```
Form Handeling
import of usestate & from "react";
  Junction Appl) &
                                                                             [firstName, set FirstName] = usestarte("");
                                         const
                                                                                 Clast Name, set Last Name] = usesterte ("");
                                         const
                                   Junction Firstehange Handler (event) &
                                                                                   setfirstMame (event.target.value);
                                     Junction Lastchange Handler (event) &
                                                                                       setlastName (event. target. value);
                                 return (
                                                    2div>
                                                                            zform>
                                                                                                    <br/>

                                                                                                     Linput
                                                                                                                         type = "text"
                                                                                                                          placeholder = "First name"
                                                                                                                         onChange = Sfirst change Handlery
                                                                                                        1>
```

468/>

type="l+ex+" placeholder= "last name" on Change = Slast change Handlery oned : Million / 4form> ); </div> (Consister of John Delo) export default App; The problem with above code is that for every input field we are defining different on Change handler junctions and multiple usestarte hooks. Let say we have input field and other form components in a very large number then writing onchanges handler for every component is not the best way to create a form. The more optimized code is given below.

function App () &

const [formData, setFormData] = useState(& dirstName: "", whole data name attribute last Name: "", we less of the form is being tracked i.e. in formDortos is provided \_ email: "", in every is Visible: true element 3); console.log (formData); iggs tropped tropped Junction change Handler (event) & through const Ename, value, checked types = event target Set Form Data (Cprev Form Data) => of return y storing the prevFormDorla, previous state of [name]: type === "checkbox"

the form with

the help of ... spread)? checked: value

operator.

3; · sealong review de about sometimentage agent and return ( By Complete restance 2dW>

(Jorm)

*<u>Cinput</u>* type="text" placeholder="first name" on Change = Schange Handlery name = "first Name" value = S.form Data. Jirist Namely 1> Il with the help of value you can track the state of individual Lbr1) element. Linput Components placeholder = "last name" on Change = of change Handler 3 name = " lastName" value = X form Data. last Namey Lbr/> cinput placeholder= "enter your email" on Change = & change Handler y name = "email" value = formData. emaily

```
Lbr/>
         Linput
             type = "checkbox"
             on Change = & change Handler 3
              name = "isvisible"
              id = "isVisible"
              checked = of form Data is Visible y
         1>
        Llaber Mm/For="isVisible">
               Am I visible?
         2/label)
     - Orm
                              'hatmifor' attribute is used
                          to attach the label with
     16/7/
                          checkbox with the help of
                        'id' attribute i.e. is Visible.
Controlled Components
 It maintains state inside the component
  (dieldset)
       21 egend> Mode: 21 legend>
       Linput
         type = "radio"
         on Change = of change Handler y
         name = "mode"
```

```
value="Online-mode"
            id = "Online-mode"
            checked= yform Data. mode === "Online-Mode"}
       1>
       Llabel " htmlfor = Online-mode > Online mode </label>
       Cinputate vilopholom $ 1 : 3:
         type="radio"
         onchange=& change Handler y
         name = "mode"
         value = "Offline-mode"
         id = "off line-mode"
         checked = of form Dorta mode === "Offline-Mode"y
      1>
      Clabel htmlFor='offline-mode') Offline Mode </abel)
     alfieldset)
     clabel n+m) For = javlar'> Favorite car < label> -
     2 select
           name = "favCar"
           id = "faular"
           value = of form Data. Jawar y
           on Change = of change Handlex 3
     Loption value= "scarpio"> Scarpio Cloption>
     Loption value: "tharr"> Tharr Goption>
    2/select)
    ebutton) Submit < (button)
4 form>
```

4div