

Xcode

常用快捷键

- command delete 删除一行
- command / 注释
- Command [/] 前后缩进
- command ctrl up down .h和.m的切换
- command 左右 行首行尾切换
- command shift o 快速查找文件
- command shift O 打开官方文档
- cmd ctrl e 统一修改某一方法内某一变量名
- edit refactor extract 把代码转换为一个方法

常用操作

pragma mark 功能说明 -

调试

- 设置断点：在想要设置断点的代码左端点击一下
- 编辑断点（加入条件）
 - condition 返回一个bool，只有bool为真的时候触发断点
 - ignore 忽略前n次断点，在n+1次触发
 - 断点出发时，xcode执行的操作
 - shell command 接收相应参数列表
 - log message 消息输出到控制台
- Symbolic Breakpoint
 - command 7
 - 程序中所有symbol出现的位置都会断点
- 监控断点
 - 我们调试程序的大部分时候都是为了看某个变量的变化，可以通过为变量添加监控断点来做到这点，找到变量第一次出现的地方，进入debug模式后variables view中右键变量，选择watch。

视图树

参考[博客](#)

lldb

参考

调试过程中是可以使用NSLog的，但是实际开发中NSLog的效率非常低下，首先编译过程非常长，然后NSLog对release性能影响非常大。

po: print object 表示对象的文本描述

p: 打印基本数据类型

call: 执行一段代码，比如call NSLog(@"%@",object);

expr: 动态执行指定表达式

bt: 打印堆栈

image lookup --address 0x0000000104c25550 查找出错地址对应的代码位置

lldb断点打印log

```
1  help
2  单个执行，显示所有命令介绍，用户自定义命令在最后
3  help print 显示print命令的使用帮助
4
5  print
6  打印参数信息
7
8  https://sourceware.org/gdb/onlinedocs/gdb/Output-Formats.html
9  // 进阶操作
10
11 expression
12 修改参数值
13 (lldb) p count
14 (NSUInteger) $4 = 12
15 (lldb)e count = 42
16 (NSUInteger) $5 = 42
17
18 watchpoint
```

19 某个数据被写入、读取时暂停程序运行

20 watchpoint **set** v -w read_write _abc3

21