

数科19班——内部资料，禁止外传——

——前端开发学习路径

1. HTML & CSS 基础

HTML

- **标签基础**：学习常用标签（如 `<div>`, ``, ``, `<a>` 等），理解标签的语义化使用。
- **表单**：掌握 `<form>` 表单元素（如 `<input>`, `<select>`, `<textarea>`），了解表单验证。
- **多媒体标签**：学习如何嵌入视频和音频（如 `<audio>`, `<video>` 标签）。
- **语义化**：理解语义化标签（如 `<header>`, `<footer>`, `<article>`, `<section>`）的作用，增强 SEO。

CSS

- **选择器**：熟练掌握基础选择器（如类选择器、ID选择器、伪类选择器等）。
- **盒模型**：理解 `content`、`padding`、`border`、`margin` 的作用和计算方式。
- **布局**：
 - **浮动布局**：理解浮动（`float`）的机制和清除浮动技巧。
 - **弹性布局**：学习 Flexbox 布局，掌握主轴/交叉轴对齐方式。
 - **网格布局**：学习 CSS Grid，适用于复杂布局场景。
- **响应式设计**：学习媒体查询（`@media`），制作适配多设备的网页。
- **动画与过渡**：了解 CSS 动画（`@keyframes`）和过渡（`transition`）效果。

2. JavaScript 基础

基础语法

- **数据类型**：了解基本数据类型（字符串、数字、布尔值、数组、对象、函数等）和类型转换。
- **运算符**：掌握算术运算符、逻辑运算符、比较运算符。
- **控制结构**：学习条件语句（`if`、`else`、`switch`）和循环结构（`for`、`while`、`do...while`）。

函数与作用域

- **函数**：定义函数的方式（声明式函数、匿名函数、箭头函数）。
- **作用域**：理解变量作用域（块级作用域、函数作用域）和闭包概念。
- **ES6+ 特性**：如解构赋值、`let` 和 `const`、模板字符串、`for...of`、箭头函数、默认参数、剩余参数。

DOM 操作与事件处理

- **DOM 树**: 理解节点的结构, 操作节点 (创建、删除、修改)。
- **事件**: 学习事件模型 (捕获、冒泡)、事件委托、常用事件 (如点击、鼠标悬停、表单提交等)。

异步编程

- **定时器**: `setTimeout` 和 `setInterval`。
 - **Promise**: 学习 Promise 基本用法、链式调用。
 - **async/await**: 掌握异步函数和处理异步逻辑。
-

3. 前端框架

Vue.js

- **基础**: 学习 Vue 的双向绑定 (`v-model`)、模板语法 (`{{}}`)、常用指令 (如 `v-if`、`v-for`)。
- **组件化**: 理解组件的创建与复用, 传递数据 (`props` 和 `emit`)。
- **Vue Router**: 实现单页面应用的路由管理, 学习路由传参、嵌套路由。
- **Vuex**: 用于全局状态管理, 学习 `state`、`getters`、`mutations`、`actions`。

React (可选)

- **基础**: 组件基础 (函数组件和类组件), JSX 语法。
 - **Hooks**: 学习常用 Hooks (如 `useState`、`useEffect`)。
 - **状态管理**: Redux、Recoil 或 MobX 的基本概念与使用。
 - **React Router**: 实现路由跳转、参数传递、嵌套路由。
-

4. 开发工具

版本控制

- **Git**: 了解 Git 基本命令 (`add`、`commit`、`push`、`pull`), 熟练使用分支管理。
- **代码托管平台**: GitHub、GitLab 或 Gitee, 用于代码的远程仓库管理和团队协作。

包管理工具

- **NPM/Yarn**: 安装、更新、卸载前端依赖包, 了解 `package.json` 文件和常用 `npm` 命令。

构建工具

- **Webpack**: 学习模块化打包、配置文件、加载器与插件的使用。
 - **Vite**: 现代化打包工具, 了解其优点并能用于开发。
-

5. 接口与数据处理

- **HTTP 基础**: 了解 HTTP 请求、响应、状态码 (200、404、500 等)。
 - **RESTful API**: 理解 RESTful 设计, 使用 RESTful 风格接口。
 - **数据请求库**: 掌握 Axios 的基本用法, 如 GET/POST 请求, 设置请求头等。
 - **JSON**: 熟悉 JSON 格式和 JSON.parse、JSON.stringify 的使用。
 - **跨域**: 理解同源策略, 了解跨域处理方法 (如 CORS、代理)。
-

6. 项目实践

- **小型项目**: 如 To-do 应用、天气查询、新闻展示等, 增强代码实战能力。
 - **组件化开发**: 将页面功能拆分成组件, 独立封装和复用。
 - **项目架构**: 搭建项目目录结构 (如分为 components、views、assets 等), 理清模块间关系。
 - **调试**: 使用 Chrome DevTools 调试页面, 分析网络请求、检查 DOM 节点、控制台输出等。
 - **代码优化**: 学习代码优化策略 (如减少 DOM 操作、优化循环、图片懒加载)。
-

7. 进阶学习

TypeScript

- **类型系统**: 学习基本类型定义 (string、number、boolean 等)。
- **接口与泛型**: 定义接口、使用泛型增强代码的灵活性。
- **类型断言**: 类型转换与安全性检测。

性能优化

- **前端性能优化**: 学习懒加载、代码分割 (如 import() 动态加载)、图片压缩等。
- **缓存**: 使用缓存策略 (如本地存储、sessionStorage 和缓存控制)。
- **服务端渲染 (SSR)**: 学习服务端渲染技术, 提高首屏加载速度。

Web 安全

- **常见安全问题**: 如 XSS (跨站脚本攻击)、CSRF (跨站请求伪造)。
 - **防御措施**: 输入过滤、转义特殊字符、Token 验证等。
-