

**Handle
Select Dropdown**

What is a Select Dropdown?

The `<select>` tag creates a dropdown menu, and each `<option>` represents a selectable item. In React, we use state to control which value is currently selected.

Basic Single Select

```
const [language, setLanguage] = useState("");

<select value={language} onChange={(e) => setLanguage(e.target.value)}>
  <option value="">-- Select --</option>
  <option value="JavaScript">JavaScript</option>
  <option value="Python">Python</option>
  <option value="C++">C++</option>
  <option value="Java">Java</option>
</select>
```

Tips:

- Don't miss value attribute in <select>.
- Don't use defaultValue in <select>. (uncontrolled component)
- Don't use selected attribute on <option>. (uncontrolled component)
- Always include a default option like -- Select -- for better UX.

Dynamic Dropdown (Using .map())

If you have many options, it's best to render them dynamically from an array.

```
const [country, setCountry] = useState("");
const countries = ["India", "USA", "Canada", "Japan"];

<select value={country} onChange={e => setCountry(e.target.value)}>
  <option value="">-- Select Country --</option>
  {countries.map(c => (
    <option key={c} value={c}>
      {c}
    </option>
  )))
</select>
```

Dependent Select (Cascading)

second dropdown's options depend on which specific item is selected in the first dropdown — also called a dependent or cascading select.

For Example: When a user selects a country, the city dropdown should show different cities for that particular country. And maybe, for some countries, even custom special cities or messages.

```
India: ["Delhi", "Mumbai", "Kolkata", "Bengaluru"],  
USA: ["New York", "Los Angeles", "Chicago"],  
Japan: ["Tokyo", "Osaka", "Kyoto"],  
Germany: ["Berlin", "Munich", "Hamburg"],
```

Thank
You