# 如何令自己创建的对象具有拷贝功能

让自定义类实现NSCopying和NSMutablecopying协议

浅拷贝

- (id)copyWithZone:(NSZone \*)zone

{

Person \*copy = [[[self class] allocWithZone:zone]init];

copy.firstName = \_firstName ;

copy.lastName = \_lastName;

return copy;

}

深拷贝

- (id) mutableCopyWithZone:(NSZone \*)zone

{

Person \*copy = [[[self class] allocWithZone:zone]init];

copy.firstName = \_firstName ;

copy.lastName = \_lastName;

return copy;

}

# 进程和线程的区别

进程：进程是系统资源分配和调度的一个独立单位

进程：线程是CPU调度的最小单元

线程的作用：执行app的代码；

 一个进程（App）至少有一个线程，这个进程叫做主线程；

进程： 在iOS 中 一个进程就是一个正在运行的一个应用程序;  比如 QQ.app  ，而且一个app只能有一个进程 不像安卓支持多个进程。

**线程**是进程的一部分，它是进程中用来**执行任务的单位**。一个进程可以拥有多个线程，但最少要有一个主线程；每个线程不能独立执行，必须有一个父进程；线程可以拥有自己的堆、栈、计数器和局部变量，一个进程里的全部线程会**共享该进程的资源**。线程是独立运行的，一个线程并不知道其它线程的存在，在线程运行过程中随时都有可能被挂起，方便其他线程运行；线程是有优先级的，当然主线程的优先级最高，在我们编程的时候也可以设置线程1执行完毕后才执行线程2。

**多线程与多进程的对比：**

1.进程之间不能共享内存，线程可以

同一个进程中的线程共享了改进程的很多资源，包括：进程虚拟空间、进程代码段、进程共有数据等，因此线程之间更容易相互通信，多线程的运行效率远远高于多进程；

2.系统创建进程的时候要为其分配系统资源，而创建线程则只需要很小一部分，因此多线程比多进程来的更加容易；

3.多线程可以充分利用处理器（双核或者多核），但是当线程数量达到上限的时候，性能就不在提升了；

4.多线程的进程中一个线程崩溃了就会导致进程崩溃，如果是主线程崩溃会导致程序崩溃；

但是多进程中子进程崩溃了不会影响到其它进程，程序稳定性更好；

5.多线程需要控制线程之间的同步，而多进程则需要控制和主进程之间的交互；

6.如果两个进程之间要相互传输大量的数据，会相当影响性能，多进程适合小数据量传输，密集运算；

**进程的并发性：**处理器每次只执行一个进程，然后在多个进程之间快速切换，依次来达到多个进程在单个处理器上同时执行的效果；

**进程的并行性：**同一时刻，处理器同时执行多个进程；

# block的本质是什么，有几种类型，有什么区别

https://juejin.cn/post/6844903893176958983

block的本质是一个结构体，有3种类型。可以通过调用class方法或者isa指针查看具体类型，最终都是继承自NSBlock类型

 \_\_NSGlobalBlock\_\_ （ \_NSConcreteGlobalBlock ）

 \_\_NSStackBlock\_\_ （ \_NSConcreteStackBlock ）

 \_\_NSMallocBlock\_\_ （ \_NSConcreteMallocBlock ）

其中三种不同的类型和环境对应如下

1. block类型环境\_\_NSGlobalBlock\_\_

2. 没有访问auto变量\_\_NSStackBlock\_\_

3. 访问了auto变量\_\_NSMallocBlock\_\_\_\_NSStackBlock\_\_调用了copy

其在内存中的分配如下对应

图示, 示意图

描述已自动生成