App.js

import React, { Component } from 'react'

import Image from './Image'

import img1 from './Be-agile-237x300.png';

import img2 from './extra-mile-237x300.png'

export default class App4 extends Component {

  render() {

    return (hm,

      <div>App4

<Image image={img1}/>

<Image image={img2}/>

<Image image="no image"/>

      </div>

    )

  }

}

Error.js

import React, { Component } from 'react'

export default class Error extends Component {

  state =

  {

    hasError : false

  }

  static getDerivedStateFromError(error)

  {

    return {hasError:true};

  }

  render() {

    if(this.state.hasError)

    {

      return <h3> ERROR </h3>;

    }

    return this.props.children;

  }

}

import React, { Component } from 'react'

export default class Image extends Component {

  render() {

    if(this.props.image ==='no image')

    {

      throw new Error("There is no image");

    }

    return (

      <div>

<img src={this.props.image} width="100px" height="100px"/>

<img src={this.props.image} width="100px" height="100px"/>

<img src={this.props.image} width="100px" height="100px"/>

   </div>

    )

  }

}

2nd Demo

import React from 'react'

import Error1 from './Error1'

export default function App5() {

  return (

    <div>App5

      <Alert message="h1"/>

    </div>

  )

}

function Alert({message})

{

  console.log(message)

if(typeof(message)!=='string' || !message){

  throw new Error("Its not string OR its empty")

}

return(

  <h1>{message}</h1> )

}

import React, { Component } from 'react'

export default class Error1 extends Component {

  constructor(props)

  {

    super(props);

    this.state=

      {hasError:false};

  }

   componentDidCatch(error,info)

   {

    this.setState({hasError:true})

   }

  render() {

    if(this.state.hasError)

    return <h1> ERROR </h1>

    return this.props.children;

  }

}