

A clock component

Wednesday, March 23, 2022 5:48 PM

- We want to display a clock on our UI which can display time.
- How do we create the component

Version #1 We can set up the UI to display current time

```
class Clock extends React.Component {
```

```
  render(){
```

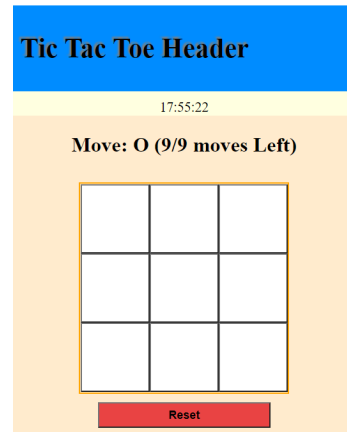
```
    var date=new Date();  
    var hour=date.getHours();  
    var min =date.getMinutes();  
    var second=date.getSeconds();
```

```
    return <div className="clock">  
      <span className="hour">{hour}</span>:  
      <span className="min">{min}</span>:  
      <span className="sec">{second}</span>  
    </div>;
```

```
  }
```

```
export default Clock;
```

- Displays the time when component was last rendered
- The time doesn't get automatically updated



How do I update the time every second?

- We can subscribe to setInterval()
- We can update the state of our component on ever setInterval call
- This subscription must be started in componentDidMount
- You should cancel subscription in componentWillUnmount

```
class Clock extends React.Component {
```

```
  state={  
    date:new Date()  
  };
```

```
  componentDidMount() {  
    this.id=setInterval(() =>{  
      this.setState({date:new Date()});  
    },1000);  
  }
```

```
  componentWillUnmount() {  
    if(this.id)  
      clearInterval(this.id);  
  }
```

```
  render(){
```

```
    function zero(x){...
```

```
    var date=this.state.date;  
    var hour=zero(date.getHours());  
    var min =zero(date.getMinutes());  
    var second=zero(date.getSeconds());
```

```
    return <div className="clock">  
      <span className="hour">{hour}</span>:
```

- Set up a timer subscription that can update your UI at required interval
- Remember the id of current timer. It will be required while unsubscribing
- Unsubscribe to any recurring event in componentWillUnmount