**App.js**

import './App.css';

import EffectHook from './HooksComponent/EffectHook';

function App() {

  return (

    <div className="App">

      <EffectHook/>

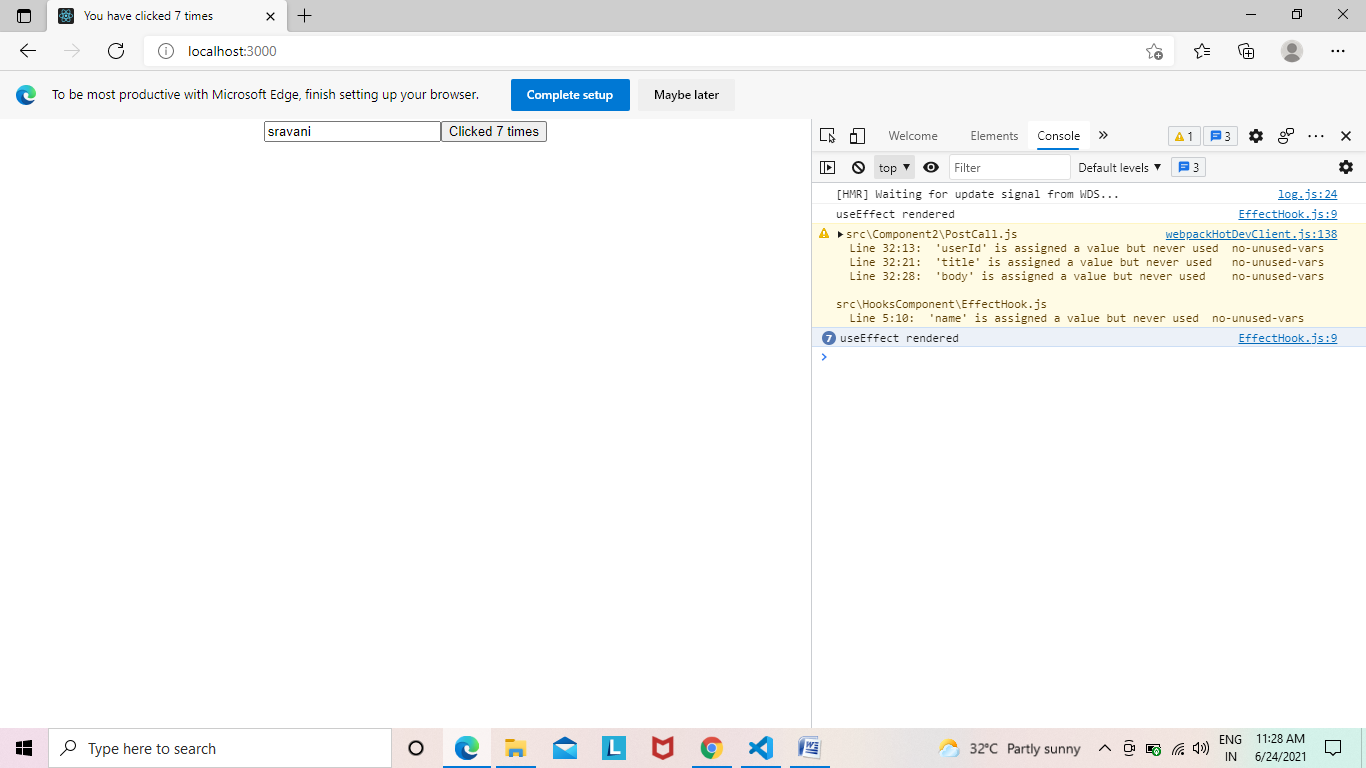
    </div>

  );

}

export default App;

**EffectHook.js**

****

**App.js**

import './App.css';

import MouseEvent from './HooksComponent/MouseEvent';

function App() {

  return (

    <div className="App">

      <MouseEvent/>

    </div>

  );

}

export default App;

**MouseEvent.js**

import React, { Component } from "react";

class MounseEvent extends Component {

  constructor(props) {

    super(props);

    this.state = {

      x: 0,

      y: 0,

    };

  }

  logMouseCordinates = (e) => {

    this.setState({ x: e.clientX, y: e.clientY });

  };

  componentDidMount() {

    window.addEventListener("mousemove", this.logMouseCordinates);

  }

  render() {

    return (

      <div>

        X- {this.state.x} Y- {this.state.y}

      </div>

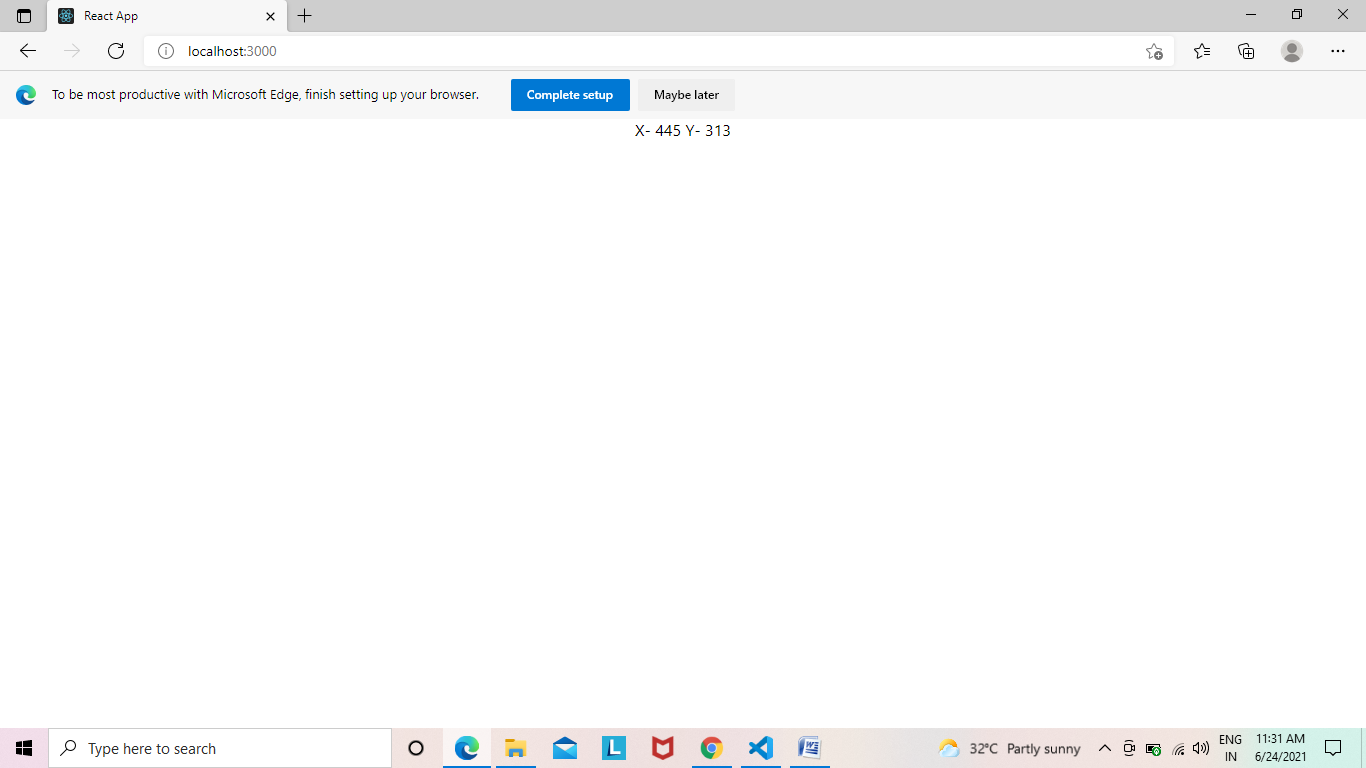
    );

  }

}

export default MounseEvent;

**Output:**

****

**App.js**

import './App.css';

import EffectMouse from './HooksComponent/EffectMouse';

function App() {

  return (

    <div className="App">

      <EffectMouse/>

    </div>

  );

}

export default App;

**EffectMouse.js**

import React, { useState, useEffect } from "react";

function EffectMouse() {

  const [x, setX] = useState(0);

  const [y, setY] = useState(0);

  const logMouseCordinates = (e) => {

    console.log("mouse event");

    setX(e.clientX);

    setY(e.clientY);

  };

  useEffect(() => {

    console.log("useEffect called");

    window.addEventListener("mousemove", logMouseCordinates);

    return () => {

      console.log("Component unmount");

      window.removeEventListener("mousemove", logMouseCordinates);

    };

  });

  return (

    <div>

      X- {x} Y- {y}

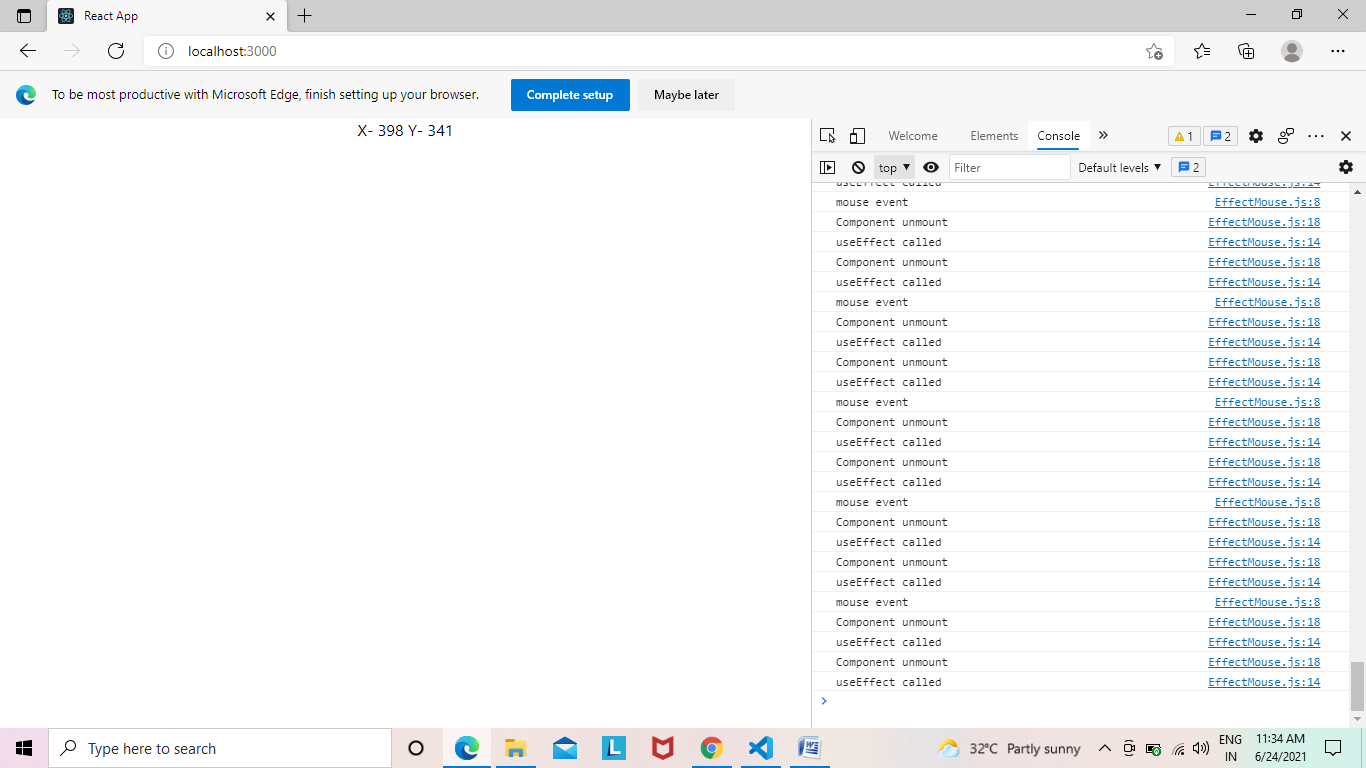
    </div>

  );

}

export default EffectMouse;

**Output:**

****

**EffectMOuse.js**

import React, { useState, useEffect } from "react";

function EffectMouse() {

  const [x, setX] = useState(0);

  const [y, setY] = useState(0);

  const logMouseCordinates = (e) => {

    console.log("mouse event");

    setX(e.clientX);

    setY(e.clientY);

  };

  useEffect(() => {

    console.log("useEffect called");

    window.addEventListener("mousemove", logMouseCordinates);

    return () => {

      console.log("Component unmount");

      window.removeEventListener("mousemove", logMouseCordinates);

    };

  }, []);

  return (

    <div>

      X- {x} Y- {y}

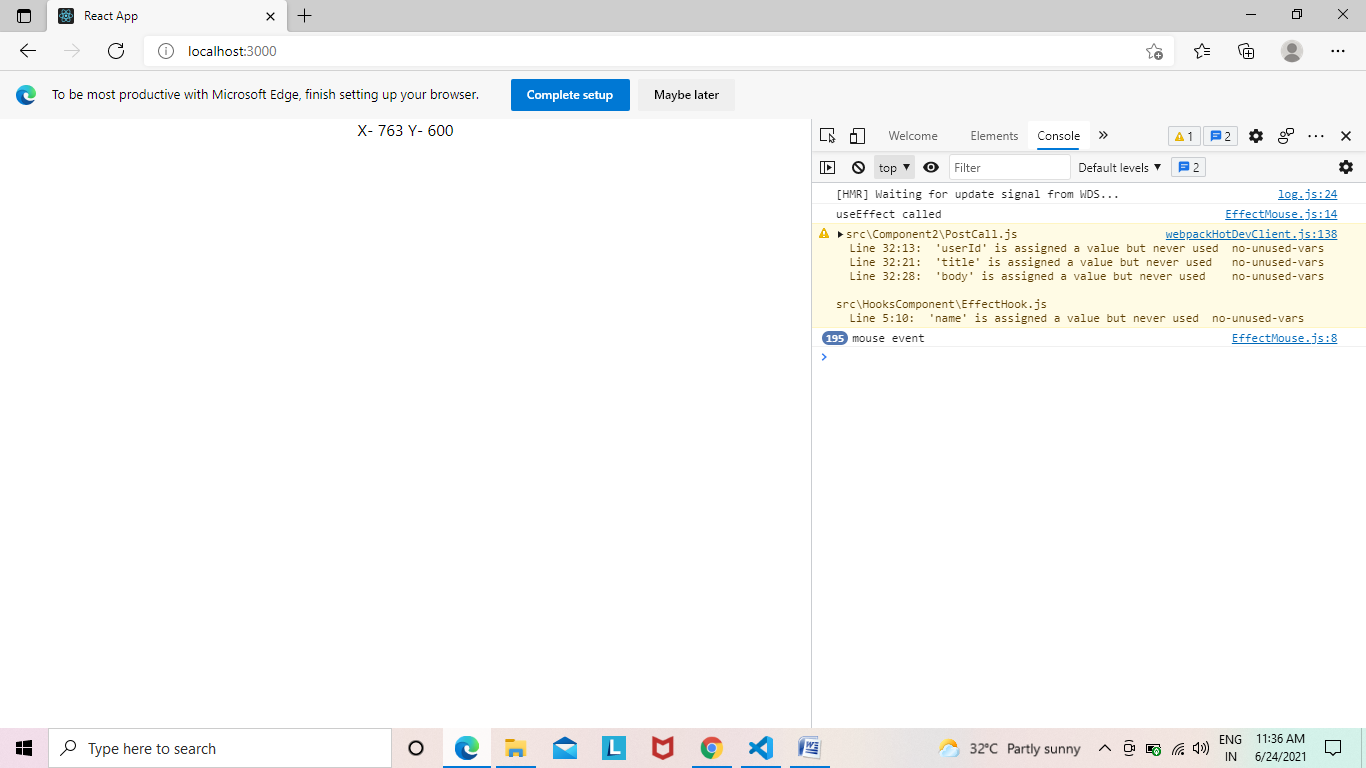
    </div>

  );

}

export default EffectMouse;

**Output:**

****

**App.js**

import './App.css';

import MouseContainer from './HooksComponent/MouseContainer';

function App() {

  return (

    <div className="App">

      <MouseContainer/>

    </div>

  );

}

export default App;

**MouseContainer.js**

import React, { useState } from "react";

import EffectMouse from "./EffectMouse";

function MouseContainer() {

  const [display, setDisplay] = useState(true);

  return (

    <div>

      <button onClick={() => setDisplay(!display)}>Toogle Display</button>

      {display && <EffectMouse />}

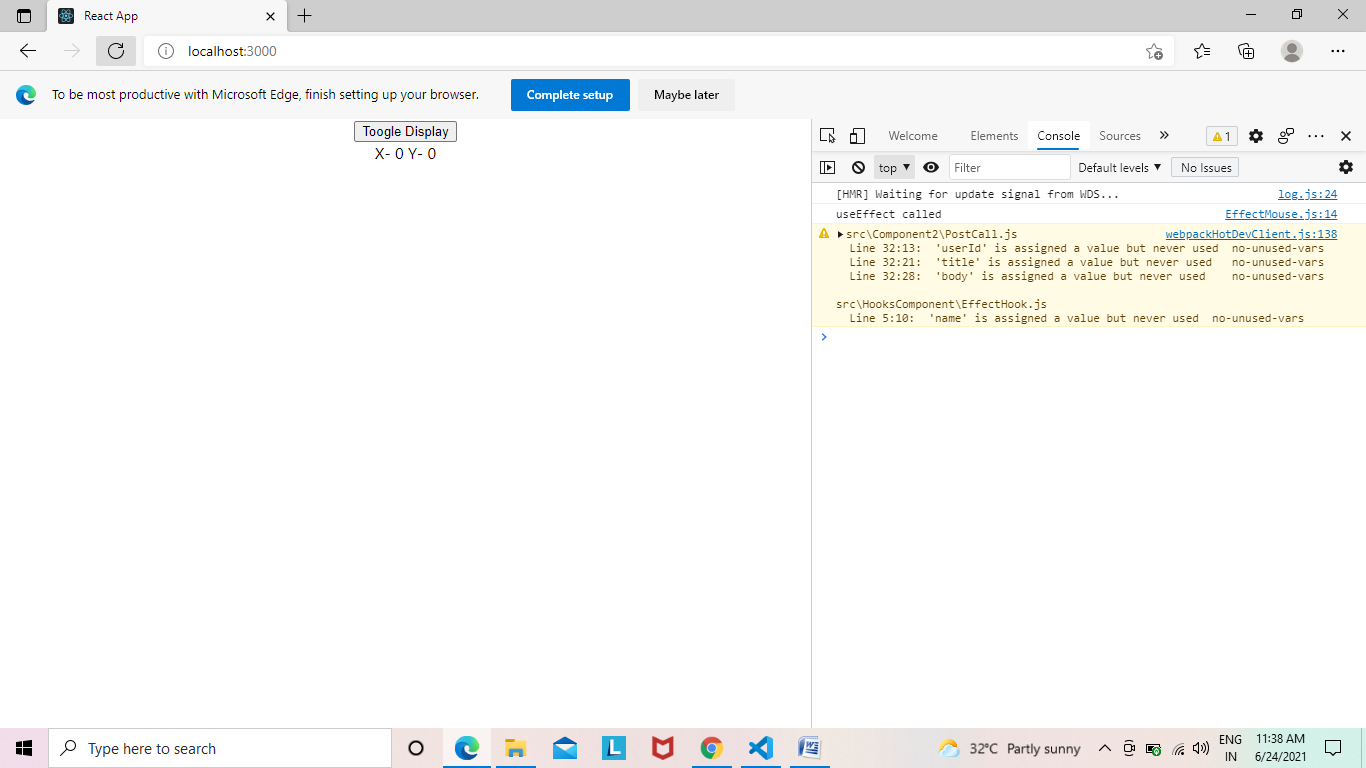
    </div>

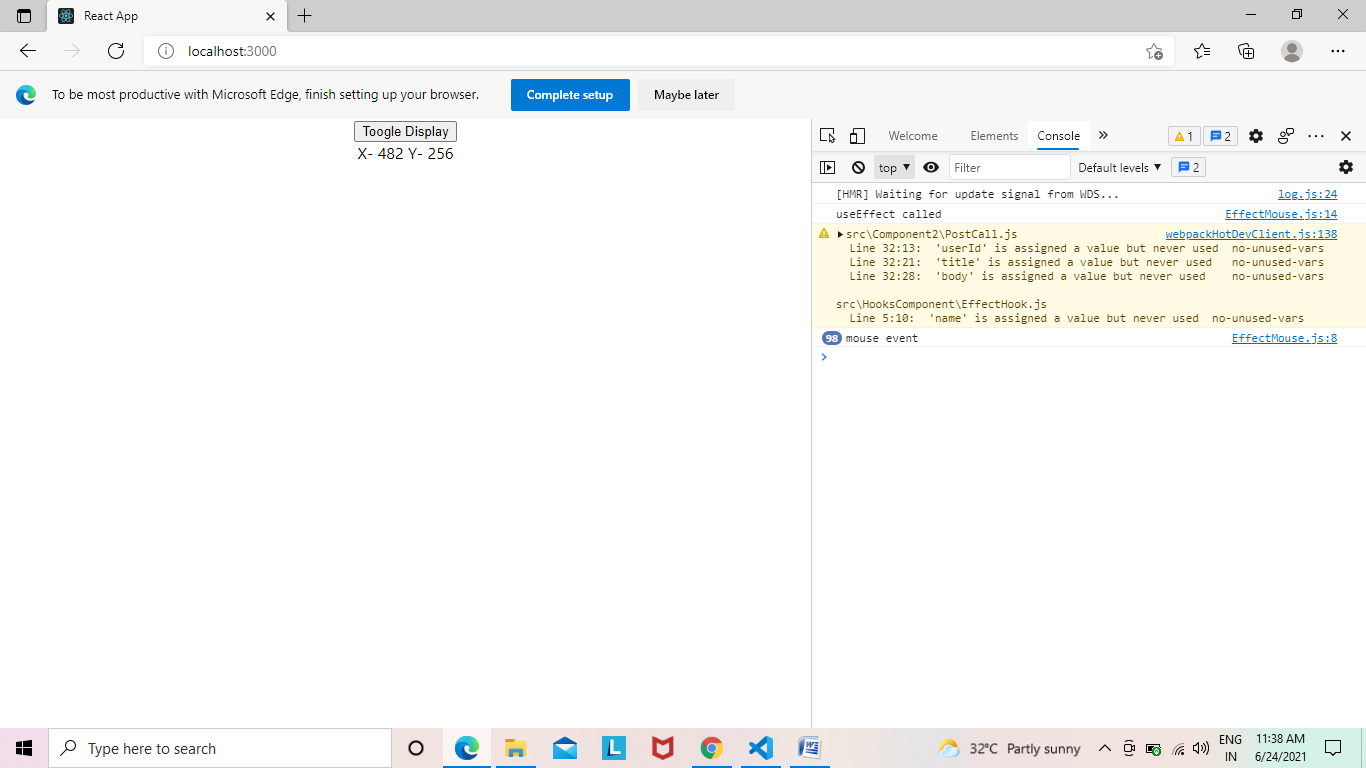
  );

}

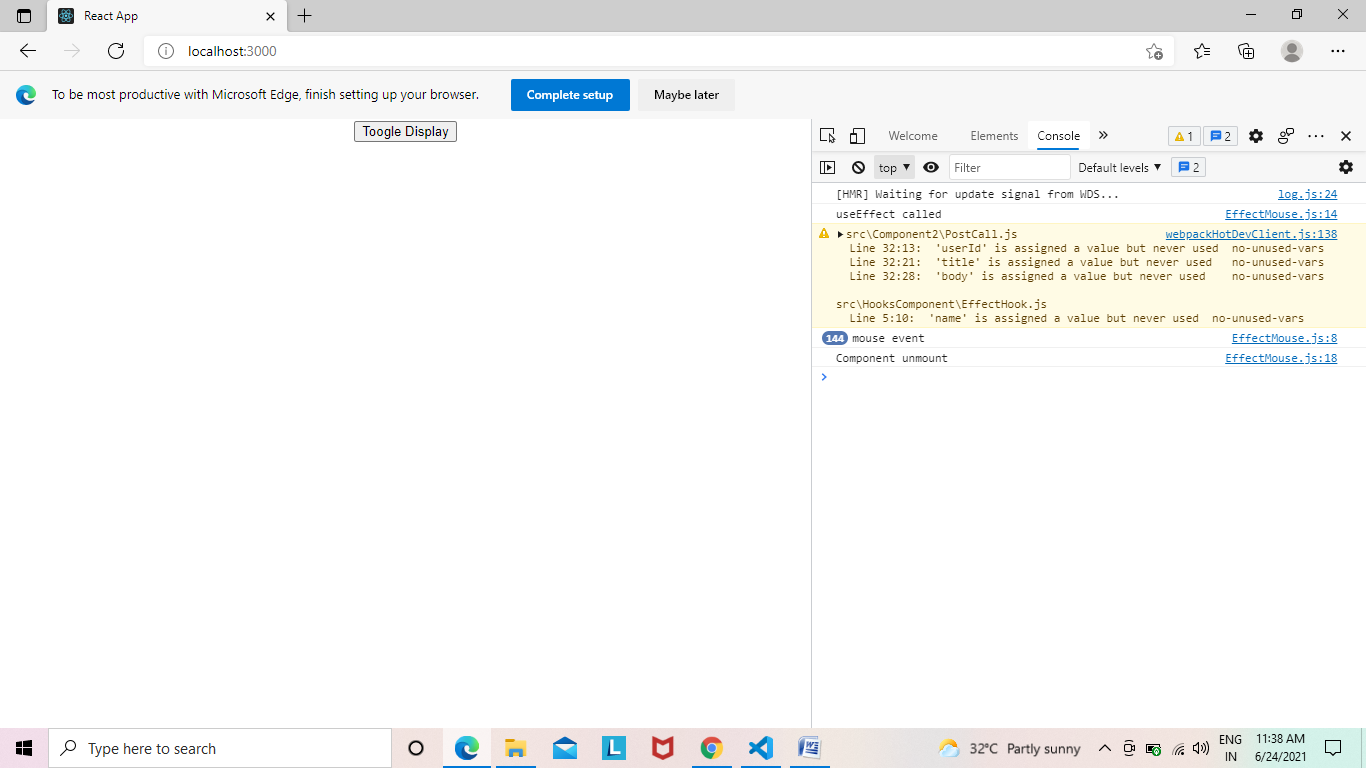
export default MouseContainer;

**Output:**

****

****

**After click on Toggle Display**

****