*Practical-6*

**Aim**: Creating a form in react js using Components

Properties & State.

**Description**:

1. Forms:

This form has the default HTML form behavior of browsing to a new page when the user submits the form. If you want this behavior in React, it just works. But in most cases, it’s convenient to have a JavaScript function that handles the submission of the form and has access to the data that the user entered into the form.

HTML form elements work a bit differently from other DOM elements in React, because form elements naturally keep some internal state.

1. Components & Properties:

Components let you split the UI into independent, reusable pieces, and think about each piece in isolation. This page provides an introduction to the idea of components. You can find a detailed component API reference here.

Conceptually, components are like JavaScript functions. They accept arbitrary inputs (called “props”) and return React elements describing what should appear on the screen.

1. State:

The state is an instance of React Component Class can be defined as an object of a set of observable properties that control the behavior of the component. In other words, the State of a component is an object that holds some information that may change over the lifetime of the component.

**Code**:

Index.js

import React from 'react';

import ReactDOM, {render} from 'react-dom';

class MyForm extends React.Component{

  constructor(props){

    super(props);

    this.state={

      username:'',

      age:null,

    };

  }

  mySubmitHandler = (event) => {

    event.preventDefault();

    let age= this.state.age;

    if(!Number(age))

    {

      alert("Age should be number")

    };

  }

  myChangeHandler = (event) =>{

    let nam = event.target.name;

    let val = event.target.value;

    this.setState({[nam]: val});

  }

render(){

  return (

    <form onSubmit={this.mySubmitHandler}>

      <h1> Hii, My name is {this.state.username} {this.state.age}</h1>

      <p> Enter your name: </p>

      <input type='text' name='username' onChange={this.myChangeHandler}/>

      <p> Enter your age: </p>

      <input type='text' name='age' onChange={this.myChangeHandler}/>

      <br/>

      <br/>

      <input type='Submit'/>

    </form>

  );

}

}

ReactDOM.render(<MyForm/>,document.getElementById('root')

);

**Output**:

