Python Text to Speech

Contents

Time required: 60 minutes

- Comment each line of code as shown in the tutorials and other code examples.
- Follow all directions carefully and accurately.
- Think of the directions as minimum requirements.

Requirements Update

Requirements for tutorials and assignments are mean to be minimum guidelines. As long as you meet the requirements, you are free to explore, expand and modify your programs. That is where a large part of higher-level learning takes place. You are beyond the need for step-by-step directions for everything.

Comment your code so I or anything else can understand and carry on with your program.

Take the concepts you have learned; start creating your own unique, creative versions of the assignments.

pyttsx3

Our first step to building a DIY Iron Man suit is to get our computer to speak.

pyttsx3 is a text-to-speech conversion library in Python. Unlike alternative libraries, it works offline. An application invokes the pyttsx3.init() factory function to get a reference to a pyttsx3. Engine instance. it is a very easy to use tool which converts the entered text into speech.

The pyttsx3 module supports two voices in Windows. One is female and the second is male which is provided by “sapi5” for windows.

There are a lot of possibilities for creative programs using this library.

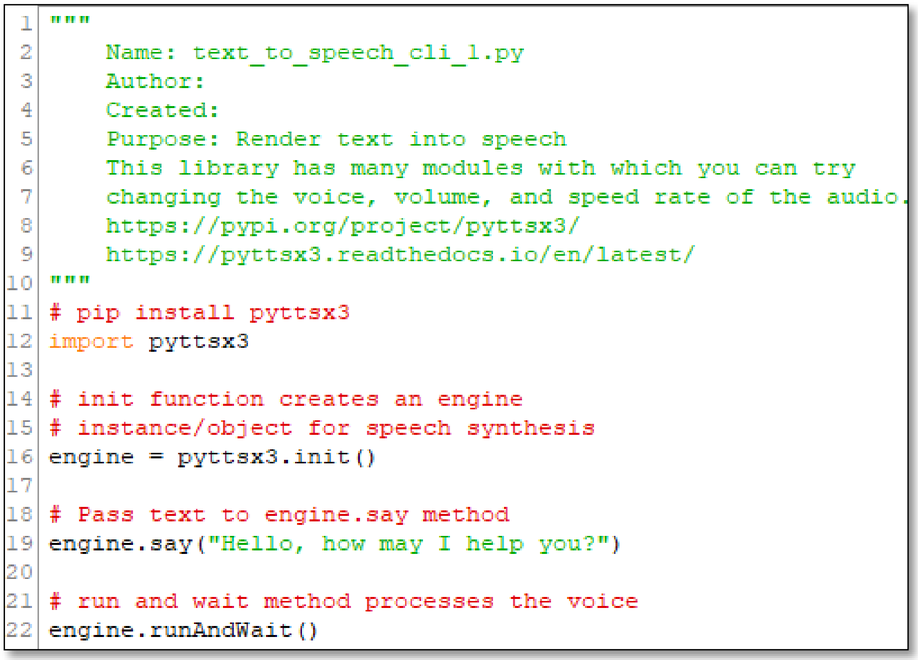
Pyttxs3 supports three TTS (Text to Speech) engines:

- sapi5 – SAPI5 on Windows
- nsss – NSSpeechSynthesizer on Mac OS X
- espeak – eSpeak on every other platform

Tutorial 1: Text to Speech CLI 1

The first step in learning a new concept is to start out simple and make sure it works.

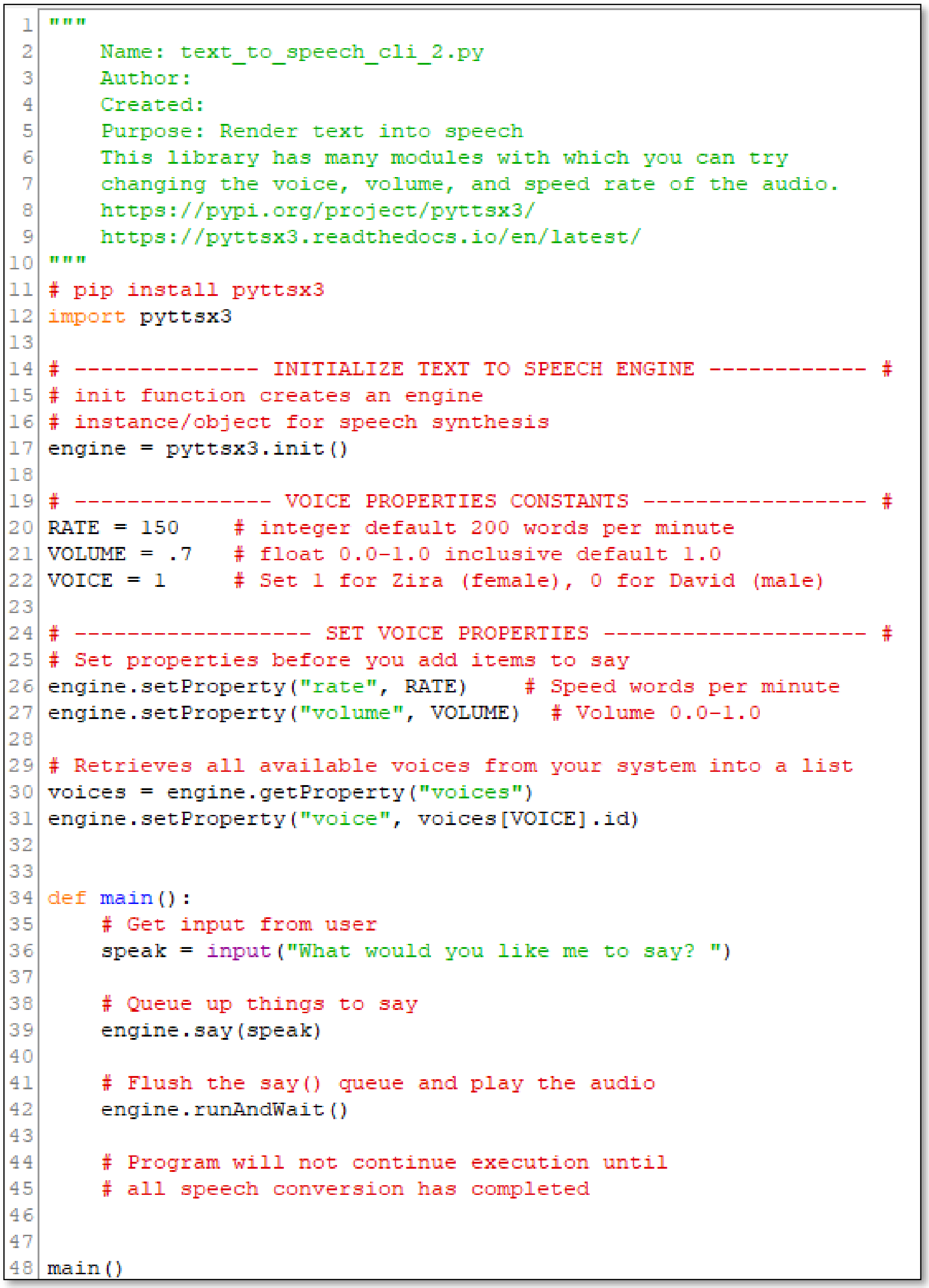
1. Open a command prompt: **pip install pyttsx3**
2. Create a Python program named **text_to_speech_cli_1.py**



```
"""
    Name: text_to_speech_cli_1.py
    Author:
    Created:
    Purpose: Render text into speech
    This library has many modules with which you can try
    changing the voice, volume, and speed rate of the audio.
    https://pypi.org/project/pyttsx3/
    https://pyttsx3.readthedocs.io/en/latest/
"""
# pip install pyttsx3
import pyttsx3

# init function creates an engine
# instance/object for speech synthesis
engine = pyttsx3.init()

# Pass text to engine.say method
engine.say("Hello, how may I help you?")

# run and wait method processes the voice
engine.runAndWait()
```

Tutorial 2: Text to Speech CLI 2

Let’s add a main() function and set some more properties.



```python
"""
    Name: text_to_speech_cli_2.py
    Author:
    Created:
    Purpose: Render text into speech
    This library has many modules with which you can try
    changing the voice, volume, and speed rate of the audio.
    https://pypi.org/project/pyttsx3/
    https://pyttsx3.readthedocs.io/en/latest/
"""
# pip install pyttsx3
import pyttsx3

# -------------- INITIALIZE TEXT TO SPEECH ENGINE ------------ #
# init function creates an engine
# instance/object for speech synthesis
engine = pyttsx3.init()

# --------------- VOICE PROPERTIES CONSTANTS ----------------- #
RATE = 150     # integer default 200 words per minute
VOLUME = .7    # float 0.0-1.0 inclusive default 1.0
VOICE = 1      # Set 1 for Zira (female), 0 for David (male)

# ------------------ SET VOICE PROPERTIES -------------------- #
# Set properties before you add items to say
engine.setProperty("rate", RATE)     # Speed words per minute
engine.setProperty("volume", VOLUME)  # Volume 0.0-1.0

# Retrieves all available voices from your system into a list
voices = engine.getProperty("voices")
engine.setProperty("voice", voices[VOICE].id)


def main():
    # Get input from user
    speak = input("What would you like me to say? ")

    # Queue up things to say
    engine.say(speak)

    # Flush the say() queue and play the audio
    engine.runAndWait()

    # Program will not continue execution until
    # all speech conversion has completed


main()
```

Assignment: Text to Speech OOP

Convert the Text to Speech program into Object Oriented Programming.

1. Create a new Python file: **text_to_speech_oop.py**

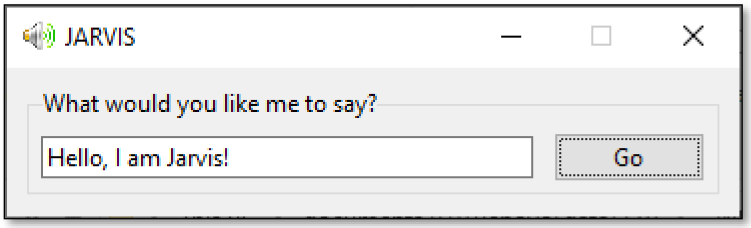
2. Copy and paste the following starter code into the new file.

3. Refer to **text_to_speech_cli_2.py** for the code to convert to OOP.

```
"""
    Name: text_to_speech_cli_2.py
    Author:
    Created:
    Purpose: Render text into speech
    This library has many modules with which you can try
    changing the voice, volume, and speed rate of the audio.
    https://pypi.org/project/pyttsx3/
    https://pyttsx3.readthedocs.io/en/latest/
"""
# pip install pyttsx3
import pyttsx3


class TextToSpeech():
    def __init__(self):
        # -------------- INITIALIZE TEXT TO SPEECH ENGINE ------------ #
        # init function creates an engine
        # instance/object for speech synthesis
        # initialize self.engine

        # Your code here

        # --------------- VOICE PROPERTIES CONSTANTS ----------------- #
        # The constants stay the same

        # Your code here

        # ------------------ SET VOICE PROPERTIES -------------------- #
        # Set properties before you add items to say
        # Use self.engine to set voice properties

        # Your code here

    def speak(self, speak):
        # self.engine say

        # Your code here

        # Program will not continue execution until
        # all speech conversion has completed

        # Your code here


def main():
    # Create text to speech object
    text_to_speech = TextToSpeech()

    while True:
        # Get input from user
        speak = input("What would you like me to say? ")

        # Queue up things to say
        text_to_speech.speak(speak)


main()
```

Extra Credit Challenge

Take your OOP and convert it to Tkinter.



What would you like me to say?

Hello, I am Jarvis!

Assignment Submission

1. Attach all program file to the assignment.

2. Submit in Blackboard.