编码规范

JAVA（后端）

1.命名规范

（1）所有的标示符都只能用 ASCII 字母（A-Z 或 a-z）、数字（0-9）和下划线“\_”。

（2）一个唯一包名的前缀总是全部小写的字母。例如：www.tonysun.cc

（3）类名是一个名词，采用大小写混合的方式，每个单词的首字母大写。例如：Tony。

（4）接口的大小写规则与类名相似：例如：Tony。

（5）方法名是一个动词或动词词组，采用大小写混合的方式，第一个单词的首字母小写，其后单词的首字母大写。例如：setNeekeName(String neekeName)。

（6）变量名第一个字母小写，任何中间单词的首字母大写。变量名应简短且可以顾名思义，易于记忆。例如：neekeName、neekeAddress。避免单个字符的变量名，除非是一次性的临时变量。

（7）常量的声明应该全部大写，每个单词之间用“\_”连接。例如：final String WWW\_TONY\_CN = "www.tonysun.cc";

2.注释规范

（1）注释尽可能使用“//”；对于所有的 Javadoc 的注释则使用“/\*\* \*/”；而临时对代码块进行注释尽量使用“/\* \*/”。

（2）所有的源文件都应该在开头有一个注释，其中列出文件名、日期和类的功能概述。

（3）每个方法必须添加文档注释（类的 main() 方法除外）。

（4）每个属性必须添加注释。

（5）代码中至少包含 15% 的注释。

（6）注释使用中文。

3.缩进排版规范

（1）避免一行的长度超过 60 个字符。

（2）使用 Intellij IDEA 的源代码的格式化功能完成代码的缩进排版(Ctrl + Alt + L)。

4.文件名规范

（1）一个 Java 源文件只能存储一个 Java 类。

（2）文件名与 Java 类名相同。

（3）一个类文件的代码行不超过 200 行。

5.声明规范

（1）一行声明一个变量。

（2）不要将不同类型变量的声明放在同一行。

（3）只在代码块的开始处声明变量。

（4）所有的变量必须在声明时初始化。

（5）避免声明的局部变量覆盖上一级声明的变量。

（6）方法与方法之间以空行分隔。

6.语句规范

（1）每行至少包含一条简单语句。

（2）在 return 语句中，返回值不使用小括号“()”括起来。

（3）if 语句总是用“{”和“}”括起来。

（4）在 for 语句的初始化或更新子句中，避免因使用三个以上变量，而导致复杂度提高。

（5）当 switch 的一个 case 顺着往下执行时（因为没有 break 语句），通常应在 break语句的位置添加注释。

Python（爬虫）

1.编码格式声明

通常，编码格式声明是必需的。如果 Python 源码文件没有声明编码格式，Python 解释器会默认使用 ASCII 编码，一旦源码文件包含非 ASCII 编码的字符，Python 解释器就会报错。以 UTF-8 为例，以下两种编码格式声明都是合乎规则的。

2.缩进

统一使用 4 个空格进行缩进。绝对不要用tab, 也不要tab和空格混用。对于行连接的情况，我一般使用 4 空格的悬挂式缩进。

3.引号

自然语言使用双引号

机器标识使用单引号

正则表达式使用双引号

4.空行

编码格式声明、模块导入、常量和全局变量声明、顶级定义和执行代码之间空两行

顶级定义之间空两行，方法定义之间空一行。

在函数或方法内部，可以在必要的地方空一行以增强节奏感，但应避免连续空行。

5.导入模块

导入总应该放在文件顶部，位于模块注释和文档字符串之后，模块全局变量和常量之前。导入应该按照从最通用到最不通用的顺序分组，分组之间空一行

6.命名规范

（1）模块尽量使用小写命名，首字母保持小写，尽量不要用下划线。

（2）类名使用驼峰（CamelCase0命名风格，首字母大写，私有类可用一个下划线开头。

（3）函数名一律小写，如有多个单词，用下划线隔开。

（4）私有函数可用一个下划线开头。

（5）变量名尽量小写, 如有多个单词，用下划线隔开。

（6）常量采用全大写，如有多个单词，使用下划线隔开。

Vue（前端）

views 命名

views 文件夹下面是由 以页面为单位的 vue 文件 或者 模块文件夹 组成的，放在 src 目录之下，与 components、assets 同级。

views 下的文件夹命名

1. views 下面的文件夹代表着模块的名字
2. 由名词组成（car、order、cart）
3. 单词只能有一个（good: car order cart）（bad: carInfo carpage）
4. 尽量是名词（good: car）（bad: greet good）
5. 以小写开头（good: car）（bad: Car）

views 下的 vue 文件命名

1. views 下面的 vue 文件代表着页面的名字
2. 放在模块文件夹之下
3. 只有一个文件的情况下不会出现文件夹，而是直接放在 views 目录下面，如 Login Home
4. 尽量是名词
5. 大写开头，开头的单词就是所属模块名字（CarDetail、CarEdit、CarList）
6. 名字至少两个单词（good: CarDetail）（bad: Car）
7. 常用结尾单词有（Detail、Edit、List、Info、Report）
8. 以 Item 结尾的代表着组件（CarListItem、CarInfoItem）

vue 方法放置顺序

1. components
2. props
3. data
4. created
5. mounted
6. activited
7. update
8. beforeRouteUpdate
9. metods
10. filter
11. computed
12. watch

method 自定义方法命名

1. 动宾短语（good：jumpPage、openCarInfoDialog）（bad：go、nextPage、show、open、login）
2. ajax 方法以 get、post 开头，以 data 结尾（good：getListData、postFormData）（bad：takeData、confirmData、getList、postForm）
3. 事件方法以 on 开头（onTypeChange、onUsernameInput）
4. init、refresh 单词除外
5. 尽量使用常用单词开头（set、get、open、close、jump）
6. 驼峰命名（good: getListData）（bad: get\_list\_data、getlistData）

data props 方法注意点

1. 使用 data 里的变量时请先在 data 里面初始化
2. props 指定类型，也就是 type
3. props 改变父组件数据 基础类型用 $emit，复杂类型直接改
4. ajax 请求数据用上 isLoading、isError 变量
5. 不命名多余数据，现在是详情页、你的数据是 ajax 请求的，那就直接声明一个对象叫 d，而不是每个字段都声明
6. 表单数据请包裹一层 form

生命周期方法注意点

1. 不在 mounted、created 之类的方法写逻辑，取 ajax 数据，
2. 在 created 里面监听 Bus 事件