**React Labs – 12**

|  |  |
| --- | --- |
|  | **React multi forms** |
|  | Generate form and element dynamically  Ref: [How to Build Dynamic Forms in React? - GeeksforGeeks](https://www.geeksforgeeks.org/how-to-build-dynamic-forms-in-react/) |

|  |  |
| --- | --- |
|  | **Task1:** Toggle dropdown list (Expand/collapse items) |
|  |  |
|  |  |

|  |  |
| --- | --- |
|  | **Input types** |
|  | export const inputTypes = [    { label: 'A Circle', value: 'text' },    { label: 'A Square', value: 'number' },    { label: 'A Rectangle', value: 'email' },    { label: 'A Triangle', value: 'password' },    { label: 'An Eclipse', value: 'date' }  ]; |
|  | const [showForm, setShowForm] = useState(false); |
|  | const toggleDropdown = (e) => {      //ToDo: Implement logic to show dropdown panel and hide the form when dropdown is open      if (!isDropDownOpen) {        setShowForm(false)      }      setIsDropDownOpen(!isDropDownOpen);      //ToDo: Implement logic to hide dropdown panel and show the form when dropdown is closed    }; |

|  |  |  |
| --- | --- | --- |
|  | **Task2: Generate form based on items selected** | |
|  |  |  |

|  |  |
| --- | --- |
|  | **Initial state** |
|  | const [selectedInputTypes, setSelectedInputTypes] = useState([]);  const [isDropDownOpen, setIsDropDownOpen] = useState(false) |

|  |  |
| --- | --- |
|  | **Handle submit** |
|  | const handleSubmit = () => {      setShowForm(true);      setIsDropDownOpen(false);      //ToDo: implement logic to render form with selected input types and hide the dropdown panel    }; |

|  |  |
| --- | --- |
|  | **Implementation** |
|  | const toggleDropdown = (e) => {      //ToDo: add the logic to show dropdown panel and hide the form when dropdown is open      if (!isDropDownOpen) {        setShowForm(false)      }      setIsDropDownOpen(!isDropDownOpen);      //ToDo: add the logic to hide dropdown panel and show the form when dropdown is closed    }; |

|  |  |
| --- | --- |
|  | **Handle input change** |
|  | const handleOptionChange = (value) => {      if (selectedInputTypes.includes(value)) {        setSelectedInputTypes(selectedInputTypes.filter(inputType => inputType !== value))      } else {        setSelectedInputTypes([...selectedInputTypes, value]);      }      //ToDo: implement logic to maintain the selected input types from the dropwdown panel    }; |

|  |  |
| --- | --- |
|  | **Handle button state** |
|  | const handleBtnState = () => {      return (selectedInputTypes.length === 0 || showForm) ? true: false;      //ToDo: implement logic to keep submit button disabled when no input type is selected from the dropdown options or when the form is already rendered    } |