

timer\src\Timer.js

```
1 import React, { useState, useEffect } from 'react';
2 import './Timer.css';
3
4 const Timer = () => {
5   const [time, setTime] = useState(0);
6   const [isRunning, setIsRunning] = useState(false);
7
8   useEffect(() => {
9     let interval = null;
10
11     if (isRunning) {
12       interval = setInterval(() => {
13         setTime((prevTime) => prevTime + 1);
14       }, 1000);
15     } else if (!isRunning && time !== 0) {
16       clearInterval(interval);
17     }
18
19     return () => clearInterval(interval);
20   }, [isRunning, time]);
21
22   const handleStartPause = () => {
23     setIsRunning(!isRunning);
24   };
25
26   const handleReset = () => {
27     setTime(0);
28     setIsRunning(false);
29   };
30
31   const formatTime = (time) => {
32     const getSeconds = `0${time % 60}`.slice(-2);
33     const minutes = Math.floor(time / 60);
34     const getMinutes = `0${minutes % 60}`.slice(-2);
35     const getHours = `0${Math.floor(time / 3600)}`.slice(-2);
36
37     if (getHours > 0) {
38       return `${getHours}:${getMinutes}:${getSeconds}`;
39     } else {
40       return `${getMinutes}:${getSeconds}`;
41     }
42   };
43
44   return (
45     <div className="timer-container">
46       <h1 className="timer-title">Timer</h1>
47       <div className="timer-display">{formatTime(time)}</div>
48       <div className="timer-controls">
49         <button className="timer-button start-pause" onClick={handleStartPause}>
50           {isRunning ? 'Pause' : 'Start'}
51         </button>
52         <button className="timer-button reset" onClick={handleReset}>
53           Reset
54         </button>
55       </div>
56     </div>
57   );
58 };
59
60 export default Timer;
```