

前端代码规范

主要参考：[阿里前端代码规范](#)

一、项目结构规范

1. 全局配置在最外层vite.config.ts。
2. 主要项目结构如下，在src文件中
 - assets：为静态资源，里面存放 images, styles, icons 等静态资源，静态资源命名格式为 kebab-case。
 - components：此目录应按照组件进行目录划分，目录命名为 KebabCase
 - api：api 中的方法名字要与后端 api url 尽量保持语义高度一致性。对于 api 中的每个方法要添加注释。
 - config：定义一些接口类型。
 - router：path除了采用kebab-case命名规范以外，必须以 / 开头，即使是children里的path也要以 / 开头。
 - views：存放页面。
 - hook：一些公用的钩子。
 - datas：模拟数据。
 - store：按照业务进行拆分不同的 js 文件。

二、编程规范

2.1 命名规范

文件命名全部采用驼峰式，有复数结构时，要采用复数命名法，缩写不用复数。

2.2 HTML规范

2.2.1 缩进

- 缩进使用 2 个空格（一个 tab）；
- 嵌套的节点应该缩进。

2.2.2 分块注释

在每一个块状元素，列表元素和表格元素后，加上一对 HTML 注释。

2.2.3 引号

使用双引号(" ")而不是单引号(' ')。

2.3 CSS规范

2.3.1 命名

- 类名使用小写字母，以**中划线**分割
- id使用**驼峰式**命名

2.3.2 选择器

- 避免使用标签名
- 使用直接子选择器
- 避免使用 ID 选择器及全局标签选择器防止污染全局样式

2.3.4 属性及值的编写

- 尽量使用缩写属性
- 每个选择器及属性独占一行
- 省略0后面的单位

2.4 LESS规范

2.4.1 代码组织

- 将公共less文件放在assets/style/base.less
- 按以下顺序组织
 - 1、@import;
 - 2、变量声明;
 - 3、样式声明;

```
@import "mixins/size.less";
@default-text-color: #333;
.page {
  width: 960px;
  margin: 0 auto;
}
```

2.4.2 编写

避免嵌套层级过多

2.5 Typescript规范

2.5.1命名

1. 使用PascalCase为类型命名。
2. 不要使用 `I` 做为接口名前缀。
3. 使用PascalCase为枚举值命名。
4. 使用camelCase为函数命名。
5. 使用camelCase为属性或本地变量命名。
6. 不要为私有属性名添加 `_` 前缀。
7. 尽可能使用完整的单词拼写命名。

2.5.2类型

1. 不要导出类型/函数，除非你要在不同的组件中共享它。
2. 不要在全局命名空间内定义类型/值。
3. 共享的类型应该在 `types.ts` 里定义。
4. 在一个文件里，类型定义应该出现在顶部。

2.5.3 `null` 和 `undefined` :

1. 使用 `undefined`，不要使用 `null`。

2.5.6风格

1. 使用`arrow函数`代替匿名函数表达式。
2. 只需要的时候才把arrow函数的参数括起来。比如，`(x) => x + x`是错误的，下面是正确的做法：
 - `x => x + x`
 - `(x,y) => x + y`
 - `<T>(x: T, y: T) => x === y`
3. 总是使用 `{}` 把循环体和条件语句括起来。
4. 开始的 `{` 总是在同一行。
 - `for (var i = 0, n = str.length; i < 10; i++) { }`
 - `if (x < 10) { }`
 - `function f(x: number, y: string): void { }`
5. 每个变量声明语句只声明一个变量（比如使用 `var x = 1; var y = 2;` 而不是 `var x = 1, y = 2;`）。
6. `else` 要在结束的 `}` 后另起一行。