In React, createRef is a method used to access the DOM nodes or React elements created in the render method. Refs provide a way to interact with the underlying DOM elements, and they can be useful in various scenarios

import React, {Component} from 'react';

export default class Form2 extends Component

{

  constructor(props)

  {

    super(props);

    this.myRef = React.createRef();

  }

  render()

  {

    return(

<form>

  user name <input type="text" />

  <br></br>

  password <input type="text" ref={this.myRef}/>

  <br></br>

  address : <input type="text"/>

</form>

    );

  }

}

import React, {Component} from 'react';

export default class Form2 extends Component

{

  constructor(props)

  {

    super(props);

    this.myRef = React.createRef();

  }

  componentDidMount()

  {

    console.log(this.myRef);

    console.log(this.myRef.current);

    this.myRef.current.focus();

  }

  render()

  {

    return(

<form>

  user name <input type="text" />

  <br></br>

  password <input type="text" ref={this.myRef}/>

  <br></br>

  address : <input type="text"/>

</form>

    );

  }

}

import React, {Component} from 'react';

export default class Form2 extends Component

{

  constructor(props)

  {

    super(props);

    this.state =

    {

      value:""

    }

    this.myRef = React.createRef();

  }

  // componentDidMount()

  // {

  //   console.log(this.myRef);

  //   console.log(this.myRef.current);

  //   this.myRef.current.focus();

  // }

  render()

  {

    return(

      <React.Fragment>

        <h2> You typed {this.state.value} </h2>

      </React.Fragment>

/\* <form>

  user name <input type="text" />

  <br></br>

  password <input type="text" ref={this.myRef}/>

  <br></br>

  address : <input type="text"/>

</form> \*/

    );

  }

}

import React, {Component} from 'react';

export default class Form2 extends Component

{

  constructor(props)

  {

    super(props);

    this.state =

    {

      value:""

    }

    this.myRef = React.createRef();

  }

  render()

  {

    return(

      <React.Fragment>

        <h2> You typed {this.state.value} </h2>

        <form>

          <input type="text" ref={this.state.value}/>

          <input type="submit" value="Submit"/>

        </form>

      </React.Fragment>

    );

  }

}

import React, {Component} from 'react';

export default class Form2 extends Component

{

  constructor(props)

  {

    super(props);

    this.state =

    {

      value:""

    };

    this.myRef = React.createRef();

  }

   handeSubmit = (event) =>

   {

     event.preventDefault();

     console.log(this.myRef.current.value);

   }

  render()

  {

    return(

      <React.Fragment>

        <h2> You typed {this.state.value} </h2>

        <form onSubmit={this.handeSubmit}>

          <input type="text" ref={this.myRef}/>

          <input type="submit" value="Submit"/>

        </form>

      </React.Fragment>

    );

  }

}

import React, {Component} from 'react';

export default class Form2 extends Component

{

  constructor(props)

  {

    super(props);

    this.state =

    {

      value:""

    };

    this.myRef = React.createRef();

  }

   handeSubmit = (event) =>

   {

     event.preventDefault();

     console.log(this.myRef.current.value);

     this.setState({value : this.myRef.current.value });

   }

  render()

  {

    return(

      <React.Fragment>

        <h2> You typed {this.state.value} </h2>

        <form onSubmit={this.handeSubmit}>

          <input type="text" ref={this.myRef}/>

          <input type="submit" value="Submit"/>

        </form>

      </React.Fragment>

    );

  }

}