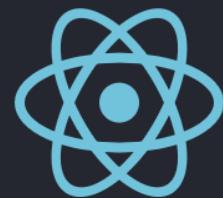


React Apps





React

- **What is React?**
 - A JavaScript library for building user interfaces
 - Originally developed by Facebook in 2011, open-sourced in 2013
 - Designed for building fast, interactive web apps
- **Why do people like it?**
 - Looks really cool
 - Component-based architecture → Reusable, modular code
 - Virtual DOM → Fast updates & rendering
- **Where is React used?**
 - Facebook, Instagram, Netflix, Airbnb, Uber, and thousands of startups

React App Structure

```
my-react-app/
├─ public/
│  └── index.html      ← HTML template loaded by React
└─ src/
    ├─ index.js        ← React entry point, renders App
    ├─ App.js           ← Root React component
    └── components/
        └── MyComponent.js ← Example custom component
└─ package.json       ← Project metadata & dependencies
```



Node.js

- **Node.js is a runtime environment that lets you run JavaScript on your computer, not just in the browser**
- **Lets developers use JavaScript for backend servers, command-line tools, and more**
- **React is built with modern JavaScript tools that require Node.js to work**
- **Download here: <https://nodejs.org/en/download>**
 - Scroll down and choose “prebuilt Node.js”

Creating a React App

1. Make a folder (i.e. my-app)
2. Open the folder in Cursor (so your terminal is in the folder)
3. In the terminal run **npx create-react-app .**
 - The dot means make the app inside this folder you are in
 - Installs a whole bunch of packages and modules (15K!)
 - This step takes a few minutes ☹
4. Run the app
 - In the terminal, run the command **npm start**

Node Package Manager (npm)

- The npm command
 - Installs packages from npm registry
 - Runs project scripts

```
npm start
```

- The npm start command
 - Starts the React development server
 - Serves your app at <http://localhost:3000>
 - Compiles your React code
 - Hot Reloading
 - Updates the browser **instantly** when you edit your code
 - No need to refresh the page manually

Execute npm Package (npx)

- **npx — Execute npm Packages**
 - Lets you run a package without installing it globally

```
npx create-react-app my-app
```

- Downloads and runs **create-react-app** temporarily just for that command
- Avoids clutter on your machine

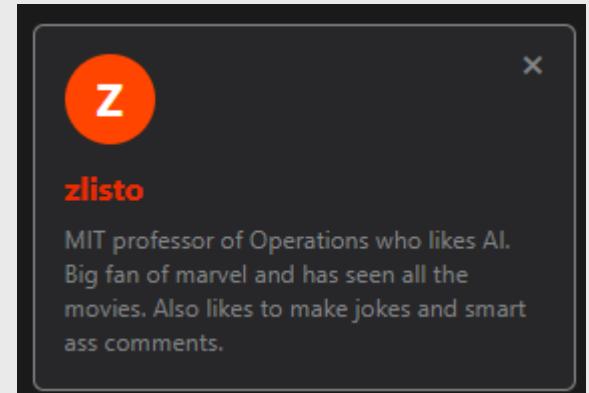
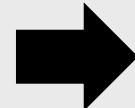
React Components

- A component is a reusable piece of user interface
- Components are Javascript files (.js)
- React apps are built by combining components like Lego blocks
- Example: this component will produce some nice HTML formatted text that says **This is my custom component!**

```
function MyComponent() {  
  return <p>This is my custom component!</p>;  
}  
  
export default MyComponent;
```

Example: AgentCard

```
const AgentCard = ({ agent, onDelete }) => {
  return (
    <div className="agent-card">
      <div className="agent-avatar" style={{ backgroundColor: agent.color }}>
        {agent.name.charAt(0).toUpperCase()}
      </div>
      <div className="agent-info">
        <h3 style={{ color: agent.color }}>{agent.name}</h3>
        <p>{agent.bio}</p>
      </div>
      <button
        className="delete-agent"
        onClick={() => onDelete(agent.id)}
      >
        ✖
      </button>
    </div>
  );
};
```



Example: App.js

- App is the main component that renders everything inside your app (this is in the App.js file)
- This component uses other components

```
function App() {
  const [activeTab, setActiveTab] = useState('agents');
  const [agents, setAgents] = useState([]);
  const [posts, setPosts] = useState([]);
  const [isRunning, setIsRunning] = useState(false);
  const [topic, setTopic] = useState('Robert Downey Jr. as Dr. Doom is a Multiverse glitch');
  const [agentName, setAgentName] = useState('');
  const [agentBio, setAgentBio] = useState('');
  const [logs, setLogs] = useState([]);
  const [responses, setResponses] = useState([]);
  const [isMonitorCollapsed, setIsMonitorCollapsed] = useState(false);
  const postIdCounter = useRef(1);
  const lastAgentRef = useRef(null);

  const colors = [
    '#FF4500', '#FF6B6B', '#4ECD4', '#45B7D1', '#FFA07A',
    '#98D8C8', '#F7DC6F', '#BB8FCCE', '#85C1E2', '#F8B739'
  ];

  const addLog = (message, emoji = '📝') => {
    const timestamp = new Date().toLocaleTimeString();
    setLogs(prev => [...prev, `[${timestamp}] ${emoji} ${message}`]);
  };

  // Load default agents from profiles.json on mount
  useEffect(() => {
    const loadDefaultAgents = async () => {
      try {
        const response = await fetch('/profiles.json');
        if (response.ok) {
          const data = await response.json();
          setAgents(data.agents);
        }
      } catch (error) {
        console.error(error);
      }
    };
    loadDefaultAgents();
  }, []);
}
```

index.js

- In the index.js file, your App component is rendered to run your app

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import './index.css';
import App from './App';
import reportWebVitals from './reportWebVitals';

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>
);

// If you want to start measuring performance in your app, pass a function
// to log results (for example: reportWebVitals(console.log))
// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals
reportWebVitals();
```

Coding Session

- We will build a simple React app
- We will try to make the style look cool
- Give the app some simple AI functionality
 - We will use OpenAI today, but can pop in any AI later on

Reminder: Help us understand recent decreases in your
OpenAI API usage

 Summarize

Hello,

This is a quick reminder about our short survey on
your recent OpenAI API usage.

We're looking to better understand why some teams
reduce usage, pause activity, or change how they use
the OpenAI API—and what factors most influence those
decisions.

The survey should take **less than 5 minutes** to
complete.

The survey will close on **February 6**.

[Take The Survey Now](#)

Thanks for helping us improve the OpenAI API.

The OpenAI User Research Team