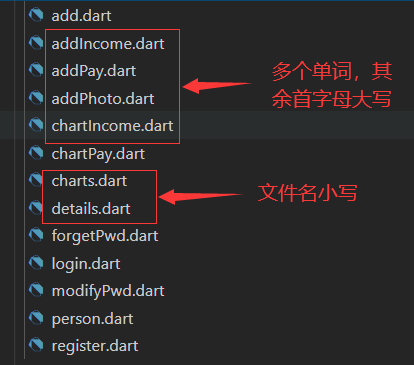
**代码编写规范**

1. **代码缩进**

代码缩进为一律为一个制表符

1. **命名规范**

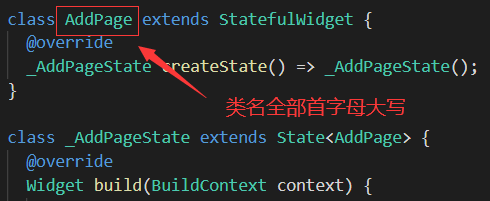
2.1 文件名首字母小写，若其由多个单词构成，除第一个单词之外，其他单词首字母大写



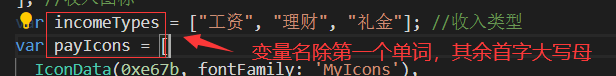
2.2 包名全部小写



2.3 类名首字母大写，若类名由多个单词构成，每个单词首字母大写



2.4 变量名、方法名首字母小写，若其由多个单词构成，除第一个单词之外，其他单词首字母大写



2.5 常量名全部大写

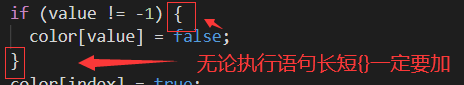
2.6 尽量不缩写，除非一看就明白的单词，如：password缩写为Pwd

2.7 尽量不使用拼音或者英语掺杂拼音

1. **代码行**

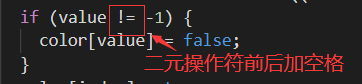
3.1 一行代码只做一件事情，如只定义一个变量，或只写一条语句。这样的代码容易阅读，并且方便于写注释。

3.2 if、for、while、do等语句自占一行，执行语句不得紧跟其后。不论执行语句有多少都要加{}。这样可以防止书写失误。



3.3 ‘，’之后要留空格，如Function(x, y, z)。如果‘;’不是一行的结束符号，其后要留空格，如for (initialization; condition; update)。

3.4 赋值操作符、比较操作符、算术操作符、逻辑操作符、位域操作符，如“=”、“+=” “>=”、“<=”、“+”、“\*”、“%”、“&&”、“||”、“<<”,“^”等二元操作符的前后应当加空格。



3.5 一元操作符如“!”、“~”、“++”、“–”、“&”（地址运算符）等前后不加空格。

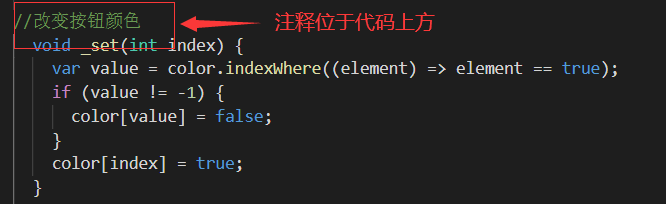
3.6 像“［］”、“.”、“->”这类操作符前后不加空格。

3.7代码行最大长度宜控制在70至80个字符以内。代码行不要过长，否则眼睛看不过来，也不便于打印。

3.8 长表达式要在低优先级操作符处拆分成新行，操作符放在新行之首（以便突出操作符）。拆分出的新行要进行适当的缩进，使排版整齐，语句可读。

1. **注释**
   1. 注释应当准确、易懂，防止注释有二义性。
   2. 尽量避免在注释中使用缩写，特别是不常用缩写。
   3. 注释的位置应放在被描述的代码右侧，且距离一个空格；或放在代码上方。





* 1. 边写代码边注释，修改代码同时修改相应的注释，以保证注释与代码的一致性。
  2. 如果代码本来就是清楚的，则不必加注释。
  3. 程序中的注释不可喧宾夺主，注释太多了会让人眼花缭乱。