1. *array.jsx*

```
import React, { Component } from 'react';


class LoginFormClass extends Component {
  constructor(props) {
    super(props);

    this.state = {
      formData: {
        email: '',
        password: '',
      },
    };
  }


  handleChangeEvent = (e) => {
    const { name, value } = e.target;
    this.setState((prevState) => ({
      formData: {
        ...prevState.formData,
        [name]: value,
      },
    }));
  };
```

```jsx
handleSubmit = (e) => {
  e.preventDefault();
  const { email, password } = this.state.formData;
  // Check if both email and password are non-empty before logging
  if (email && password) {
    console.log('Form Data:', this.state.formData);
    // You can add further logic for form submission here
  }
};


render() {
  const { email, password } = this.state.formData;


  return (
    <form onSubmit={this.handleSubmit}>
      <div>
        <label htmlFor="email">Email:</label>
        <input
          type="email"
          id="email"
          name="email"
          value={email}
          onChange={this.handleChangeEvent}
          required
        />
      </div>
      <div>
        <label htmlFor="password">Password:</label>
```

```jsx
      <input
        type="password"
        id="password"
        name="password"
        value={password}
        onChange={this.handleChangeEvent}
        required
      />
    </div>
    <button type="submit">Submit</button>
  </form>
  );
 }
}

export default LoginFormClass;
```

2. Boolean.jsx

```jsx
import React, { Component } from 'react';

class ToggleButton extends Component {
 constructor(props) {
  super(props);

  this.state = {
   isToggled: false,
  };
```

```jsx
  }

  handleToggle = () => {

    this.setState((prevState) => ({

      isToggled: !prevState.isToggled,

    }));

  };


  render() {

    return (

      <div>

        <button onClick={this.handleToggle}>Toggle</button>

        <p>{this.state.isToggled ? 'Toggle is ON' : 'Toggle is OFF'}</p>

      </div>

    );

  }

}


export default ToggleButton;
```

3. *String.jsx*

```jsx
import React, { Component } from 'react';

class DynamicContent extends Component {

  constructor(props) {

    super(props);

    this.state = {

      inputValue: '',

    };
```

```
    }

  handleChange = (e) => {

    this.setState({

      inputValue: e.target.value,

    });

  };

  render() {

    return (

      <div>

        <label htmlFor="input">Enter Text: </label>

        <input

          type="text"

          id="input"

          value={this.state.inputValue}

          onChange={this.handleChange}

        />

        <p>The entered Text : {this.state.inputValue}</p>

      </div>

    );

  }

}

export default DynamicContent;
```