

1. Importar al proyecto

```
npm i @reduxjs/toolkit react-redux
```

2. Crear archivos

```
src
├── components
│   └── store
│       ├── indexRedux.js
│       ├── file1-slice.js
│       ├── file2-slice.js
│       └── file3-slice.js
```

3. Preparar los archivos Slice

file1-slice.js

```
1 import {createSlice} from '@reduxjs/toolkit'

2 const file1InitialState = {
  variable1: 0,
  variable2: [],
  variable3: '',
}

3 const file1Slice = createSlice({
  name: 'file1',
  initialState: file1InitialState,
  reducers: {
    method1(state, action) {
      state.variable1 = action.payload.newValue
    }
    method2(state) {
      state.variable2 ++
    }
  }
})

4 export default file1Slice
```

4. Trabajar indexRedux.js

indexRedux.js

```
1 import {configureStore} from '@reduxjs/toolkit'

import file1Slice from './file1-slice'
import file2Slice from './file2-slice'
import file3Slice from './file3-slice'
import file4Slice from './file4-slice'

2 const store = configureStore({
  reducer: {
    file1: file1Slice
    file2: file2Slice
    file3: file3Slice
    file4: file4Slice
  }
})

3 export const file1Actions = file1Slice.actions
export const file2Actions = file2Slice.actions
export const file3Actions = file3Slice.actions
export const file4Actions = file4Slice.actions
export default store
```

USAR

5. Wrap el componente padre

index.js Nivel más alto de la app

```
1 import store from './store/indexRedux.js'
import {provider} from 'react-redux'

2 ReactDOM.render(
  <Provider store={store}>
    <App />
  </Provider>,
  document.getElementById('root')
);
```

6. Usar en un componente

Component.js

```
1 import {useSelector, useDispatch} from 'react-redux'
import {file1Actions} from '../store/indexRedux.js'

2 const Component = () => {
  const dispatch = useDispatch();

  const variable1 = useSelector( state => state.file1.variable1 );
  const variable2 = useSelector( state => state.file1.variable2 );

  const funcHandler = () => dispatch(file1Actions.method( {newValue: 5} ))

  return (...)
}
```