**Question- React component that shows the current time to the user.**

import {useEffect, useState} from 'react';

function Time(){

const [date, setDate] = useState(new Date());

useEffect(() => {

const timer = setInterval(() => setDate(new Date()), 1000)

return () => clearInterval(timer);

});

return (

<p>{`${date.toLocaleTimeString()}`}</p>

);

}

export default Time;

***Exact Center div***

**Using flex layout module**

.parent {

  display: flex;

  justify-content: center;

  align-items: center;

  width: 100%;

  height: 100vh;

}

**Using grid layout module**

.parent {

  display: grid;

  width: 100%;

  height: 100vh;

  place-items: center;

}

[How to Center a Div Using CSS Grid — SitePoint](https://www.sitepoint.com/css-grid-center-element/" \l ":~:text=Setting%20Up%201%201.%20Center%20a%20Div%20with,Center%20a%20Div%20with%20CSS%20Grid%20and%20place-items)

**Using position properties**

.parent {

  position: relative;

  width: 100%;

  height: 100vh;

}

.child {

  position: absolute;

  height: 100px;

  top: 50%;

  left: 50%;

  border: 1px solid black;

  transform: translate(-50%, -50%);

  width: 100px;

}

**React decrement timer**

import React, { useEffect, useState, useRef } from 'react';

function Timer() {

  const [val, setVal] = useState(10);

  const timer = useRef();

  const start = () => {

    timer.current = setInterval(() => {

      setVal((prevVal) => {

        if (prevVal > 0) {

          return prevVal - 1;

        }

        clearInterval(timer.current);

        return 0;

      });

    }, 1000);

  };

  const stop = () => {

    clearInterval(timer.current);

  };

  const reset = () => {

    setVal(10);

  };

  return (

    <>

      <p>{val? val: 'Time out'}</p>

      <button onClick={start}>Start</button>

      <button onClick={stop}>Stop</button>

      <button onClick={reset}>Reset</button>

    </>

  );

}

export default Timer;

**React increment timer**

import React, {useEffect, useState, useRef} from 'react';

function Timer(){

  const [val, setVal] = useState(0);

  const timer = useRef();

  const start = () => {

    timer.current = setInterval( () => {

      setVal(prevVal => prevVal + 1);

    }, 1000);

  }

  const stop = () => {

    clearInterval(timer.current);

  }

  const reset = () => {

    setVal(0);

  }

  return (

    <>

      <p>{val}</p>

      <button onClick={start}>Start</button>

      <button onClick={stop}>Stop</button>

      <button onClick={reset}>Reset</button>

    </>

  );

}

export default Timer;