ErrorBoundary.js

import React, { Component } from "react";

class ErrorBoundary extends Component {

  constructor(props) {

    super(props);

    this.state = {

      hasError: false,

    };

  }

  static getDerivedStateFromError(error){

    return{

        hasError:true

    }

  }

  componentDidCatch(error,info){

      console.log(error)

      console.log(info)

  }

  render() {

    if(this.state.hasError){

        return <h1>Something went wrong</h1>

    }

    return this.props.children;

  }

}

export default ErrorBoundary;

Hero.js

import React from 'react'

function Hero({heroName}) {

    if(heroName=='Joker'){

        throw new Error('Not a hero!')

    }

    return(

        <div>

            {heroName}

        </div>

    )

}

export default Hero

App.js

import './App.css';

import ErrorBoundary from './Component/ErrorBoundary';

import Hero from './Component/Hero';

function App() {

  return (

    <div className="App">

      <ErrorBoundary>

      <Hero heroName="Batman" />

      <Hero heroName="Superman" />

      <Hero heroName="Joker" />

      </ErrorBoundary>

    </div>

  );

}

export default App;



