

```
////////////////////////////////////
//                               盈烨Objective-C代码规范
////////////////////////////////////

// 作者: Cking iOS 开发团队
// 整理: Zheng Xu
// 日期: 2012.12.7
// 版本: 0.9

// 参考资料:
// Apple: Coding Guidelines for Cocoa
// Google: Objective-C Style Guide
// Three20: Source code style guildelines

#pragma mark - 1.格式化 -

//-----
// 1.1 指针“*”号的位置。
//-----

NSString *varName;

//-----
// 1.2 在 - OR + 和返回值之间留1个空格,方法名和第一个参数间不留空格。
// 花括号的位置可以是以下任意一种, 但要求坚持一种风格。
//-----

- (void)doSomethingWithString:(NSString *)theString
{

}

// OR
- (void)doOtherthingWithString:(NSString *)theString {

}

//-----
// 1.3 当参数过长时,每个参数占用一行,以冒号对齐。
//-----

- (void)doSomethingWith:(NSString *)theFoo
                      rect:(CGRect)theRect
                      interval:(float)theInterval
{

}

//-----
// 1.4 如果方法名比参数名短,每个参数占用一行,至少缩进4个字符,
// 且为垂直对齐(而非使用冒号 对齐)
//-----

- (void)short:(NSString *)theFoo
    longKeyword:(CGRect)theRect
    evenLongerKeyword:(float)theInterval
{

}

#pragma mark - 2.命名 -

//-----
// 2.1 类名(及其category name 和 protocol name)的首字母大写,
// 写使用首字母大写的形式分割单词
//-----

//-----
// 2.2 方法名的首字母小写,且使用首字母大写的形式分割单词。
//-----

- (void)sendData
{

}

//-----
// 2.3 getter的方法名和变量名应相同。不允许使用“get”前缀。
//-----

// 正确
- (NSString *)foo
{
    return @"foo";
}

// 错误
- (NSString *)getFoo
{
    return @"foo";
}

//-----
// 2.4 成员变量使用“_”作为前缀
//-----
NSString *_varName;

#pragma mark - 3.现代语法(下次Seminar讨论内容) -

//-----
// 3.1 默认情况声明属性不再需要@synthesize
//-----

//-----
// 3.1 成员方法不再需要前置声明
//-----

//-----
// 3.3 NSNumber赋值
//-----

- (void)logBoolNum
{
    NSNumber *boolNum = @YES;
    NSLog(@"%@", boolNum); // 输出:1 (@YES: 1 @NO: 0)
}

- (void)logFloatNum
{
    NSNumber *floatNum = @1.3;
    NSLog(@"%@", floatNum); // 输出1.3
}

//-----
// 3.4 数组使用
//-----

- (void)useArray
{
    NSArray *array = @[@"foo", @1.3, @YES, @[@"hello", @"world!"]];
    NSLog(@"%@", array[1]); // 输出:1.3
    NSLog(@"%@", array[3][1]); // 输出:world!
}

//-----
// 3.5 字典使用
//-----

- (void)useDictionary
{
    NSDictionary *dict =
        @{@"key1":@"value1", @"key2" :@[@"key3":@"value3"]};
    NSLog(@"%@", dict[@"key1"]); // 输出value1
    NSLog(@"%@", dict[@"key2"][@"key3"]); // 输出value3
}
```