目标

了解 class Method(类方法)与 instance Method(实例方法)区别

掌握 @property 的使用

掌握对象的生命周期—创建、初始化、销毁

1、了解 class Method(类方法)与 instance Method(实例方法)区别

／／定义

@interface YCProjectManager : NSObject

//打印该对象的相关信息

-(void)print;

+(void)print2;

@end

//实现

@implementation YCProjectManager

static double salary=5000.0; //工资

#pragma mark - 打印该对象的相关信息

-(void)print

{

NSLog(@"姓名: %@ ,性别: %@ ,部门 :%@ ",name,sex,department);

}

+(void)print2

{

NSLog(@"工资 %lf ",salary);

}

@end

／／调用

//YCProjectManager 类名 通过类名调用的方法叫”类方法”

//pm2 对象名 通过对象名|实例名调用的方法叫”实例方法”

YCProjectManager \*pm2 =[[YCProjectManager alloc]init];

//对象名可以调用 – 开头的方法

[pm2 print];

//[pm2 print2];

//类名可以调用 + 开头的方法

[YCProjectManager print2];

在类的声明中，如果方法定义的时候由 - 开头，由 + 开头表示不同的意义。

* 能够控制，只能由 对象来调用这个方法，

方法内部可以直接使用类中所有的属性。

+ 能够控制，只能由 类名调用这个方法，

方法内部能够调用的静态的属性 static 关键字修饰的

self 的用法

2、掌握 @property 的使用

头文件中声明

//想省略 getter setter 方法

//OC 2.0 中提供了一个很好的替代方法，关键字 @property

//一个@property 替代两个方法 setXXX xxx

@property NSString \*name;

@property int age;

@property NSString \*number;

实现文件中实现

//实现 属性的 getter/setter 方法

//用 @synthesize 同步关键字

@synthesize name,age,number;

3、掌握对象的生命周期—创建、初始化、销毁

对象：具有明确属性和行为的个体；

通过类模版，分配的一块堆内存；

内存分配、使用、释放这些过程，就叫做生命周期。(一个人从生到死，叫一个生命周期)

对象的生命周期对应着内存的生命周期：

统统在 NSObject 类中定义好了.

创建 🡪 内存的分配

alloc 方法

使用 -> 内存的初始化

init

销毁 -> 内存的释放

dealloc