_ and "year" not in q_):
            for x in q_:
                for y in x:
                    if y.isdigit():
                        x_.append(y)
            n=int(''.join(x_))
            for i in l:
                if "week" in i:
                    j=l.index(i)
                elif "day" in i:
                    k=l.index(i)
            if k>j:
                p=n*7
                print(p,"Days")
```

```
                    _v_="It is",p,"Days"
                elif j>k:
                    p=n/7
                    if n%7!=0:
                        if p>1:
                            p_=math.ceil(p)-1
                            d=round((p-p_)*7)
                        else:
                            p_=p
                            d=round((p_)*7)
                        print(p_,"Weeks &",d,"Days, But to be
Accurate",p,"Weeks")
                        _v_="It is",p_,"Weeks &",d,"Days, But to be
Accurate",p,"Weeks"
                    else:
                        print(int(p),"Weeks")
                        _v_="It is",int(p),"Weeks"
                if v=="YES":
                    s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            elif ("week" in q_ and "month" in q_) and ("sec"
not in q_ and "min" not in q_ and " h" not in q_ and "day" not
in q_ and "year" not in q_):
                for x in q_:
                    for y in x:
                        if y.isdigit():
                            x_.append(y)
                n=int(''.join(x_))
                for i in l:
```

```
                    if "month" in i:
                        j=l.index(i)
                    elif "week" in i:
                        k=l.index(i)
                if k>j:
                    p=n*4
                    print(p,"Week")
                    _v_="It is",p,"Week"
                elif j>k:
                    p=n/4
                    if n%4!=0:
                        if p>1:
                            p_=math.ceil(p)-1
                            d=round((p-p_)*4)
                        else:
                            p_=p
                            d=round((p_)*4)
                        print(d)
                        print(p_,"Months &",d,"Weeks, But to be
Accurate",p,"Months")
                        _v_="It is","Months &",d,"Weeks, But to be
Accurate",p,"Months"
                    else:
                        print(int(p),"Months")
                        _v_="It is",int(p),"Months"
                if v=="YES":
                    s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            elif ("week" in q_ and "year" in q_) and ("sec" not
```

```
in q_ and "min" not in q_ and " h" not in q_ and "day" not in
q_ and "month" not in q_):
                    for x in q_:
                        for y in x:
                            if y.isdigit():
                                x_.append(y)
                    n=int(''.join(x_))
                    for i in l:
                        if "year" in i:
                            j=l.index(i)
                        elif "week" in i:
                            k=l.index(i)
                    if k>j:
                        p=n*4*12
                        print(p,"Weeks")
                        _v_="It is",p,"Weeks"
                    elif j>k:
                        p=n/(4*12)
                        if n%(4*12)!=0:
                            if p>1:
                                p_=math.ceil(p)-1
                                d=round((p-p_)*12)
                            else:
                                p_=p
                                d=round((p_)*12)
                            print(p_,"Years &",d,"Months, But to be
Accurate",p,"Years")
                            _v_="It is",p_,"Years &",d,"Months, But to be
Accurate",p,"Years"
                        else:
                            print(int(p),"Years")
                            _v_="It is",int(p),"Years"
                    if v=="YES":
                        s=gTTS(text="hmm "+str(_v_),lang='en-uk')
```

```
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                elif ("month" in q_ and "year" in q_) and ("sec" not
in q_ and "min" not in q_ and " h" not in q_ and "day" not in
q_ and "week" not in q_):
                    for x in q_:
                        for y in x:
                            if y.isdigit():
                                x_.append(y)
                    n=int(''.join(x_))
                    for i in l:
                        if "year" in i:
                            j=l.index(i)
                        elif "month" in i:
                            k=l.index(i)
                    if k>j:
                        p=n*12
                        print(p,"Months")
                        _v_="It is",p,"Months"
                    elif j>k:
                        p=n/12
                        if n%12!=0.0:
                            if p>1:
                                p_=math.ceil(p)-1
                                d=round((p-p_)*12)
                            else:
                                p_=p
                                d=round((p_)*12)
                            print(p_,"Years &",d,"Months, But to be
Accurate",p,"Years")
```

```
                    _v_="It is",p_,"Years &",d,"Months, But to be
Accurate",p,"Years"
                else:
                    print(int(p),"Years")
                    _v_="It is",int(p),"Years"
            if v=="YES":
                s=gTTS(text="hmm "+str(_v_),lang='en-uk')
                s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            else:
                try:
                    try:
                        l= q_.split()
                        try:
                            l.remove("sid")
                            w=' '.join(l)
                        except:
                            l.remove("glenda")
                            w=' '.join(l)
                    except:
                        w=q_
                    print("Just Sit Back And Hold Tight!")
                    webbrowser.open("https://google.com/
search?q={}".format(w))
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
                    keyboard.press_and_release('win + Right')
```

```
                time.sleep(1)
            except requests.exceptions.ConnectionError:
                print(" \t⚠Oops...Make Sure That You Are
Connected With Internet⚠ ")
                if v == "YES":
                    playsound(r"C:\Python
Files\Audiofiles\offline.mp3")
        except:
            try:
                try:
                    l= q_.split()
                    try:
                        l.remove("sid")
                        w=' '.join(l)
                    except:
                        l.remove("glenda")
                        w=' '.join(l)
                except:
                    w=q_
                print("Just Sit Back And Hold Tight!")
                webbrowser.open("https://google.com/
search?q={}".format(w))
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Right')
                time.sleep(1)
            except requests.exceptions.ConnectionError:
                print(" \t⚠Oops...Make Sure That You Are
Connected With Internet⚠ ")
                if v == "YES":
                    playsound(r"C:\Python
```

```
Files\Audiofiles\offline.mp3")
        elif "picture" in q_ or "image" in q_ or ("show me" in
q_ and "in " not in q_):
            if "sid" in q_ or "glenda" in q_:
                l= q_.split()
                try:
                    l.remove("sid")
                    w=' '.join(l)
                except:
                    l.remove("glenda")
                    w=' '.join(l)
            else:
                w=q_
            webbrowser.open("https://google.com/
search?q={} pictures".format(w))
            time.sleep(1)
            keyboard.press_and_release('win + Up')
            time.sleep(1)
            keyboard.press_and_release('win + Up')
            time.sleep(1)
            keyboard.press_and_release('win + Right')
            time.sleep(1)
            print("Here's the Pics...")
            try:
                if v == "YES":
                    s=gTTS(text="hmm these are the pictures from
the web",lang='en-uk')
                    s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except:
```

```
            pass

        elif (("summarize" in q_ or "summary" in q_ or "short"
in q_ or "small" in q_ or "info" in q_) and ("about" in q_ or "
on" in q_)) or (("tell" in q_ or "show" in q_ or "about" in q_ or
"on" in q_) and ("short" in q_ or "small" in q_)):
            try:
                if "sid" in q_ or "glenda" in q_:
                    l= q_.split()
                    try:
                        l.remove("sid")
                        w=' '.join(l)
                    except:
                        l.remove("glenda")
                        w=' '.join(l)
                else:
                    w=q_
                r = requests.get("https://google.com/search?q={}
wikipedia".format(w))
                s = BeautifulSoup(r.content, 'html.parser')
                e=[]
                for a_ in s.find_all('a', href=True):
                    e.append(a_['href'])
                x=[]
                for t in e:
                    if "/url?q=https://en.wikipedia.org/wiki/" in t:
                        x.append(e.index(t))
                l=e[x[0]]
                i=l.index('&')
                q=l[37:i]
                if '%' in q:
                    print("Just Sit Back And Hold Tight!")
                    webbrowser.open("https://google.com/
search?q={}".format(w))
```

```
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Right')
                time.sleep(1)
            else:
                print("\t                    \||~~~~||||",q,"
||||~~~~||/ \n")
                r=requests.get("https://en.wikipedia.org/wiki/
{}".format(q))
                s = BeautifulSoup(r.text,"html.parser")
                y=[]
                try:
                    i_=s.find(class_='infobox')
                    try:
                        for __l__ in i_.findAll('tr'):
                            y.append(__l__.text)
                            try:
                                g=''
                                g_=[]
                                for h in (__l__.text):
                                    if h.isupper() or h.isdigit():
                                        if h.isdigit():
                                            g_.append(((__l__.text).index(h)))
                                            if ((__l__.text).index(h))==g_[0]:
                                                g=g+' '+h
                                            else:
                                                g=g+h
                                        elif h.isupper():
                                            g=g+' '+h
                                        else:
                                            g=g+h
```

```
                        else:
                            g=g+h
                    print("∴",g)
                except:
                    print("₪",__l__.text)
            except:
                try:
                    for __l__ in i_.find():
                        print("₪",__l__.text)
                        y.append(__l__.text)
                except:
                    for __l__ in i_.findAll('td'):
                        print("₪",__l__.text)
                        y.append(__l__.text)
        except:
            try:
                i_=s.find(class_='infobox vcard')
                try:
                    for __l__ in i_.findAll('tr'):
                        y.append(__l__.text)
                        try:
                            g=''
                            g_=[]
                            for h in (__l__.text):
                                if h.isupper() or h.isdigit():
                                    if h.isdigit():
                                        g_.append(((__l__.text).index(h)
))
                                        if ((__l__.text).index(h))==g_[0]:
                                            g=g+' '+h
                                        else:
                                            g=g+h
                                    elif h.isupper():
                                        g=g+' '+h
```

```
                                    else:
                                        g=g+h
                                else:
                                    g=g+h
                            print("·.",g)
                        except:
                            print("₪",__l__.text)
                    except:
                        try:
                            for __l__ in i_.find():
                                print("₪",__l__.text)
                                y.append(__l__.text)
                        except:
                            for __l__ in i_.findAll('td'):
                                print("₪",__l__.text)
                                y.append(__l__.text)
                except:
                    try:
                        i_=s.find(class_='infobox vevent')
                        try:
                            for __l__ in i_.findAll('tr'):
                                y.append(__l__.text)
                                try:
                                    g=''
                                    g_=[]
                                    for h in (__l__.text):
                                        if h.isupper() or h.isdigit():
                                            if h.isdigit():
                                                g_.append(((__l__.text).
index(h)))
                                                if ((__l__.text).index(h))
==g_[0]:
                                                    g=g+' '+h
                                                else:
```

```
                                    g=g+h
                                elif h.isupper():
                                    g=g+' '+h
                                else:
                                    g=g+h
                            else:
                                g=g+h
                        print("∴",g)
                    except:
                        print("₪",__l__.text)
                except:
                    try:
                        for __l__ in i_.find():
                            print("₪",__l__.text)
                            y.append(__l__.text)
                    except:
                        for __l__ in i_.findAll('td'):
                            print("₪",__l__.text)
                            y.append(__l__.text)
            except:
                try:
                    i_=s.find(class_='infobox geography
vcard')
                    try:
                        for __l__ in i_.findAll('tr'):
                            y.append(__l__.text)
                            try:
                                g=''
                                g_=[]
                                for h in (__l__.text):
                                    if h.isupper() or h.isdigit():
                                        if h.isdigit():
                                            g_.append(((__l__.text).
index(h)))
```

```
                                        if ((__l__.text).index(h))
==g_[0]:
                                            g=g+' '+h
                                        else:
                                            g=g+h
                                    elif h.isupper():
                                        g=g+' '+h
                                    else:
                                        g=g+h
                                else:
                                    g=g+h
                            print("·.",g)
                        except:
                            print("₪",__l__.text)
                    except:
                        try:
                            for __l__ in i_.find():
                                print("₪",__l__.text)
                                y.append(__l__.text)
                        except:
                            for __l__ in i_.findAll('td'):
                                print("₪",__l__.text)
                                y.append(__l__.text)
                except:
                    try:
                        i_=s.find(class_='infobox biography
vcard')
                        try:
                            for __l__ in i_.findAll('tr'):
                                y.append(__l__.text)
                                try:
                                    g=''
                                    g_=[]
                                    for h in (__l__.text):
```

```
                                    if h.isupper() or h.isdigit():
                                        if h.isdigit():
                                            g_.append(((__l__.text).
index(h)))
                                            if ((__l__.text).index(h))
==g_[0]:
                                                g=g+' '+h
                                            else:
                                                g=g+h
                                        elif h.isupper():
                                            g=g+' '+h
                                        else:
                                            g=g+h
                                    else:
                                        g=g+h
                                print("·.",g)
                            except:
                                print("₪",__l__.text)
                        except:
                            try:
                                for __l__ in i_.find():
                                    print("₪",__l__.text)
                                    y.append(__l__.text)
                            except:
                                for __l__ in i_.findAll('td'):
                                    print("₪",__l__.text)
                                    y.append(__l__.text)
                    except:
                        try:
                            i_=s.find(class_='infobox biota')
                            try:
                                for __l__ in i_.findAll('tr'):
                                    y.append(__l__.text)
                                    try:
```

```
                                    g=''
                                    g_=[]
                                    for h in (__l__.text):
                                       if h.isupper() or h.isdigit():
                                          if h.isdigit():
                                             g_.append(((__l__.text).
index(h)))
                                             if ((__l__.text).index(h))
==g_[0]:
                                                g=g+' '+h
                                             else:
                                                g=g+h
                                          elif h.isupper():
                                             g=g+' '+h
                                          else:
                                             g=g+h
                                       else:
                                          g=g+h
                                    print("·.",g)
                                 except:
                                    print("₪",__l__.text)
                              except:
                                 try:
                                    for __l__ in i_.find():
                                       print("₪",__l__.text)
                                       y.append(__l__.text)
                                 except:
                                    for __l__ in i_.findAll('td'):
                                       print("₪",__l__.text)
                                       y.append(__l__.text)
                           except:
                              try:
                                 i_=s.find(class_='infobox
hproduct')
```

```
                                    try:
                                       for __l__ in i_.findAll('tr'):
                                          y.append(__l__.text)
                                          try:
                                             g=''
                                             g_=[]
                                             for h in (__l__.text):
                                                if h.isupper() or h.
isdigit():
                                                   if h.isdigit():
                                                      g_.append(((__l__.
text).index(h)))
                                                   if ((__l__.text).
index(h))==g_[0]:
                                                      g=g+' '+h
                                                   else:
                                                      g=g+h
                                                elif h.isupper():
                                                   g=g+' '+h
                                                else:
                                                   g=g+h
                                             else:
                                                g=g+h
                                          print("·.",g)
                                       except:
                                          print("₪",__l__.text)
                                    except:
                                       try:
                                          for __l__ in i_.find():
                                             print("₪",__l__.text)
                                             y.append(__l__.text)
                                       except:
                                          for __l__ in i_.findAll('td'):
                                             print("₪",__l__.text)
```

```
                                    y.append(__l__.text)
                                except:
                                    try:
                                        i_=s.find(class_='infobox
bordered')
                                        try:
                                            for __l__ in i_.findAll('tr'):
                                                y.append(__l__.text)
                                                try:
                                                    g=''
                                                    g_=[]
                                                    for h in (__l__.text):
                                                        if h.isupper() or h.
isdigit():
                                                            if h.isdigit():
                                                                g_.append(((__l__.
text).index(h)))
                                                                if ((__l__.text).
index(h))==g_[0]:
                                                                    g=g+' '+h
                                                                else:
                                                                    g=g+h
                                                            elif h.isupper():
                                                                g=g+' '+h
                                                            else:
                                                                g=g+h
                                                        else:
                                                            g=g+h
                                                    print("·.",g)
                                                except:
                                                    print("₪",__l__.text)
                                        except:
                                            try:
                                                for __l__ in i_.find():
```

```
                print("₪",__l__.text)
                y.append(__l__.text)
            except:
                for __l__ in i_.findAll('td'):
                print("₪",__l__.text)
                y.append(__l__.text)
        except:
            try:
                i_=s.find(class_='infobox
vcard plainlist')
                try:
                for __l__ in i_.findAll('tr'):
                y.append(__l__.text)
                try:
                g=''
                g_=[]
                for h in (__l__.text):
                if h.isupper() or h.
isdigit():

                if h.isdigit():
                g_.append(((__l__.
text).index(h)))

                if ((__l__.text).
index(h))==g_[0]:

                g=g+' '+h
                else:
                g=g+h
                elif h.isupper():
                g=g+' '+h
                else:
                g=g+h
            else:
                g=g+h
            print("·.",g)
```

```
                                    except:
                                        print("₪",__l__.text)
                                except:
                                    try:
                                        for __l__ in i_.find():
                                            print("₪",__l__.text)
                                            y.append(__l__.text)
                                    except:
                                        for __l__ in i_.findAll('td'):
                                            print("₪",__l__.text)
                                            y.append(__l__.text)
                            except:
                                try:
                                    i_=s.find(class_='vertical-
navbox nowraplinks')
                                    for __l__ in i_.find():
                                        print("↴",__l__.text)
                                        y.append(__l__.text)
                                except:
                                    try:
                                        i_=s.find(class_='vertical-
navbox vcard')
                                        for __l__ in i_.find():
                                            print("↴",__l__.text)
                                            y.append(__l__.text)
                                    except:
                                        try:
                                            i_=s.
find(class_='infobox vevent haudio')
                                            try:
                                                for __l__ in i_.
findAll('tr'):
                                                    y.append(__l__.text)
                                                    try:
```

```
                                                    g=''
                                                    g_=[]
                                                    for h in (__l__.
text):
                                                       if h.isupper()
or h.isdigit():
                                                          if h.isdigit():
                                                             g_.
append(((__l__.text).index(h)))
                                                             if ((__l__.
text).index(h))==g_[0]:
                                                                g=g+' '+h
                                                             else:
                                                               g=g+h
                                                          elif h.
isupper():
                                                             g=g+' '+h
                                                          else:
                                                             g=g+h
                                                       else:
                                                             g=g+h
                                                    print("·.",g)
                                                 except:
                                                    print("₪",__l__.
text)
                                           except:
                                              try:
                                                 for __l__ in i_.find():
                                                    print("₪",__l__.
text)
                                                    y.append(__l__.
text)
                                              except:
                                                 for __l__ in i_.
```

```
findAll('td'):
                                                    print("₪",__l__.
text)
                                                    y.append(__l__.
text)
                                                except:
                                                   try:
                                                      i_=s.
find(class_='vertical-navbox nowraplinks hlist')
                                                      try:
                                                         for __l__ in i_.
findAll('tr'):
                                                    y.append(__l__.
text)
                                                    try:
                                                       g=''
                                                       g_=[]
                                                       for h in (__l__.
text):
                                                          if h.isupper()
 or h.isdigit():
                                                             if h.
isdigit():
                                                                g_.
append(((__l__.text).index(h)))
                                                                if ((__l__.
text).index(h))==g_[0]:
                                                                   g=g+'
'+h
                                                                else:
                                                                   g=g+h
                                                             elif h.
isupper():
                                                                g=g+' '+h
```

```
                                    else:
                                        g=g+h
                                else:
                                    g=g+h
                            print("·.",g)
                        except:
                            print("₪",__l__.
text)
                    except:
                        try:
                            for __l__ in i_.
find():
                                print("₪",__l__.
text)
                                y.append(__l__.
text)
                        except:
                            for __l__ in i_.
findAll('td'):
                                print("₪",__l__.
text)
                                y.append(__l__.
text)
                except:
                    r=requests.get("
https://en.wikipedia.org/wiki/{}".format(q))
                    s = BeautifulSoup(r.
text,"html.parser")
                    p=s.select('p')[0:3]
                    i='\n'.join([o.text for
o in p])
                    print(i)
                    try:
                        for x in range(0,3):
```

```
                                                    if len(p[x].text)
>100:
                                                        f=(p[x].text)
                                                if v == "YES":
                                                    s=gTTS(text="
hmm {}".format(f),lang='en-uk')
                                                    s.save(r"
C:\Python Files\Audiofiles\VoiceOver.mp3")
                                                    playsound(r"
C:\Python Files\Audiofiles\VoiceOver.mp3")
                                                    os.remove("
C:\Python Files\Audiofiles\VoiceOver.mp3")
                                                except:
                                                    pass
            if v == "YES":
                s=gTTS(text="hmm Here is the gist about {}".
format(q),lang='en-uk')
                s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            if len(y)<3:
                r=requests.get("https://en.wikipedia.org/wiki/
{}".format(q))
                s = BeautifulSoup(r.text,"html.parser")
                p=s.select('p')[0:3]
                i='\n'.join([o.text for o in p])
                try:
                    for x in range(0,3):
                        if len(p[x].text)>100:
                            f=(p[x].text)
                    if v == "YES":
```

```
                    s=gTTS(text="hmm {}".format(f),lang='en-
uk')
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                except:
                    pass
                if len(i)<50:
                    webbrowser.open("https://www.google.com/
search?q={}".format(w))
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
                    keyboard.press_and_release('win + Right')
                    time.sleep(1)
                else:
                    print(i)
            except requests.exceptions.ConnectionError:
                print(" \t⚠Oops...Make Sure That You Are
Connected With Internet⚠ ")
                if v == "YES":
                    playsound(r"C:\Python Files\Audiofiles\offline.
mp3")
        elif ("tell" in q_ or "show" in q_ or "get" in q_ or "give"
in q_ or "brief" in q_ or "big note" in q_ or "long note" in q_ or
"large note" in q_ or "huge note" in q_) and ("about" in q_ or "
on" in q_):
            try:
                try:
```

```
    l= q_.split()
    try:
        l.remove("sid")
        w=' '.join(l)
    except:
        l.remove("glenda")
        w=' '.join(l)
except:
    w=q_
try:
    l=w.split()
    try:
        i=l.index("about")+1
        j=l.index("of")
        w__=' '.join(l[i:j])
    except:
        try:
            i=l.index("about")+1
            j=l.index("in")
            w__=' '.join(l[i:j])
        except:
            try:
                i=l.index("on")+1
                j=l.index("of")
                w__=' '.join(l[i:j])
            except:
                i=l.index("on")+1
                j=l.index("in")
                w__=' '.join(l[i:j])
    w_ = str(w__).casefold()
    try:
        i=l.index("of")+1
        u=' '.join(l[i:])
    except:
```

```
                    try:
                        if "in brief" not in q_ or " in big note" not in
q_ or "in long note" not in q_ or "in large note" not in q_ or "
in huge note" not in q_:
                            i=l.index("in")+1
                            u=' '.join(l[i:])
                    except:
                        pass
                r = requests.get("https://google.com/
search?q={} wikipedia".format(u))
                s = BeautifulSoup(r.content, 'html.parser')
                e=[]
                for a_ in s.find_all('a', href=True):
                    e.append(a_['href'])
                x=[]
                for t in e:
                    if "/url?q=https://en.wikipedia.org/wiki/" in
t:
                        x.append(e.index(t))
                l=e[x[0]]
                i=l.index('&')
                q=l[37:i]
                if '%' in q:
                    print("Just Sit Back And Hold Tight!")
                    webbrowser.open("https://google.com/
search?q={}".format(w))
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
                    keyboard.press_and_release('win + Right')
                    time.sleep(1)
                r=requests.get("https://en.wikipedia.org/wiki/
```

```
{}".format(q))
                    s = BeautifulSoup(r.text,"html.parser")
                    z=[]
                    y=[]
                    i_=s.find(class_='infobox')
                    for __l__ in i_.findAll('tr'):
                       y.append(__l__.text)
                       if w_ in (__l__.text).casefold():
                          z.append(__l__.text)
                    if "president" in w_ and len(z[0]) > 53:
                       p=(y).index(z[1])
                    else:
                       p=(y).index(z[0])
                    r = requests.get("https://google.com/
search?q={} wikipedia".format(y[p]))
                    s = BeautifulSoup(r.content, 'html.parser')
                    e=[]
                    for a_ in s.find_all('a', href=True):
                       e.append(a_['href'])
                    x=[]
                    for t in e:
                       if "/url?q=https://en.wikipedia.org/wiki/" in
t:
                          x.append(e.index(t))
                    l=e[x[0]]
                    i=l.index('&')
                    q=l[37:i]
                    if '%' in q:
                       print("Just Sit Back And Hold Tight!")
                       webbrowser.open("https://google.com/
search?q={}".format(w))
                       time.sleep(1)
                       keyboard.press_and_release('win + Up')
                       time.sleep(1)
```

```
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Right')
                time.sleep(1)
            else:
                print("\t                    \||~~~~||||",q," 
||||~~~~||/ \n")
                r=requests.get("https://en.wikipedia.org/
wiki/{}".format(q))
                s = BeautifulSoup(r.text,"html.parser")
                if "brief" in q_ or "big note" in q_ or "long note"
 in q_ or "large note" in q_ or "huge note" in q_:
                    p=s.select('p')
                    i='\n'.join([o.text for o in p])
                    print(i)
                    try:
                        for x in range(0,10):
                            if len(p[x].text)>100:
                                f_=(p[x+1].text)
                                _f_=(p[x+2].text)
                                __f__=(p[x+3].text)
                        if v == "YES":
                            s=gTTS(text="hmm {}".
format(f+f_+_f_+__f__),lang='en-uk')
                            s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                            playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                            os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    except:
                        pass
                else:
                    p=s.select('p')[0:6]
```

```
                    i='\n'.join([o.text for o in p])
                    print(i)
                    try:
                        for x in range(0,6):
                            if len(p[x].text)>100:
                                f=(p[x].text)
                        if v == "YES":
                            s=gTTS(text="hmm {}".format(f),
lang='en-uk')
                            s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                            playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                            os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    except:
                        pass
            except:
                r = requests.get("https://google.com/
search?q={} wikipedia".format(w))
                s = BeautifulSoup(r.content, 'html.parser')
                e=[]
                for a_ in s.find_all('a', href=True):
                    e.append(a_['href'])
                x=[]
                for t in e:
                    if "/url?q=https://en.wikipedia.org/wiki/" in
t:
                        x.append(e.index(t))
                l=e[x[0]]
                i=l.index('&')
                q=l[37:i]
                if '%' in q:
                    print("Just Sit Back And Hold Tight!")
```

```
                webbrowser.open("https://google.com/
search?q={}".format(w))
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Right')
                time.sleep(1)
            else:
                print("\t                    \||~~~~||||",q,"
||||~~~~||/ \n")
                r=requests.get("https://en.wikipedia.org/
wiki/{}".format(q))
                s = BeautifulSoup(r.text,"html.parser")
                a=[]
                p=s.select('p')[0:6]
                i=''.join([o.text for o in p])
                for b in i:
                    for c in b:
                        a.append(c)
                if "brief" in q_ or "big note" in q_ or "long note"
 in q_ or "large note" in q_ or "huge note" in q_:
                    p=s.select('p')
                    i='\n'.join([o.text for o in p])
                    try:
                        for x in range(0,10):
                            if len(p[x].text)>100:
                                f=(p[x].text)
                                f_=(p[x+1].text)
                                _f_=(p[x+2].text)
                                __f__=(p[x+3].text)
                        if v == "YES":
                            s=gTTS(text="hmm {}".
```

```
format(f+f_+_f_+__f__),lang='en-uk')
                        s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    except:
                        pass
                elif len(a)<150:
                    i_=s.find(class_='infobox')
                    y=[]
                    for __l__ in i_.findAll('tr'):
                        y.append(__l__.text)
                        try:
                            g=''
                            g_=[]
                            for h in (__l__.text):
                                if h.isupper() or h.isdigit():
                                    if h.isdigit():
                                        g_.append(((__l__.text).index(h)))
                                        if ((__l__.text).index(h))==g_[0]:
                                            g=g+' '+h
                                        else:
                                            g=g+h
                                    elif h.isupper():
                                        g=g+' '+h
                                    else:
                                        g=g+h
                                else:
                                    g=g+h
                            print("·.",g)
                        except:
                            print("₪",__l__.text)
```

```
                    else:
                        p=s.select('p')[0:6]
                        i='\n'.join([o.text for o in p])
                        print(i)
                        try:
                            for x in range(0,6):
                                if len(p[x].text)>100:
                                    f=(p[x].text)
                            if v == "YES":
                                s=gTTS(text="hmm {}".format(f),
lang='en-uk')
                                s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        except:
                            pass
            except requests.exceptions.ConnectionError:
                print(" \t⚠Oops...Make Sure That You Are
Connected With Internet⚠ ")
                if v == "YES":
                    playsound(r"C:\Python Files\Audiofiles\offline.
mp3")
        elif (("what" in q_ or "who" in q_) and ("is" in q_ or "
are" in q_)) and ("of" in q_ or " in" in q_ or "for" in q_):
            try:
                l= q_.split()
                try:
                    try:
                        i=l.index("the")+1
                        j=l.index("of")
                        w=' '.join(l[i:j])
```

```
    except:
        try:
            i=l.index("the")+1
            j=l.index("in")
            w=' '.join(l[i:j])
        except:
            i=l.index("the")+1
            j=l.index("for")
            w=' '.join(l[i:j])
except:
    try:
        try:
            i=l.index("is")+1
            j=l.index("of")
            w=' '.join(l[i:j])
        except:
            try:
                i=l.index("is")+1
                j=l.index("in")
                w=' '.join(l[i:j])
            except:
                i=l.index("is")+1
                j=l.index("for")
                w=' '.join(l[i:j])
    except:
        try:
            i=l.index("are")+1
            j=l.index("of")
            w=' '.join(l[i:j])
        except:
            try:
                i=l.index("are")+1
                j=l.index("in")
                w=' '.join(l[i:j])
```

```
                except:
                    i=l.index("are")+1
                    j=l.index("for")
                    w=' '.join(l[i:j])
            w_ = str(w).casefold()
            try:
                i=l.index("of")+1
                u=' '.join(l[i:])
            except:
                try:
                    i=l.index("in")+1
                    u=' '.join(l[i:])
                except:
                    i=l.index("for")+1
                    u=' '.join(l[i:])
            try:
                if ("birth" in q_ or "born" in q_) and ("day" in q_
or "anniversary" in q_ or "place" in q_ or "location" in q_):
                    w_='born'
                elif ("Assassination" in q and ((("died" in q_ or "
death" in q_) and ("day" in q_ or "anniversary")) or "die" in
q_ or "killed" in q_ or "murder" in q_ or "death" in q_ or "
assassination" in q_)) or "started" in q_ or "end" in q_ or "
period" in q_:
                    w_='date'
                elif (("died" in q_ or "death" in q_) and ("day" in
q_ or "anniversary" or "place" in q_ or "location" in q_)) or "
die" in q_ or "killed" in q_ or "murder" in q_ or "death" in q_
or "assassinat" in q_:
                    w_='died'
                elif "construct" in q_ or "open" in q_ or "build"
in q_ or "built" in q_:
                    w_='open'
                elif "release" in q_ or "out" in q_:
```

```
                w_="release"
            elif "locat" in q_:
                w_='location'
        except:
            w_ = str(w).casefold()
        try:
            r = requests.get("https://google.com/
search?q={} wikipedia".format(u))
            s = BeautifulSoup(r.content, 'html.parser')
            e=[]
            for a_ in s.find_all('a', href=True):
                e.append(a_['href'])
            x=[]
            for t in e:
                if "/url?q=https://en.wikipedia.org/wiki/" in
t:
                    x.append(e.index(t))
            l=e[x[0]]
            i=l.index('&')
            q=l[37:i]
            if '%' in q:
                print("Just Sit Back And Hold Tight!")
                webbrowser.open("https://google.com/
search?q={}".format(w))
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Right')
                time.sleep(1)
            a=b=0
            for o in range(len(q)):
                if q[o].isalpha():
```

```
                    a=a+1
                elif q[o].isdigit():
                    b=b+1
                else:
                    pass
            if a<4 and b>=1:
                l=e[x[1]]
                i=l.index('&')
                q=l[37:i]
            else:
                pass
            r=requests.get("https://en.wikipedia.org/wiki/
{}".format(q))
            s = BeautifulSoup(r.text,"html.parser")
            _w_=w_.split()
            z=[]
            y=[]
            try:
                i_=s.find(class_='infobox geography vcard')
                try:
                    try:
                        for __l__ in i_.find():
                            y.append(__l__.text)
                            if w_ in (__l__.text).casefold():
                                z.append(__l__.text)
                    except:
                        try:
                            for __l__ in i_.find():
                                y.append(__l__.text)
                                if _w_[0] in (__l__.text).casefold():
                                    z.append(__l__.text)
                        except:
                            for __l__ in i_.find():
                                y.append(__l__.text)
```

```
                if _w_[1] in (__l__.text).casefold():
                    z.append(__l__.text)
        except:
            try:
                for __l__ in i_.findAll('tr'):
                    y.append(__l__.text)
                    if w_ in (__l__.text).casefold():
                        z.append(__l__.text)
            except:
                try:
                    for __l__ in i_.findAll('tr'):
                        y.append(__l__.text)
                        if _w_[0] in (__l__.text).casefold():
                            z.append(__l__.text)
                except:
                    for __l__ in i_.findAll('tr'):
                        y.append(__l__.text)
                        if _w_[1] in (__l__.text).casefold():
                            z.append(__l__.text)
    except:
        try:
            i_=s.find(class_='infobox')
            try:
                try:
                    for __l__ in i_.find():
                        y.append(__l__.text)
                        if w_ in (__l__.text).casefold():
                            z.append(__l__.text)
                except:
                    try:
                        for __l__ in i_.find():
                            y.append(__l__.text)
                            if _w_[0] in (__l__.text).casefold():
                                z.append(__l__.text)
```

```
            except:
                for __l__ in i_.find():
                    y.append(__l__.text)
                    if _w_[1] in (__l__.text).casefold():
                        z.append(__l__.text)
        except:
            try:
                for __l__ in i_.findAll('tr'):
                    y.append(__l__.text)
                    if w_ in (__l__.text).casefold():
                        z.append(__l__.text)
            except:
                try:
                    for __l__ in i_.findAll('tr'):
                        y.append(__l__.text)
                        if _w_[0] in (__l__.text).casefold():
                            z.append(__l__.text)
                except:
                    for __l__ in i_.findAll('tr'):
                        y.append(__l__.text)
                        if _w_[1] in (__l__.text).casefold():
                            z.append(__l__.text)
    except:
        try:
            i_=s.find(class_='infobox vcard')
            try:
                try:
                    for __l__ in i_.find():
                        y.append(__l__.text)
                        if w_ in (__l__.text).casefold():
                            z.append(__l__.text)
                except:
                    try:
                        for __l__ in i_.find():
```

```
                                y.append(__l__.text)
                                if _w_[0] in (__l__.text).casefold():
                                    z.append(__l__.text)
                            except:
                                for __l__ in i_.find():
                                    y.append(__l__.text)
                                    if _w_[1] in (__l__.text).casefold():
                                        z.append(__l__.text)
                        except:
                            try:
                                for __l__ in i_.findAll('tr'):
                                    print(__l__.text)
                                    y.append(__l__.text)
                                    if w_ in (__l__.text).casefold():
                                        z.append(__l__.text)
                            except:
                                try:
                                    for __l__ in i_.findAll('tr'):
                                        y.append(__l__.text)
                                        if _w_[0] in (__l__.text).casefold():
                                            z.append(__l__.text)
                                except:
                                    for __l__ in i_.findAll('tr'):
                                        y.append(__l__.text)
                                        if _w_[1] in (__l__.text).casefold():
                                            z.append(__l__.text)
                    except:
                        try:
                            i_=s.find(class_='infobox biography
vcard')
                            try:
                                try:
                                    for __l__ in i_.find():
                                        y.append(__l__.text)
```

```
                        if w_ in (__l__.text).casefold():
                            z.append(__l__.text)
                    except:
                        try:
                            for __l__ in i_.find():
                                y.append(__l__.text)
                                if _w_[0] in (__l__.text).
casefold():
                                    z.append(__l__.text)
                        except:
                            for __l__ in i_.find():
                                y.append(__l__.text)
                                if _w_[1] in (__l__.text).
casefold():
                                    z.append(__l__.text)
                else:
                    try:
                        for __l__ in i_.findAll('tr'):
                            y.append(__l__.text)
                            if w_ in (__l__.text).casefold():
                                z.append(__l__.text)
                    except:
                        try:
                            for __l__ in i_.findAll('tr'):
                                y.append(__l__.text)
                                if _w_[0] in (__l__.text).
casefold():
                                    z.append(__l__.text)
                        except:
                            for __l__ in i_.findAll('tr'):
                                y.append(__l__.text)
                                if _w_[1] in (__l__.text).
casefold():
                                    z.append(__l__.text)
```

```
                except:
                    try:
                        i_=s.find(class_='infobox vevent')
                        try:
                            try:
                                for __l__ in i_.find():
                                    y.append(__l__.text)
                                    if w_ in (__l__.text).casefold():
                                        z.append(__l__.text)
                            except:
                                try:
                                    for __l__ in i_.find():
                                        y.append(__l__.text)
                                        if _w_[0] in (__l__.text).
casefold():
                                            z.append(__l__.text)
                                except:
                                    for __l__ in i_.find():
                                        y.append(__l__.text)
                                        if _w_[1] in (__l__.text).
casefold():
                                            z.append(__l__.text)
                        except:
                            try:
                                for __l__ in i_.findAll('tr'):
                                    y.append(__l__.text)
                                    if w_ in (__l__.text).casefold():
                                        z.append(__l__.text)
                            except:
                                try:
                                    for __l__ in i_.findAll('tr'):
                                        y.append(__l__.text)
                                        if _w_[0] in (__l__.text).
casefold():
```

```
                                    z.append(__l__.text)
                                except:
                                    for __l__ in i_.findAll('tr'):
                                        y.append(__l__.text)
                                        if _w_[1] in (__l__.text).
casefold():
                                            z.append(__l__.text)
                            except:
                                try:
                                    i_=s.find(class_='infobox biota')
                                    try:
                                        try:
                                            for __l__ in i_.findAll('tr'):
                                                y.append(__l__.text)
                                                if w_ in (__l__.text).casefold():
                                                    z.append(__l__.text)
                                        except:
                                            try:
                                                for __l__ in i_.findAll('tr'):
                                                    y.append(__l__.text)
                                                    if _w_[0] in (__l__.text).
casefold():
                                                        z.append(__l__.text)
                                            except:
                                                for __l__ in i_.findAll('tr'):
                                                    y.append(__l__.text)
                                                    if _w_[1] in (__l__.text).
casefold():
                                                        z.append(__l__.text)
                                    except:
                                        try:
                                            b_='True'
                                            for __l__ in i_.findAll('td'):
                                                y.append(__l__.text)
```

```
                        if w_ in (__l__.text).casefold():
                            z.append(__l__.text)
                    except:
                        try:
                            for __l__ in i_.findAll('td'):
                                y.append(__l__.text)
                                if _w_[0] in (__l__.text).
casefold():
                                    z.append(__l__.text)
                        except:
                            for __l__ in i_.findAll('td'):
                                y.append(__l__.text)
                                if _w_[1] in (__l__.text).
casefold():
                                    z.append(__l__.text)
                except:
                    try:
                        b_='False'
                        i_=s.find(class_='infobox
hproduct')
                        try:
                            try:
                                for __l__ in i_.findChild():
                                    y.append(__l__.text)
                                    if w_ in (__l__.text).
casefold():
                                        z.append(__l__.text)
                            except:
                                try:
                                    for __l__ in i_.findChild():
                                        y.append(__l__.text)
                                        if _w_[0] in (__l__.text).
casefold():
                                            z.append(__l__.text)
```

```
                except:
                    for __l__ in i_.findChild():
                        y.append(__l__.text)
                        if _w_[1] in (__l__.text).
casefold():
                            z.append(__l__.text)
            except:
                try:
                    for __l__ in i_.findAll('td'):
                        y.append(__l__.text)
                        if w_ in (__l__.text).
casefold():
                            z.append(__l__.text)
                except:
                    try:
                        for __l__ in i_.findAll('td'):
                            y.append(__l__.text)
                            if _w_[0] in (__l__.text).
casefold():
                                z.append(__l__.text)
                    except:
                        for __l__ in i_.findAll('td'):
                            y.append(__l__.text)
                            if _w_[1] in (__l__.text).
casefold():
                                z.append(__l__.text)
        except:
            try:
                i_=s.find(class_='infobox
bordered')
                try:
                    try:
                        for __l__ in i_.findAll('tr'):
                            y.append(__l__.text)
```

```
                    if w_ in (__l__.text).
casefold():
                        z.append(__l__.text)
                except:
                    try:
                        for __l__ in i_.findAll('tr'):
                            y.append(__l__.text)
                            if _w_[0] in (__l__.text).
casefold():
                                z.append(__l__.text)
                    except:
                        for __l__ in i_.findAll('tr'):
                            y.append(__l__.text)
                            if _w_[1] in (__l__.text).
casefold():
                                z.append(__l__.text)
            except:
                try:
                    c_='True'
                    for __l__ in i_.findAll('td'):
                        y.append(__l__.text)
                        if w_ in (__l__.text).
casefold():
                            z.append(__l__.text)
                except:
                    try:
                        for __l__ in i_.findAll('td'):
                            y.append(__l__.text)
                            if _w_[0] in (__l__.text).
casefold():
                                z.append(__l__.text)
                    except:
                        for __l__ in i_.findAll('td'):
                            y.append(__l__.text)
```

```
                                    if _w_[1] in (__l__.text).
casefold():
                                        z.append(__l__.text)
                                except:
                                    try:
                                        c_='False'
                                        i_=s.find(class_='infobox
vcard plainlist')
                                        try:
                                            try:
                                                for __l__ in i_.find():
                                                    y.append(__l__.text)
                                                    if w_ in (__l__.text).
casefold():
                                                        z.append(__l__.text)
                                            except:
                                                try:
                                                    for __l__ in i_.find():
                                                        y.append(__l__.text)
                                                        if _w_[0] in (__l__.text)
.casefold():
                                                            z.append(__l__.text)
                                                except:
                                                    for __l__ in i_.find():
                                                        y.append(__l__.text)
                                                        if _w_[1] in (__l__.text)
.casefold():
                                                            z.append(__l__.text)
                                        except:
                                            try:
                                                for __l__ in i_.findAll('tr'):
                                                    y.append(__l__.text)
                                                    if w_ in (__l__.text).
casefold():
```

```
                                        z.append(__l__.text)
                                except:
                                    try:
                                        for __l__ in i_.
findAll('tr'):
                                            y.append(__l__.text)
                                            if _w_[0] in (__l__.text)
.casefold():
                                                z.append(__l__.text)
                                    except:
                                        for __l__ in i_.
findAll('tr'):
                                            y.append(__l__.text)
                                            if _w_[1] in (__l__.text)
.casefold():
                                                z.append(__l__.text)

                            except:
                                try:
                                    i_=s.find(class_='infobox
vevent haudio')
                                    try:
                                        try:
                                            for __l__ in i_.find():
                                                y.append(__l__.text)
                                                if w_ in (__l__.text).
casefold():
                                                    z.append(__l__.text)
                                        except:
                                            try:
                                                for __l__ in i_.find():
                                                    y.append(__l__.text)
                                                    if _w_[0] in (__l__.
text).casefold():
```

```
                                        z.append(__l__.
text)
                                    except:
                                        for __l__ in i_.find():
                                            y.append(__l__.text)
                                            if _w_[1] in (__l__.
text).casefold():
                                                z.append(__l__.
text)
                                except:
                                    try:
                                        for __l__ in i_.
findAll('tr'):
                                            y.append(__l__.text)
                                            if w_ in (__l__.text).
casefold():
                                                z.append(__l__.text)
                                    except:
                                        try:
                                            for __l__ in i_.
findAll('tr'):
                                                y.append(__l__.text)
                                                if _w_[0] in (__l__.
text).casefold():
                                                    z.append(__l__.
text)
                                        except:
                                            for __l__ in i_.
findAll('tr'):
                                                y.append(__l__.text)
                                                if _w_[1] in (__l__.
text).casefold():
                                                    z.append(__l__.
text)
```

```
                                        except:
                                            try:
                                                i_=s.find(class_='vertical-
navbox nowraplinks')
                                                try:
                                                    try:
                                                        for __l__ in i_.find():
                                                            y.append(__l__.text)
                                                            if w_ in (__l__.text).
casefold():
                                                                z.append(__l__.
text)
                                                    except:
                                                        try:
                                                            for __l__ in i_.find():
                                                                y.append(__l__.
text)
                                                                if _w_[0] in (__l__.
text).casefold():
                                                                    z.append(__l__.
text)
                                                        except:
                                                            for __l__ in i_.find():
                                                                y.append(__l__.
text)
                                                                if _w_[1] in (__l__.
text).casefold():
                                                                    z.append(__l__.
text)
                                                except:
                                                    try:
                                                        for __l__ in i_.
findAll('tr'):
                                                            y.append(__l__.text)
```

```
                                                    if w_ in (__l__.text).
casefold():
                                                        z.append(__l__.
text)
                                                except:
                                                    try:
                                                        for __l__ in i_.
findAll('tr'):
                                                            y.append(__l__.
text)
                                                            if _w_[0] in (__l__.
text).casefold():
                                                                z.append(__l__.
text)
                                                    except:
                                                        for __l__ in i_.
findAll('tr'):
                                                            y.append(__l__.
text)
                                                            if _w_[1] in (__l__.
text).casefold():
                                                                z.append(__l__.
text)
                                            except:
                                                try:
                                                    i_=s.
find(class_='vertical-navbox vcard')
                                                    try:
                                                        try:
                                                            for __l__ in i_.find():
                                                                y.append(__l__.
text)
                                                                if w_ in (__l__.
text).casefold():
```

```
text)



find():

text)

(__l__.text).casefold():

append(__l__.text)


find():

text)

(__l__.text).casefold():

append(__l__.text)



findAll('tr'):

text)

text).casefold():

text)
                    z.append(__l__.
            except:
                try:
                    for __l__ in i_.
                        y.append(__l__.
                        if _w_[0] in
                            z.
                except:
                    for __l__ in i_.
                        y.append(__l__.
                        if _w_[1] in
                            z.
        except:
            try:
                for __l__ in i_.
                    y.append(__l__.
                    if w_ in (__l__.
                        z.append(__l__.
            except:
                try:
                    for __l__ in i_.
```

```
findAll('tr'):
                                                    y.append(__l__.
text)
                                                    if _w_[0] in
(__l__.text).casefold():
                                                        z.
append(__l__.text)
                                                except:
                                                    for __l__ in i_.
findAll('tr'):
                                                    y.append(__l__.
text)
                                                    if _w_[1] in
(__l__.text).casefold():
                                                        z.
append(__l__.text)
                                            except:
                                                i_=s.
find(class_='vertical-navbox nowraplinks hlist')
                                                try:
                                                    try:
                                                        for __l__ in i_.find():
                                                            y.append(__l__.
text)
                                                            if w_ in (__l__.
text).casefold():
                                                                z.append(__l__.
text)
                                                    except:
                                                        try:
                                                            for __l__ in i_.
find():
                                                                y.append(__l__.
text)
```

```
(__l__.text).casefold():

append(__l__.text)


find():

text)

(__l__.text).casefold():

append(__l__.text)



findAll('tr'):

text)

text).casefold():

text)



findAll('tr'):

text)

(__l__.text).casefold():

append(__l__.text)
                    if _w_[0] in

                        z.

                except:
                    for __l__ in i_.

                        y.append(__l__.

                        if _w_[1] in

                            z.

            except:
                try:
                    for __l__ in i_.

                        y.append(__l__.

                        if w_ in (__l__.

                            z.append(__l__.

                except:
                    try:
                        for __l__ in i_.

                            y.append(__l__.

                            if _w_[0] in

                                z.

                    except:
```

```
                                                    for __l__ in i_.
findAll('tr'):
                                                        y.append(__l__.
text)
                                                        if _w_[1] in
(__l__.text).casefold():
                                                            z.
append(__l__.text)
                    if "president" in w_ and len(z[0]) > 53:
                        p=(y).index(z[1])
                    elif "Show map" in z[0]:
                        p=(y).index(z[1])
                    elif i_==s.find(class_='infobox geography vcard')
 and len(z[0]) > 95:
                        try:
                            p=(y).index(z[2])
                        except:
                            try:
                                p=(y).index(z[1])
                            except:
                                p=(y).index(z[0])
                    else:
                        p=(y).index(z[0])
                    g=''
                    for h in y[p]:
                        if h.isupper() or h.isdigit():
                            if (y[p][((y[p].index(h))-1)]).islower() or
(y[p][((y[p].index(h))+1)]).islower():
                                g=g+' '+h
                            elif (y[p][((y[p].index(h))-1)]).isdigit() or
(y[p][((y[p].index(h))+1)]).isdigit() or (y[p][((y[p].index(h))
+1)]) is not ' ':
                                g=g+h
                            elif (y[p][((y[p].index(h))-1)]).isupper() or
```

```
(y[p][((y[p].index(h))+1)]).isupper() or (y[p][((y[p].
index(h))+1)]) is not ' ':
                        g=g+h
                    else:
                        g=g+h
                y[p]=g
                if "•" in y[p]:
                    print("〜",y[p],"• 〜")
                elif i_==s.find(class_='vertical-navbox
nowraplinks hlist') or i_==s.find(class_='vertical-navbox
nowraplinks') or i_==s.find(class_='vertical-navbox vcard'):
                    print("↴",y[p])
                else:
                    print(y[p])
                try:
                    if v == "YES":
                        s=gTTS(text="hmm {}".format(y[p]),
lang='en-uk')
                        s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                except:
                    pass
                try:
                    if i_==s.find(class_='infobox bordered') and
c_=='True':
                        y_=y.index('Chemical formula\n')
                        if y_<p:
                            print("-⇴►",y[p+1])
                        else:
                            pass
```

```
                elif i_==s.find(class_='infobox biota') and
b_=='True':
                    print("-⇏►",y[p+1])
                try:
                    if v == "YES":
                        s=gTTS(text="hmm {}".format(y[p+1]),
lang='en-uk')
                        s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                except:
                    pass
            except:
                pass
            try:
                if "•" not in y[p]:
                    if "•" in y[p+1]:
                        print("\n")
                        print(y[p+1])
                        try:
                            if v == "YES":
                                s=gTTS(text="hmm {}".
format(y[p+1]),lang='en-uk')
                                s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        except:
                            pass
```

```
                    if "•" in y[p+2] and "•" in y[p+1]:
                        print("\n")
                        print(y[p+2])
                        try:
                            if v == "YES":
                                s=gTTS(text="hmm {}".
format(y[p+2]),lang='en-uk')
                                s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        except:
                            pass
                    if "•" in y[p+3] and ("•" in y[p+1] and "•" in
y[p+2]):
                        print("\n")
                        print(y[p+3])
                        try:
                            if v == "YES":
                                s=gTTS(text="hmm {}".
format(y[p+3]),lang='en-uk')
                                s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        except:
                            pass
                    if "•" in y[p+4] and ("•" in y[p+1] and "•" in
y[p+2] and "•" in y[p+3]):
                        print("\n")
```

```
                        print(y[p+4])
                        try:
                            if v == "YES":
                                s=gTTS(text="hmm {}".
format(y[p+4]),lang='en-uk')
                                s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        except:
                            pass
                    if "•" in y[p+5] and ("•" in y[p+1] and "•" in
y[p+2] and "•" in y[p+3] and "•" in y[p+4]):
                        print("\n")
                        print(y[p+5])
                        try:
                            if v == "YES":
                                s=gTTS(text="hmm {}".
format(y[p+5]),lang='en-uk')
                                s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        except:
                            pass
                    if "•" in y[p+6] and ("•" in y[p+1] and "•" in
y[p+2] and "•" in y[p+3] and "•" in y[p+4] and "•" in y[p+5]):
                        print("\n")
                        print(y[p+6])
                        try:
```

```
                        if v == "YES":
                            s=gTTS(text="hmm {}".
format(y[p+6]),lang='en-uk')
                            s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                            playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                            os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        except:
                            pass
                    except:
                        pass
                except:
                    try:
                        l= q_.split()
                        try:
                            l.remove("sid")
                            w=' '.join(l)
                        except:
                            l.remove("glenda")
                            w=' '.join(l)
                    except:
                        w=q_
                    print("Just Sit Back And Hold Tight!")
                    webbrowser.open("https://google.com/
search?q={}".format(w))
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
                    keyboard.press_and_release('win + Right')
                    time.sleep(1)
```

```
            except:
                try:
                    try:
                        l= q_.split()
                        try:
                            l.remove("sid")
                            w=' '.join(l)
                        except:
                            l.remove("glenda")
                            w=' '.join(l)
                    except:
                        w=q_
                    print("Just Sit Back And Hold Tight!")
                    webbrowser.open("https://google.com/
search?q={}".format(w))
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
                    keyboard.press_and_release('win + Right')
                    time.sleep(1)
                except requests.exceptions.ConnectionError:
                    print(" \t⚠Oops...Make Sure That You Are
Connected With Internet⚠ ")
                    if v == "YES":
                        playsound(r"C:\Python
Files\Audiofiles\offline.mp3")
            elif "what is" in q_:
                try:
                    try:
                        if "who" in q_ or "glenda" in q_:
                            l= q_.split()
                            try:
```

```
                    l.remove("sid")
                    w=' '.join(l)
                except:
                    l.remove("glenda")
                    w=' '.join(l)
            else:
                w=q_
            r = requests.get("https://google.com/
search?q={} wikipedia".format(q_))
            s = BeautifulSoup(r.content, 'html.parser')
            e=[]
            for a_ in s.find_all('a', href=True):
                e.append(a_['href'])
            x=[]
            for t in e:
                if "/url?q=https://en.wikipedia.org/wiki/" in
t:
                    x.append(e.index(t))
            l=e[x[0]]
            i=l.index('&')
            q=l[37:i]
            if '%' in q:
                print("Just Sit Back And Hold Tight!")
                webbrowser.open("https://google.com/
search?q={}".format(w))
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Right')
                time.sleep(1)
            else:
                print("\t                    \||~~~~||||",q,"
```

```
||||~~~~||/ \n")
                    r=requests.get("https://en.wikipedia.org/
wiki/{}".format(q))
                    s = BeautifulSoup(r.text,"html.parser")
                    p=s.select('p')[0:6]
                    i='\n'.join([o.text for o in p])
                    if len(i) > 100:
                        print(i)
                        try:
                            for x in range(0,6):
                                if len(p[x].text)>100:
                                    f=(p[x].text)
                            if v == "YES":
                                s=gTTS(text="hmm {}".format(f),
lang='en-uk')
                                s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        except:
                            pass
                    else:
                        g = requests.get("https://google.com/
search?q={} in cambridge dictionary".format(w))
                        _s_ = BeautifulSoup(g.content, 'html.parser')
                        y=[]
                        for a in _s_.find_all('a', href=True):
                            y.append(a['href'])
                        z=[]
                        for t in y:
                            if "/url?q=https://dictionary.cambridge.
org/dictionary/english/" in t:
```

```
                z.append(y.index(t))
            l=y[z[0]]
            j=l.index('&')
            k=l[59:j]
            if k == "":
                print("ERROR 404:Not Found!")
            else:
                try:
                    r = requests.get("https://dictionary.
cambridge.org/dictionary/english/{}".format(k))
                    print(r)
                    s = BeautifulSoup(r.content, 'html.
parser')
                    print(k)
                    i= s.find(class_="ddef_h")
                    e=[]
                    for b in i.findAll():
                        e.append(b.text)
                    x=[]
                    for h in e:
                        if ":" in h:
                            x.append(e.index(h))
                    p_=e[x[0]]
                    print("As Per Cambridge Dictionary:",
p_)
                except:
                    pass
                try:
                    r_ = requests.get("https://www.
oxfordlearnersdictionaries.com/definition/english/{}".
format(k))
                    s_ = BeautifulSoup(r_.content, 'html.
parser')
                    i_= s_.find(class_="def")
```

```
                print("\nAs Per Oxford Dictionary:",(i_.
extract()).text)
            except:
                pass
            try:
                r__ = requests.get("https://www.
macmillandictionary.com/dictionary/british/{}".format(k))
                s__ = BeautifulSoup(r__.content, 'html.
parser')
                i__= s__.find(class_="DEFINITION")
                print("\nAs Per Macmillan Dictionary:",
(i__.extract()).text)
            except:
                pass
        try:
            try:
                if len((i_.extract()).text)>len((i__.
extract()).text):
                    m=(i_.extract()).text
                else:
                    m=(i__.extract()).text
            except:
                try:
                    m=(i_.extract()).text
                except:
                    m=(i__.extract()).text
            if v == "YES":
                s=gTTS(text="hmm {} means that {}".
format(k,m),lang='en-uk')
                s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
```

```
Files\Audiofiles\VoiceOver.mp3")
                    except:
                        pass
                except:
                    webbrowser.open("https://google.com/
search?q={}".format(w))
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
                    keyboard.press_and_release('win + Right')
                    time.sleep(1)
            except requests.exceptions.ConnectionError:
                print(" \t⚠Oops...Make Sure That You Are
Connected With Internet⚠ ")
                if v == "YES":
                    playsound(r"C:\Python Files\Audiofiles\offline.
mp3")
        elif "who is" in q_ or "how is" in q_ or "how was" in q_
or "who was" in q_ or "how did" in q_:
            try:
                try:
                    try:
                        try:
                            l= q_.split()
                            try:
                                l.remove("sid")
                                w=' '.join(l)
                            except:
                                l.remove("glenda")
                                w=' '.join(l)
                        except:
                            w=q_
                        r = requests.get("https://google.com/
```

```
search?q={} wikipedia".format(w))
                s = BeautifulSoup(r.content, 'html.parser')
                e=[]
                for a_ in s.find_all('a', href=True):
                   e.append(a_['href'])
                x=[]
                for t in e:
                   if "/url?q=https://en.wikipedia.org/wiki/" in
t:
                      x.append(e.index(t))
                l=e[x[0]]
                i=l.index('&')
                q=l[37:i]
                if '%' in q:
                   print("Just Sit Back And Hold Tight!")
                   webbrowser.open("https://google.com/
search?q={}".format(w))
                   time.sleep(1)
                   keyboard.press_and_release('win + Up')
                   time.sleep(1)
                   keyboard.press_and_release('win + Up')
                   time.sleep(1)
                   keyboard.press_and_release('win + Right')
                   time.sleep(1)
                else:
                   print("\t                    \||~~~~||||",q,"
||||~~~~||/ \n")
                   r=requests.get("https://en.wikipedia.org/
wiki/{}".format(q))
                   s = BeautifulSoup(r.text,"html.parser")
                   p=s.select('p')[0:6]
                   i='\n'.join([o.text for o in p])
                   if len(i)>100:
                      print(i)
```

```
                    try:
                        for x in range(0,6):
                            if len(p[x].text)>100:
                                f=(p[x].text)
                        if v == "YES":
                            s=gTTS(text="hmm {}".format(f),
lang='en-uk')
                            s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                            playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                            os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    except:
                        pass
                else:
                    r=requests.get("https://en.wikipedia.org/
wiki/{}".format(q))
                    s = BeautifulSoup(r.text,"html.parser")
                    i_=s.find(class_='infobox')
                    y=[]
                    for __l__ in i_.findAll('tr'):
                        y.append(__l__.text)
                        try:
                            g=''
                            g_=[]
                            for h in (__l__.text):
                                if h.isupper() or h.isdigit():
                                    if h.isdigit():
                                        g_.append(((__l__.text).index(h)))
                                        if ((__l__.text).index(h))==g_[0]:
                                            g=g+' '+h
                                        else:
                                            g=g+h
```

```
                                    elif h.isupper():
                                        g=g+' '+h
                                    else:
                                        g=g+h
                                else:
                                    g=g+h
                            print("·.·",g)
                        except:
                            print("₪",__l__.text)
                except:
                    webbrowser.open("https://google.com/
search?q={}".format(w))
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
                    keyboard.press_and_release('win + Right')
                    time.sleep(1)
            except requests.exceptions.ConnectionError:
                print(" \t⚠Oops...Make Sure That You Are
Connected With Internet⚠ ")
                if v == "YES":
                    playsound(r"C:\Python Files\Audiofiles\offline.
mp3")
        elif "when was" in q_ or "where was" in q_ or "where
is" in q_ or "where did" in q_ or "when is" in q_ or "when did"
 in q_:
            try:
                l= q_.split()
                try:
                    i=l.index("was")+1
                    w=' '.join(l[i:])
                except:
```

```
                i=l.index("is")+1
                w=' '.join(l[i:])
            try:
                if l[len(l)-1]=='?' or l[len(l)-1]=='.' or l[len(l)-
1]=='!':
                    i=len(l)-2
                else:
                    i=len(l)-1
                u=l[i]
            except:
                pass
            w_ = str(u).casefold()
            if '?' in w_ or '.' in w_ or '.' in w_ or ',' in w_:
                o=len(w_)-1
                w_=w_[0:o]
            else:
                w_ = str(u).casefold()
            try:
                r = requests.get("https://google.com/
search?q={} wikipedia".format(w))
                s = BeautifulSoup(r.content, 'html.parser')
                e=[]
                for a_ in s.find_all('a', href=True):
                    e.append(a_['href'])
                x=[]
                for t in e:
                    if "/url?q=https://en.wikipedia.org/wiki/" in
t:
                        x.append(e.index(t))
                l=e[x[0]]
                i=l.index('&')
                q=l[37:i]
                print(q)
                if "when" in q_:
```

```
                    if ("birth" in q_ or "born" in q_) and ("day" in
q_ or "anniversary" in q_):
                        w_='born'
                    elif ("Assassination" in q and ((("died" in q_
or "death" in q_) and ("day" in q_ or "anniversary")) or "die"
in q_ or "killed" in q_ or "murder" in q_ or "death" in q_ or "
assas" in q_)) or "started" in q_ or "end" in q_ or "period" in
q_:
                        w_='date'
                    elif (("died" in q_ or "death" in q_) and ("day"
in q_ or "anniversary")) or "die" in q_ or "killed" in q_ or "
murder" in q_ or "death" in q_ or "assas" in q_:
                        w_='died'
                    elif "construct" in q_ or "open" in q_ or "build"
in q_ or "built" in q_:
                        w_='open'
                    elif "release" in q_ or "out" in q_:
                        w_="release"
                elif "where" in q_:
                    if ("birth" in q_ or "born" in q_) and ("place" in
q_ or "location" in q_):
                        w_='born'
                    elif (("died" in q_ or "death" in q_) and ("place"
 in q_ or "location" in q_)):
                        w_='died'
                    else:
                        w_='location'
                if '%' in q:
                    print("Just Sit Back And Hold Tight!")
                    webbrowser.open("https://google.com/
search?q={}".format(w))
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
```

```
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Right')
                time.sleep(1)
            a=b=0
            for o in range(len(q)):
                if q[o].isalpha():
                    a=a+1
                elif q[o].isdigit():
                    b=b+1
                else:
                    pass
            if a<4 and b>=1:
                l=e[x[1]]
                i=l.index('&')
                q=l[37:i]
            else:
                pass
            r=requests.get("https://en.wikipedia.org/wiki/
{}".format(q))
            s = BeautifulSoup(r.text,"html.parser")
            _w_=w_.split()
            z=[]
            y=[]
            try:
                i_=s.find(class_='infobox geography vcard')
                try:
                    for __l__ in i_.findAll('tr'):
                        y.append(__l__.text)
                        if w_ in (__l__.text).casefold():
                            z.append(__l__.text)
                except:
                    for __l__ in i_.find():
                        y.append(__l__.text)
```

```
                    if w_ in (__l__.text).casefold():
                        z.append(__l__.text)

                except:
                    try:
                        i_=s.find(class_='infobox')
                        try:
                            for __l__ in i_.findAll('tr'):
                                y.append(__l__.text)
                                if w_ in (__l__.text).casefold():
                                    z.append(__l__.text)
                        except:
                            for __l__ in i_.find():
                                y.append(__l__.text)
                                if w_ in (__l__.text).casefold():
                                    z.append(__l__.text)
                    except:
                        try:
                            i_=s.find(class_='infobox vcard')
                            try:
                                for __l__ in i_.findAll('tr'):
                                    y.append(__l__.text)
                                    if w_ in (__l__.text).casefold():
                                        z.append(__l__.text)
                            except:
                                for __l__ in i_.find():
                                    y.append(__l__.text)
                                    if w_ in (__l__.text).casefold():
                                        z.append(__l__.text)
                        except:
                            try:
                                i_=s.find(class_='infobox biography
vcard')
                                try:
```

```
                    for __l__ in i_.findAll('tr'):
                        y.append(__l__.text)
                        if w_ in (__l__.text).casefold():
                            z.append(__l__.text)
                    except:
                        for __l__ in i_.find():
                            y.append(__l__.text)
                            if w_ in (__l__.text).casefold():
                                z.append(__l__.text)
                except:
                    try:
                        i_=s.find(class_='infobox vevent')
                        try:
                            for __l__ in i_.findAll('tr'):
                                y.append(__l__.text)
                                if w_ in (__l__.text).casefold():
                                    z.append(__l__.text)
                        except:
                            for __l__ in i_.find():
                                y.append(__l__.text)
                                if w_ in (__l__.text).casefold():
                                    z.append(__l__.text)
                    except:
                        try:
                            i_=s.find(class_='infobox biota')
                            for __l__ in i_.findAll('tr'):
                                y.append(__l__.text)
                                if w_ in (__l__.text).casefold():
                                    z.append(__l__.text)
                        except:
                            try:
                                b_='False'
                                i_=s.find(class_='infobox
hproduct')
```

```
                    for __l__ in i_.findChild():
                        y.append(__l__.text)
                        if w_ in (__l__.text).casefold():
                            z.append(__l__.text)
                except:
                    try:
                        i_=s.find(class_='infobox
bordered')
                        for __l__ in i_.findAll('tr'):
                            y.append(__l__.text)
                            if w_ in (__l__.text).casefold():
                                z.append(__l__.text)
                    except:
                        try:
                            c_='False'
                            i_=s.find(class_='infobox
vcard plainlist')
                            try:
                                for __l__ in i_.findAll('tr'):
                                    y.append(__l__.text)
                                    if w_ in (__l__.text).
casefold():
                                        z.append(__l__.text)
                            except:
                                for __l__ in i_.find():
                                    y.append(__l__.text)
                                    if w_ in (__l__.text).
casefold():
                                        z.append(__l__.text)
                        except:
                            try:
                                i_=s.find(class_='infobox
vevent haudio')
                                try:
```

```
                                    for __l__ in i_.findAll('tr'):
                                        y.append(__l__.text)
                                        if w_ in (__l__.text).
casefold():
                                            z.append(__l__.text)
                                    except:
                                        for __l__ in i_.find():
                                            y.append(__l__.text)
                                            if w_ in (__l__.text).
casefold():
                                                z.append(__l__.text)
                                except:
                                    try:
                                        i_=s.find(class_='vertical-
navbox nowraplinks')
                                        try:
                                            for __l__ in i_.
findAll('tr'):
                                                y.append(__l__.text)
                                                if w_ in (__l__.text).
casefold():
                                                    z.append(__l__.text)
                                        except:
                                            for __l__ in i_.find():
                                                y.append(__l__.text)
                                                if w_ in (__l__.text).
casefold():
                                                    z.append(__l__.text)
                                    except:
                                        try:
                                            i_=s.
find(class_='vertical-navbox vcard')
                                            try:
                                                for __l__ in i_.
```

```
findAll('tr'):
                                                    y.append(__l__.text)
                                                    if w_ in (__l__.text).
casefold():
                                                       z.append(__l__.
text)
                                                 except:
                                                    for __l__ in i_.find():
                                                       y.append(__l__.text)
                                                       if w_ in (__l__.text).
casefold():
                                                       z.append(__l__.
text)
                                              except:
                                                 i_=s.
find(class_='vertical-navbox nowraplinks hlist')
                                                 try:
                                                    for __l__ in i_.
findAll('tr'):
                                                       y.append(__l__.text)
                                                       if w_ in (__l__.text).
casefold():
                                                       z.append(__l__.
text)
                                                 except:
                                                    print("26")
                                                    for __l__ in i_.find():
                                                       y.append(__l__.text)
                                                       if w_ in (__l__.text).
casefold():
                                                       z.append(__l__.
text)
                    if "president" in w_ and len(z[0]) > 53:
                       p=(y).index(z[1])
```

```
                elif "Show map" in z[0]:
                    p=(y).index(z[1])
                elif i_==s.find(class_='infobox geography vcard')
 and len(z[0]) > 95:
                    try:
                        p=(y).index(z[2])
                    except:
                        try:
                            p=(y).index(z[1])
                        except:
                            p=(y).index(z[0])
                else:
                    p=(y).index(z[0])
                g=''
                for h in y[p]:
                    if h.isupper() or h.isdigit():
                        if (y[p][((y[p].index(h))-1)]).islower() or
(y[p][((y[p].index(h))+1)]).islower():
                            g=g+' '+h
                        elif (y[p][((y[p].index(h))-1)]).isalpha() and
(y[p][((y[p].index(h))+1)]).isdigit():
                            g=g+' '+h
                        else:
                            g=g+h
                    else:
                        g=g+h
                y[p]=g
                if "•" in y[p]:
                    print("29")
                    print("〜",y[p],"• 〜")
                elif i_==s.find(class_='vertical-navbox
nowraplinks hlist') or i_==s.find(class_='vertical-navbox
nowraplinks') or i_==s.find(class_='vertical-navbox vcard'):
                    print("30")
```

```
                print("↴",y[p])
            else:
                print("31")
                print(y[p])
            try:
                if v == "YES":
                    s=gTTS(text="hmm {}".format(y[p]),
lang='en-uk')
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except:
                pass
            try:
                if i_==s.find(class_='infobox bordered') and
c_=='True':
                    y_=y.index('Chemical formula\n')
                    if y_<p:
                        print("-⊷►",y[p+1])
                    else:
                        pass
                elif i_==s.find(class_='infobox biota') and
b_=='True':
                    print("-⊷►",y[p+1])
                try:
                    if v == "YES":
                        s=gTTS(text="hmm {}".format(y[p+1]),
lang='en-uk')
                        s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        playsound(r"C:\Python
```

```
Files\Audiofiles\VoiceOver.mp3")
                        os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    except:
                        pass
                except:
                    pass
                try:
                    if "•" not in y[p]:
                        if "•" in y[p+1]:
                            print("\n")
                            print(y[p+1])
                            try:
                                if v == "YES":
                                    s=gTTS(text="hmm {}".
format(y[p+1]),lang='en-uk')
                                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                            except:
                                pass
                        if "•" in y[p+2] and "•" in y[p+1]:
                            print("\n")
                            print(y[p+2])
                            try:
                                if v == "YES":
                                    s=gTTS(text="hmm {}".
format(y[p+2]),lang='en-uk')
                                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                    playsound(r"C:\Python
```

```
Files\Audiofiles\VoiceOver.mp3")
                                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                            except:
                                pass
                        if "•" in y[p+3] and ("•" in y[p+1] and "•" in
y[p+2]):
                            print("\n")
                            print(y[p+3])
                            try:
                                if v == "YES":
                                    s=gTTS(text="hmm {}".
format(y[p+3]),lang='en-uk')
                                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                            except:
                                pass
                        if "•" in y[p+4] and ("•" in y[p+1] and "•" in
y[p+2] and "•" in y[p+3]):
                            print("\n")
                            print(y[p+4])
                            try:
                                if v == "YES":
                                    s=gTTS(text="hmm {}".
format(y[p+4]),lang='en-uk')
                                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                    os.remove("C:\Python
```

```
Files\Audiofiles\VoiceOver.mp3")
                        except:
                            pass
                    if "•" in y[p+5] and ("•" in y[p+1] and "•" in
y[p+2] and "•" in y[p+3] and "•" in y[p+4]):
                        print("\n")
                        print(y[p+5])
                        try:
                            if v == "YES":
                                s=gTTS(text="hmm {}".
format(y[p+5]),lang='en-uk')
                                s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        except:
                            pass
                    if "•" in y[p+6] and ("•" in y[p+1] and "•" in
y[p+2] and "•" in y[p+3] and "•" in y[p+4] and "•" in y[p+5]):
                        print("\n")
                        print(y[p+6])
                        try:
                            if v == "YES":
                                s=gTTS(text="hmm {}".
format(y[p+6]),lang='en-uk')
                                s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        except:
```

```
                        pass
                    except:
                        pass
                except:
                    try:
                        l= q_.split()
                        try:
                            l.remove("sid")
                            w=' '.join(l)
                        except:
                            l.remove("glenda")
                            w=' '.join(l)
                    except:
                        w=q_
                    print("Just Sit Back And Hold Tight!")
                    webbrowser.open("https://google.com/
search?q={}".format(w))
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
                    keyboard.press_and_release('win + Up')
                    time.sleep(1)
                    keyboard.press_and_release('win + Right')
                    time.sleep(1)
            except requests.exceptions.ConnectionError:
                print(" \t⚠Oops...Make Sure That You Are
Connected With Internet⚠ ")
                if v == "YES":
                    playsound(r"C:\Python Files\Audiofiles\offline.
mp3")
        elif (("hey" in q_ or "hi" in q_ or "hay" in q_ or "ok" in
q_) and ("siri" in q_ or "google" in q_ or "cortana" in q_ or "
bixby" in q_ or "alexa" in q_)):
            r=["For clearification: I'm Sid", "That's Awkward!", "
```

```
That's Awful", "I did'nt Expect that!", "You're dissappointing
Me!", "You're here with different Person in a different
Platform", "I thought you Recognise me!"]
            r_=random.randint(0,6)
            t_=r[r_]
            print(t_)
            r1=["That is Awkward!", "That is Awful", "I did'nt
Expect that!", "You're dissappointing Me!", "You're here
with different Person in a different Platform", "I thought you
Recognise me!"]
            r1_=random.randint(0,5)
            t=r1[r1_]
            if v == "YES":
               s=gTTS(text="hmm,{}".format(t),lang='en-uk')
               s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
               playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
               os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
         elif "scan file" in q_ or "scan folder" in q_ or "scan disk"
 in q_ or "scan location" in q_ or " scan" in q_:
            try:
               if v == "YES":
                  s=gTTS(text="Enter the Location",lang='en-uk')
                  s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                  playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                  os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except requests.exceptions.ConnectionError:
               pass
            try:
```

```
            l=input("Enter the location:")
            os.chdir('{}'.format(l))
            for dirpath, dirnames, filenames in os.walk(os.
getcwd()):
                print("File Name:", filenames)
                print("Current path:", dirpath)
                print("Files available:", dirnames)
            print("Scanning Completed! These are the
Results")
            try:
                if v == "YES":
                    s=gTTS(text="Scanning Completed! These are
the Results",lang='en-uk')
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except requests.exceptions.ConnectionError:
                pass
        except:
            print("Error: InValid Location")
            try:
                if v == "YES":
                    s=gTTS(text="Couldn't pass on procedures
due to Invalid location ",lang='en-uk')
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except requests.exceptions.ConnectionError:
```

```
                pass
        elif "iam" in q_ or "i'm" in q_:
            print("I'm Glad to Acknowledge that!")
            try:
                if v == "YES":
                    s=gTTS(text="I'm Glad to Acknowledge that!",
lang='en-uk')
                    s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except requests.exceptions.ConnectionError:
                pass
        elif "sid" in q_ and "you" in q_:
            print("Hmm that was UnExpected!")
            try:
                if v == "YES":
                    s=gTTS(text="hmm he didn't expect that!",
lang='en-uk')
                    s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except requests.exceptions.ConnectionError:
                pass
        elif "my" in q_ and "is" in q_:
            print("I'm Glad to Acknowledge that!")
            try:
                if v == "YES":
                    s=gTTS(text="I'm Glad to Acknowledge that!",
```

```
lang='en-uk')
                s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except requests.exceptions.ConnectionError:
                pass
        elif ("open" in q_ and ("search" in q_ or "look" in q_ or "
find" in q_)) or (("search" in q_ or "look" in q_ or "find" in q_
or "some" in q_ or "get" in q_) and ("on" in q_ or "in" in q_ or "
for" in q_ or "over" in q_ or "from" in q_)) or (("on" in q_ or "
in" in q_ or "over") and "for" in q_):
            try:
                if "sid" in q_ or "glenda" in q_:
                    l= q_.split()
                    try:
                        l.remove("sid")
                        w=' '.join(l)
                    except:
                        l.remove("glenda")
                        w=' '.join(l)
                else:
                    w=q_
                l= w.split()
                try:
                    i=l.index("open")+1
                    j=l.index("and")
                    w_=' '.join(l[i:j])
                    i_=l.index("for")+1
                    q=' '.join(l[i_:])
                except:
                    try:
```

```
    i_=l.index("for")+1
    j_=l.index("in")
    q=' '.join(l[i_:j_])
    i_=l.index("in")+1
    w_=' '.join(l[i:])
  except:
    try:
      i_=l.index("look")+1
      j_=l.index("for")
      q=' '.join(l[i_:j_])
      i=l.index("in")+1
      w_=' '.join(l[i:])
    except:
      try:
        i_=l.index("search")+1
        j_=l.index("for")
        q=' '.join(l[i_:j_])
        i=l.index("in")+1
        w_=' '.join(l[i:])
      except:
        try:
          i_=l.index("find")+1
          j_=l.index("in")
          q=' '.join(l[i_:j_])
          i=l.index("in")+1
          w_=' '.join(l[i:])
        except:
          try:
            i_=l.index("search")+1
            j_=l.index("in")
            q=' '.join(l[i_:j_])
            i_=l.index("in")+1
            w_=' '.join(l[i:])
          except:
```

```
try:
   i=l.index("in")+1
   j=l.index("for")
   w_=' '.join(l[i:j])
   i_=l.index("for")+1
   q=' '.join(l[i_:])
except:
   try:
      i=l.index("on")+1
      j=l.index("search")
      w_=' '.join(l[i:j])
      i_=l.index("for")+1
      q=' '.join(l[i_:])
   except:
      try:
         i=l.index("on")+1
         j=l.index("look")
         w_=' '.join(l[i:j])
         i_=l.index("for")+1
         q=' '.join(l[i_:])
      except:
         try:
            i_=l.index("some")+1
            j_=l.index("in")
            q=' '.join(l[i_:j_])
            i=l.index("in")+1
            w_=' '.join(l[i:])
         except:
            try:
               i_=l.index("some")+1
               j_=l.index("from")
               q=' '.join(l[i_:j_])
               i=l.index("from")+1
               w_=' '.join(l[i:])
```

```
                except:
                    try:
                        i_=l.index("some")+1
                        j_=l.index("on")
                        q=' '.join(l[i_:j_])
                        i=l.index("on")+1
                        w_=' '.join(l[i:])
                    except:
                        try:
                            i=l.index("to")+1
                            j=l.index("and")
                            w_=' '.join(l[i:j])
                            i_=l.index("for")+1
                            q=' '.join(l[i_:])
                        except:
                            try:
                                i=l.index("over")+1
                                j=l.index("for")
                                w_=' '.join(l[i:j])
                                i_=l.index("for")+1
                                q=' '.join(l[i_:])
                            except:
                                try:
                                    i=l.index("me")+1
                                    j=l.index("in")
                                    q=' '.join(l[i:j])
                                    i_=l.index("in")+1
                                    w_=' '.join(l[i_:])
                                except:
                                    i=l.index("search")
+1
                                    j=l.index("for")
                                    w_=' '.join(l[i:j])
                                    i_=l.index("for")+1
```

```
                                                    q=' '.join(l[i_:])
l_=w_.split()
if w_ == "" or q == "" or len(l_) > 1:
    try:
        i=l.index("in")+1
        j=l.index("search")
        w_=' '.join(l[i:j])
        i_=l.index("for")+1
        q=' '.join(l[i_:])
    except:
        try:
            i=l.index("on")+1
            j=l.index("search")
            w_=' '.join(l[i:j])
            i_=l.index("for")+1
            q=' '.join(l[i_:])
        except:
            try:
                i=l.index("in")+1
                j=l.index("find")
                w_=' '.join(l[i:j])
                i_=l.index("find")+1
                q=' '.join(l[i_:])
            except:
                try:
                    i=l.index("on")+1
                    j=l.index("find")
                    w_=' '.join(l[i:j])
                    i_=l.index("find")+1
                    q=' '.join(l[i_:])
                except:
                    try:
                        i_=l.index("for")+1
                        j_=l.index("from")
```

```
    q=' '.join(l[i_:j_])
    i=l.index("from")+1
    w_=' '.join(l[i:])
  except:
    try:
      i=l.index("in")+1
      j=l.index("search")
      w_=' '.join(l[i:j])
      i_=l.index("for")+1
      q=' '.join(l[i_:])
    except:
      try:
        i=l.index("in")+1
        j=l.index("look")
        w_=' '.join(l[i:j])
        i_=l.index("for")+1
        q=' '.join(l[i_:])
      except:
        try:
          i=l.index("in")+1
          j=l.index("search")
          w_=' '.join(l[i:j])
          i_=l.index("search")+1
          q=' '.join(l[i_:])
        except:
          try:
            i=l.index("in")+1
            j=l.index("look")
            w_=' '.join(l[i:j])
            i_=l.index("look")+1
            q=' '.join(l[i_:])
          except:
            try:
              i=l.index("in")+1
```

```
            j=l.index("for")
            w_=' '.join(l[i:j])
            i_=l.index("for")+1
            q=' '.join(l[i_:])
        except:
            try:
                i_=l.index("few")+1
                j_=l.index("in")
                q=' '.join(l[i_:j_])
                i=l.index("in")+1
                w_=' '.join(l[i:])
            except:
                try:
                    i_=l.index("few")+1
                    j_=l.index("from")
                    q=' '.join(l[i_:j_])
                    i=l.index("from")+1
                    w_=' '.join(l[i:])
                except:
                    try:
                        i_=l.index("few")+1
                        j_=l.index("on")
                        q=' '.join(l[i_:j_])
                        i=l.index("on")+1
                        w_=' '.join(l[i:])
                    except:
                        try:
                            i=l.index("look")+1
                            j=l.index("for")
                            w_=' '.join(l[i:j])
                            i_=l.index("for")+1
                            q=' '.join(l[i_:])
                        except:
                            try:
```

```
                                                    i=l.index("
search")+1
                                                    j=l.index("for")
                                                    w_=' '.join(l[i:j])
                                                    i_=l.index("for")
+1
                                                    q=' '.join(l[i_:])
                                                 except:
                                                    try:
                                                       i=l.index("me")
+1
                                                       j=l.index("on")
                                                       q=' '.join(l[i:j])
                                                       i_=l.index("on"
)+1
                                                       w=' '.join(l[i_:])
                                                    except:
                                                       pass
                r = requests.get("https://google.com/
search?q={}".format(w))
                s = BeautifulSoup(r.content, 'html.parser')
                e=[]
                for a_ in s.find_all('a', href=True):
                   e.append(a_['href'])
                x=[]
                for t in e:
                   if "/url?q=https://www." in t:
                      x.append(e.index(t))
                l=e[x[0]]
                j=l[19:].index('.')
                w=l[19:19+j]
                l_=q.split()
                if w_==" " or w_=="":
                   w_=w
```

```
            elif w in l_ or "in" in l_ or "from" in l_ or "on" in l_:
                if w in l_:
                    l_.remove(w)
                if "in" in l_:
                    l_.remove("in")
                if "on" in l_:
                    l_.remove("on")
                if "for" in l_:
                    l_.remove("for")
                q=' '.join([x for x in l_ if x in l_])
                w_=w
            print(q,l_)
            if Counter(w)==Counter(w_) or len(w)==len(w_)
or w[0]==w_[0] or w[len(w)-1]==w_[len(w)-1] or " " not in
w_:
                w_=w
            else:
                w_=q
            print(w)
            print("Just Sit Back And Hold Tight!")
            print("Here We Are Moving to {}.com".format(w))
            if w is True:
                pass
            elif "amazon" in w:
                webbrowser.open("https://www.amazon.in/
s?k={}".format(q))
            elif "snapdeal" in w:
                webbrowser.open("https://www.snapdeal.
com/search?keyword={}".format(q))
            elif "ebay" in w:
                webbrowser.open("https://www.ebay.com/
sch/i.html?_from=R40&_trksid=m570.l1313&_nkw={}".
format(q))
            elif "olx" in w:
```

```
                webbrowser.open("https://www.olx.in/items/
q-{}?isSearchCall=true".format(q))
            elif "paytm" in w:
                webbrowser.open("https://paytmmall.com/
shop/search?q={}".format(q))
            elif "myntra" in w:
                webbrowser.open("https://www.myntra.com/
{}".format(q))
            elif "bigbasket" in w:
                webbrowser.open("https://www.bigbasket.
com/ps/?q={}".format(q))
            elif "pepperfry" in w:
                webbrowser.open("https://www.pepperfry.
com/site_product/search?q={}+&as=0&src=os".format(q))
            elif "alibaba" in w:
                webbrowser.open("https://www.alibaba.com/
showroom/{}.
html?fsb=y&IndexArea=product_en&CatId=&SearchText={}
&isGalleryList=G".format(q,q))
            elif "aliexpress" in w:
                webbrowser.open("https://www.aliexpress.
com/premium/{}.
html?d=y&origin=y&catId=0&SearchText={}".format(q,q))
            elif "yahoo" in w:
                webbrowser.open("https://www.search.yahoo.
com/search?p={}".format(q))
            elif "google" in w and "news" in w:
                webbrowser.open("https://news.google.com/
search?q={}&hl=en-US&gl=US&ceid=US%3Aen".format(q))
            else:
                webbrowser.open("https://www.{}.com/
search?q={}".format(w,q))
            time.sleep(1)
            keyboard.press_and_release('win + Up')
```

```
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Right')
                time.sleep(1)
                if v == "YES":
                    s=gTTS(text="hmm These are the results for {}
in {}".format(q,w),lang='en-uk')
                    s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except requests.exceptions.ConnectionError:
                print(" \t⚠Oops...Make Sure That You Are
Connected With Internet⚠")
                if v == "YES":
                    playsound(r"C:\Python Files\Audiofiles\offline.
mp3")
        elif "open" in q_ or "go to" in q_ or "take me" in q_ or "
move to" in q_:
            try:
                try:
                    if "sid" in q_ or "glenda" in q_:
                        l= q_.split()
                        try:
                            l.remove("sid")
                            w=' '.join(l)
                        except:
                            l.remove("glenda")
                            w=' '.join(l)
                    else:
                        w=q_
```

```
            except:
                pass
            if (".com" in q_ or ("dot" in q_ and "com" in q_) or "
amazon" in q_ or "youtube" in q_ or "yahoo" in q_ or "
bigbasket" in q_ or "bing" in q_ or "aliexpress" in q_ or "ebay"
 in q_ or  "flipkart" in q_ or "snapdeal" in q_ or "browser" in
q_ or "search engine" in q_ or "web" in q_ or ("book" in q_
and "my" in q_ and "show" in q_)) and ("app" not in q_ and "
sett" not in q_):
                try:
                    r = requests.get("https://google.com/
search?q={}".format(w))
                    s = BeautifulSoup(r.content, 'html.parser')
                    e=[]
                    for a_ in s.find_all('a', href=True):
                        e.append(a_['href'])
                    x=[]
                    x_=[]
                    for t in e:
                        if "/url?q=https://www." in t:
                            x.append(e.index(t))
                        elif "/url?q=https://in." in t and ".mail." not
in t:
                            x_.append(e.index(t))
                        else:
                            pass
                    try:
                        if x[0]<x_[0]:
                            l=e[x[0]]
                            j=l[19:].index('.')
                            w=l[19:19+j]
                        else:
                            l=e[x_[0]]
                            j=l[18:].index('.')
```

```
                    w=l[18:18+j]
                except:
                    l=e[x[0]]
                    j=l[19:].index('.')
                    w=l[19:19+j]
                print("Just Sit Back And Hold Tight!")
                if "yandex" in q_:
                    w="yandex"
                elif "baibu" in q_:
                    w="baidu"
                if "private" in q_ and "browser":
                    w="duckduckgo"
                print("We Are Heading To {}...".format(w))
                webbrowser.open("https://www.{}.com".
format(w))
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Right')
                time.sleep(1)
                if v == "YES":
                    s=gTTS(text="hmm Here we are moving to
{}dot com".format(w),lang='en-uk')
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except:
                l=q_.split()
                try:
```

```
                    i=l.index("open")+1
                    w=' '.join(l[i:])
                except:
                    i=l.index("to")+1
                    w=' '.join(l[i:])
                print("Just Sit Back And Hold Tight!")
                print("We Are Heading To {}...".format(w))
                webbrowser.open("https://www.{}".
format(w))
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Right')
                time.sleep(1)
            else:
                l= w.split()
                try:
                    i=l.index("open")+1
                    j=l.index("app")+1
                    w=' '.join(l[i:j])
                except:
                    try:
                        i=l.index("open")+1
                        w=' '.join(l[i:])
                    except:
                        try:
                            i=l.index("to")+1
                            j=l.index("app")+1
                            w=' '.join(l[i:j])
                        except:
                            i=l.index("to")+1
                            w=' '.join(l[i:])
```

```
                keyboard.press_and_release('win')
                time.sleep(1)
                keyboard.write('{}'.format(w))
                time.sleep(1)
                keyboard.press_and_release('Enter')
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Right')
                time.sleep(1)
                print("Just Sit Back And Hold Tight!")
                print("Opening {}...".format(w))
                if v == "YES":
                    s=gTTS(text="hmm And this is {}".format(w),
lang='en-uk')
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except:
                try:
                    try:
                        l= q_.split()
                        try:
                            l.remove("sid")
                            w=' '.join(l)
                        except:
                            l.remove("glenda")
                            w=' '.join(l)
                    except:
```

```
                w=q_
                print("Just Sit Back And Hold Tight!")
                webbrowser.open("https://google.com/
search?q={}".format(w))
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Up')
                time.sleep(1)
                keyboard.press_and_release('win + Right')
                time.sleep(1)
                try:
                    if v == "YES":
                        s=gTTS(text="hmm These are results for {}
in internet".format(w),lang='en-uk')
                        s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                except:
                    pass
            except requests.exceptions.ConnectionError:
                print(" \t⚠Oops...Make Sure That You Are
Connected With Internet⚠")
                if v == "YES":
                    playsound(r"C:\Python
Files\Audiofiles\offline.mp3")
        elif "thanks" in q_ or "thank you" in q_:
            if "very" in q_:
                print("So nice of you, you're Welcome")
                if v == "YES":
                    s=gTTS(text="hmm so nice of you, you're
```

```
welcome",lang='en-uk')
                    s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                else:
                    print("You're Welcome")
                    if v == "YES":
                        s=gTTS(text="hmm you're welcome",lang='en-
uk')
                        s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                        playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            elif "restart" in q_ or "reboot" in q_:
                print("ReBooting...")
                try:
                    if v == "YES":
                        s=gTTS(text="hmm Computer Rebooting now!",
lang='en-uk')
                        s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                        playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                        os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                except requests.exceptions.ConnectionError:
                    pass
                os.system('shutdown -t 0 -r -f')
            elif "shut" in q_ and "down" in q_:
```

```
            print("Shutting Down...")
            try:
               if v == "YES":
                  s=gTTS(text="hmm Computer Shutting down!",
lang='en-uk')
                  s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                  playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                  os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except requests.exceptions.ConnectionError:
               pass
            os.system('shutdown /s /t 1')
         elif ("log" in q_ and "out" in q_) or ("sign" in q_ and "
out" in q_):
            print("SigningOut...")
            try:
               if v == "YES":
                  s=gTTS(text="hmm Computer Signing out!",
lang='en-uk')
                  s.save(r"C:\Python Files\Audiofiles\VoiceOver.
mp3")
                  playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                  os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except requests.exceptions.ConnectionError:
               pass
            os.system('shutdown -l')
         else:
            try:

               if "sid" in q_ or "glenda" in q_:
```

```
                l= q_.split()
                try:
                    l.remove("sid")
                    w=' '.join(l)
                except:
                    l.remove("glenda")
                    w=' '.join(l)
            else:
                w=q_
            print("Just Sit Back And Hold Tight!")
            webbrowser.open("https://google.com/
search?q={}".format(q_))
            time.sleep(1)
            keyboard.press_and_release('win + Up')
            time.sleep(1)
            keyboard.press_and_release('win + Up')
            time.sleep(1)
            keyboard.press_and_release('win + Right')
            time.sleep(1)
            try:
                if v == "YES":
                    s=gTTS(text="hmm these are Results from
Internet! about {}".format(q_),lang='en-uk')
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
            except:
                try:
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    if v == "YES":
```

```
                    s=gTTS(text="hmm these are Results from
Internet! about {}".format(q_),lang='en-uk')
                    s.save(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    playsound(r"C:\Python
Files\Audiofiles\VoiceOver.mp3")
                    os.remove("C:\Python
Files\Audiofiles\VoiceOver.mp3")
                except:
                    pass
            except requests.exceptions.ConnectionError:
                print(" \t⚠Oops...Make Sure That You Are
Connected With Internet⚠")
                if v == "YES":
                    playsound(r"C:\Python Files\Audiofiles\offline.
mp3")
        finally:
            pass
#=========================================
==#
_q_=input().casefold()
if _q_ == "":
    print("Hi I'm Sid, Virtual Assistant \n So,How can I help
you?”)
    process()
else:
    pass
‘’’
First Finished & Tested Prototype       |*| 20/04/2020-03:52
AM
First Updated & Tested Prototype        |*| 21/04/2020-12:51
AM
Second Updated & Tested Prototype   |*| 04/06/2020--
09:16 PM
```

Third Updated & Tested Prototype |*| 08/07/2020-09:46 PM
Forth Updated & Tested Prototype |*| 13/07/2020-10:39 PM
Fifth Updated & Tested Prototype |*| 14/07/2020-10:16 PM
Sixth Updated & Tested Prototype |*| 04/08/2020--05:32 PM
Seventh Updated & Tested Prototype |*| 04/09/2020--01:53 AM
Eighth Updated & Tested Prototype |*| 20/09/2020--04:13 AM
Ninth Updated & Tested Prototype |*| 20/09/2020--03:42 PM
Tenth Updated & Tested Prototype |*| 22/09/2020-04:18 AM
Eleventh Updated & Tested Prototype|*| 07/12/2020--09:34 PM
‘’’

# OUTPUT

## BIBLIOGRAPHY


- w3schools- https://www.w3schools.com/python/
- GeeksforGeeks- https://www.geeksforgeeks.org/python-programming-language/