

Dart const常量构造函数详解

原创

野猿新一



于 2019-10-17 10:37:18 发布

6808

收藏

版权

分类专栏:

Dart

文章标签:

Dart

const

常量构造函数



Dart 专栏收录该内容

0 订阅

12 篇文章

订阅专栏

常量 **构造函数** 总结如下几点:

- 常量构造函数需以const关键字修饰
- const构造函数必须用于成员变量都是final的类
- 构建常量实例必须使用定义的常量构造函数
- 如果实例化时不加const修饰符, 即使调用的是常量构造函数, 实例化的对象也不是常量实例

下面结合实例, 对以上几点加以说明

正确的常量构造函数定义

根据以上的总结, 定义一个Point类, 包含一个常量构造函数, 注意其成员变量都是final类型, 且构造函数用const修饰

```
1 class Point {  
2     final int x;  
3     final int y;  
4     const Point(this.x, this.y);  
5 }
```

常量构造函数需以const关键字修饰

如下代码定义一个const对象, 但是调用的构造方法不是const修饰的, 则会报 **The constructor being called isn't a const constructor.**错误

```
1 void main() {  
2     const point = Point(1, 2); // 报错  
3 }  
4  
5 class Point {  
6     final int x;  
7     final int y;  
8     Point(this.x, this.y);  
9 }
```

const构造函数必须用于成员变量都是final的类

如下代码中成员变量x为非final, 会报 **Can't define a const constructor for a class with non-final fields.**错误

```
1 class Point {  
2     int x;  
3     final int y;  
4     const Point(this.x, this.y);  
5 }
```

构建常量实例必须使用定义的常量构造函数

如下代码, 定义一个const对象, 但是调用的却是非常量构造函数, 会报 **The constructor being called isn't a const constructor.**错误

```
1 void main() {
2   var point = const Point(1, 2); // 报错
3   print(point.toString());
4 }
5
6 class Point {
7   int x;
8   int y;
9   Point(this.x, this.y); // 非const构造函数
10
11   String toString() {
12     return 'Point(${x}, ${y})';
13   }
14 }
```

如果实例化时不加const修饰符，即使调用的是常量构造函数，实例化的对象也不是常量实例

如下代码，用常量构造函数构造一个对象，但是未用const修饰，那么该对象就不是const常量，其值可以再改变

```
1 void main() {
2   var point = Point(1, 2); // 调用常量构造函数，但是未定义成常量
3   print(point.toString());
4   point = Point(10, 20); // 可以重新赋值，说明定义的变量为非常量
5   print(point.toString());
6 }
7
8 class Point {
9   final int x;
10  final int y;
11  const Point(this.x, this.y);
12
13  String toString() {
14    return 'Point(${x}, ${y})';
15  }
16 }
```