React Project with JavaScript + SWC

Overview

A React project with JavaScript + SWC means you're using:

- React — for building your user interface
- JavaScript — as your programming language (not TypeScript)
- SWC — as your compiler/transpiler instead of Babel.

What is SWC?

SWC (Speedy Web Compiler) is a super-fast JavaScript/TypeScript compiler written in Rust. It's used to transform modern JavaScript (ES6+) and JSX (React syntax) into older JavaScript so browsers can understand it. It's an alternative to Babel, but much faster — especially in large projects.

Why use SWC in a React project?

• Speed: Compiles much faster than Babel.
• Modern: Supports the latest JavaScript and React features.
• JSX Support: Can handle JSX syntax out of the box (with the right config).
• Lightweight: Less overhead and quicker builds for dev and production.

Where is this setup used?

• Tools like Next.js use SWC by default (since v12+).
• You can also use it with Vite or Webpack (via swc-loader).

Example: swc config for a React + JS Project

```
{
  "jsc": {
    "parser": {
      "syntax": "ecmascript",
      "jsx": true
    },
    "transform": {
      "react": {
        "runtime": "automatic"
      }
```

```
      }
    },
    "module": {
      "type": "es6"
    }
  }
}
```

Summary

A React project with JavaScript + SWC is simply a React app written in JavaScript where SWC — not Babel — is used to handle JSX and modern syntax compilation.