当一个进程想为其它进程提供服务时，就需要通过进程间通信的方 式来提供服务

APP1所在进程有一个对象object1，其中有一个方法method1。 APP2所在的另外一个进程，想使用object1的method1方法。binder可能帮助我们在APP2所在进程拿到一个object1对象的引 用，使我们能够像调用本地对象一样，通过object1.method1()直接调用。利用binder，我们可以突破进程的限制，将对象传给其它进程， 让其它进程方便调用对象的方法。

一个Binder系统由四部分组成：Binder客户端、Binder服务端、Binder驱动、服务登记查询模块

**Binder客户端**：想要使用服务的进程

**Binder服务端**：实际提供服务的进程

**Binder驱动**：我们在客户端先通过Binder拿到一个服务端进程中的一个对象的引用，通过这个引用，直接 调用对象的方法获取结果。在这个引用对象执行方法时，它是先将方法调用的请求传给binder驱动；然后binder驱动再将请求传给服务端进程；服务端 进程收到请求后，调用服务端“真正”的对象来执行所调用的方法；得出结果后，将结果发给binder驱动；binder驱动再将结果发给我们的客户端；最 终，我们在客户端进程的调用就有了返回值。Binder驱动，相当于一个中转者的角色。通过这个中转者的帮忙，我们就可以调用其它进程中的对象。

**服务登记查询模块**：我们调用其它进程里面的对象时，首先要获取这个对象。这个对象其实代表了另外一个进程能给我 们提供什么样的服务（再直接一点，就是：对象中有哪些方法可以让客户端进程调用）。首先服务端进程要在某个地方注册登记一下，告诉系统我有个对象可以公开 给其它进程来提供服务。当客户端进程需要这个服务时，就去这个登记的地方通过查询来找到这个对象。