11. SOURCE CODE

```
#nexusassistant.py
from tkinter import *
import textwrap
import os
import random
import datetime
import wikipedia
import time
import json
from googlesearch import search
from wikipedia.exceptions import WikipediaException
from bs4 import BeautifulSoup
from youtubesearchpython import VideosSearch
import requests
import math
import pyttsx3
import threading
import speech_recognition as sr
root = Tk()
root.config(bg="light grey")
root.geometry('1000x550+10+500')
root.title('Nexus')
root.iconbitmap('img/nexus.ico')
# Main window
canvas = Canvas(root, width=550, height=100, bg="snow")
canvas.grid(row=0, column=0, columnspan=2)
canvas.place(x=10, y=10, width=980, height=500)
# Set pyttsx3
engine = pyttsx3.init()
engine.setProperty('rate', 150)
engine.setProperty('volume', 1)
# Load json file with commands
with open("commands.json", "r", encoding='utf-8') as f:
COMMAND = json.load(f)
messages = []
class Me:
def __init__(self, master, message=""):
self.master = master
self.frame = Frame(master, bg="cyan")
self.i = self.master.create_window(
900, 200, window=self.frame, anchor="ne"
)
Label(self.frame, text=textwrap.fill(message, 100), font=(
"Sogue", 15), bg="cyan").grid(row=1, column=0, sticky="w", padx=1, pady=3)
root.update_idletasks()
ask.delete(0, END)
# Apply commands
```

```
if 'translate' in message:
translator(message)
elif 'wikipedia' in message:
Search(message)
elif 'timer' in message:
Timer(message)
elif 'timer' in message:
Timer(message)
elif 'google' in message:
Google(message)
elif 'note' in message:
Note(message)
elif 'video' in message:
find_video(message)
put_answer('Have a nice day!')
elif 'what is' in message:
x = Math_Operations(message)
put_answer(str(x))
elif 'weather' in message:
try:
temp, w = weather(message)
put_answer(f"The temperature is {temp}°C")
put_answer(w)
except Exception:
put_answer("I can't find your place")
else:
keys = COMMAND.keys()
for key in keys:
if message in key and len(message) > 1 or key == "error":
commands = COMMAND.get(key)
for command in commands[0]:
os.startfile(command)
answer = random.choice(commands[1])
put_answer(answer)
if "rock" in message or "scissors" in message or "paper" in message:
result = Rock(message, answer)
put_answer(result)
elif 'tell me a joke' in message or "joke" in message:
canvas.move(ALL, 0, -120)
elif 'clear' in message:
canvas.move(ALL, -1000, 0)
elif 'time' in message:
Time = datetime.datetime.now().strftime("%H:%M:%S")
put_answer(Time)
elif 'stop' in message or 'bay' in message or 'bey' in message or 'see you soon' in
message:
root.quit()
break
```

```
answers = []
class Assistant:
def __init__(self, master, answer=""):
self.master = master
self.frame = Frame(master, bg="dodger blue")
self.i = self.master.create_window(
20, 250 + 15, window=self.frame, anchor="nw")
Label(self.frame, text=textwrap.fill(answer, 100), font=("Sogue", 15),
bg="dodger blue").grid(row=1, column=0, sticky="w", padx=1, pady=3)
root.update_idletasks()
# Functions
def send_message():
canvas.move(ALL, 0, -110)
message = get_audio()
me = Me(canvas, message)
messages.append(me)
# ask.delete(0, END)
asking = random.choice(['What can I do for you!', 'How can I help?'])
put_answer(asking)
def write_message():
canvas.move(ALL, 0, -110)
me = Me(canvas, message=ask.get())
messages.append(me)
# ask.delete(0, END)
asking = random.choice(['What can I do for you!', 'How can I help?'])
put_answer(asking)
def put_answer(answer):
assistant = Assistant(canvas, answer=answer)
answers.append(assistant)
canvas.move(ALL, 0, -110)
canvas.update()
q = threading.Thread(target=speak(answer))
q.start()
def key(event=None):
q = threading.Thread(target=send_message)
q.start()
def key1(event=None):
q = threading.Thread(target=write_message)
q.start()
def wishMe():
hour = datetime.datetime.now().hour
if hour >= 0 and hour < 12:
put_answer("Hello user, Good Morning")
put_answer("How can I help?")
elif hour >= 12 and hour < 18:
put_answer("Hi user, Good Afternoon")
put_answer("How can I help?")
else:
```

```
put_answer("Hello User, Good Evening")
put_answer("How can I help?")
def Timer(message):
t = message.replace("timer", "")
try:
time.sleep(int(t))
put_answer('Time is over')
except Exception:
put_answer("Write your time")
def Rock(message, answer):
if message == 'rock' or message == 'paper' or message == 'scissors':
if answer == 'Rock' and message == 'paper':
result = 'You won'
elif answer == 'Paper' and message == 'rock':
result = 'I won'
elif answer == 'Scissors' and message == 'rock':
result = 'You won'
elif answer == 'Rock' and message == 'scissors':
result = 'I won'
elif answer == 'Paper' and message == 'scissors':
result = 'You won'
elif answer == 'Scissors' and message == 'paper':
result = 'I won'
else:
result = 'Draw'
result = 'Draw'
return result
def Search(message):
put_answer('Searching Wikipedia...')
statement = message.replace("wikipedia", "")
try:
results = wikipedia.summary(statement, sentences=3)
put_answer(results)
canvas.move(ALL, 0, -len(results)*0.68)
except WikipediaException:
put_answer('I can not find it')
def Google(message):
query = message.replace("google ", "")
j = search(query)
put_answer(f"Google {query}")
os.startfile(j[0])
# root.clipboard_clear()
# root.clipboard_append(j[0])
# root.update()
canvas.move(ALL, 0, -20)
def Note(message):
message = message.replace('note ', '')
root.clipboard_clear()
root.clipboard_append(message)
root.update()
os.startfile(COMMAND[0]['note'][0])
put_answer('Paste a note!')
video = message.replace("video ", "")
videosSearch = VideosSearch(video, limit=1)
info = videosSearch.result()
url = f"https://www.youtube.com/watch?v={info['result'][0]['id']}"
os.startfile(url)
def get_audio():
r = sr.Recognizer()
with sr.Microphone() as source:
audio = r.listen(source)
def get_audio():
said = ""
try:
said = r.recognize_google(audio)
except Exception:
put_answer('Error')
return said.lower()
def speak(text):
engine.say(text)
engine.runAndWait()
def Math_Operations(operation):
ex = operation.replace("what is ", "")
try:
if "sin" in ex:
ex = ex.replace("sin ", "")
result = math.sin(float(ex))
elif "cos" in ex:
ex = ex.replace("cos ", "")
result = math.cos(float(ex))
elif "factorial" in ex:
ex = ex.replace("factorial ", "")
result = math.factorial(int(ex))
# (Possibly more operations follow in the full code)
return result
except Exception:
return "I don't understand"
elif "binary" in ex:
ex = ex.replace("binary ", "")
ex = bin(int(ex))
result = ex.replace("0b", "")
else:
result = str(eval(ex))
{
"open": [
[],
[
"What you want to open?"
]
],
"open blender": [
[
"your_path"
],
[
"Blender has been opened!"
]
],
"open chrome": [
[
"C:\\Program Files\\Google\\Chrome\\Application\\chrome.exe"
],
[
"Chrome has been opened!"
]
],
"open file": [
[
"your_path"
],
[
"your_path"
]
],
"open gimp": [
[
"your_path"
],
[
"Gimp has been opened!"
]
],
"open minecraft": [
[
"your_path"
],
[
"Have a nice game"
]
],
"open word": [
[
"C:\\Program Files\\Microsoft Office\\Office15\\WINWORD.EXE"
],
[
"Word has been opened!"
]
],
"podcast": [
[
"your_path"
],
[
"Good listening"
]
],
"programming": [
[
"your_path"
],
[
"You can program!"
]
],
"school": [
[
"www.amjaincollege.com"
],
[
"Good luck"
]
],
"weekend": [
[
"your_path",
"your_path",
"your_path"
],
[
"Have a nice weekend."
]
]
}
"hi hello hey": [
[],
[
"Hello",
"Hi",
"Hey"
]
],
"how are you": [
[],
[
"I am fine",
"Great",
"Ok",
"Good",
"Super"
]
],
"how old are you": [
[],
[
"I was born 31.1 2023."
]
],
"open blender": [
[
"your_path"
],
[
"Blender has been opened!"
]
],
"open cmd": [
[
"your_path"
],
[
"CMD has been opened"
]
],
"open control panel": [
[
"your_path"
],
[
"Control has been opened"
]
],
"open excel": [
[
"your_path"
],
[
"Excel has been opened"
]
],
"open gmail": [
[
"your_path"
],
[
"Gmail has been opened!"
]
],
"open gimp": [
[
"your_path"
],
[
"Gimp has been opened!"
]
],
"open minecraft": [
[
"your_path"
],
[
"Have a nice game"
]
],
"open seznam": [
[
"your_path"
],
[
"Have a nice day"
]
],
"open teams": [
[
"your_path"
],
[
"Good luck"
]
],
"open youtube": [
[
"youtube.com"
],
[
"Have a nice day"
]
],
"send mail": [
[
"your_path"
],
[
"Gmail has been opened!"
]
],
"tell me a joke": [
[],
[
]
],
"what's your name": [
[],
```