

Countdown Timer

We can use the time functions to create a simple timer. For this, we will get the current time in milliseconds using the `millis()` method.

```
int startTime;
int endTime = 5000;
int timePassed;

void setup() {
  size(200, 200);
  background(0);
  resetTimer();
}

void draw() {
  timePassed = millis() - startTime;
  if (timePassed > endTime) {
    background(255, 0, 0);
  }
}

void resetTimer() {
  startTime = millis();
}

void keyPressed() {
  if (key == ' ') {
    resetTimer();
  }
}
```

Image and Sound

We can put together our previous lesson to make a buzzer that makes sound after the time is up.

```
import processing.sound.*;

PImage buzzer;
SoundFile buzzerSound;

int startTime;
int endTime = 5000;
int timePassed;

PVector buzzerLocation;
int buzzerSize;
```

```

boolean runBuzzer = true;

void setup() {
  size(200, 200);
  background(0);
  buzzer = loadImage("buzzer.jpg");
  buzzerLocation = new PVector(width/2, height/2);
  buzzerSize = width/2;
  buzzerSound = new SoundFile(this, "buzzerSound.mp3");
  resetTimer();
}

void draw() {
  imageMode(CENTER);
  image(buzzer, buzzerLocation.x, buzzerLocation.y, buzzerSize, buzzerSize);
  if (runBuzzer) {
    tickTimer();
    checkTime();
  }
}

void resetTimer() {
  startTime = millis();
  background(0);
  runBuzzer = true;
}

void tickTimer() {
  timePassed = millis() - startTime;
}

void checkTime() {
  if (timePassed > endTime) {
    timeUp();
  }
}

void timeUp() {
  background(255, 0, 0);
  image(buzzer, buzzerLocation.x, buzzerLocation.y, buzzerSize, buzzerSize);
  buzzerSound.play();
  runBuzzer = false;
}

void mousePressed() {
  float buzzerLeft = buzzerLocation.x - buzzerSize/2;
  float buzzerRight = buzzerLocation.x + buzzerSize/2;
  float buzzerTop = buzzerLocation.y - buzzerSize/2;
  float buzzerBottom = buzzerLocation.y + buzzerSize/2;
  boolean overBuzzer = mouseX > buzzerLeft && mouseX < buzzerRight && mouseY > buzzerTop && mouseY < buzzerBottom &&

```

```
boolean overBuzzer = mouseX < BUZZERRIGHT && mouseX > BUZZERLEFT && mouseY < BUZZERBOTTOM &&
if (overBuzzer) {
    resetTimer();
}
}
```