import Alert from 'react-bootstrap/Alert';

function AdditionalContentExample() {

return (

<Alert variant="success">

<Alert.Heading>Hey, nice to see you</Alert.Heading>

<p>

Aww yeah, you successfully read this important alert message. This

example text is going to run a bit longer so that you can see how

spacing within an alert works with this kind of content.

</p>

<hr />

<p className="mb-0">

Whenever you need to, be sure to use margin utilities to keep things

nice and tidy.

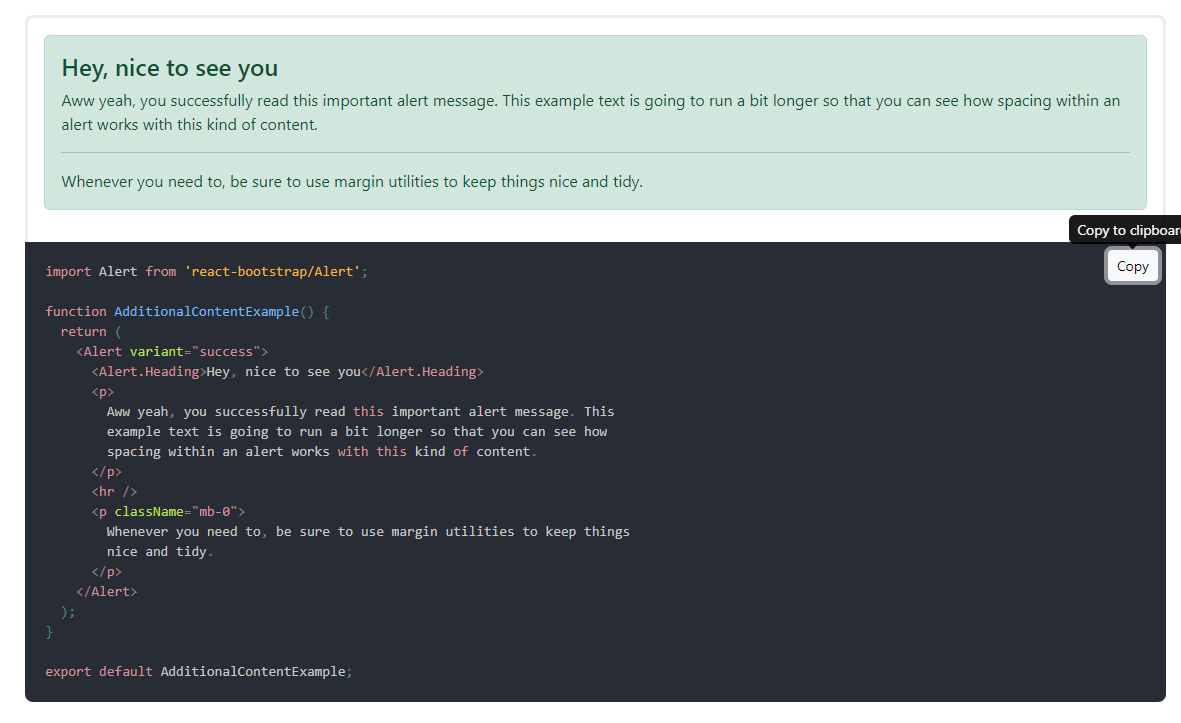
</p>

</Alert>

);

}

export default AdditionalContentExample;



import React, { useState } from 'react';

import Alert from 'react-bootstrap/Alert';

import Button from 'react-bootstrap/Button';

function AlertDismissibleExample() {

const [show, setShow] = useState(true);

if (show) {

return (

<Alert variant="danger" onClose={() => setShow(false)} dismissible>

<Alert.Heading>Oh snap! You got an error!</Alert.Heading>

<p>

Change this and that and try again. Duis mollis, est non commodo

luctus, nisi erat porttitor ligula, eget lacinia odio sem nec elit.

Cras mattis consectetur purus sit amet fermentum.

</p>

</Alert>

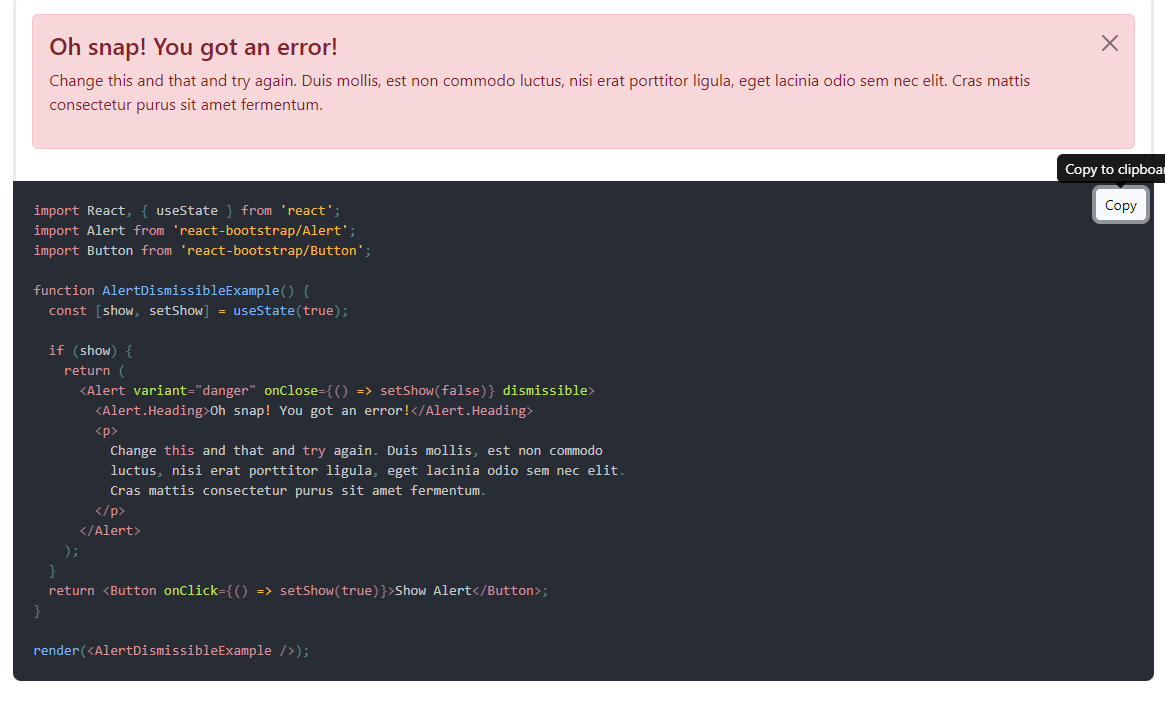
);

}

return <Button onClick={() => setShow(true)}>Show Alert</Button>;

}

render(<AlertDismissibleExample />);



import "~slick-carousel/slick/slick.css";   
import "~slick-carousel/slick/slick-theme.css";

import React, { Component } from "react";

import Slider from "react-slick";

export default class SimpleSlider extends Component {

render() {

const settings = {

dots: true,

infinite: true,

speed: 500,

slidesToShow: 1,

slidesToScroll: 1

};

return (

<div>

<h2> Single Item</h2>

<Slider {...settings}>

<div>

<h3>1</h3>

</div>

<div>

<h3>2</h3>

</div>

<div>

<h3>3</h3>

</div>

<div>

<h3>4</h3>

</div>

<div>

<h3>5</h3>

</div>

<div>

<h3>6</h3>

</div>

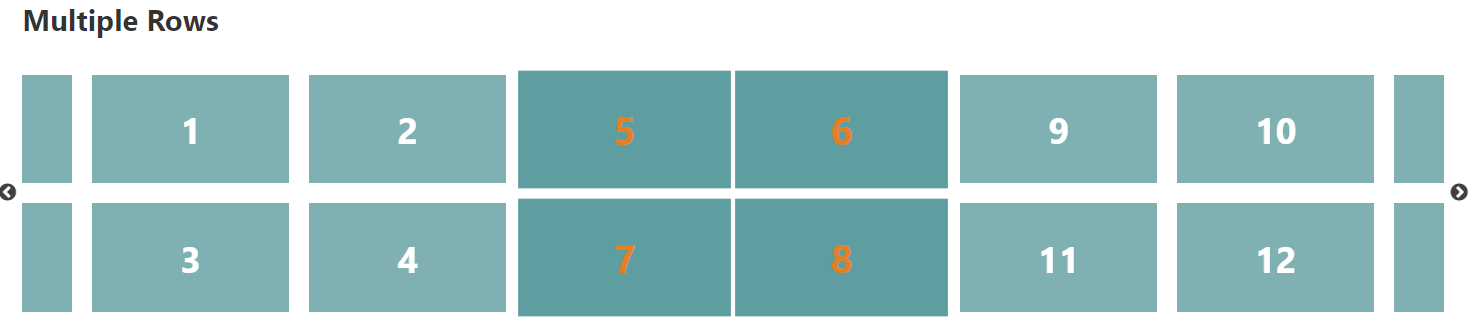
</Slider>

</div>

);

}

}



import React, { Component } from "react";

import Slider from "react-slick";

export default class MultipleRows extends Component {

render() {

const settings = {

className: "center",

centerMode: true,

infinite: true,

centerPadding: "60px",

slidesToShow: 3,

speed: 500,

rows: 2,

slidesPerRow: 2

};

return (

<div>

<h2>Multiple Rows</h2>

<Slider {...settings}>

<div>

<h3>1</h3>

</div>

<div>

<h3>2</h3>

</div>

<div>

<h3>3</h3>

</div>

<div>

<h3>4</h3>

</div>

<div>

<h3>5</h3>

</div>

<div>

<h3>6</h3>

</div>

<div>

<h3>7</h3>

</div>

<div>

<h3>8</h3>

</div>

<div>

<h3>9</h3>

</div>

<div>

<h3>10</h3>

</div>

<div>

<h3>11</h3>

</div>

<div>

<h3>12</h3>

</div>

<div>

<h3>13</h3>

</div>

<div>

<h3>14</h3>

</div>

<div>

<h3>15</h3>

</div>

<div>

<h3>16</h3>

</div>

</Slider>

</div>

);

}

}

const name1 = state.Namefield;

const mob1 = state.Mobilefield;

const mail1 = state.Emailfield;

if (name1.value !== "") {

const newName = [...state.fields, name1];

setState({

fields: newName,

Namefield: {

value: '',

error: ''

}

});

}

if (mob1.value !== "") {

const newMob = [...state.fields, mob1];

setState({

fields: newMob,

Mobilefield: {

value: '',

error: ''

}

});

}

if (mail1.value !== "") {

const newMail = [state.fields, mail1];

setState({

fields: newMail,

Emailfield: {

value: '',

error: ''

}

});

}

}

const handleName = (e) => {

setState({

...state,

Namefield: {

value: e.target.value,

error: false

}

})

};

return (

<div className="App">

....

</div>

);

};

export default App;