

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1" />
<title>Pomodoro Clock</title>
<style>
  body {
    font-family: Arial, sans-serif;
    text-align: center;
    margin-top: 100px;
    background: #f0f0f0;
    color: #333;
  }
  #timer {
    font-size: 80px;
    margin-bottom: 20px;
  }
  #mode {
    font-size: 24px;
    margin-bottom: 40px;
  }
  button {
    font-size: 18px;
    padding: 10px 20px;
    margin: 0 10px;
    cursor: pointer;
  }
</style>
</head>
<body>
```

```
<h1>Pomodoro Clock</h1>
```

```
<div id="mode">Work</div>
```

```
<div id="timer">25:00</div>
```

```
<button id="pause">Pause</button>
```

```
<button id="stop">Stop</button>
```

```
<audio id="ringtone" src="https://actions.google.com/sounds/v1/alarms/alarm_clock.ogg" preload="auto"></audio>
```

```
<script>
```

```
  const workDuration = 25 * 60;
```

```
  const breakDuration = 5 * 60;
```

```
  let currentTime = workDuration;
```

```
  let isWork = true;
```

```
  let timerInterval = null;
```

```
  let isPaused = false;
```

```
  const timerEl = document.getElementById('timer');
```

```
  const modeEl = document.getElementById('mode');
```

```
  const pauseBtn = document.getElementById('pause');
```

```
  const stopBtn = document.getElementById('stop');
```

```
  const ringtone = document.getElementById('ringtone');
```

```
  function updateTimerDisplay() {
```

```
    let minutes = Math.floor(currentTime / 60);
```

```
    let seconds = currentTime % 60;
```

```
    timerEl.textContent =
```

```
      (minutes < 10 ? '0' + minutes : minutes) + ':' +
```

```
      (seconds < 10 ? '0' + seconds : seconds);
```

```
  }
```

```
  function switchMode() {
```

```
isWork = !isWork;
currentTime = isWork ? workDuration : breakDuration;
modeEl.textContent = isWork ? 'Work' : 'Break';
updateTimerDisplay();
}
```

```
function playRingtone() {
  ringtone.currentTime = 0;
  ringtone.play();
}
```

```
function tick() {
  if (!isPaused) {
    currentTime--;
    if (currentTime < 0) {
      playRingtone();
      switchMode();
    } else {
      updateTimerDisplay();
    }
  }
}
```

```
function startTimer() {
  if (!timerInterval) {
    timerInterval = setInterval(tick, 1000);
  }
}
```

```
pauseBtn.addEventListener('click', () => {
  if (isPaused) {
    isPaused = false;
    pauseBtn.textContent = 'Pause';
  }
});
```

```
    } else {  
        isPaused = true;  
        pauseBtn.textContent = 'Resume';  
    }  
});
```

```
stopBtn.addEventListener('click', () => {  
    clearInterval(timerInterval);  
    timerInterval = null;  
    isPaused = false;  
    pauseBtn.textContent = 'Pause';  
    isWork = true;  
    currentTime = workDuration;  
    modeEl.textContent = 'Work';  
    updateTimerDisplay();  
});
```

```
// Start timer automatically
```

```
startTimer();  
updateTimerDisplay();
```

```
</script>
```

```
</body>
```

```
</html>
```