any name

## Buzzer Interfacing with Raspberry Pi: Playing the Harry Potter Theme  
  
\*\*Aim:\*\* To interface a buzzer with a Raspberry Pi and play the Harry Potter theme melody using a Python program.  
  
\*\*Materials Required:\*\*  
  
\* Raspberry Pi (any model)  
\* Buzzer  
\* Jumper wires (male to female)  
\* Breadboard (optional)  
  
\*\*Code Snippet:\*\*  
  
```python  
import RPi.GPIO as GPIO  
import time  
  
# Define GPIO pin for the buzzer  
BUZZER\_PIN = 17  
  
# Set up GPIO  
GPIO.setmode(GPIO.BCM)  
GPIO.setup(BUZZER\_PIN, GPIO.OUT)  
  
# Define the Harry Potter theme melody notes and durations  
notes = [261.63, 293.66, 329.63, 349.23, 392, 440, 493.88, 523.25]  
durations = [0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5]  
  
# Function to play a note  
def play\_note(note, duration):  
 GPIO.output(BUZZER\_PIN, GPIO.HIGH)  
 time.sleep(duration)  
 GPIO.output(BUZZER\_PIN, GPIO.LOW)  
  
# Play the melody  
for i in range(len(notes)):  
 play\_note(notes[i], durations[i])  
  
# Clean up GPIO  
GPIO.cleanup()  
```  
  
\*\*Working:\*\*  
  
\* The program defines a GPIO pin for the buzzer and sets it as an output.  
\* It then defines a list of notes and durations for the Harry Potter theme melody.  
\* The `play\_note` function generates a tone by turning the buzzer on and off for the specified duration.  
\* The program iterates through the notes and durations, calling the `play\_note` function to generate the melody.  
\* Finally, the GPIO pins are cleaned up.  
  
\*\*Procedure:\*\*  
  
1. Connect the buzzer to the chosen GPIO pin (here, pin 17) using jumper wires.  
2. Ensure the buzzer is connected to a 5V power source.  
3. Save the code snippet as a Python file (e.g., "harry\_potter.py").  
4. Run the program using the command `python harry\_potter.py`.  
5. Listen to the buzzer playing the Harry Potter theme.  
  
\*\*Result:\*\*  
  
The buzzer played the Harry Potter theme melody, showcasing successful interfacing and programming of the buzzer with the Raspberry Pi.  
  
\*\*Learning:\*\*  
  
Students learned about interfacing electronic components with the Raspberry Pi using GPIO pins, understanding the concept of generating tones using a buzzer, and writing Python code to control the hardware and produce a melody.