

iOS初学者须知的Swift String的类型转换方法?

作者: [balala](#)

原文链接: <https://ld246.com/article/1449728683461>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

随着swift2.0的问世，越来越多的开发者使用这门语言进行iOS应用开发，在开发中我们常常需要用到种字符串、类、接口等等，今天小编和大家分享的就是swift2.0中String的类型转换方法，一起来看吧。

其实，swift语言属于强类型语言，但简单的String类型转换其实还是比较方便的，最暴力的就是强制型转换，百试百灵：

```
func toString<T>(a: T) -> String{
// Way one, format
//return "(a)"
// Way two, new string
return String(a)
}
toString(123)
toString(0.007)
toString([4, 8, 32, 5 ])
toString("test".startIndex)
```

同样依葫芦画瓢也能转回来：

```
Int("123")
Float("3.1")
```

但这个太简单、太有局限性了！建议还是用下面这个，利用了New的时候format方法，简直就是sprintf()的替代品！

```
// Way 1, 格式化转String
String(format:"%i, 0x%x, 0o%o, %.2f", 31, 31, 31, 3.1)
// Way 2
String().stringByAppendingFormat("%x",31)
String().stringByAppendingFormat("%o",31)
```

同样的，有了sprintf()，怎么能没有sscanf()??

```
//十六进制String转整数
var res:UInt32 = 0
NSScanner.localizedScannerWithString(str).scanHexInt(&res)
```

当然，这个用起来就要麻烦许多，类似的一堆API:

```
NSScanner.localizedScannerWithString(str).scanInt(&res)
NSScanner.localizedScannerWithString(str).scanFloat(&res)
NSScanner.localizedScannerWithString(str).scanDouble(&res)
NSScanner.localizedScannerWithString(str).scanInteger(&res)

NSScanner.localizedScannerWithString(str).scanHexFloat(&res)
```

最后，虽然ASCII码现在不大常用了，但相信也有开发者喜欢ASCII码的转换函数吧：

```
/**
ASCII value to char
*/
func ASCII ( n : Int) -> String? {
guard (n >= 0 && n < 255 ) else {
return nil
}
return String(format:@"%c", n)
}

/**
ASCII char to int value
*/
func ASCII (c : String) -> Int8?{
let str:NSString = c
let n = str.UTF8String[0]
guard n>=0 else {
return nil
}
return n
}

ASCII(70)
ASCII("B")
```

以上就是swift2.0中string类型转换相关操作方法，强烈建议大家在自己的环境中去尝试实现上面的些方法，看看效果如何。自有自己动手实践了，才能更好的转变成自己的东西。

相关文章：《Swift2.0 中String的正则匹配详解》

<http://www.maiziedu.com/group/article/10076/>