

html 规范:

1. 正确的文档类型;
2. 使用小写元素名,看起来清爽且容易编写;
3. 关闭每个需要关闭的标签;
4. 关闭空的 HTML 元素,如<meta />;
5. 属性名用小写;
6. 属性值用双引号;
7. 图片添加 alt、title 属性;
8. 避免一行代码过长,尽量少于 80 个字符;
9. 不要无缘无故添加空行;
10. 缩进建议使用四个空格,可以使用 Tab;
11. H5 中不推荐省略<html><body><head>;
12. 多行注释第一个字符缩进两个空格,更易阅读;
13. 载入 js 和样式表时,type 不是必需的;
14. 页面尽量不要直接写 js, css 代码;
15. 不要有重复 id 和空 id, 空 class, 和空属性;

(1)页面结构

容器: container

页头: header

内容: content/container

页面主体: main

页尾: footer

导航: nav

侧栏: sidebar

栏目: column

页面外围控制整体佈局宽度: wrapper

左右中: left right center

(2)导航

导航: nav

主导航: mainnav

子导航: subnav

顶导航: topnav

边导航: sidebar

左导航: leftsidebar

右导航: rightsidebar

菜单: menu

子菜单: submenu

标题: title

摘要: summary

(3)功能

标志: logo

广告: banner

登陆: login

登录条: loginbar

注册: register

搜索: search

功能区: shop

标题: title

加入: joinus

状态: status

按钮: btn

滚动: scroll

标籤页: tab

文章列表: list

提示信息: msg

当前的: current

小技巧: tips

图标: icon

注释: note

指南: guild

服务: service

热点: hot

新闻: news

下载: download

投票: vote

合作伙伴: partner

友情链接: link

版权: copyright

css 规范:

1. sass 编写只能一个父级 class 包括起来，一个页面不要有多个全局属性；
2. 不要有 id 作为选择器；
3. 减少过多的层级选择；
4. 命名规范：一律小写，横线-来为选择器命名，尽量是英文单词；
5. 尽量使用缩写模式，以及属性值 0.8rem 写成 .8rem；
6. 盒子编写一般格式：定位—宽高一字体—颜色—其他；
7. 移动端尽量使用 flex 布局；

js 规范:

1. 命名规范：全局变量为大写，类首字母大写；
2. 结构清晰：运算符前后加空格，括号后加分号，冒号后加空格，将右花括号独立放在一行；
3. 键值对最后不要添加逗号；
4. ``或者\\拼接字符串；

5. 注释：尽量写有意义；
6. 编码：统一 utf-8；
7. 配置 eslint 进行开发；

```
// 导入插件
// import jq from './jquery.html';
(function(){
    // 接口
    let name_request = '../..//成绩列表.json';

    let Ep = {
        // 初始化
        init(){
            this.handler();
        },
        // 事件句柄
        handler(){
            // that代理this
            // let that = this;
            // 事件委托
            this.second_menu.addEventListener('click',function(event){
                var evt = event || window.event;
                var target = evt.target || evt.srcElement;
                if(target.nodeName.toLowerCase() == 'div'){
                    // ...
                }
            });
            // 调用接口
            this.get.get_url({}, function(data){
                // ...业务
            })
        },
        // 继续添加方法
        load(){
            // 渲染
        },
        // 统一接口位置
        get:{
            get_url(args, callback){
                $.post(name_request, args, function(data){
                    callback(data)
                })
            }
        }
    };
    // 立即执行
    Ep.init();
})();
```

Vue 教程

1. 官方教程

<https://cn.vuejs.org/v2/guide/>