

OC

类和对象

面向对象
以事物为中心,事物相互协作共同完成
考虑完成该事物需要哪些对象参与

Objective-C
特性 封装,继承,多态
C语言的超集

类和对象的概念和区别

类
具有相同特征和行为的事物的抽象
特征 如姓名,性别
行为 拍电影,潜规则

对象
是类的实例,(事物)一切为对象
类是对象的类型

类和对象的定义

面向对象

定义
接口部分 对外声明类的特征和行为(.h文件中)
实现部分 行为的具体实现(.m文件中)

接口部分
标志 @interface...@end 声明类的实例变量(成员变量)和方法,即特征行为
包含内容 类名,父类名,实例变量(成员变量),方法

接口代码

```
@interface Teacher : NSObject
{
    NSString *_name;
    NSString *_gender;
    NSInteger _age;
}
-(void)teaching;
-(void)getSalary;
@end
```

实现部分
标志 @implementation...@end 类的行为的具体实现(方法的实现)
实现代码

```
@implementation Teacher
- (void)teaching {
    NSLog(@"我正在进行教学活动");
}
- (void)getSalary {
    NSLog(@"这个月发的不少,可以吃大餐了");
}
@end
```

对象的创建

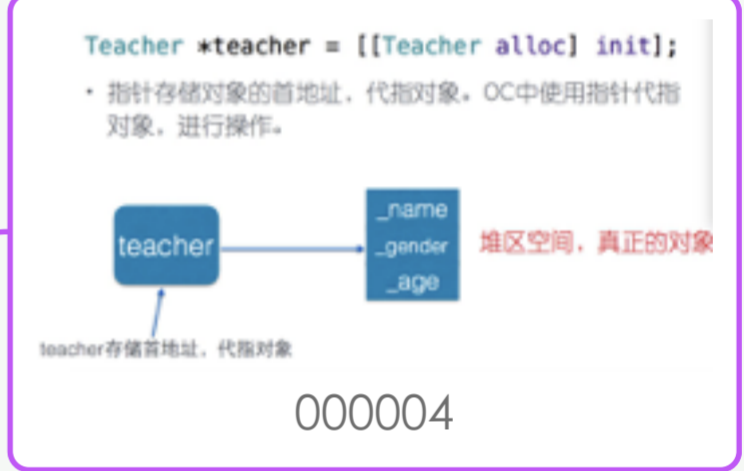
分配内存 在堆区分配内存
初始化 设置初始值

开辟空间
+ (instancetype)alloc;
+ 表示方法是类方法,只能类调用,instancetype返回值可以用id(任意对象类型)替代

初始化
- (instancetype)init;
- 对象的方法,对象调用,instancetype 返回值类型

instancetype和id区别 前者只能作为返回值和参数,id还可以定义变量

```
Teacher *teacher = [[Teacher alloc];
teacher = [teacher init];
Teacher *teacher = [[Teacher alloc] init];
```



OC中指针代指对象

实例变量可见度
@public 都可以访问
@private 只有本类可以访问
@protected 只有本类和子类可以访问,系统默认可见度

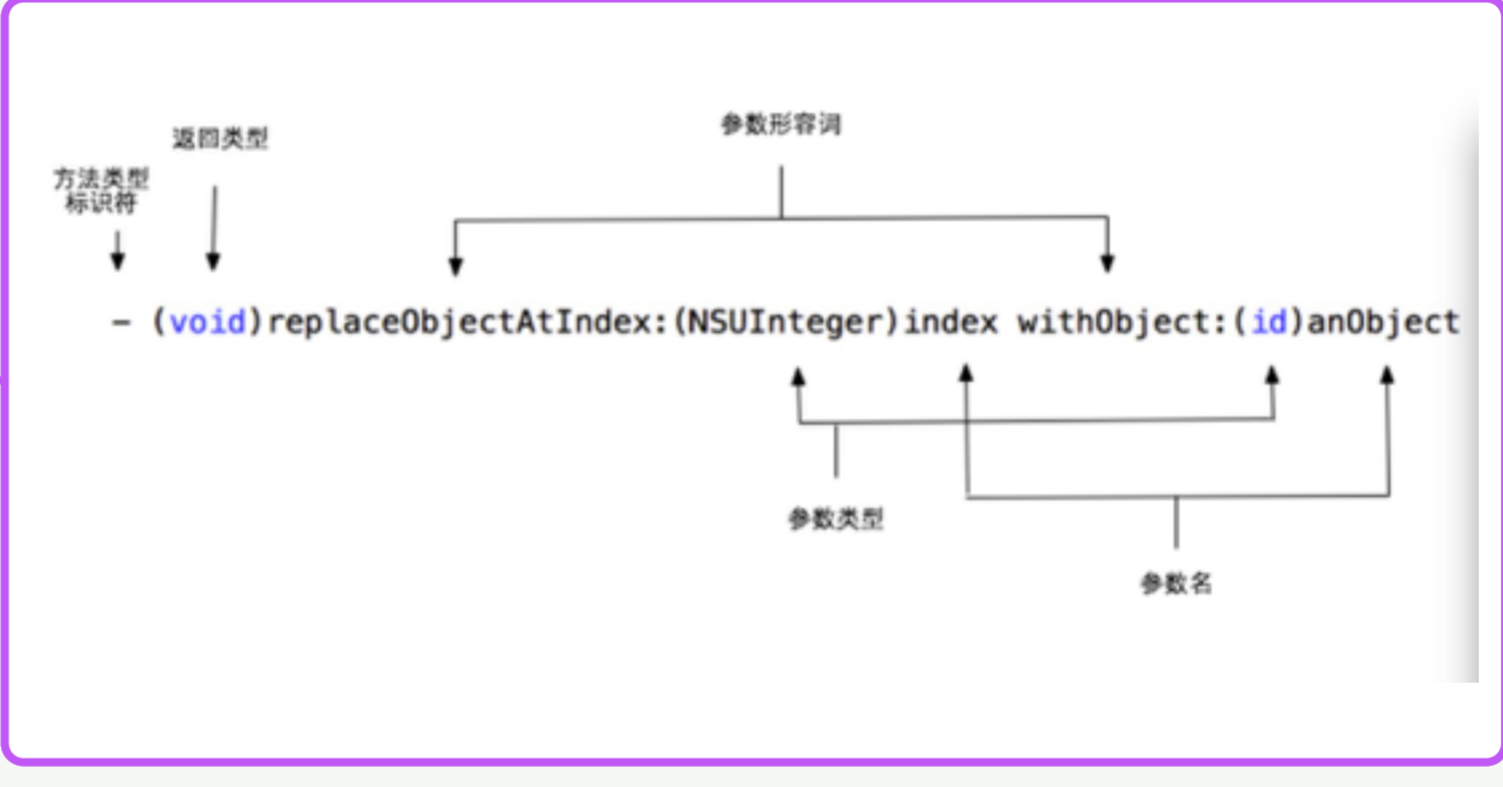
	本类	子类	其他类
@public	可以直接访问实例变量	可以直接访问实例变量	可以访问实例变量
@private	可以直接访问实例变量	不可以访问实例变量	不可以访问实例变量
@protected	可以直接访问实例变量	可以直接访问实例变量	不可以访问实例变量

方法

(+号)类方法 只能被类使用 + (instancetype)alloc;
类方法中不能使用实例变量 - (instancetype)init;

(-号)实例方法 只能对象使用

+/- 方法



方法名
类中不能出现同名方法
“:”标识参数,不能省略,有冒号必须有参数.
冒号属于方法名的一部分

方法声明

```
@interface Teacher : NSObject
{
    NSString *_name;
    NSString *_gender;
    NSInteger _age;
}
//声明一个多参数的方法
-(void)setName:(NSString *)name gender:(NSString *)gender age:(NSInteger)age;
@end
```

```
@implementation Teacher
- (void)setName:(NSString *)name gender:(NSString *)gender age:(NSInteger)age {
    _name = name;
    _gender = gender;
    _age = age;
}
@end
```

setter和getter方法

定义
为单一实例变量赋值的方法是setter(设置器)
获取单一实例变量值的方法称作getter(访问器)

书写格式

与实例变量关系
类部操作都是实例变量
每一个实例变量都需要一对setter getter方法

OC中规定了setter和getter的书写格式。
比如: 实例变量 NSInteger _age;
setter的书写格式如下:
- (void)setAge:(NSInteger)age;
即set+首字母大写的实例变量名(忽略下划线)。
getter的书写格式如下:
- (NSInteger)age;
即返回值类型与变量类型一致,方法名与实例变量名相同(忽略下划线)。

总结
面向对象和面向过程是两种不同的编程思想,思考角度不同,前者以对象为中心,后者以事件为中心。
类是对象的抽象,对象是类的实例。面向对象编程中,先创建对应的类,再由类生成对象,完成我们的开发。
类包含接口和实现两个部分,对象创建分两步:创建对象、初始化。
实例变量有3种常见的可见度:@public、@protected、@private。
方法是OC的核心,采用消息机制(cece meage)。
“-”meage由对象来调用,“-”meage由类来调用。
setter和getter方法访问的是单一实例变量。