

# React 培训大纲

---

本系列培训旨在凯捷沈阳 RDC 普及 React 及相关生态的基础知识、使用方法以及深层次的原理，扩宽开发人员的技术栈，为未来可能到来的 React 项目做前期准备。

课程主要分为两大部分，基础和进阶：

- 基础部分主要是帮助受训人员了解 React，如何搭建 React 脚手架，如何开发 React 应用、周边生态的使用等；
- 进阶部分主要是基础部分的扩展，深入浅出 React 部分原理，如何优化 React 应用等；

其他附加：

- 在培训过程中，适当普及 ES6+、Typescript 以及函数式编程；
- 在每次课程的最后布置作业；
- 提供实战类的项目；

课程周期：3 周半（共 7 次课，14 课时）；

课程频率：2 次 / 周；

单节课程的时长：2 小时；

课程展现形式：PPT（Slidev）+ coding；

课程培训方式：线上；

**实战项目：简单购物车功能（暂定）**

Git 地址：<https://github.com/jeffersonRibeiro/react-shopping-cart>

Demo 演示：<https://react-shopping-cart-67954.firebaseio.com/>

## 基础部分

主要内容如下：

- What is React and updates between v17 and v18
- What is JSX
- React Components
  - Class Components
  - Function Components
  - Controlled Components
  - Uncontrolled Components
- Components Interacting
  - State
  - Props
  - Ref
  - Context
  - PubSub
- Lifecycle Methods
  - Old Lifecycle
  - New Lifecycle
- Hooks
  - APIS
  - How to use hooks

## 进阶部分

主要内容如下：

- React 哲学
- Create React App
- With Typescript
- React Router
- React Context
  - Redux
  - Zustand

- Jotai
- CSS in JS and CSS Module
  - CSS in JS
  - CSS Module
  - Template String Styled
- Optimized Components
  - HOC and Render Props
  - Children API
- Custom Hooks
- Umijs

## 培训计划

- React Introduction (React, CRA Cli, JSX vs Vue, Lifecycle vs Vue, Components - Class Component, Function Component, Hooks, Component Libraries - Antd / Element / Material UI, Umijs);
- Components Interacting (Create Components, State, Props, Ref, Interaction: Parent and Children, between Childrens (state hoisting));
- Hooks (Function Component, How to use hooks, How to create custom hooks, vs Vue3);
- How to style your components (CSS in JS, CSS Module, Style Scoped, How to create styled function in Template String)
- React Router and Typescript (React Router, History and Browser, typescript, refactoring with Typescript);
- States Management (redux / Zustand / Jotai / React Context);
- React Philosophy and Optimize React App (How to write a component elegantly, HOC, Render Props, reducing re-render using Children, useMemo, memo and useCallback)