**Forms and Validations  
  
- Form Interactions and Validations are managed by using event and data binding techniques.  
- React uses event and data binding implicitly without any additional libraries.  
- React events are SyntheticEvents that map to DOM events.  
- React can also use style and class binding for validation styles.  
  
Keyboard Events Key Properties  
------------------------ ---------------------   
onKeyUp                         keyCode      
onKeyDown                   charCode  
onKeyPress which  
 shiftKey Note: onkeyUp & onKeyDown fails incase of altKey   
 identifying keycode. ctrlKey   
  
Ex: Form and Validations  
  
*import { useState } from "react";  
  
export default function FormComponent(){  
  
    const [users,setUsers] = useState([  
        {UserId:'john'},  
        {UserId:'john12'},  
        {UserId:'david'},  
        {UserId:'john\_nit'}  
    ]);  
    const [userMsg, setUserMsg] = useState('');  
    const [isUserValid, setUserValid] = useState(false);  
    const [pwdMsg, setPwdMsg] = useState('');  
    const [capsStatus, setCapsStatus] = useState(false);  
    const [cityMsg, setCityMsg] = useState('');  
    const [userDetails, setUserDetails] = useState({UserId:'', Password:'', City:''});  
    function VerifyUserId(e){  
        for(var user of users)  
        {  
            if(user.UserId==e.target.value) {  
                setUserMsg('User Id Taken - Try Another');  
                setUserValid(false);  
                break;  
            } else {  
                setUserMsg('User Id Available');  
                setUserValid(true);  
            }  
        }  
    }  
    function HideUserMsg(e){  
        if(e.target.value=="") {  
            setUserMsg('User Id Required');  
        } else {  
            setUserMsg('');  
        }  
    }  
  
    function VerifyPassword(e){  
        if(e.target.value.match(/(?=.\*[A-Z])\w{4,10}/))  
        {  
            setPwdMsg('Strong Password');  
        } else {  
            if(e.target.value.length<4){  
                setPwdMsg('Poor Password');  
            } else {  
                setPwdMsg('Weak Password');  
            }  
        }  
    }   
    function HidePasswordMsg(){  
        setPwdMsg('');  
        setCapsStatus(false);  
    }  
  
    function VerifyCaps(e){  
        if(e.keyCode>=65 && e.keyCode<=90 || e.which>=65 && e.which<=90) {***

***//which is used to identify all keyboard & it avoids the keyboard compatibility issues.  
            setCapsStatus(true);  
        } else {  
            setCapsStatus(false);  
        }  
    }  
    function VerifyCity(e){  
        if(e.target.value=="notcity") {  
            setCityMsg('Please Select a City');  
        } else {  
            setCityMsg('');  
        }  
    }  
    function HandleUserChange(e) {  
        setUserDetails({  
            UserId : e.target.value,  
            Password: userDetails.Password,  
            City: userDetails.City  
        })  
    }  
    function HandlePasswordChange(e) {  
        setUserDetails({  
            UserId : userDetails.UserId,  
            Password: e.target.value,  
            City: userDetails.City  
        })  
    }  
    function RegisterClick(){  
        alert(JSON.stringify(userDetails));  
    }  
  
    return(  
        <div className="container-fluid">  
            <h2>Register User</h2>  
            <dl>  
                <dt>User Id</dt>  
                <dd><input type="text" onChange={HandleUserChange} onBlur={HideUserMsg} onKeyUp={VerifyUserId} /></dd>  
                <dd className={(isUserValid==true)?'text-success':'text-danger'} >{userMsg}</dd>  
                <dt>Password</dt>  
                <dd>  
                    <input type="password" onChange={HandlePasswordChange} onKeyPress={VerifyCaps} onBlur={HidePasswordMsg} onKeyUp={VerifyPassword} />  
                </dd>  
                <dd>{pwdMsg}</dd>  
                <dd className={(capsStatus==true)?'d-block':'d-none'}>  
                    <span className="text-warning"><span className="bi bi-exclamation-triangle"></span> Caps ON</span>  
                </dd>  
                <dt>Your City</dt>  
                <dd>  
                    <select onChange={VerifyCity}>  
                        <option value="notcity">Select Your City</option>  
                        <option value="Delhi">Delhi</option>  
                        <option value="Hyd">Hyd</option>  
                    </select>  
                </dd>  
                <dd className="text-danger">{cityMsg}</dd>  
            </dl>  
            <button onClick={RegisterClick}>Register</button>  
        </div>  
    )  
}***