

# 【Flutter】Dart 数据类型 ( dynamic 数据类型 )

原创 韩曙亮 于 2021-02-06 14:09:04 发布 3061 收藏 1 版权

分类专栏: Flutter 文章标签: Flutter



Flutter 专栏收录该内容

13 订阅

139 篇文章

订阅专栏

## 文章目录

- 一、dynamic 数据类型
- 二、dynamic 变量无法进行语法检查
- 三、dynamic 变量运行时类型修改
- 四、完整代码示例
- 五、相关资源

Dart 语言中有 `dynamic` , `var` , `Object` , 3 种数据类型 ;

## 一、dynamic 数据类型

**dynamic 数据类型** : `dynamic` 是 Dart 语言中的 **动态数据类型** , 通过 `dynamic` 关键字定义的变量 , 在 **编译时不会管定义的数据类型** , **运行时才处理变量的类型** , 根据变量赋值的类型推测当前的 `dynamic` 数据类型的运行时数据类型 , `dynamic` 变量只有在运行时才知道数据类型 ;

**dynamic 变量定义 :**

```
1 | dynamic 变量名称 = 变量值
```

**dynamic 变量代码示例 :**

```
1 | dynamicDemo(){
2 |     dynamic d = "CSDN";
3 |
4 |     // 打印 dynamic 变量的运行时类型
5 |     print(d.runtimeType);
6 |     // 打印 dynamic 变量值
7 |     print(d);
8 | }
```

**运行结果 :**

```
1 | String
2 | CSDN
```

## 二、dynamic 变量无法进行语法检查

通过 dynamic 变量可以定义任何数据类型，在编译时会使用 Dart 语言的静态语法检查失效；

如：先定义了 dynamic 变量，赋值了一个字符串，然后调用一个不属于该字符串对象的方法，这里肯定出错了，但是静态的语法检查肯定检查不出来，只能在最后运行报错时知道这里出现了错误；

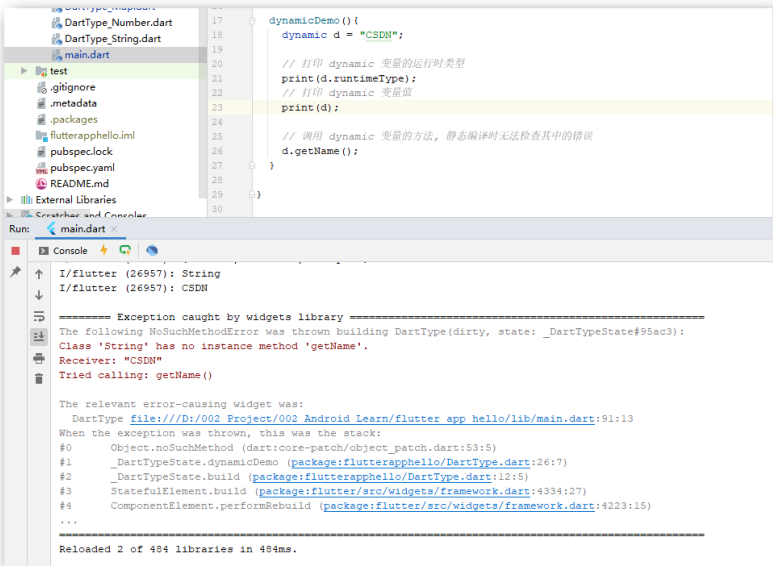
```
dynamicDemo() {
  dynamic d = "CSDN";

  // 打印 dynamic 变量的运行时类型
  print(d.runtimeType);
  // 打印 dynamic 变量值
  print(d);

  // 调用 dynamic 变量的方法，静态编译时无法检查其中的错误
  d.getName();
}
```

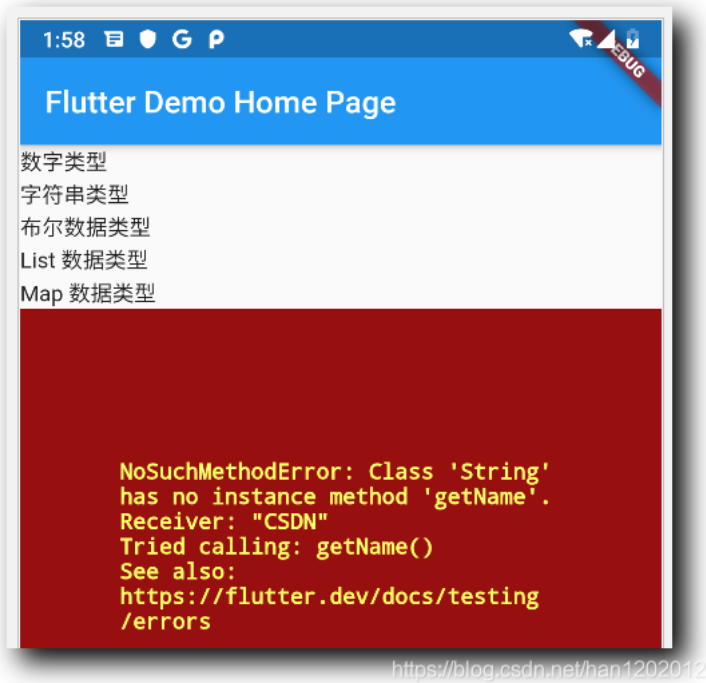
<https://blog.csdn.net/han1202012>

运行时出现错误：



<https://blog.csdn.net/han1202012>

这 Flutter 界面中也有报错信息：



### 三、dynamic 变量运行时类型修改

dynamic 数据类型 [运行时类型可以改变](#)，上述的 dynamic d 类型初始化为 String 类型，中途将整型赋值给该变量，此时该变量的类型变成了 int 类型；

代码示例：先为 dynamic 变量赋值了 String 类型值“CSDN”，然后为 dynamic 变量赋值了 int 类型值 666；

```
1 dynamicDemo(){
2   dynamic d = "CSDN";
3
4   // 打印 dynamic 变量的运行时类型
5   print(d.runtimeType);
6   // 打印 dynamic 变量值
7   print(d);
8
9   // 调用 dynamic 变量的方法，静态编译时无法检查其中的错误，运行时会报错
10  //d.getName();
11
12  // 为 dynamic 变量赋值 int 数据
13  d = 666;
14  // 打印 dynamic 变量的运行时类型
15  print(d.runtimeType);
16  // 打印 dynamic 变量值
17  print(d);
18 }
```

运行结果：

```
1 String
2 CSDN
3 int
4 666
```

## 四、完整代码示例

---

```
1 import 'package:flutter/material.dart';
2
3 class DartType extends StatefulWidget {
4   @override
5   _DartTypeState createState() => _DartTypeState();
6 }
7
8 class _DartTypeState extends State<DartType> {
9   @override
10  Widget build(BuildContext context) {
11
12    dynamicDemo();
13
14    return Container();
15  }
16
17  dynamicDemo(){
18    dynamic d = "CSDN";
19
20    // 打印 dynamic 变量的运行时类型
21    print(d.runtimeType);
22    // 打印 dynamic 变量值
23    print(d);
24
25    // 调用 dynamic 变量的方法，静态编译时无法检查其中的错误，运行时会报错
26    //d.getName();
27
28    // 为 dynamic 变量赋值 int 数据
29    d = 666;
30    // 打印 dynamic 变量的运行时类型
31    print(d.runtimeType);
32    // 打印 dynamic 变量值
33    print(d);
34  }
35
36 }
37
```

运行结果：

```
1 String
2 CSDN
3 int
4 666
```

## 五、相关资源

---

参考资料：

- Dart 开发者官网：<https://api.dart.dev/>

- **Flutter** 中文网 ( 非官方, 翻译的很好 ) : <https://flutterchina.club/>,  
<http://flutter.axuer.com/docs/>
- **Flutter** 官网 : <https://flutter.dev/> ( 被墙 )
- 官方 **GitHub** 地址 : <https://github.com/flutter>
- **Flutter** 相关问题 : <https://flutterchina.club/faq/> ( 入门阶段推荐看一遍 )

博客源码下载 :

- **GitHub** 地址 : [https://github.com/han1202012/flutter\\_app\\_hello](https://github.com/han1202012/flutter_app_hello) ( 随博客进度一直更新, 有可能没有本博客的源码 )
- 博客源码快照 : <https://download.csdn.net/download/han1202012/15110705> ( 本篇博客的源码快照, 可以找到本博客的源码 )