

打破Java定律:无需创建对象计算机等级考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/645/2021_2022__E6_89_93_E7_A0_B4Java_c97_645397.htm 我们知道，在Java基本教程中有一个定律告诉我们：所有的对象都必须创建；或者说：使用对象之前必须创建，但是现在我们可以不必一定遵循这个定律了，我们可以从Ioc容器中直接获得一个对象然后直接使用，无需事先创建它们。这种变革，就如同我们无需考虑对象销毁一样；因为Java的垃圾回收机制帮助我们实现了对象销毁；现在又无需考虑对象创建，对象的创建和销毁都无需考虑了，这给编程带来的影响是巨大的。我们从一个简单例子开始，有一个普通类B代码如下：

```
public class B implements BI{
    AInfterface a.
    public B(AInfterface a){ this.a = a }
    public void invoke(){ /** ...*/ }
```

有两种使用B的方式：普通无Ioc容器的调用方式：

```
BI b = new B(new A()).//需要在生成B实例之前 生成A等实例
b.invoke().
```

使用Ioc容器(Jdon框架)的革命调用方式：

```
BI b = (BI) WebAppUtil.getService( " b " ).
b.invoke().
```

上面两种方式重要区别：前者需要照顾B类中A类的实例化，如果B类中调用不只A类一个，还有更多其他类如C/D/E等类，这样，你在使用B类时，还需要研究其他类的创建，如果C/D/E这些类不是你自己编写，你还需要翻阅它们的API说明，研究它们应该如何创建？是使用New 还是工厂模式 还是单态调用？这时，你会感叹：哇，有没有搞错？我只不过是为了使用B类中一个小小的方法，就花去我这么多时间和精力？当我们使用第二种方式时，就无需花很多精力和时间考虑A/C/D/E等类的创建。使用Ioc容器，你再也不必做这些僵化愚蠢的工作了

，我们只需从ioc容器中抓取一个类然后直接使用它们。当然，在使用之前，我们需要做一个简单的配置，把你将来需要使用的类全部告诉ioc容器，例如Jdon框架的ioc容器配置jdonframework.xml如下: 100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com