

css规范

和谐而论，编写组件前请认真研读此规范，然后再开始编码！

组织规范

FrozenUI 的模块组织方式分为四个部分：

base（reset + type + animation + icon + mixin + variable）

util（arrowlink + border + grid + layout）

component（UI组件，包括css组件和js插件中使用的）

vip（业务组件）

可以参考下面的规范在此基础上扩展出自己的业务组件。

命名规范

统一使用ui为前缀作为命名空间。

class 名称中只能出现小写字母和破折号（dashe）（不是下划线，也不是驼峰命名法）。破折号应当用于相关 class 的命名（类似于命名空间）（例如 .ui-btn 和 .ui-btn-danger）。

避免过度任意的简写。 .btn 代表 button，但是 .b 不能表达任何意思。

class 名称应当尽可能短，并且意义明确，不要使用表现形式（presentational）的名称。

基于最近的父 class 或基本（base） class 作为新 class 的前缀。

模块名如果是两个单词组合使用简单连写的方式，如ui-tooltips而不是ui-tool-tips。

`.ui-btn-lg.disabled`

{命名空间}-{模块}-{属性}.{状态}

`.ui-dialog-cnt`

{命名空间}-{模块}-{子模块}

描述属性的class应该基于当前的class，层级不超过3级

`.ui-form-item ui-form-item-link`

{命名空间}-{模块}-{属性}.{状态}

除了常用状态的class，不能直接使用其他不带前缀的class，如

```
<div class="ui-btn primary">确认</div>
```

应该写为

```
<div class="ui-btn ui-btn-primary">确认</div>
```

不强制所有class都必须带父class，简单的组件可以直接使用子class，如

```
<span class="ui-txt-highlight">ui-txt-highlight</span>
```

而不是

```
<span class="ui-txt ui-txt-highlight">ui-txt-highlight</span>
```

因为是在移动端使用，dom结构并不会太复杂，因此不建议过多使用class嵌套，可以直接使用标签名的直接使用标签名。

如建议使用

```
<ul class="ui-list">
  <li>1</li>
  <li>2</li>
  <li>3</li>
</ul>
```

而不使用

```
<ul class="ui-list">
  <li class="ui-list-item">1</li>
  <li class="ui-list-item">2</li>
  <li class="ui-list-item">3</li>
</ul>
```

####状态类:

单词	意思
show	显示
hide	隐藏
current	当前状态
active	激活态
checked	选中态
selected	已选中状态
disabled	失效状态
done	完成状态
focus	聚焦状态
blur	失去焦点状态

约定的一些简写:

缩写	原单词
-s	small

- lg large
- l left
- r right
- t top
- b bottom
- thumb thumbnail
- img images
- nav navigation
- cnt content
- hd header
- bd body
- ft footer
- txt text
- btn button
- multi 多个
- info 信息内容

#####一些常用的属性或模块:

单词	意思
-wrap	外层
-default	默认样式
-pure	简版
-stable	稳的, 用于灰色背景
-border/-outline	带边框的
-halve	两等分
-trisect	三等分
-cover	通栏
-tiled	平铺
-vertical	垂直
-horizontal	横向
-divider	分割
-muted	弱的
-group	组的
-halve	2等分
-trisect	3等分
-info	信息
-news	消息
-success	成功的
-warn	警告的
-highlight	高亮的

-item	子元素
-title	标题
-subtitle	小标题
-state	状态
-guide	引导性
-link	链接
-bar	横块

只要词义表达了组件要实现的功能或者要表现出来的的外观就可以了。

例如：

```
<div class="ui-tooltips ui-tooltips-warn">
  <div class="ui-tooltips-cnt ui-tooltips-cnt-link">
    <i></i>当前网路不可用，请检查你的网路设置。
  </div>
</div>
```

SASS编写

避免不必要的嵌套。只有在需要给父元素增加样式并且同时存在多个子元素时才需要考虑嵌套。

字体颜色，字体大小，行高，border颜色，背景颜色等基本属性作为变量，并mixin常用代码段。

CSS编码规范

参考<http://alloyteam.github.io/code-guide/#css>