

# 연습문제 12 풀이

## 소스파일 구성

components 디렉토리의 `NewItem.js`와 `Spinner.js`는 이전 예제의 파일을 재사용함.

```
.
├── components
│   ├── NewItem.js
│   └── Spinner.js
├── pages
│   └── News.js
├── slices
│   └── NewsSlice.js
├── App.js
├── index.js
└── store.js
```

## 소스코드

slices/NewsSlice.js

```
import { createSlice, createAsyncThunk } from '@reduxjs/toolkit'
import axios from 'axios';

export const getNewsList = createAsyncThunk("NewsSlice/getNewsList", async
(payload, { rejectWithValue }) => {
  let result = null;

  try {
    result = await axios.get('http://localhost:3001/news');
  } catch (err) {
    result = rejectWithValue(err.response);
  }

  return result;
});

const NewsSlice = createSlice({
  name: 'news',
  initialState: {
    data: null,
    loading: false,
    error: null
  },
  reducers: {},
  extraReducers: {
    [getNewsList.pending]: (state, { payload }) => {
```

```

        return { ...state, loading: true }
      },
      [getNewsList.fulfilled]: (state, { payload }) => {
        return {
          data: payload?.data,
          loading: false,
          error: null
        }
      },
      [getNewsList.rejected]: (state, { payload }) => {
        return {
          data: payload?.data,
          loading: false,
          error: {
            code: payload?.status ? payload.status : 500,
            message: payload?.statusText ? payload.statusText :
'Server Error'
          }
        }
      }
    },
  },
});

export default NewsSlice.reducer;

```

## store.js

```

import { configureStore } from '@reduxjs/toolkit';

import NewsSlice from './slices/NewsSlice';

const store = configureStore({
  reducer: {
    news: NewsSlice
  },
  middleware: (getDefaultMiddleware) =>
getDefaultMiddleware({serializableCheck: false}),
  devTools: true
});

export default store;

```

## index.js

```

import React from 'react';
import ReactDOM from 'react-dom/client';
import App from './App';
import { BrowserRouter } from 'react-router-dom';

```

```
import { Provider } from 'react-redux';
import store from './store';

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    <Provider store={store}>
      <BrowserRouter>
        <App />
      </BrowserRouter>
    </Provider>
  </React.StrictMode>
);
```

## App.js

```
import React, { memo } from 'react';

import News from './pages/News';

const App = memo(() => {
  return (
    <div>
      <h1>Redux News Viewer</h1>
      <hr />
      <News />
    </div>
  );
});

export default App;
```