

# Use CPX's capacitive touch A1 to A7 and the speaker to create a where is thumbkin? tune.

## Why It's Learned

This activity helps you understand how **capacitive touch sensors** and **built-in sound features** work in the CPX board.

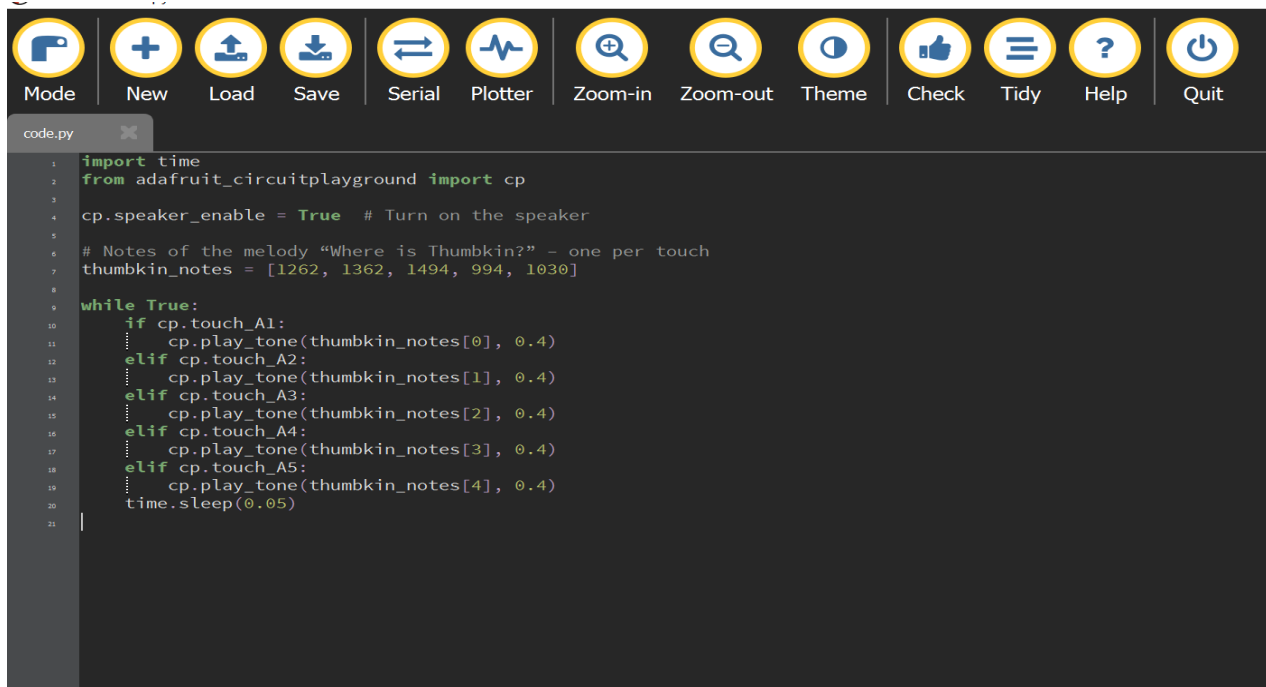
By learning this, you explore how to use **touch inputs** to control actions (like playing notes).

It also helps in learning the concept of **interactive inputs and outputs** — touch as input, sound as output.

## What Is Learned

- How to **enable and use the built-in speaker** in CPX.
- How to use **touch pads (A1 to A5)** as inputs.
- How to create and use an **array (list)** to store sound frequencies.
- How to play **different tones** based on which pad is touched.
- Understanding the **while True loop** for continuous checking of user input.

## Code



```
1 import time
2 from adafruit_circuitplayground import cp
3
4 cp.speaker_enable = True # Turn on the speaker
5
6 # Notes of the melody "Where is Thumbkin?" - one per touch
7 thumbkin_notes = [1262, 1362, 1494, 994, 1030]
8
9 while True:
10     if cp.touch_A1:
11         cp.play_tone(thumbkin_notes[0], 0.4)
12     elif cp.touch_A2:
13         cp.play_tone(thumbkin_notes[1], 0.4)
14     elif cp.touch_A3:
15         cp.play_tone(thumbkin_notes[2], 0.4)
16     elif cp.touch_A4:
17         cp.play_tone(thumbkin_notes[3], 0.4)
18     elif cp.touch_A5:
19         cp.play_tone(thumbkin_notes[4], 0.4)
20     time.sleep(0.05)
21
```

## Code Explanation (Line by Line)

import time

→ Adds the time module for small delays.

```
from adafruit_circuitplayground import cp
```

→ Imports the CPX library to use touch pads and speaker.

```
cp.speaker_enable = True
```

→ Turns on the built-in speaker to make sound.

```
thumbkin_notes = [1262, 1362, 1494, 994, 1030]
```

→ Creates an array (list) that stores five sound frequencies in hertz (Hz). Each value represents a musical note for the song “Where is Thumbkin?”.

```
while True:
```

→ Starts an infinite loop so the board keeps checking for touches forever.

```
if cp.touch_A1:
```

→ If pad A1 is touched, play the first note (1262 Hz).

```
elif cp.touch_A2:
```

→ If pad A2 is touched, play the second note (1362 Hz).

```
elif cp.touch_A3:
```

→ If pad A3 is touched, play the third note (1494 Hz).

```
elif cp.touch_A4:
```

→ If pad A4 is touched, play the fourth note (994 Hz).

```
elif cp.touch_A5:
```

→ If pad A5 is touched, play the fifth note (1030 Hz).

```
time.sleep(0.05)
```

→ Waits for a short time to avoid overlapping inputs.

## Output

- When the program runs, you can **touch the A1–A5 pads** on the Circuit Playground Express board.
- Each touch plays a **different musical note** through the built-in speaker.
- You can use your fingers to make a small tune, just like playing a mini piano.

Example:

Touch A1 → Sound 1

Touch A2 → Sound 2

Touch A3 → Sound 3

Touch A4 → Sound 4

Touch A5 → Sound 5