
Java 编程面试题锦囊

@Author:jilongliang @Date:2013-3

@博客@Email:jilongliang@sina.com

一、常见题目

1.Java 关键字列表 (依字母排序 共 51 组):

abstract, assert, boolean, break, byte, case, catch, char, class, const, continue, default, do, double, else, enum, extends, final, finally, float, for, if, implements, import, instanceof, int, interface, long, native, new, package, private, protected, public, return, short, static, strictfp, super, switch, synchronized, this, throw, throws, transient, try, void, volatile, while

2Java 保留字列表 (依字母排序 共 14 组): Java 保留字是指现有 Java 版本尚未使用 但以后版本可能会作为关键字使用.

byValue, cast, false, future, generic, inner, operator, outer, rest, true, var, goto, const, null

3.请你描述封装! (PS 可以不用答)

答;抽象: 将属性私有化 提供共有访问私有属性,抽象包括两个方面, 一是过程抽象, 二是数据抽象。

PS: 现实的步骤:

- 修改属性的可见性来限制对属性的访问
- 为每个属性创建一对赋值方法和取值方法用于对这些属性的访问
- 在赋值和取值方法中加入对属性的存取限制

4 请你描述继承:

继承是一种联结类的层次模型,并且允许和鼓励类的重用,它提供了一种明确表述共性的方法.对象的一个新类可以从现有的类中派生,这个过程称为类继承.新类继承了原始类的特性,新类称为原始类的派生类(子类),而原始类称为新类的基类(父类).派生类可以从它的基类那里继承方法和实例变量,并且类可以修改或增加新的方法使之更适合特殊的需要.

5 请你描述封装:

封装是把过程和数据包围起来,对数据的访问只能通过已定义的界面.面向对象计算始于这个基本概念,即现实世界可以被描绘成一系列完全自治、封装的对象,这些对象通过一个受保护的接口访问其他对象.

6. 请你描述多态性:

多态性是指允许不同类的对象对同一消息作出响应.多态性包括参数化多态性和包含多态性.多态性语言具有灵活、抽象、行为共享、代码共享的优势,很好的解决了应用程序函数同名问题.

多态-概念

有了前面的基础,我们再看多态就相对容易了,请大家看我在 Demo3_12.java 上改动一下代码.从而引出多态的概念。

所谓多态,就是指一个引用(类型)在不同情况下的多种状态.你也可以这样理解:多态是指通过指向父类的指针,来调用在不同子类中实现的方法。

7 请你描述抽象:

抽象就是忽略一个主题中与当前目标无关的那些方面,以便更充分地注意与当前目标有关的方面.抽象并不打算了解全部问题,而只是选择其中的一部分,暂时不用部分细节.抽象包括两个方面,一是过程抽象,二是数据抽象.

8、String 是最基本的数据类型吗?

基本数据类型包括 byte、int、char、long、float、double、boolean 和 short.

java.lang.String 类是 **final** 类型的,因此**不可以继承这个类、不能修改这个类**.为了提高效率节省空间,我们应该用 StringBuffer 类

9、int 和 Integer 有什么区别

Java 提供两种不同的类型: 引用类型和原始类型 (或内置类型) .

Int 是 java 的原始数据类型,Integer 是 java 为 int 提供的封装类.Java 为每个原始类型提供了封装类.

原始类型	封装类
boolean	Boolean
char	Character
byte	Byte
short	Short
int	Integer
long	Long
float	Float

10、String 和 StringBuffer 的区别

PS: JAVA 平台提供了两个类: String 和 StringBuffer,它们可以储存和操作字符串,即包含多个字符的字符数据.

答:String 类提供了**数值不可改变**的字符串. StringBuffer 类提供的字符串**进行修改**.当**你知道**字符数据**要改变的时候**你就可以使用 StringBuffer.典型地,你可以使用 StringBuffers 来动态构造字符数据

11、运行时异常与一般异常有何异同?

答;异常表示程序运行过程中可能出现的非正常状态,运行时异常表示虚拟机的通常操作中可能遇到的异常,是一种**常见运行错误**.java 编译器要求方法必须声明抛出可能发生的非运行时异常,但是并不要求必须声明抛出**未被捕获的运行时异常**.

12、Collection 和 Collections 的区别

答:Collection 是集合类的**上级接口**,继承他的接口主要有 Set 和 List.

Collections 是针对集合类的一个**帮助类**,他提供一系列静态方法实现对各种集合的搜索、排序、线程安全化等操作

13、说出 Servlet 的生命周期,并说出 Servlet 和 CGI 的区别.

Servlet 被服务器实例化后,容器运行其 init 方法,请求到达时运行其 service 方法,service 方法自动派遣**运行与请求对应的 doXXX 方法** (doGet,doPost) 等,当服务器决定**将实例销毁**的时候调用其 destroy 方法.

与 cgi 的区别在于 servlet 处于服务器进程中,它通过**多线程**方式运行其 service 方法,**一个实例可以服务于多个请求,并且其实例一般不会销毁**,而 CGI 对每个请求都产生新的进程,服务完成后就销毁,所以效率上低于 servlet.

14、说出 ArrayList,Vector,LinkedList 的存储性能和特性

ArrayList 和 Vector 都是使用数组方式存储数据,此数组元素数大于实际存储的数据以便增加和插入元素,它们都允许直接按序号索引元素,但是插入元素要涉及数组元素移动等内存操作,所以索引数据快而插入数据慢,Vector 由于使用了 synchronized 方法（线程安全）,通常性能上较 ArrayList 差,而 LinkedList 使用双向链表实现存储,按序号索引数据需要进行前向或后向遍历,但是插入数据时只需要记录本项的前后项即可,所以插入速度较快.

15、EJB 是基于哪些技术实现的？并说出 SessionBean 和 EntityBean 的区别,StatefulBean 和 StatelessBean 的区别.

EJB 包括 Session Bean、Entity Bean、Message Driven Bean,基于 JNDI、RMI、JAT 等技术实现.

SessionBean 在 J2EE 应用程序中被用来完成一些服务器端的业务操作,例如访问数据库、调用其他 EJB 组件.EntityBean 被用来代表应用系统中用到的数据.

对于客户机,SessionBean 是一种非持久性对象,它实现某些在服务器上运行的业务逻辑.

对于客户机,EntityBean 是一种持久性对象,它代表一个存储在持久性存储器中的实体的对象视图,或是一个由现有企业应用程序实现的实体.

Session Bean 还可以再细分为 Stateful Session Bean 与 Stateless Session Bean ,这两种的 Session Bean 都可以将系统逻辑放在 method 之中执行,不同的是 Stateful Session Bean 可以记录呼叫者的状态,因此通常来说,一个使用者会有一个相对应的 Stateful Session Bean 的实体.Stateless Session Bean 虽然也是逻辑组件,但是他却不负责记录使用者状态,也就是说当使用者呼叫 Stateless Session Bean 的时候,EJB Container 并不会找寻特定的 Stateless Session Bean 的实体来执行这个 method.换言之,很可能数个使用者在执行某个 Stateless Session Bean 的 methods 时,会是同一个 Bean 的 Instance 在执行.从内存方面来看, Stateful Session Bean 与 Stateless Session Bean 比较, Stateful Session Bean 会消耗 J2EE Server 较多的内存,然而 Stateful Session Bean 的优势却在于他可以维持使用者的状态.

16、&和&&的区别.

&是位运算符,表示按位与运算,&&是逻辑运算符,表示逻辑与（and）.

17、HashMap 和 Hashtable 的区别.(答以下分别三点即可)

HashMap

- 1.Jdk1.2 之后推出,是新类
- 2.采用异步处理方式,性能较高,属于非线程安全
- 3.允许设置值 null

Hashtable

- 1.Jdk1.0 之后推出,是旧类
- 2.采用同步处理方式,性能较低,属于线程安全
- 3.不允许设置值 null,否则会报 NullPointerException.

HashMap 是 Hashtable 的轻量级实现（非线程安全的实现）,他们都完成了 Map 接口,主要区别在于 HashMap 允许空（null）键值（key）,由于非线程安全,效率上可能高于 Hashtable.

HashMap 允许将 null 作为一个 entry 的 key 或者 value,而 Hashtable 不允许.

HashMap 把 Hashtable 的 contains 方法去掉了,改成 containsvalue 和 containsKey.因为 contains 方法容易让人引起误解.

Hashtable 继承自 Dictionary 类,而 HashMap 是 Java1.2 引进的 Map interface 的一个实现.

最大的不同是,Hashtable 的方法是 Synchronize 的,而 HashMap 不是,在多个线程访问 Hashtable 时,不需要自己为它的方法实现同步,而 HashMap 就必须为之提供外同步.
Hashtable 和 HashMap 采用的 hash/rehash 算法都大概一样,所以性能不会有很大的差异.

18、final, finally, finalize 的区别.

final 用于声明属性,方法和类,分别表示属性不可变,方法不可覆盖,类不可继承.
finally 是异常处理语句结构的一部分,表示总是执行.
finalize 是垃圾回收前收尾工作,是 Object 类定义的

19、sleep() 和 wait() 有什么区别?

sleep 是线程类 (Thread) 的方法,导致此线程暂停执行指定时间,给执行机会给其他线程,但是监控状态依然保持,到时后会自动恢复.调用 sleep 不会释放对象锁.

wait 是 Object 类的方法,对此对象调用 wait 方法导致本线程放弃对象锁,进入等待此对象的等待锁定池,只有针对此对象发出 notify 方法 (或 notifyAll) 后本线程才进入对象锁定池准备获得对象锁进入运行状态.

20、Overload 和 Override 的区别.Overloaded 的方法是否可以改变返回值的类型?

OverLoad: 重载

- 1.在一个类定义若干个方法
- 2.方法名称都相同,但是参数的类型或个数不相同

Override: 重写

- 1.重写是父类与子类之间多态性的一种表现
- 2.子类定义了与父类方法,参数类型或个数相同的

重载的时候不是依靠返回值类型区分的,而靠参数

21、error 和 exception 有什么区别?

error 表示是由 JVM 进行处理的,是有 JVM 出错的。比如说内存溢出.不可能指望程序能处理这样的情况.

exception 表示一种设计或实现问题,可以用程序进行处理的,使用 try catch 处理

21、同步和异步有何异同,在什么情况下分别使用他们? 举例说明.

如果数据将在线程间共享.例如正在写的以后可能被另一个线程读到,或者正在读的数据可能已经被另一个线程写过了,那么这些数据就是共享数据,必须进行同步存取.

当应用程序在对象上调用了一个需要花费很长时间来执行的方法,并且不希望让程序等待方法的返回时,就应该使用异步编程,在很多情况下采用异步途径往往更有效率.

23、abstract class 和 interface 有什么区别?

- 1.由抽象方法和常量,变量,全局常量,构造方法,普通方法组成
- 2.使用 abstract 声明
- 3.子类要通过 extends 继承抽象类,子类如果不是抽象类,则不想覆写抽象类中的全部抽象方法

4.存在单继承局限,一个子类只能继承一个抽象类

5.抽象类可以实现若干个接口

- 1.全部由抽象方法和全局常量组成

- 2.使用 interface 声明
- 3.子类通过 implements 实现接口,子类如果不是抽象类,则不想覆写抽象类中的全部抽象方法
- 4.不存在单继承局限,一个类可以同时实现多个接口
- 5.接口不能继承一个抽象类,但是允许继承多个接口

24、heap 和 stack 有什么区别.

栈是一种线形集合,其添加和删除元素的操作应在同一段完成.栈按照后进先出的方式进行处理.

堆是栈的一个组成元素

25、forward 和 redirect 的区别

forward 是服务器请求资源,服务器直接访问目标地址的 URL,把那个 URL 的响应内容读取过来,然后把这些内容再发给浏览器,浏览器根本不知道服务器发送的内容是从哪儿来的,所以它的地址栏中还是原来的地址.

redirect 就是服务端根据逻辑,发送一个状态码,告诉浏览器重新去请求那个地址,一般来说浏览器会用刚才请求的所有参数重新请求,所以 session,request 参数都可以获取.

26、EJB 与 JAVA BEAN 的区别?

Java Bean 是可复用的组件,对 Java Bean 并没有严格的规范,理论上讲,任何一个 Java 类都可以是一个 Bean.但通常情况下,由于 Java Bean 是被容器所创建(如 Tomcat)的,所以 Java Bean 应具有一个无参的构造器,另外,通常 Java Bean 还要实现 Serializable 接口用于实现 Bean 的持久性.Java Bean 实际上相当于微软 COM 模型中的本地进程内 COM 组件,它是不能被跨进程访问的.Enterprise Java Bean 相当于 DCOM,即分布式组件.它是基于 Java 的远程方法调用(RMI)技术的,所以 EJB 可以被远程访问(跨进程、跨计算机).但 EJB 必须被布署在诸如 Webspere、WebLogic 这样的容器中,EJB 客户从不直接访问真正的 EJB 组件,而是通过其容器访问.EJB 容器是 EJB 组件的代理,EJB 组件由容器所创建和管理.客户通过容器来访问真正的 EJB 组件.

27、Static Nested Class 和 Inner Class 的不同.

使用 static 声明的内部类就是外部类,可以通过外部类,内部类直接访问
普通的内部类是不能直接被外部所访问的,需要外部类实例再找到内部类实例

28、JSP 中动态 INCLUDE 与静态 INCLUDE 的区别?

动态 INCLUDE 用 jsp:include 动作实现 <jsp:include page="included.jsp" flush="true" />它总是会检查所含文件中的变化,适合用于包含动态页面,并且可以带参数.

静态 INCLUDE 用 include 伪码实现,定不会检查所含文件的变化,适用于包含静态页面<%@include file="included.htm" %>

29、什么时候用 assert.

Jdk1.4 之后添加的新类,assert 表示断言,程序执行到某个地方之后值肯定是预计好的啦!一般开发中很少用到 assert,要想用断言,则必须使用 ea 参数

30、GC 是什么? 为什么要有 GC?

GC: 垃圾收集

使用 GC 可以进行空间的释放操作

31、`short s1 = 1; s1 = s1 + 1;`有什么错? `short s1 = 1; s1 += 1;`有什么错?

`short s1 = 1; s1 = s1 + 1;` (`s1+1` 运算结果是 `int` 型,需要强制转换类型)
`short s1 = 1; s1 += 1;` (可以正确编译)

32、`Math.round(11.5)`等於多少? `Math.round(-11.5)`等於多少?

`Math.round(11.5)`==12 `Math.round(-11.5)`== -11
`round` 方法返回与参数最接近的长整数,参数加 1/2 后求其 `floor`.

33、`String s = new String("xyz");`创建了几个 `String Object`?

答: 两个, 一个是匿名, 一个是 `new` 实例化创建产生

34、EJB 包括 (`SessionBean`,`EntityBean`) 说出他们的生命周期,及如何管理事务的?

SessionBean: `Stateless Session Bean` 的生命周期是由容器决定的,当客户机发出请求要建立一个 `Bean` 的实例时,EJB 容器不一定要创建一个新的 `Bean` 的实例供客户机调用,而是随便找一个现有的实例提供给客户机.当客户机第一次调用一个 `Stateful Session Bean` 时,容器必须立即在服务器中创建一个新的 `Bean` 实例,并关联到客户机上,以后此客户机调用 `Stateful Session Bean` 的方法时容器会把调用分派到与此客户机相关联的 `Bean` 实例.

EntityBean: `Entity Beans` 能存活相对较长的时间,并且状态是持续的.只要数据库中的数据存在,`Entity beans` 就一直存活.而不是按照应用程序或者服务进程来说的.即使 EJB 容器崩溃了,`Entity beans` 也是存活的.`Entity Beans` 生命周期能够被容器或者 `Beans` 自己管理.

EJB 通过以下技术管理实务: 对象管理组织 (OMG) 的对象实务服务 (OTS),Sun Microsystems 的 `Transaction Service (JTS)`、`Java Transaction API (JTA)` ,开发组 (X/Open) 的 `XA` 接口.

35、应用服务器有那些?

`BEA WebLogic Server`,`IBM WebSphere Application Server`,`Oracle9i Application Server`,`jBoss`,`Tomcat`

36、给我一个你最常见到的 `runtime exception`.

`ArithmeticException`, `ArrayStoreException`, `BufferOverflowException`, `BufferUnderflowException`,
`CannotRedoException`, `CannotUndoException`, `ClassCastException`, `CMMException`,
`ConcurrentModificationException`, `DOMException`, `EmptyStackException`, `IllegalArgumentException`,
`IllegalMonitorStateException`, `IllegalPathStateException`, `IllegalStateException`, `ImageOpException`,
`IndexOutOfBoundsException`, `MissingResourceException`, `NegativeArraySizeException`,
`NoSuchElementException`, `NullPointerException`, `ProfileDataException`, `ProviderException`,
`RasterFormatException`, `SecurityException`, `SystemException`, `UndeclaredThrowableException`,
`UnmodifiableSetException`, `UnsupportedOperationException`

37、接口是否可继承接口? 抽象类是否可实现(`implements`)接口? 抽象类是否可继承实体类 (`concrete class`)?

接口可以继承接口.抽象类可以实现(`implements`)接口,抽象类是否可继承实体类,但前提是实体类必须有明确的构造函数.

38、`List`, `Set`, `Map` 是否继承自 `Collection` 接口?

`List`,`Set` 是,`Map` 不是

39、说出数据连接池的工作机制是什么？

J2EE 服务器启动时会建立一定数量的池连接,并一直维持不少于此数目的池连接.客户端程序需要连接时,池驱动程序会返回一个未使用的池连接并将其标记为忙.如果当前没有空闲连接,池驱动程序就新建一定数量的连接,新建连接的数量有配置参数决定.当使用的池连接调用完成后,池驱动程序将此连接标记为空闲,其他调用就可以使用这个连接.

40、abstract 的 method 是否可同时是 static,是否可同时是 native,是否可同时是 synchronized?

都不能

41、数组有没有 length()这个方法? String 有没有 length()这个方法?

数组没有 length()这个方法,有 length 的属性.String 有 length()这个方法.

42、Set 里的元素是不能重复的,那么用什么方法来区分重复与否呢? 是用==还是 equals()? 它们有何区别?

Set 里的元素是不能重复的,那么用 iterator()方法来区分重复与否.equals()是判读两个 Set 是否相等.

equals()和==方法决定引用值是否指向同一对象 equals()在类中被覆盖,为的是当两个分离的对象的内容和类型相配的话,返回真值.

“==”比较两个变量本身值,即两个对象在内存中的首地址 ==是用来对比地址是否相等
“equals()”比较字符串中所包含的内容是否相同

43、构造器 Constructor 是否可被 override?

构造器 Constructor 不能被继承,因此不能重写 Overriding,但可以被重载 Overloading.

44、是否可以继承 String 类?

String 类是 final 类故不可以继承.

45、switch 是否能作用在 byte 上,是否能作用在 long 上,是否能作用在 String 上?

switch (expr1) 中,expr1 是一个整数表达式.因此传递给 switch 和 case 语句的参数应该是 int、short、char 或者 byte.long,string 都不能作用于 switch.

46、try {}里有一个 return 语句,那么紧跟在这个 try 后的 finally {}里的 code 会不会被执行,什么时候被执行,在 return 前还是后?

会执行,在 return 前执行.

47、编程题:用最有效率的方法算出 2 乘以 8 等於几?

2 << 3

48、两个对象值相同(x.equals(y) == true),但却可有不同的 hash code,这句话对不对?

不对,有相同的 hash code.

49、当一个对象被当作参数传递到一个方法后,此方法可改变这个对象的属性,并可返回变化后的结果,那么这里到底是值传递还是引用传递?

是值传递.Java 编程语言只有值传递参数.当一个对象实例作为一个参数被传递到方法中时,参数的值就是对该对象的引用.对象的内容可以在被调用的方法中改变,但对象的引用是永远不会改变的.

50、当一个线程进入一个对象的一个 `synchronized` 方法后,其它线程是否可进入此对象的其它方法?

不能,一个对象的一个 `synchronized` 方法只能由一个线程访问.

51、编程题: 写一个 Singleton 出来.

Singleton 模式主要作用是保证在 Java 应用程序中,一个类 Class 只有一个实例存在.

一般 Singleton 模式通常有几种形式:

第一种形式: 定义一个类,它的构造函数为 `private` 的,它有一个 `static` 的 `private` 的该类变量,在类初始化时实例化,通过一个 `public` 的 `getInstance` 方法获取对它的引用,继而调用其中的方法.

其他形式:

定义一个类,它的构造函数为 `private` 的,所有方法为 `static` 的.

一般认为第一种形式要更加安全些

```
public class Singleton {
    /** 一个构造方法*/
    public Singleton() {
    }
    /** 方法一*/
    public static Singleton sing = new Singleton();
    synchronized public static Singleton getInst() {
        return sing;
    }
    /** 方法二使用时生成实例,提高了效率/
    public static Singleton singleton = null;
    public static Singleton getInstance() {
        if (singleton == null) {
            Synchronized(Singleton.class) {
                singleton = new Singleton();
            }
        }
        return singleton;
    }
}
```

52、Java 的接口和 C++的虚类的相同和不同处.

由于 Java 不支持多继承,而有可能某个类或对象要使用分别在几个类或对象里面的方法或属性,现有的单继承机制就不能满足要求.与继承相比,接口有更高的灵活性,因为接口中没有任何实现

代码.当一个类实现了接口以后,该类要实现接口里面所有的方法和属性,并且接口里面的属性在默认状态下面都是 `public static`,所有方法默认情况下是 `public`.一个类可以实现多个接口.

53、Java 中的异常处理机制的简单原理和应用.

当 JAVA 程序违反了 JAVA 的语义规则时,JAVA 虚拟机就会将发生的错误表示为一个异常.违反语义规则包括 2 种情况.一种是 JAVA 类库内置的语义检查.例如数组下标越界,会引发 `IndexOutOfBoundsException`;访问 `null` 的对象时会引发 `NullPointerException`.另一种情况就是 JAVA 允许程序员扩展这种语义检查,程序员可以创建自己的异常,并自由选择何时用 `throw` 关键字引发异常.所有的异常都是 `java.lang.Throwable` 的子类.

54、垃圾回收的优点和原理.并考虑 2 种回收机制.

将无用的对象空间进行释放,
两种回收: 自动回收, 手工调用 `System.gc()`; 方法实际上是调用 `System.gc()`;就相当于调用了 `Runtime.getRuntime().gc()`;方法

55、请说出你所知道的线程同步的方法.

`wait()`:使一个线程处于等待状态,并且释放所持有的对象的 `lock`.
`sleep()`:使一个正在运行的线程处于睡眠状态,是一个静态方法,调用此方法要捕捉 `InterruptedException` 异常.
`notify()`:唤醒一个处于等待状态的线程,注意的是在调用此方法的时候,并不能确切的唤醒某一个等待状态的线程,而是由 JVM 确定唤醒哪个线程,而且不是按优先级.
`Allnotify()`:唤醒所有处入等待状态的线程,注意并不是给所有唤醒线程一个对象的锁,而是让它们竞争.

56、你所知道的集合类都有哪些? 主要方法?

最常用的集合类是 `List` 和 `Map`.`List` 的具体实现包括 `ArrayList` 和 `Vector`,它们是可变大小的列表,比较适合构建、存储和操作任何类型对象的元素列表.`List` 适用于按数值索引访问元素的情形.
`Map` 提供了一个更通用的元素存储方法.`Map` 集合类用于存储元素对(称作"键"和"值"),其中每个键映射到一个值.

57、描述一下 JVM 加载 class 文件的原理机制?

JVM 中类的装载是由 `ClassLoader` 和它的子类来实现的,Java `ClassLoader` 是一个重要的 Java 运行时系统组件.它负责在运行时查找和装入类文件的类.

58、char 型变量中能不能存贮一个中文汉字?为什么?

能,因为 java 中以 `unicode` 编码,一个 `char` 占 16 个字节,所以放一个中文是没问题的

59、多线程有几种实现方法,都是什么?同步有几种实现方法,都是什么?

多线程有两种实现方法,分别是继承 `Thread` 类与实现 `Runnable` 接口
同步的实现方面有两种,分别是 `synchronized`,`wait` 与 `notify`

60、JSP 的内置对象及方法.

`request` 表示 `HttpServletRequest` 对象.它包含了有关浏览器请求的信息,并且提供了几个用于获取 `cookie`, `header`, 和 `session` 数据的有用的方法.

`response` 表示 `HttpServletResponse` 对象,并提供了几个用于设置送回浏览器的响应的方法(如 cookies,头信息等)

`out` 对象是 `javax.jsp.JspWriter` 的一个实例,并提供了几个方法使你能用于向浏览器回送输出结果.

`pageContext` 表示一个 `javax.servlet.jsp.PageContext` 对象.它是用于方便存取各种范围的名字空间、servlet 相关的对象的 API,并且包装了通用的 servlet 相关功能的方法.

`session` 表示一个请求的 `javax.servlet.http.HttpSession` 对象.Session 可以存贮用户的状态信息

`applicaton` 表示一个 `javax.servle.ServletContext` 对象.这有助于查找有关 servlet 引擎和 servlet 环境的信息

`config` 表示一个 `javax.servlet.ServletConfig` 对象.该对象用于存取 servlet 实例的初始化参数.

`page` 表示从该页面产生的一个 servlet 实例

61、线程的基本概念、线程的基本状态以及状态之间的关系

线程指在程序执行过程中,能够执行程序代码的一个执行单位,每个程序至少都有一个线程,也就是程序本身.

Java 中的线程有四种状态分别是: **运行、就绪、挂起、结束**.

62、JSP 的常用指令

```
<%@page language="java" contentType="text/html; charset=gb2312" session="true" buffer="64kb"
autoFlush="true" isThreadSafe="true" info="text" errorPage="error.jsp" isErrorPage="true"
isELIgnored="true" pageEncoding="gb2312" import="java.sql.*"%>
isErrorPage(是否能使用 Exception 对象),isELIgnored(是否忽略表达式)
<%@include file="filename"%>
<%@taglib prefix="c" uri="http://....."%>
```

63、什么情况下调用 doGet()和 doPost()?

Jsp 页面中的 form 标签里的 method 属性为 **get** 时调用 `doGet()`,为 **post** 时调用 `doPost()`.

64、servlet 的生命周期

web 容器加载 servlet,生命周期开始.通过调用 servlet 的 `init()`方法进行 servlet 的初始化.通过调用 `service()`方法实现,根据请求的不同调用不同的 `do***()`方法.结束服务,web 容器调用 servlet 的 `destroy()`方法.

65、如何现实 servlet 的单线程模式

```
<%@ page isThreadSafe="false"%>
```

66、页面间对象传递的方法

`request,session,application,cookie` 等

67、JSP 和 Servlet 有哪些相同点和不同点,他们之间的联系是什么?

JSP 是 Servlet 技术的扩展,本质上是 Servlet 的简易方式,更强调应用的外表达.JSP 编译后是"类 servlet".Servlet 和 JSP 最主要的不同点在于,
Servlet 的应用逻辑是在 Java 文件中,并且完全从表示层中的 HTML 里分离开来.而 JSP 的情况是

Java 和 HTML 可以组合成一个扩展名为.jsp 的文件。

JSP 侧重于视图,Servlet 主要用于控制逻辑。

68、JSP 四种会话跟踪技术

page 当前页面有效

request 浏览器对服务器的一次请求有效,服务器返回请求后失效

session 在服务器规定会话最长时间范围内有效,对浏览器窗口和其子窗口。

application 就是对整个正在运行的项目有效了。

会话作用域 ServletsJSP 页面描述

page 否是代表与一个页面相关的对象和属性.一个页面由一个编译好的 Java servlet 类(可以带有任何的 include 指令,但是没有 include 动作)表示.这既包括 servlet 又包括被编译成 servlet 的 JSP 页面

request 是代表与 Web 客户机发出的一个请求相关的对象和属性.一个请求可能跨越多个页面,涉及多个 Web 组件(由于 forward 指令和 include 动作的关系)

session 是代表与用于某个 Web 客户机的一个用户体验相关的对象和属性.一个 Web 会话可以也经常跨越多个客户机请求

application 是代表与整个 Web 应用程序相关的对象和属性.这实质上是跨越整个 Web 应用程序,包括多个页面、请求和会话的一个全局作用域

69、Request 对象的主要方法:

setAttribute(String name,Object): 设置名字为 name 的 request 的参数值

getAttribute(String name): 返回由 name 指定的属性值

getAttributeNames(): 返回 request 对象所有属性的名字集合,结果是一个枚举的实例

getCookies(): 返回客户端的所有 Cookie 对象,结果是一个 Cookie 数组

getCharacterEncoding(): 返回请求中的字符编码方式

getContentLength(): 返回请求的 Body 的长度

getHeader(String name): 获得 HTTP 协议定义的文件头信息

getHeaders(String name): 返回指定名字的 request Header 的所有值,结果是一个枚举的实例

getHeaderNames(): 返回所以 request Header 的名字,结果是一个枚举的实例

getInputStream(): 返回请求的输入流,用于获得请求中的数据

getMethod(): 获得客户端向服务器端传送数据的方法

getParameter(String name): 获得客户端传送给服务器端的有 name 指定的参数值

getParameterNames(): 获得客户端传送给服务器端的所有参数的名字,结果是一个枚举的实例

getParameterValues(String name): 获得有 name 指定的参数的所有值

getProtocol(): 获取客户端向服务器端传送数据所依据的协议名称

getQueryString(): 获得查询字符串

getRequestURI(): 获取发出请求字符串的客户端地址

getRemoteAddr(): 获取客户端的 IP 地址

getRemoteHost(): 获取客户端的名字

getSession([Boolean create]): 返回和请求相关 Session

getServerName(): 获取服务器的名字

getServletPath(): 获取客户端所请求的脚本文件的路径

getServerPort(): 获取服务器的端口号
removeAttribute(String name): 删除请求中的一个属性

70、J2EE 是技术还是平台还是框架?

J2EE 本身是一个标准,一个为企业分布式应用的开发提供的标准平台.

J2EE 也是一个框架,包括 JDBC、JNDI、RMI、JMS、EJB、JTA 等技术.

71、我们在 web 应用开发过程中经常遇到输出某种编码的字符,如 iso8859-1 等,如何输出一个某种编码的字符串?

```
public String translate (String str) {  
    String tempStr = "";  
    try {  
        tempStr = new String(str.getBytes("ISO-8859-1"), "GBK");  
        tempStr = tempStr.trim();  
    }  
    catch (Exception e) {  
        System.err.println(e.getMessage());  
    }  
    return tempStr;  
}
```

72、简述逻辑操作(&,&|,&^)与条件操作(&&,&||)的区别.

区别主要答两点:

- a. 条件操作只能操作布尔型的,而逻辑操作不仅可以操作布尔型,而且可以操作数值型
- b. 逻辑操作不会产生短路

73、XML 文档定义有几种形式? 它们之间有何本质区别? 解析 XML 文档有哪几种方式?

a: 两种形式 dtd schema,

b: 本质区别:schema 本身是 xml 的,可以被 XML 解析器解析(这也是从 DTD 上发展 schema 的根本目的),

c: 有 DOM,SAX,STAX 等

DOM: 处理大型文件时其性能下降的非常厉害.这个问题是由 DOM 的树结构所造成的,这种结构占用的内存较多,而且 DOM 必须在解析文件之前把整个文档装入内存,适合对 XML 的随机访问

SAX: 不现于 DOM,SAX 是事件驱动型的 XML 解析方式.它顺序读取 XML 文件,不需要一次全部装载整个文件.当遇到像文件开头,文档结束,或者标签开头与标签结束时,它会触发一个事件,用户通过在其回调事件中写入处理代码来处理 XML 文件,适合对 XML 的顺序访问

STAX: Streaming API for XML (StAX)

74、简述 synchronized 和 java.util.concurrent.locks.Lock 的异同?

主要相同点: Lock 能完成 synchronized 所实现的所有功能

主要不同点: Lock 有比 synchronized 更精确的线程语义和更好的性能.synchronized 会自动释放锁,而 Lock 一定要求程序员手工释放,并且必须在 finally 从句中释放.

75、EJB 的角色和三个对象

一个完整的基于 EJB 的分布式计算结构由六个角色组成,这六个角色可以由不同的开发商提供,每个角色所作的工作必须遵循 Sun 公司提供的 EJB 规范,以保证彼此之间的兼容性.

这六个角色分别是 EJB 组件开发者 (Enterprise Bean Provider) 、应用组合者 (Application Assembler)、部署者 (Deployer)、EJB 服务器提供者 (EJB Server Provider)、EJB 容器提供者 (EJB Container Provider)、系统管理员 (System Administrator)
三个对象是 Remote (Local) 接口、Home (LocalHome) 接口,Bean 类

76、EJB 容器提供的服务

主要提供声明周期管理、代码产生、持续性管理、安全、事务管理、锁和并发管理等服务.

77、EJB 规范规定 EJB 中禁止的操作有哪些?

- 1.不能操作线程和线程 API(线程 API 指非线程对象的方法如 notify,wait 等)
- 2.不能操作 awt,
- 3.不能实现服务器功能,
- 4.不能对静态属生存取,
- 5.不能使用 IO 操作直接存取文件系统,
- 6.不能加载本地库.
- 7.不能将 this 作为变量和返回,
- 8.不能循环调用.

78、remote 接口和 home 接口主要作用

remote 接口定义了业务方法,用于 EJB 客户端调用业务方法.
home 接口是 EJB 工厂用于创建和移除查找 EJB 实例

79、bean 实例的生命周期

对于 Stateless Session Bean、Entity Bean、Message Driven Bean 一般存在缓冲池管理,而对于 Entity Bean 和 Statefull Session Bean 存在 Cache 管理,通常包含创建实例,设置上下文、创建 EJB Object (create)、业务方法调用、remove 等过程,对于存在缓冲池管理的 Bean,在 create 之后实例并不从内存清除,而是采用缓冲池调度机制不断重用实例,而对于存在 Cache 管理的 Bean 则通过激活和去激活机制保持 Bean 的状态并限制内存中实例数量.

80、EJB 的激活机制

以 Stateful Session Bean 为例: 其 Cache 大小决定了内存中可以同时存在的 Bean 实例的数量,根据 MRU 或 NRU 算法,实例在激活和去激活状态之间迁移,激活机制是当客户端调用某个 EJB 实例业务方法时,如果对应 EJB Object 发现自己没有绑定对应的 Bean 实例则从其去激活 Bean 存储中 (通过序列化机制存储实例) 回复 (激活) 此实例.状态变迁前会调用对应的 ejbActive 和 ejbPassivate 方法.

81、EJB 的几种类型

会话 (Session) Bean ,实体 (Entity) Bean 消息驱动的 (Message Driven) Bean

会话 Bean 又可分为有状态 (Stateful) 和无状态 (Stateless) 两种
实体 Bean 可分为 Bean 管理的持续性 (BMP) 和容器管理的持续性 (CMP) 两种

82、客户端调用 EJB 对象的几个基本步骤

设置 JNDI 服务工厂以及 JNDI 服务地址系统属性,查找 Home 接口,从 Home 接口调用 Create 方法创建 Remote 接口,通过 Remote 接口调用其业务方法.

83、如何给 weblogic 指定大小的内存?

在启动 Weblogic 的脚本中 (位于所在 Domain 对应服务器目录下的 startServerName),增加 set MEM_ARGS=-Xms32m -Xmx200m,可以调整最小内存为 32M,最大 200M

84、如何设定的 weblogic 的热启动模式(开发模式)与产品发布模式?

可以在管理控制台中修改对应服务器的启动模式为开发或产品模式之一.或者修改服务的启动文件或者 commenv 文件,增加 set PRODUCTION_MODE=true.

85、如何启动时不需输入用户名与密码?

修改服务启动文件,增加 WLS_USER 和 WLS_PW 项.也可以在 boot.properties 文件中增加加密过的用户名和密码.

86、在 weblogic 管理制台中对一个应用域(或者说是一个网站,Domain)进行 jms 及 ejb 或连接池等相关信息进行配置后,实际保存在什么文件中?

保存在此 Domain 的 config.xml 文件中,它是服务器的核心配置文件.

87、说说 weblogic 中一个 Domain 的缺省目录结构?比如要将一个简单的 helloWorld.jsp 放入何目录下,然的在浏览器上就可打入 http://主机:端口号/helloword.jsp 就可以看到运行结果了? 又比如这其中用到了一个自己写的 javaBean 该如何办?

Domain 目录服务器目录 applications,将应用目录放在此目录下将可以作为应用访问,如果是 Web 应用,应用目录需要满足 Web 应用目录要求,jsp 文件可以直接放在应用目录中,JavaBean 需要放在应用目录的 WEB-INF 目录的 classes 目录中,设置服务器的缺省应用将可以实现在浏览器上无需输入应用名.

88、在 weblogic 中发布 ejb 需涉及到哪些配置文件

不同类型的 EJB 涉及的配置文件不同,都涉及到的配置文件包括 ejb-jar.xml,weblogic-ejb-jar.xmlCMP 实体 Bean 一般还需要 weblogic-cmp-rdbms-jar.xml

89、如何在 weblogic 中进行 ssl 配置与客户端的认证配置或说说 j2ee(标准)进行 ssl 的配置

缺省安装中使用 DemoIdentity.jks 和 DemoTrust.jks KeyStore 实现 SSL,需要配置服务器使用 Enable SSL,配置其端口,在产品模式下需要从 CA 获取私有密钥和数字证书,创建 identity 和 trust keystore,装载获得的密钥和数字证书.可以配置此 SSL 连接是单向还是双向的.

90、如何查看在 weblogic 中已经发布的 EJB?

可以使用管理控制台,在它的 Deployment 中可以查看所有已发布的 EJB

91、CORBA 是什么?用途是什么?

CORBA 标准是公共对象请求代理结构(Common Object Request Broker Architecture),由对象管理组织 (Object Management Group,缩写为 OMG)标准化.它的组成是接口定义语言(IDL), 语言绑定(binding;也译为联编)和允许应用程序间互操作的协议.其目的为: 用不同的程序设计语言书写在不同的进程中运行,为不同的操作系统开发.

92、说说你所熟悉或听说过的 j2ee 中的几种常用模式?及对设计模式的一些看法

Session Facade Pattern: 使用 SessionBean 访问 EntityBean

Message Facade Pattern: 实现异步调用

EJB Command Pattern: 使用 Command JavaBeans 取代 SessionBean,实现轻量级访问

Data Transfer Object Factory: 通过 DTO Factory 简化 EntityBean 数据提供特性

Generic Attribute Access: 通过 AttributeAccess 接口简化 EntityBean 数据提供特性

Business Interface: 通过远程(本地)接口和 Bean 类实现相同接口规范业务逻辑一致性

EJB 架构的设计好坏将直接影响系统的性能、可扩展性、可维护性、组件可重用性及开发效率.项目越复杂,项目队伍越庞大则越能体现良好设计的重要性.

93、说说在 weblogic 中开发消息 Bean 时的 persistent 与 non-persistent 的差别

persistent 方式的 MDB 可以保证消息传递的可靠性,也就是如果 EJB 容器出现问题而 JMS 服务器依然会将消息在此 MDB 可用的时候发送过来,而 non-persistent 方式的消息将被丢弃.

94、Servlet 执行时一般实现哪几个方法?

```
public void init(ServletConfig config)
public ServletConfig getServletConfig()
public String getServletInfo()
public void service(ServletRequest request,ServletResponse response)
public void destroy()
```

95、j2ee 常用的设计模式? 说明工厂模式.Java 中的 23 种设计模式:

Factory (工厂模式), Builder (建造模式), Factory Method (工厂方法模式),
Prototype (原始模型模式), Singleton (单例模式), Facade (门面模式),
Adapter (适配器模式), Bridge (桥梁模式), Composite (合成模式),
Decorator (装饰模式), Flyweight (享元模式), Proxy (代理模式),
Command (命令模式), Interpreter (解释器模式), Visitor (访问者模式),
Iterator (迭代子模式), Mediator (调停者模式), Memento (备忘录模式),
Observer (观察者模式), State (状态模式), Strategy (策略模式),

Template Method (模板方法模式), Chain Of Responsibility (责任链模式)

工厂模式: 工厂模式是一种经常被使用到的模式,根据工厂模式实现的类可以根据提供的数据生成一组类中某一个类的实例,通常这一组类有一个公共的抽象父类并且实现了相同的方法,但是这些方法针对不同的数据进行了不同的操作.首先需要定义一个基类,该类的子类通过不同的方法实现了基类中的方法.然后需要定义一个工厂类,工厂类可以根据条件生成不同的子类实例.当得到子类的实例后,开发人员可以调用基类中的方法而不必考虑到底返回的是哪一个子类的实例.

96、EJB 需直接实现它的业务接口或 Home 接口吗,请简述理由.

远程接口和 Home 接口不需要直接实现,他们的实现代码是由服务器产生的,程序运行中对应实现类会作为对应接口类型的实例被使用.

97、排序都有哪些方法？请列举.用 JAVA 实现一个快速排序.

排序的方法有：[插入排序](#)（直接插入排序、希尔排序），[交换排序](#)（冒泡排序、快速排序），[选择排序](#)（直接选择排序、堆排序），[归并排序](#)，[分配排序](#)（箱排序、基数排序）

顺口溜：冒、择、路、兮、快、归、堆

冒(冒泡) 择(选择) 路(插入) 兮(希尔) 快(快速) 归(归并) 堆(堆排序)

快速排序的伪代码.

//使用快速排序方法对 $a[0:n-1]$ 排序

从 $a[0:n-1]$ 中选择一个元素作为 `middle`, 该元素为支点

把余下的元素分割为两段 `left` 和 `right`, 使得 `left` 中的元素都小于等于支点, 而 `right` 中的元素都大于等于支点

递归地使用快速排序方法对 `left` 进行排序

递归地使用快速排序方法对 `right` 进行排序

所得结果为 `left+middle+right`

99、请对以下在 J2EE 中常用的名词进行解释(或简单描述)

web 容器：给处于其中的应用程序组件 (JSP, SERVLET) 提供一个环境, 使 JSP, SERVLET 直接更容器中的环境变量接口交互, 不必关注其它系统问题. 主要有 WEB 服务器来实现. 例如：TOMCAT, WEBLOGIC, WEBSphere 等. 该容器提供的接口严格遵守 J2EE 规范中的 WEB APPLICATION 标准. 我们把遵守以上标准的 WEB 服务器就叫做 J2EE 中的 WEB 容器.

EJB 容器：Enterprise java bean 容器. 更具有行业领域特色. 他提供给运行在其中的组件 EJB 各种管理功能. 只要满足 J2EE 规范的 EJB 放入该容器, 马上就会被容器进行高效率的管理. 并且可以通过现成的接口来获得系统级别的服务. 例如邮件服务、事务管理.

JNDI：(Java Naming & Directory Interface) JAVA 命名目录服务. 主要提供的功能是：提供一个目录系统, 让其它各地的应用程序在其上面留下自己的索引, 从而满足快速查找和定位分布式应用程序的功能.

JMS：(Java Message Service) JAVA 消息服务. 主要实现各个应用程序之间的通讯. 包括点对点和广播.

JTA：(Java Transaction API) JAVA 事务服务. 提供各种分布式事务服务. 应用程序只需调用其提供的接口即可.

JAF：(Java Action FrameWork) JAVA 安全认证框架. 提供一些安全控制方面的框架. 让开发者通过各种部署和自定义实现自己的个性安全控制策略.

RMI/IIOP：(Remote Method Invocation /internet 对象请求中介协议) 他们主要用于通过远程调用服务. 例如, 远程有一台计算机上运行一个程序, 它提供股票分析服务, 我们可以在本地计算机上实现对其直接调用. 当然这是要通过一定的规范才能在异构的系统之间进行通信. RMI 是 JAVA 特有的.

100、JAVA 语言如何进行异常处理, 关键字：throws, throw, try, catch, finally 分别代表什么意义？在 try 块中可以抛出异常吗？

Java 通过面向对象的方法进行异常处理, 把各种不同的异常进行分类, 并提供了良好的接口. 在 Java 中, 每个异常都是一个对象, 它是 Throwable 类或其它子类的实例. 当一个方法出现异常后便抛出一个异常对象, 该对象中包含有异常信息, 调用这个方法可以捕获到这个异常并进行处

理.Java 的异常处理是通过 5 个关键词来实现的: try、catch、throw、throws 和 finally.一般情况下是用 try 来执行一段程序,如果出现异常,系统会抛出 (throws) 一个异常,这时候你可以通过它的类型来捕捉 (catch) 它,或最后 (finally) 由缺省处理器来处理.

用 try 来指定一块预防所有"异常"的程序.紧跟在 try 程序后面,应包含一个 catch 子句来指定你想要捕捉的"异常"的类型.

throw 语句用来明确地抛出一个"异常".

throws 用来标明一个成员函数可能抛出的各种"异常".

Finally 为确保一段代码不管发生什么"异常"都被执行一段代码.

可以在一个成员函数调用的外面写一个 try 语句,在这个成员函数内部写另一个 try 语句保护其他代码.每当遇到一个 try 语句,"异常"的框架就放到堆栈上面,直到所有的 try 语句都完成.如果下一级的 try 语句没有对某种"异常"进行处理,堆栈就会展开,直到遇到有处理这种"异常"的 try 语句.

101、一个".java"源文件中是否可以包括多个类(不是内部类)?有什么限制?

可以.必须只有一个类名与文件名相同.

102、MVC 的各个部分都有那些技术来实现?如何实现原理?

Model :JavaBean 业务处理, 数据处理

View: 显示的 jsp, 最好不存在任何 Scriptlet, 可以使用表情消除

Controller: 控制接收 View 传递的内容和调用 JavaBean, 并根据 JavaBean 的处理结果进行跳转

实现原理: 所有 JSP 将内容 给 Servlet, servlet 接收这些内容并调用相应的 JavaBean 进行业务及数据的处理一般 JavaBean 有专人负责开发, 完成业务的操作流程, 而 servlet 调用只是接口并根据这些接口的操作结果将内容交给 JSP 进行显示, 但是在显示的时候尽可能少出现 Scriptlet 代码

103、java 中有几种方法可以实现一个线程? 用什么关键字修饰同步方法? stop()和 suspend()方法为何不推荐使用?

有两种实现方法,分别是继承 Thread 类与实现 Runnable 接口

用 synchronized 关键字修饰同步方法

反对使用 stop(),是因为它不安全.它会解除由线程获取的所有锁定,而且如果对象处于一种不连贯状态,那么其他线程能在那种状态下检查和修改它们.结果很难检查出真正的问题所在.suspend()方法容易发生死锁.调用 suspend()的时候,目标线程会停下来,但却仍然持有在这之前获得的锁定.此时,其他任何线程都不能访问锁定的资源,除非被"挂起"的线程恢复运行.对任何线程来说,如果它们想恢复目标线程,同时又试图使用任何一个锁定的资源,就会造成死锁.所以不应该使用 suspend(),而应在自己的 Thread 类中置入一个标志,指出线程应该活动还是挂起.若标志指出线程应该挂起,便用 wait()命其进入等待状态.若标志指出线程应当恢复,则用一个 notify()重新启动线程.

104、java 中有几种类型的流? JDK 为每种类型的流提供了一些抽象类以供继承,请说出他们分别是哪些类?

字节流,字符流.字节流继承于 InputStream OutputStream,字符流继承于 InputStreamReader OutputStreamWriter.在 java.io 包中还有许多其他的流,主要是为了提高性能和使用方便.

105、java 中会存在内存泄漏吗,请简单描述.

会,如: `int i,i2; return (i-i2);` //when i 为足够大的正数,i2 为足够大的负数.结果会造成溢位,导致错误.

106、java 中实现多态的机制是什么?

方法的重写 Overriding 和重载 Overloading 是 Java 多态性的不同表现.重写 Overriding 是父类与子类之间多态性的一种表现,重载 Overloading 是一个类中多态性的一种表现.

107、垃圾回收器的基本原理是什么? 垃圾回收器可以马上回收内存吗? 有什么办法主动通知虚拟机进行垃圾回收?

对于 GC 来说,当程序员创建对象时,GC 就开始监控这个对象的地址、大小以及使用情况.通常,GC 采用有向图的方式记录和管理堆(heap)中的所有对象.通过这种方式确定哪些对象是"可达的",哪些对象是"不可达的".当 GC 确定一些对象为"不可达"时,GC 就有责任回收这些内存空间.可以.程序员可以手动执行 `System.gc()`,通知 GC 运行,但是 Java 语言规范并不保证 GC 一定会执行.

108、静态变量和实例变量的区别?

```
static i = 10; //常量
class A a; a.i=10; //可变
```

109、什么是 java 序列化,如何实现 java 序列化?

序列化就是一种用来处理对象流的机制,所谓对象流也就是将对象的内容进行流化.可以对流化后的对象进行读写操作,也可将流化后的对象传输于网络之间.序列化是为了解决在对对象流进行读写操作时所引发的问题.序列化的实现: 将需要被序列化的类实现 `Serializable` 接口,该接口没有需要实现的方法,implements `Serializable` 只是为了标注该对象是可被序列化的,然后使用一个输出流(如: `FileOutputStream`)来构造一个 `ObjectOutputStream`(对象流)对象,接着,使用 `ObjectOutputStream` 对象的 `writeObject(Object obj)`方法就可以将参数为 `obj` 的对象写出(即保存其状态),要恢复的话则用输入流.

110、是否可以从一个 static 方法内部发出对非 static 方法的调用?

不可以,如果其中包含对象的 `method()`; 不能保证对象初始化.

111、写 clone()方法时,通常都有一行代码,是什么?

Clone 有缺省行为,`super.clone()`;他负责产生正确大小的空间,并逐位复制.

108、在 JAVA 中,如何跳出当前的多重嵌套循环?

用 `break; return` 方法.

112、List、Map、Set 三个接口,存取元素时,各有什么特点?

List 以特定次序来持有元素,可有重复元素
Set 无法拥有重复元素,内部排序.

Map 保存 key-value 值,value 可多值.

113、J2EE 是什么？

J2EE 是 Sun 公司提出的多层(multi-tiered),分布式(distributed),基于组件(component-base)的企业级应用模型 (enterprise application model)

.在这样一个应用系统中,可按照功能划分为不同的组件,这些组件又可在不同计算机上,并且处于相应的层次(tier)中.所属层次包括客户层(client tier)组件,web 层和组件,Business 层和组件,企业信息系统(EIS)层.

114、UML 方面

标准建模语言 UML.用例图,静态图(包括类图、对象图和包图),行为图,交互图(顺序图,合作图),实现图.

115、说出一些常用的类,包,接口,请各举 5 个

常用的类: `BufferedReader` `BufferedWriter` `FileReader` `FileWriter` `String` `Integer`

常用的包: `java.lang` `java.awt` `java.io` `java.util` `java.sql`

常用的接口: `Remote` `List` `Map` `Document` `NodeList`

116、开发中都用到了那些设计模式?用在什么场合?

每个模式都描述了一个在我们的环境中不断出现的问题,然后描述了该问题的解决方案的核心.通过这种方式,你可以无数次地使用那些已有的解决方案,无需在重复相同的工作.主要用到了 MVC 的设计模式.用来开发 JSP/Servlet 或者 J2EE 的相关应用.简单工厂模式等.

117、jsp 有哪些动作?作用分别是什么?

JSP 共有以下 6 种基本动作

`jsp:include`: 在页面被请求的时候引入一个文件.

`jsp:useBean`: 寻找或者实例化一个 JavaBean.

`jsp:setProperty`: 设置 JavaBean 的属性.

`jsp:getProperty`: 输出某个 JavaBean 的属性.

`jsp:forward`: 把请求转到一个新的页面.

`jsp:plugin`: 根据浏览器类型为 Java 插件生成 OBJECT 或 EMBED 标记.

118、Anonymous Inner Class (匿名内部类) 是否可以 extends(继承)其它类,是否可以 implements(实现)interface(接口)?

可以继承其他类或完成其他接口, 因为匿名内部类是在抽象类和接口的基础上发展而来, 在 swing 编程中常用此方式.

119、应用服务器与 WEB SERVER 的区别?

应用服务器: `Weblogic`、`Tomcat`、`Jboss` WEB SERVER: `IIS`、`Apache`

120、BS 与 CS 的联系与区别. (PS 答蓝色字体即可)

C/S 是 Client/Server 的缩写.服务器通常采用高性能的 PC、工作站或小型机,并采用大型数据库系统,如 Oracle、Sybase、Informix 或 SQL Server.客户端需要安装专用的客户端软件.

B/S 是 Browser/Server 的缩写,客户机上只要安装一个浏览器 (Browser),如 Netscape Navigator

或 Internet Explorer,服务器安装 Oracle、Sybase、Informix 或 SQL Server 等数据库.在这种结构下,用户界面完全通过 WWW 浏览器实现,一部分事务逻辑在前端实现,但是主要事务逻辑在服务器端实现.浏览器通过 Web Server 同数据库进行数据交互.

C/S 与 B/S 区别:

1. 硬件环境不同:

C/S 一般建立在专用的网络上,小范围里的网络环境,局域网之间再通过专门服务器提供连接和数据交换服务.

B/S 建立在广域网之上的,不必是专门的网络硬件环境,例与电话上网,租用设备.信息自己管理.有比 C/S 更强的适应范围,一般只要有操作系统和浏览器就行

2. 对安全要求不同

C/S 一般面向相对固定的用户群,对信息安全的控制能力很强.一般高度机密的信息系统采用 C/S 结构适宜.可以通过 B/S 发布部分可公开信息.

B/S 建立在广域网之上,对安全的控制能力相对弱,可能面向不可知的用户.

3. 对程序架构不同

C/S 程序可以更加注重流程,可以对权限多层次校验,对系统运行速度可以较少考虑.

B/S 对安全以及访问速度的多重的考虑,建立在需要更加优化的基础之上.比 C/S 有更高的要求 B/S 结构的程序架构是发展的趋势,从 MS 的 .Net 系列的 BizTalk 2000 Exchange 2000 等,全面支持网络的构件搭建的系统.SUN 和 IBM 推的 JavaBean 构件技术等,使 B/S 更加成熟.

4. 软件重用不同

C/S 程序可以不可避免的整体性考虑,构件的重用性不如在 B/S 要求下的构件的重用性好.

B/S 对的多重结构,要求构件相对独立的功能.能够相对较好的重用.就象买来的餐桌可以再利用,而不是做在墙上的石头桌子

5. 系统维护不同

C/S 程序由于整体性,必须整体考察,处理出现的问题以及系统升级.升级难.可能是再做一个全新的系统

B/S 构件组成,方面构件个别的更换,实现系统的无缝升级.系统维护开销减到最小.用户从网上自己下载安装就可以实现升级.

6. 处理问题不同

C/S 程序可以处理用户面固定,并且在相同区域,安全要求高需求,与操作系统相关.应该都是相同的系统

B/S 建立在广域网上,面向不同的用户群,分散地域,这是 C/S 无法作到的.与操作系统平台关系最小.

7. 用户接口不同

C/S 多是建立的 Window 平台上,表现方法有限,对程序员普遍要求较高

B/S 建立在浏览器上,有更加丰富和生动的表现方式与用户交流.并且大部分难度减低,减低开发成本.

8. 信息流不同

C/S 程序一般是典型的中央集权的机械式处理,交互性相对低

B/S 信息流向可变化,B-B B-C B-G 等信息、流向的变化,更像交易中心.

121、Linux 下线程,GDI 类的解释.

LINUX 实现的就是基于核心轻量级进程的"一对一"线程模型,一个线程实体对应一个核心轻量级进程,而线程之间的管理在核外函数库中实现.

GDI 类为图像设备编程接口类库.

123、STRUTS 的应用(如 STRUTS 架构)

Struts 是采用 Java Servlet/JavaServer Pages 技术,开发 Web 应用程序的开放源码的 framework. 采用 Struts 能开发出基于 MVC(Model-View-Controller)设计模式的应用构架. Struts 有如下的主要功能:一.包含一个 controller servlet,能将用户的请求发送到相应的 Action 对象.二.JSP 自由 tag 库,并且在 controller servlet 中提供关联支持,帮助开发员创建交互式表单应用.三.提供了一系列实用对象: XML 处理、通过 Java reflection APIs 自动处理 JavaBeans 属性、国际化的提示和消息.

124、Jdo 是什么?

JDO 是 Java 对象持久化的新的规范,为 java data object 的简称,也是一个用于存取某种数据仓库中的对象的标准 API.JDO 提供了透明的对象存储,因此对开发人员来说,存储数据对象完全不需要额外的代码(如 JDBC API 的使用).这些繁琐的例行工作已经转移到 JDO 产品提供商身上,使开发人员解脱出来,从而集中时间和精力在业务逻辑上.另外,JDO 很灵活,因为它可以在任何数据底层上运行.JDBC 只是面向关系数据库(RDBMS)JDO 更通用,提供到任何数据底层的存储功能,比如关系数据库、文件、XML 以及对象数据库(ODBMS)等等,使得应用可移植性更强.

125、内部类可以引用他包含类的成员吗?有没有什么限制?

一个内部类对象可以访问创建它的外部类对象的内容

126、WEB SERVICE 名词解释.JSWDL 开发包的介绍.JAXP、JAXM 的解释.SOAP、UDDI,WSDL 解释.

Web Service Web Service 是基于网络的、分布式的模块化组件,它执行特定的任务,遵守具体的技术规范,这些规范使得 Web Service 能与其他兼容的组件进行互操作.

JAXP(Java API for XML Parsing) 定义了 Java 中使用 DOM, SAX, XSLT 的通用的接口.这样在你的程序中你只要使用这些通用的接口,当你需要改变具体的实现时候也不需要修改代码.

JAXM(Java API for XML Messaging) 是为 SOAP 通信提供访问方法和传输机制的 API.

WSDL 是一种 XML 格式,用于将网络服务描述为一组端点,这些端点对包含面向文档信息或面向过程信息的信息进行操作.这种格式首先对操作和消息进行抽象描述,然后将其绑定到具体的网络协议和消息格式上以定义端点.相关的具体端点即组合成为抽象端点(服务).

SOAP 即简单对象访问协议(Simple Object Access Protocol),它是用于交换 XML 编码信息的轻量级协议.

UDDI 的目的是为电子商务建立标准; UDDI 是一套基于 Web 的、分布式的、为 Web Service 提供的、信息注册中心的实现标准规范,同时也包含一组使企业能将自身提供的 Web Service 注册,以使别的企业能够发现的访问协议的实现标准.

127、设计 4 个线程,其中两个线程每次对 j 增加 1,另外两个线程对 j 每次减少 1.写出程序. 以下程序使用内部类实现线程,对 j 增减的时候没有考虑顺序问题.

```
public class ThreadTest1 {
    private int j;
    public static void main(String args[]) {
        ThreadTest1 tt = new ThreadTest1();
        Inc inc = tt.new Inc();
        Dec dec = tt.new Dec();
        for (int i = 0; i < 2; i++) {
            Thread t = new Thread(inc);
            t.start();
```

```

        t = new Thread(dec);
        t.start();
    }
}

private synchronized void inc() {
    j++;
    System.out.println(Thread.currentThread().getName() +
"-inc:" + j);
}

private synchronized void dec() {
    j--;
    System.out.println(Thread.currentThread().getName() +
"-dec:" + j);
}

class Inc implements Runnable {
    public void run() {
        for (int i = 0; i < 100; i++) {
            inc();
        }
    }
}

class Dec implements Runnable {
    public void run() {
        for (int i = 0; i < 100; i++) {
            dec();
        }
    }
}
}

```

128、Java 有没有 goto?

java 中的 goto 保留字,现在[没有在 java 中使用](#).

129、启动一个线程是用 run()还是 start()?

使用 start()方法, 因为要通知 JVM 进行 CPU 资源分类

130、MVC 的好处

- 1 各施其职 互不干涉
- 2.利于开发中的分工
- 3.有利于组件的重用

模型(**Model**)可以分为业务模型和数据模型,他们代表应用程序的业务逻辑和状态
 视图(**View**)视图提供可交互的客户界面,向客户显示模型数据
 控制器(**Controller**)响应客户的请求, 根据客户的请求来操作模型 并把模型的响应结果由视图展现给客户.

131、javaBean 有什么好处

1 封装数据 2 封装业务
满足 JavaBean 的要求有几点呢？ A. 是一个公有类 并提供 无参的公有的构造方法 B. 属性是私有的
3. 具有公有的访问属性的 <code>get</code> 和 <code>set</code> 方法
要满足构造方法的条件有哪些呢？ A. 方法名与类名相同 B. 没有返回类型

132、100个和尚吃100个馒头,大和尚每人吃4个,小和尚每4人吃一个,问,大,小和尚各有多少人?

假设大和尚有100个（全都是大和尚），那么可以吃掉400个馒头（缺少300个）。
而每用一个和尚替换一个大和尚，就会少吃 $4 - 0.25 = 3.75$ 个馒头，而实际比假设少吃了300个馒头，则共有 $300 / 3.75 = 80$ 个小和尚。
所以：大和尚20个，小和尚80个。

133、Tomcat 的优化经验

答：去掉对 `web.xml` 的监视，把 `jsp` 提前编辑成 `Servlet`。

134、MVC 的各个部分都有那些技术来实现?如何实现?

答：MVC 是 Model—View—Controller 的简写。Model 代表的是应用的业务逻辑（通过 `JavaBean`，`EJB` 组件实现），View 是应用的表示面（由 `JSP` 页面产生），Controller 是提供应用的处理过程控制（一般是一个 `Servlet`），通过这种设计模型把应用逻辑，处理过程和显示逻辑分成不同的组件实现。这些组件可以进行交互和重用

135. 在 eclipse 中调试时，怎样查看一个变量的值

在要查看的变量前先设置断点，然后选中变量，右键选 `debug as-->Java Application`，打开 `debug` 透视图，这时在 `Variables` 窗口中可以看到变量当前的值。

136. 什么是 Web Service ?

Web Service 就是一个网络组件（一个可以通过网络访问的程序）。
它有一个或多个端口（Port），这些端口用于接收客户端的请求，并返回响应请求和响应的 都是一种基于 XML 的消息。
不过这种消息遵循特定的格式（SOAP）

137、Java 的==与 equals 的区别

“==”比较两个变量本身的值，即两个对象在内存中的首地址。

“equals()”比较字符串中所包含的内容是否相同

```
package com.jse.org.core.eq;

public class Test1 {
    public static void main(String[] args) {
        String a = "abc";
        String b = "abc";
        //这个是创建了两个对象,一个是匿名,一个是new实例创建
        String c = new String("abC");
        if (a == b) {
            System.out.println("==是用来对比地址是否相等!");
        }
        if (a.equals(b)) {
            if (b.equalsIgnoreCase(c)) //忽略大小写
                System.out.println("equals用来对比空间内容是否相等!");
        }
    }
}
```

138、局部变量、公有变量、静态变量的区别

局部变量:程序里面不一定就只是一个程序,也可能调用其他子程序,只在子程序里面声明并起到作用

全局变量就是对于整个程序,都可以调用的变量

静态变量:指变量的地址不变,也就是说变量

139 有无限多的水，一个 3 升的桶子和一个 5 升的桶子，怎样准确称出 4 升的水

先把5升桶到入7升桶,在装5升继续到入7升桶,则5升桶剩3升,把7升倒了,将3升倒如7升桶,在装5升加入7升桶,因为已经有3升了,所以加了4升就满了,5升桶正好剩1升.

140.有两个房间，一个房间有3个开关，另外一个房间有3个灯泡，只允许进房间一次，问怎么分出哪个开关控制哪个灯

先开两个，然后过一段时间关一个，进房间里亮的是开着的开关，暗的但热的是先开后关的开关，另外一个是不快过的开关

141. java 中 list 和 set 怎么区别？

1)List 和 Set 都是接口。他们各自有自己的实现类，

有无顺序的实现类，也有有顺序的实现类。

2)最大的不同就是 List 是可以重复的。而 Set 是不能重复的。

List 适合经常追加数据，插入，删除数据。但随即取数效率比较低。

Set 适合经常地随即储存，插入，删除。但是在遍历时效率比较低

142. struts 与 spring 的区别

struts: 是一种基于 MVC 模式的一个 web 层的处理。

Spring:提供了通用的服务, ioc/di aop,关心的不仅仅 web 层, 应当 j2ee 整体的一个服务, 可以很容易融合不同的技术 struts hibernate ibatis ejb remote springJDBC springMVC

143. 两种跳转方式分别是什么? 有什么区别?

1. 服务器跳转:<jsp:forward>跳转之后地址栏不改变, 可以传递 request 属性, 实际上对 RequestDispatcher 接口的封装
2. 客户端跳转, response.sendRedirect();跳转之后地址栏发生改变

144. 请你说出类集框架的完整结构

类集框架最大的接口: Collection, Map, Iterator, Enumeration
Collection: 存放单个值 List: 允许有重复内容 ArrayList: 异步处理, 新的操作类, 非线程安全 Vector: 同步处理, 旧的操作类, 线程安全, 支持 Enumeration 输出 Set: 不允许有重复内容, 靠 hashCode() 和 equals() 进行存放的验证 HashSet: 无序存放 TreeSet: 有序存放, 按 Comparable 排序
Map: 存放一对值 HashMap: 新的类, 异步处理, 非线程安全, 允许为 null Hashable: 旧的类, 同步处理, 非线程安全, 不允许为 null Properties: 属性操作类 TreeMap: 有序排列, 按照 key 排序, 根据 Comparable 指定排序规则
Iterator: 迭代输出, 依靠 Collection 接口中的 iterator 方法输出, 是新的输出标准
Enumeration: 旧的输出操作.

145小明一家过一座桥, 过桥时是黑夜, 所以必须有灯。现在小明过桥要1秒, 小明的弟弟要3秒, 小明的爸爸要6秒, 小明的妈妈要8秒, 小明的爷爷要12秒。每次此桥最多可过两人, 而过桥的速度依过桥最慢者而定, 而且灯在点燃后30秒就会熄灭。问: 小明一家如何过桥?

- 1、小明与弟弟过桥——3秒;
 - 2、小明回——1秒;
 - 3、妈妈与爷爷过桥——12秒;
 - 4、弟弟回——3秒;
 - 5、小明与爸爸过桥——6秒;
 - 6、小明回——1秒;
 - 7、小明与弟弟过桥——3秒。
- 总计: 29秒。
- 5与7可以调换顺序, 总时间不变。

146.下面有一个电话号码是18912012901请你用正则表达式写校验一下！

答: `d{11}` //匹配国内电话号码: `d{3}-d{8}|d{4}-d{7}`

147. 下面那几个是正确的.

```
String s1="你好"; int i=3;s1+=i; //
String s2="你好"; int y=3; if(i==s2){} //Error
String s3="你好"; int k=3; s3=k+s3;
String s4="你好"; int z=3; s4=z+; //Error
String s5=null; int o=(s5!=null)&&(s5.length()>0)?s5.length():0;
```

148. 用一枚硬币进行实验，连续抛三次，结果都是同一面的概率是多少

设正面为 a，反面为 b

所有的可能性为

abb

aba

aab

aaa

baa

bab

bba

bbb

都是同一面的概率自然是25%

149 据说有人给酒肆的老板娘出了一个难题：此人明明知道店里只有两个舀酒的勺子，分别能舀 7 两和 11 两酒，却硬要老板娘卖给他 2 两酒。聪明的老板娘毫不含糊，用这两个勺子在酒缸里舀酒，并倒来倒去，居然量出了 2 两酒，聪明的你能做到吗？

先量 11 两倒入 7 两勺，则 11 两勺余 4 两酒，将 7 两勺空出，将此 11 两酒勺中的 4 两酒倒入 7 两勺，则此时 7 两勺尚有 3 两容量，再量取 11 两酒继续倒入 7 两勺满为止，则此时 11 两勺内为 8 两酒，倒空 7 两勺，将 11 两勺内 8 两剩酒倒入 7 两勺满为止则 11 两勺内余下 1 两酒，将此 1 两酒倒入客人容器，再重复一次就有 2 两酒了。

150.有一天中午一位农妇，拿着一篮鸡蛋来到市场上，第一位顾客买了全部的一半再加半个，第二位顾客买了第一次剩下部分一半再加半个，第三位顾客买了第二次剩下部分一半再加半个，第四位顾客买了第三次剩下部分一半再加半个，第五位顾客买了第四次剩下部分一半再加半个，第六位顾客买了第五次剩下部分一半再加半个，他发现自己的跟其他客户的鸡蛋买了刚好整个，而且农妇刚好卖了所有的鸡蛋，问农妇一共拿了多少个鸡蛋市场？

151.tomcat 中能运行 ejb 吗

Tomcat 是 web container,支持 jsp & servlet, apache 是 http Server.

Weblogic Jboss 才是 ejb container, 支持 EJB

3、protected: protected对于子女、朋友来说,就是public的,可以自由使用,没有任何限制,而对于其他的外部class, protected就变成private。

作用域 当前类 同一package 子孙类 其他package

public ✓ ✓ ✓ ✓

protected ✓ ✓ ✓ ×

friendly ✓ ✓ × ×

private ✓ × × ×

【152】final, 在 Java 中有哪些作用

final 可以修饰类,函数,变量

修饰类时:意思是修饰类为最终类,任何类不能再继承它。

修改函数时:意思是该方法在类里是最终方法,继承它的方法不能去重写它。

修饰变量时:意思是被修饰的变量是常量或者叫终态变量,一旦初始化后就不能再改变 变量的值

153Servlet 与 Applet、JSP 的区别?

applet 是客户端的程序,它们可以[直接嵌入](#)到网页或者其他特定的容器中,并能够产生特殊的效果。

servlet 是服务端器的程序,用于实现服务器功能的扩充,响应客户端请求

jsp 则是对 [servlet](#) 简化,用于方便输出动态页面 实质上就是 servlet

154 Synchronized 和 java.util.concurrent.locks.Lock 的区别

主要相同点 : Lock 能完成 Synchronized 所实现的所有功能。

主要不同点 : Lock 有比 Synchronized 更精确的县城予以和更好的性能.

Synchronized 会自动释放锁,但是 Lock 一定要求程序员手工释放,并且必须在 finally 从句中释放.synchronized 修饰方法时 表示同一个对象在不同的线程中 表现为同步队列,如果实例化不同的对象 那么 synchronized 就不会出现同步效果了

155TRUNCATE,DELETE,DROP 放在一起比较:

TRUNCATE TABLE: 删除内容、释放空间但不删除定义。

DELETE TABLE: 删除内容不删除定义, 不释放空间。

DROP TABLE: 删除内容和定义, 释放空间。

Hibernate

一、Hibernate的get与load

Hibernate中有两个极为相似的方法get()与load(),他们都可以指定通过指定的实体类与ID从数据库中读取数据,并返回对应的实例,但Hibernate不会搞两个完全一样的方法的,它们间的不同在于:

1. 如果找不到符合条件的纪录, get()方法将返回null. 而load()将会报出ObjectNotFoundException.
2. load()方法可以返回实体的代理类实例, 而get()永远只返回实体类.
3. load()方法可以充分利用二级缓存和内部缓存的现有数据, 而get()方法只在内部缓存中进行查找, 如没有发现对应数据将跳过二级缓存, 直接调用SQL完成查找.

二、getCurrentSession()与openSession()的区别

1. 采用getCurrentSession()创建的session会绑定到当前线程中, 而采用openSession()创建的session则不会
2. 采用getCurrentSession()创建的session在commit或rollback时会自动关闭, 而采用openSession()创建的session必须手动关闭

三、hibernate工作原理及为什么要用?

原理:

1. 读取并解析配置文件
2. 读取并解析映射信息 orm, 创建 SessionFactory
3. 打开Session
4. 创建事务Transaction
5. 持久化操作
6. 提交事务
7. 关闭Session
8. 关闭SessionFactory

为什么要用:

1. 对JDBC访问数据库的代码做了封装, 大大简化了数据访问层繁琐的重复性代码。
2. Hibernate是一个基于JDBC的主流持久化框架, 是一个优秀的ORM实现, 大程度的简化DAO层的编码工作
3. hibernate使用Java反射机制, 而不是字节码增强程序来实现透明性。
4. hibernate的性能非常好, 因为它是个轻量级框架。映射的灵活性很出色。它支持各种关系数据库, 从一对一到多对多的各种复杂关系。

四、Hibernate是如何延迟加载?

1. Hibernate2延迟加载实现: a) 实体对象 b) 集合 (Collection)
2. Hibernate3 提供了属性的延迟加载功能

当Hibernate在查询数据的时候,数据并没有存在与内存中,当程序真正对数据的操作时,对象才存在与内存中,就实现了延迟加载,他节省了服务器的内存开销,从而提高了服务器的性能。

3. Hibernate中怎样实现 类之间的关系?(如: 一对多、多对多的关系)
类与类之间的关系主要体现在表与表之间的关系进行操作,它们都市对对象进行操作,我们程序中 把所有的表与类都映射在一起, 它们通过配置文件中的many-to-one、one-to-many、many-to-many、

五. 说下Hibernate的缓存机制

1. 内部缓存存在Hibernate中又叫一级缓存, 属于应用事物级缓存
2. 二级缓存:

- a) 应用及缓存
- b) 分布式缓存

条件: 数据不会被第三方修改、数据大小在可接受范围、数据更新频率低、同一数据被系统频繁使用、非 关键数据

二级缓存就是只需要查一次数据库,在次操作数据不用去查询了,它会首先找缓存,看看有没有,如果有就直接用,没有的时候再去查!

c) 第三方缓存的实现

六、Hibernate的查询方式

Sql、Criteria,object composition

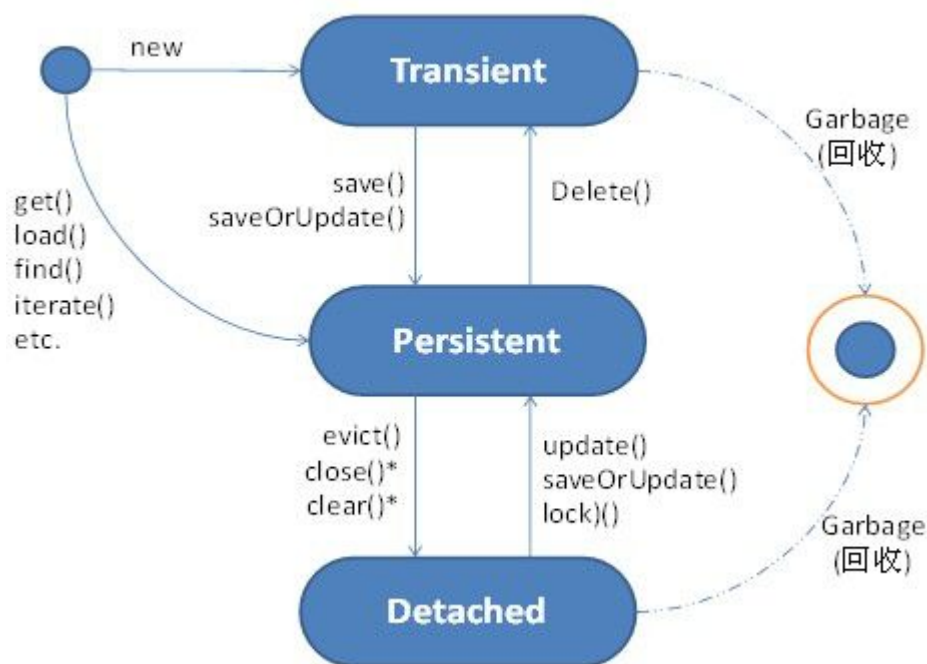
Hql:

- 1、 属性查询
- 2、 参数查询、命名参数查询
- 3、 关联查询
- 4、 分页查询
- 5、 统计函数

七、 如何优化Hibernate?

1. 使用双向一对多关联,不使用单向 一对多
2. 灵活使用单向一对多关联
3. 不用一对一,用多对一取代
4. 配置对象缓存,不使用集合缓存
5. 一对多集合使用 Bag,多对多集合使用Set
6. 继承类使用显式多态
7. 表字段要少,表关联不要怕多,有二级缓存撑腰

八、hibernate生命周期



瞬时对象

Transient Objects使用new 操作符初始化的对象不是立刻持久的,它们的状态是瞬时的,也就是它们没有任何跟数据库表相关联系的行为,主要应用不再引用这些对象,它们的状态将会丢失,并由垃圾回收机制回收

持久化对象

Persist Objects 持久实例是任何具有数据库标识的实例,它有持久化管理器Session统一管理,持久实例是在事务中进行操作的 它们的状态在事务结束同数据库进行同步,当事务提交时通过执行sql insert update delete 语句把内存中的状态同步到数据库

离线对象

Detached Objects Session关闭之后,持久化对象就变成离线对象,离线表示这个对象不能再与数据库保持同步,它们不再受hibernate管理

瞬时和离线对象

共同点: 没有被session管理

异点: 瞬时是数据库没有语句对应记录 离线是DB是有语句对应的记录只是session的缓存清掉啦

Spring

一、spring工作机制及为什么要用?

1.spring mvc请所有的请求都提交给DispatcherServlet,它会委托应用系统的其他模块负责负责对请求进行真正的处理工作。

<ol style="list-style-type: none">2. DispatcherServlet 查询一个或多个 HandlerMapping, 找到处理请求的 Controller.3. DispatcherServlet 请求提交到目标 Controller4. Controller 进行业务逻辑处理后, 会返回一个 ModelAndView5. DispatcherServlet 查询一个或多个 ViewResolver 视图解析器, 找到 ModelAndView 对象指定的视图对象6. 视图对象负责渲染返回给客户端。
--

为什么用:

{AOP 让开发人员可以创建非行为性的关注点, 称为横切关注点, 并将它们插入到应用程序代码中。使用 AOP 后, 公共服务 (比如日志、持久性、事务等) 就可以分解成方面并应用到域对象上, 同时不会增加域对象的对象模型的复杂性。

IOC 允许创建一个可以构造对象的应用环境, 然后向这些对象传递它们的协作对象。正如单词 倒置 所表明的, IOC 就像反过来的 JNDI。没有使用一堆抽象工厂、服务定位器、单元素 (singleton) 和直接构造 (straight construction), 每一个对象都是用其协作对象构造的。因此是由容器管理协作对象 (collaborator)。

Spring 即使是一个 AOP 框架, 也是一个 IOC 容器。Spring 最好的地方是它有助于您替换对象。有了 Spring, 只要用 JavaBean 属性和配置文件加入依赖性 (协作对象)。然后可以很容易地在需要时替换具有类似接口的协作对象。}

Spring 框架是一个分层架构, 由 7 个定义良好的模块组成。Spring 模块构建在核心容器之上, 核心容器定义了创建、配置和管理 bean 的方式, 如图 1 所示。

组成 Spring 框架的每个模块 (或组件) 都可以单独存在, 或者与其他一个或多个模块联合实现。每个模块的功能如下:

☆ **核心容器**: 核心容器提供 Spring 框架的基本功能。核心容器的主要组件是 BeanFactory, 它是工厂模式的实现。BeanFactory 使用控制反转 (IOC) 模式将应用程序的配置和依赖性规范与实际的应用程序代码分开。

☆ **Spring 上下文**: Spring 上下文是一个配置文件, 向 Spring 框架提供上下文信息。Spring 上下文包括企业服务, 例如 JNDI、EJB、电子邮件、国际化、校验和调度功能。

☆ **Spring AOP**: 通过配置管理特性, Spring AOP 模块直接将面向方面的编程功能集成到了 Spring 框架中。所以, 可以很容易地使 Spring 框架管理的任何对象支持 AOP。Spring AOP 模块为基于 Spring 的应用程序中的对象提供了事务管理服务。通过使用 Spring AOP, 不用依赖 EJB 组件, 就可以将声明性事务管理集成到应用程序中。

☆ **Spring DAO**: JDBC DAO 抽象层提供了有意义的异常层次结构, 可用该结构来管理异常处理和不同数据库供应商抛出的错误消息。异常层次结构简化了错误处理, 并且极大地降低了需要编写的异常代码数量 (例如打开和关闭连接)。Spring DAO 的面向 JDBC 的异常遵从通用的 DAO 异常层次结构。

☆ **Spring ORM**: Spring 框架插入了若干个 ORM 框架,从而提供了 ORM 的对象关系工具,其中包括 JDO、Hibernate 和 iBatis SQL Map。所有这些都遵从 Spring 的通用事务和 DAO 异常层次结构。

☆ **Spring Web 模块**: Web 上下文模块建立在应用程序上下文模块之上,为基于 Web 的应用程序提供了上下文。所以, Spring 框架支持与 Jakarta Struts 的集成。Web 模块还简化了处理多部分请求以及将请求参数绑定到域对象的工作。

☆ **Spring MVC 框架**: MVC 框架是一个全功能的构建 Web 应用程序的 MVC 实现。通过策略接口, MVC 框架变成为高度可配置的, MVC 容纳了大量视图技术,其中包括 JSP、Velocity、Tiles、iText 和 POI。

Spring 框架的功能可以用在任何 J2EE 服务器中,大多数功能也适用于不受管理的环境。Spring 的核心要点是:支持不绑定到特定 J2EE 服务的可重用业务和数据访问对象。毫无疑问,这样的对象可以在不同 J2EE 环境 (Web 或 EJB)、独立应用程序、测试环境之间重用。

二 spring工作机制及为什么要用?

1.spring mvc请所有的请求都提交给DispatcherServlet,它会委托应用系统的其他模块负责负责对请求进行真正的处理工作。

2.DispatcherServlet 查询一个或多个HandlerMapping,找到处理请求的Controller.

3.DispatcherServlet请请求提交到目标 Controller

4.Controller进行业务逻辑处理后,会返回一个ModelAndView

5.Dispathcher查询 一个或多个ViewResolver视图解析器,找到ModelAndView对象指定的视图对象

6.视图对象负责渲染返回给客户端。

为 什么用:

{AOP 让开发人员可以创建非行为性的关注点,称为横切关注点,并将它们插入到应用程序代码中。使用 AOP 后,公共服务 (比如日志、持久性、事务等)就可以分解成方面并应用到域对象上,同时不会增加域对象的对象模型的复杂性。

IOC 允许创建一个可以构造对象的应用环境,然后向这些对象传递它们的协作对象。正如单词 倒置 所表明的,IOC 就像反过来的 JNDI。没有使用一堆抽象工厂、服务定位器、单元素 (singleton) 和直接构造 (straight construction),每一个对象都是用其协作对象构造的。因此是由容器管理协作对象 (collaborator)。

Spring即使一个 AOP框架,也是一IOC容器。 Spring 最好的地方是它有助于您替换对象。有了 Spring,只要用 JavaBean 属性和配置文件加入依赖性 (协作对象)。然后可以很容易地在需要时替换具有类似接口的协作对象。}

三、什么是IOC和AOP

1) IoC 就是 Inversion of Control,控制反转。在 Java 开发中, **IoC 意味着将你设计好的类**

交给系统去控制，而不是在你的类内部控制。这称为控制反转。

2)Aspect-oriented programming (AOP) ，从名字上说类似oop Object Oriented Programming。AOP是OOP的延续，是Aspect Oriented Programming的缩写，意思是面向方面编程

将一切实物都抽象的看做是多个实体的抽象体，而每个不同类型的抽象体都能够作为这个实物的一种实现机制的表现，从而在业务拓展时减少对原有代码的维护，取而代之的则是 增加->切换 的操作。

对 AOP 一些概念的理解：

- 1.切面(aspect)：指横切性关注点的抽象即为切面（系统模块化的一个切面或领域），它与类相似，只是两者的关注点不一样，类是对物体特征的抽象，而切面是横切性关注点的抽象，如日志记录。
- 2.连接点(Joinpoint)：所谓连接点是指那些被拦截到的点。在 spring 中，只支持拦截方法，因为 spring 只支持方法类型的连接点，连接点可以是方法、field、类构造器或异常抛出点。
- 3.通知(Advice)：切面的实际实现，他通知系统新的行为。是指在什么连接点做什么动作。
- 4.切入点(Pointcut)：表达式表示的连接点集合，表达式是一个规则用来表示与连接点匹配的规则。
- 5.引入(Introduction)：在不修改类代码的前提下，为类添加新的方法和属性，分静态和动态引入。
- 6.目标对象(target)：被通知的对象。
- 7.织入(Weave)：将切面应用到目标对象从而创建一个新代理对象的过程。织入发生在目标对象生命周期的多个点上。

四、事务的几种传播特性

1. `propagation_required`：如果存在一个事务，则支持当前事务.如果没有事务则开启
2. `propagation_supports`：如果存在一个事务，支持当前事务.如果没有事务，则非事务的执行
3. `propagation_mandatory`：如果已经存在一个事务，支持当前事务.如果没有一个活动的事务，则抛出异常.
4. `propagation_requires_new`：总是开启一个新的事务.如果一个事务已经存在，则将这个存在的事务挂起.
5. `propagation_not_supported`：总是非事务地执行，并挂起任何存在的事务.
6. `propagation_never`：总是非事务地执行，如果存在一个活动事务，则抛出异常
7. `propagation_nested`：如果一个活动的事务存在，则运行在一个嵌套的事务中. 如果没有活动事务，则按TransactionDefinition.PROPROPAGATION_REQUIRED 属性执行

Struts

一、Struts工作机制？为什么要使用Struts？

工作机制：

Struts的工作流程：

在web应用启动时就会加载初始化 ActionServlet,ActionServlet从struts-config.xml文件中读取配置信息,把它们存放到各种配置对象

当 `ActionServlet` 接收到一个客户请求时, 将执行如下流程。

- 1) 检索和用户请求匹配的 `ActionMapping` 实例, 如果不存在, 就返回请求路径无效信息;
- 2) 如果 `ActionForm` 实例不存在, 就创建一个 `ActionForm` 对象, 把客户提交的表单数据保存到 `ActionForm` 对象中;
- 3) 根据配置信息决定是否需要表单验证。如果需要验证, 就调用 `ActionForm` 的 `validate()` 方法;
- 4) 如果 `ActionForm` 的 `validate()` 方法返回 `null` 或返回一个不包含 `ActionMessage` 的 `ActionErrors` 对象, 就表示表单验证成功;
- 5) `ActionServlet` 根据 `ActionMapping` 所包含的映射信息决定将请求转发给哪个 `Action`, 如果相应的 `Action` 实例不存在, 就先创建这个实例, 然后调用 `Action` 的 `execute()` 方法;
- 6) `Action` 的 `execute()` 方法返回一个 `ActionForward` 对象, `ActionServlet` 在把客户请求转发给 `ActionForward` 对象指向的 JSP 组件;
- 7) `ActionForward` 对象指向 JSP 组件生成动态网页, 返回给客户;

为什么要用:

JSP、Servlet、JavaBean 技术的出现给我们构建强大的企业应用系统提供了可能。但用这些技术构建的系统非常的繁乱, 所以在此之上, 我们需要一个规则、一个把这些技术组织起来的规则, 这就是框架, `Struts` 便应运而生。

基于 `Struts` 开发的应用由 3 类 组件构成: 控制器组件、模型组件、视图组件

二、Struts 的 validate 框架是如何验证的?

在 `struts` 配置文件中配置具体的错误提示, 再在 `FormBean` 中的 `validate()` 方法具体调用。

三、说下 Struts 的设计模式

MVC 模式: web 应用程序启动时就会加载并初始化 `ActionServlet`。用户提交表单时, 一个配置好的 `ActionForm` 对象被创建, 并被填入表单相应的数据, `ActionServlet` 根据 `Struts-config.xml` 文件配置好的设置决定是否需要表单验证, 如果需要就调用 `ActionForm` 的 `validate()` 验证后选择将请求发送到哪个 `Action`, 如果 `Action` 不存在, `ActionServlet` 会先创建这个对象, 然后调用 `Action` 的 `execute()` 方法。`execute()` 从 `ActionForm` 对象中获取数据, 完成业务逻辑, 返回一个 `ActionForward` 对象, `ActionServlet` 再把客户请求转发给 `ActionForward` 对象指定的 jsp 组件, `ActionForward` 对象指定的 jsp 生成动态的网页, 返回给客户。

四、struts1 与 struts2 本质区别

1 在 **Action 实现类方面** 的对比: `Struts 1` 要求 `Action` 类继承一个抽象基类; `Struts 1` 的一个具体问题是使用抽象类编程而不是接口。 `Struts 2` `Action` 类可以实现一个 `Action` 接口, 也可以实现其他接口, 使可选和定制的服务成为可能。 `Struts 2` 提供一个 `ActionSupport` 基类去实现常用的接口。即使 `Action` 接口不是必须实现的, 只有一个包含 `execute` 方法的 POJO 类都可以用作 `Struts 2` 的 `Action`。

2 **线程模式方面** 的对比: `Struts 1` `Action` 是单例模式并且必须是线程安全的, 因为仅有 `Action` 的一个实例来处理所有的请求。单例策略限制了 `Struts 1` `Action` 能做的事, 并且要在开发时特别小心。 `Action` 资源必须是线程安全的或同步的; `Struts 2`

Action对象为每一个请求产生一个实例，因此 没有线程安全问题。

3 **Servlet依赖方面的对比**: Struts 1 Action依赖于 Servlet API，因为Struts 1 Action的execute方法中有HttpServletRequest和 HttpServletResponse方法。Struts 2 Action不再依赖于Servlet API，从而允许Action 脱离Web容器运行，从而降低了测试 Action 的难度。当然，如果 Action 需要直接访问HttpServletRequest和HttpServletResponse参数，Struts 2 Action仍然可以访问它们。但是，大部分时候，Action都无需直接访问HttpServletRequest和HttpServletResponse，从而给开发者更多 灵活的选择。

4 **可测性方面的对比**: 测试Struts 1 Action的一个主要问题 是execute方法依赖于Servlet API，这使得Action的测试要依赖于Web容器。为了脱离Web容器测试Struts 1的Action，必须借助于第三方扩展：Struts TestCase，该扩展下包含了系列的Mock对象(模拟了HttpServletRequest和HttpServletResponse对象)，从而 可以脱离Web容器测试Struts 1的Action类。Struts 2 Action可以通过初始化、设置属性、调用方法来测试。

5 **封装请求参数的对比**: Struts 1使用ActionForm对象封装用户的请求参数，所有的ActionForm必须继承一个基类：ActionForm。普通的JavaBean不能用作ActionForm，因此，开发者必须创建大量的ActionForm类封装用户请求参数。虽然Struts 1提供了动态ActionForm来简化ActionForm的开发，但依然需要在配置文件中定义ActionForm；Struts 2直接使用Action 属性来封装用户请求属性，避免了开发者需要大量开发ActionForm类的烦琐，实际上，这些属性还可以是包含子属性的Rich对象类型。如果开发者依然怀念Struts 1 ActionForm的模式，Struts 2提供了ModelDriven模式，可以让开发者使用单独的Model对象来封装用户请求参数，但该Model对象无需继承任何Struts 2基类，是一个POJO，从而降低了代码污染。

6 **表达式语言方面的对比**: Struts 1整合了JSTL，因此可以使用JSTL表达式语言。这种表达式语言有基本对象图遍历，但在对集合和索引属性的支持上则功能不强；Struts 2可以使用JSTL，但它整合了一种更强大和灵活的表达式语言：OGNL (Object Graph Notation Language)，因此，Struts 2下的表达式语言功能更加强大。

7 **绑定值到视图的对比**: Struts 1使用标准JSP机制把对象绑定到视图页面；Struts 2使用“ValueStack”技术，使标签库能够访问值，而不需要把对象和视图页面绑定在一起。

8 **类型转换的对比**: Struts 1 ActionForm 属性通常都是String类型。Struts 1使用Commons-Beanutils进行类型转换，每个类一个转换器，转换器是不可配置的；Struts 2使用OGNL进行类型转换，支持基本数据类型和常用对象之间的转换。

9 **数据校验的对比**: Struts 1支持在ActionForm重写validate方法中手动校验，或者通过整合Commons-Beanutils validator框架来完成数据校验。Struts 2支持通过重写validate方法进行校验，也支持整合XWork校验框架进行校验。

10 **Action执行控制的对比**: Struts 1支持每一个模块对应一个请求处理（即生命周期的概念），但是模块中的所有Action必须共享相同的生命周期。Struts 2支持通过拦截器堆栈（Interceptor Stacks）为每一个Action创建不同的生命周期。开发者可以根据需要创建相应堆栈，从而和不同的Action一起使用。

struts2 有IoC概念，struts1没有

struts2 每个请求创建一个Action实例，

struts1则都是使用同一个Action实例

struts2 的Action可以是纯POJO，不依赖于Web API（易于测试）

struts1 依赖于 Web API （依赖容器测试）
struts1 的概念用Action ActionForm Mapping 等
struts2 有Action Result （ActionForm已经变成了反射赋值!）

五、MVC流程图和struts1/Strurts2运行机制

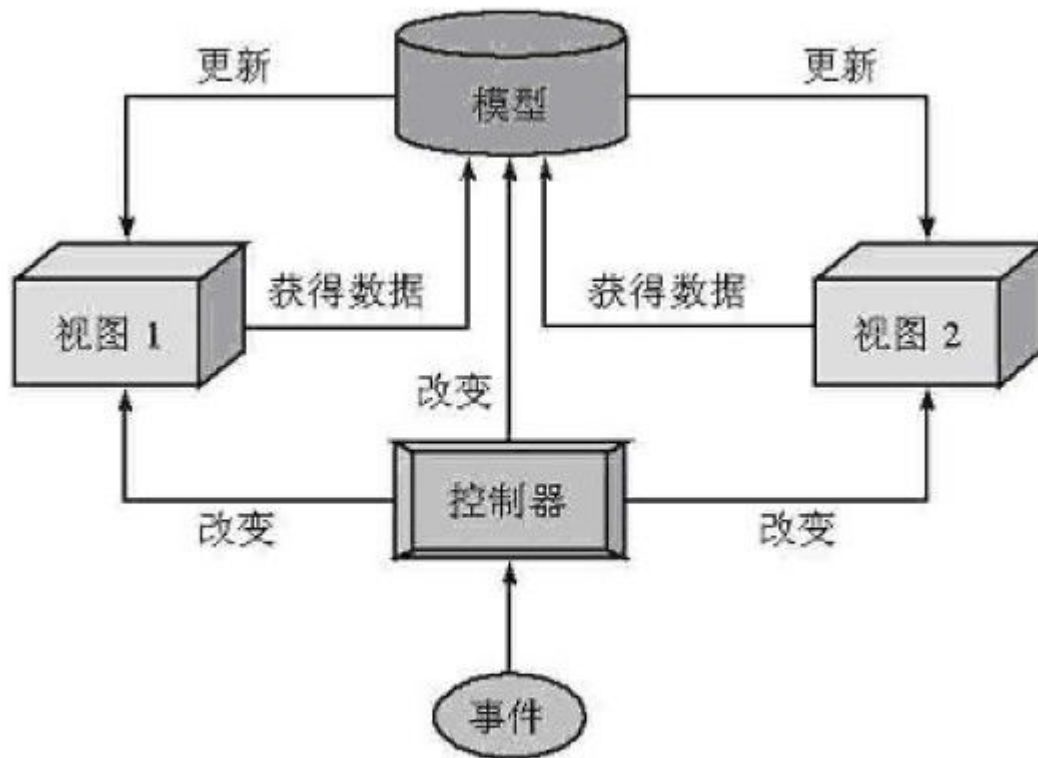


图 1.6 MVC 结构

Struts 1 的程序运行流程如图 1.7 所示。

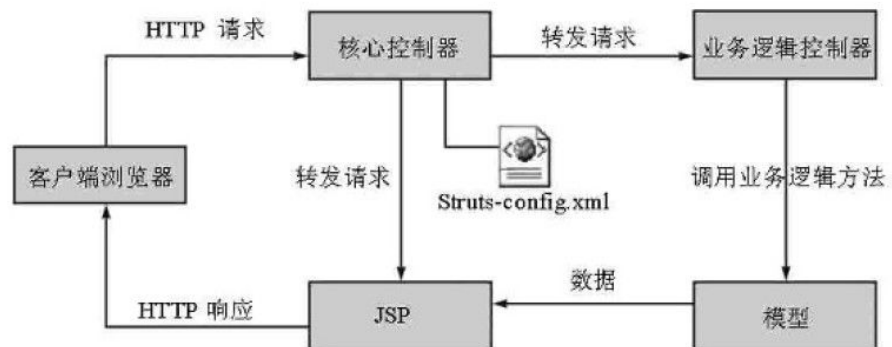
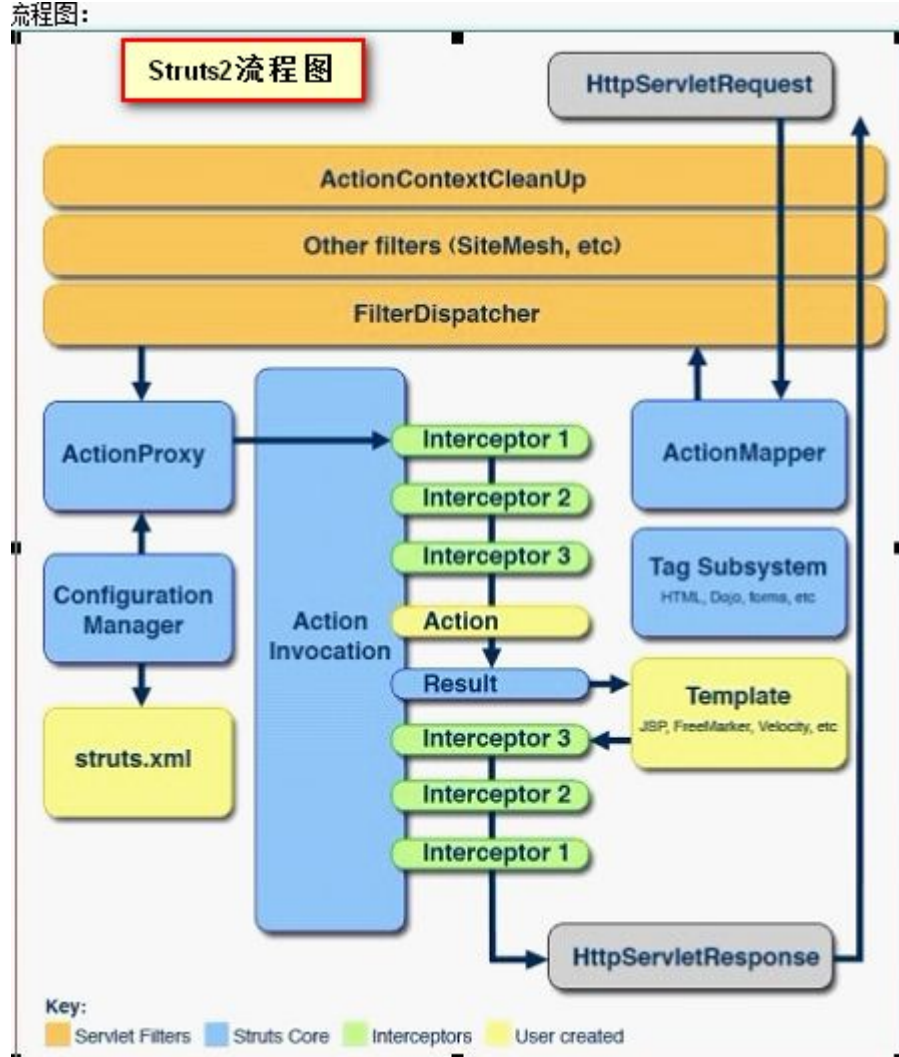


图 1.7 Struts 1 的程序运行流程

流程图:



六、Struts1与Struts2区别 (PS: 这里建议答蓝色字体即可, 由于太多, 我觉得也不用答太多, 能体现他们的区别也差不多啦! 因为实在太多了o(∩_∩)o哈哈~.. 有时候答太多也未必有利. 当然了你能答完应该也很好!)

特性	Struts 1	Struts 2
Action 类	Struts 1要求 Action 类要扩展自一个抽象基类。Struts 1的一个共有的问题是面向抽象类编程而不是面向接口编程。	Struts 2的 Action 类实现了一个 Action 接口, 连同其他接口一起来实现可选择和自定义的服务。Struts 2提供一个名叫 ActionSupport 的基类来实现一般使用的接口。虽然, Action 接口不是必须的。任何使用 execute 方法的 POJO 对象可以被当作 Struts 2的 Action 对象来使用。
线程模型	Struts 1 Action 类是单例类, 因为只有一个实例 控制所有的请求。单例类策略造成了一定的限制并且给开发带来了额外的烦	Struts 2 Action 对象为每一个请求都实例化对象, 所以没有线程安全的问题。(实践中, servlet 容器产生许多丢弃的对象对于每一个请求, 多于一个的对象并不影响垃圾收集)

	恼。Action 资源必须是线程安全或者同步的。	
Servlet 依赖	Struts 1的 Action 类依赖于 servlet API 以为 HttpServletRequest 和 HttpServletResponse 作为参数传给 execute 方法当 Action 被调用时。	Struts 2的 Action 不和容器有关。Servlet 上下文被表现为简单的 Maps，允许 Action 被独立的测试。Struts 2的 Action 可以访问最初的请求和相应，如果需要的话。然而，其他的架构元素减少或者排除直接访问 HttpServletRequest 或者 HttpServletResponse 的需要。
易测性	测试 Struts 1的主要障碍是 execute 方法暴露了 Servlet API。第三方的扩展，Struts 测试用例，提供 Struts 1的集合对象。	Struts 2的 Action 可以通过实例化 Action 来测试，设置属性，然后调用方法。依赖注入的支持也是测试变得更简单。
接受输入	Struts 1使用 ActionForm 对象来捕获输入。象 Action 一样，所有的 ActionForm 必须扩展基类。因为其他的 JavaBean 不能作为 ActionForm 使用，开发者经常创建多余的类来捕获输入。DynaBeans 可以被用来作为替代 ActionForm 的类来创建。但是开发者可以重新描述已经存在的 JavaBean。	Struts 2 Action 属性作为输入属性，排除第二个输入对象的需要。输入属性可能有丰富的对象类型这些类型有他们自己的属性。Action 的属性可以通过标签库来访问。Struts 2也支持 ActionForm 形式。丰富的对象类型，包含业务或者域对象，可以被当作输入或者输出对象来使用。模型驱动特性简化标签对 POJO 输入对象的引用。
表达式语言	Struts 1整合 JSTL，所以它使用 JSTL 的表达式语言。表达式语言有基本的图形对象移动，但是相对很弱的集合和被索引的属性支持。	Struts 2使用 OGNL，但是框架也支持更强大和更灵活的表达式，叫做“对象图形符号语言”（OGNL）。
将值绑定要视图上	Struts 1使用标准 JSP 机制来绑定对象到页面上下文。	Struts 2使用“ValueStack”技术为了标签库可以不用链接你的视图到对象的表现类型来访问值。ValueStack 策略允许重用视图。
类型转换	Struts 1的 ActionForm 属性经常都是 String 的。Struts 1使用 Commons-Beanutils 来类型转换。转换每一个类，并不是为每一个实例配置。	Struts 2使用 OGNL 来类型转换。框架包含转换器来为基本的和共同的对象类型和原始类型。
验证	Struts 1支持手动验证凭借	Struts 2支持手动验证凭借 validate 方法和

	ActionForm 的 validate 方法, 或者通过扩展的公用验证器。类可以有不同验证上下文来未相同的类, 但是不能不能包括验证子对象。	XWork 验证框架。Xwork 验证框架支持一连串的验证子属性使用的验证为了属性类的类型和严正上下文而定义。
Action 执行的控制	Struts 1支持独立的请求处理器对于每一个模型, 但是所有在模型中的 Action 必须共享同一个生命周期。	Struts 2支持在每一个 Action 基础上凭借拦截栈创建不同的生命周期。自定义栈可以被创建并且使用不同的所需 的 Action。

数据库

一、请你用oracle的sql语句删除表的数据,在删除的时候请你用在oracle新建一个tb_A表字段至少要2个以上字段随便写

```
delete from 表名
create table tb_A
(
  Id int , --ID
  Name varchar2(20), ---注意是varchar2() 不是varchar
  sal number(4),      --薪水
);
```

二、删除一个表里面所有重复的Id

```
delete from dbo.t_test where id in ( select id from t_test group by
id having count(id) > 1)
```

说明:自己用数据库新建一个表试一试这个功能看是不是成功 sqlserver2005(我试过是行的) oracle mysql由你们去试一试

例如有个test表,有Id,name 字段,请你用sql查询写最大的8条数据.

```
select top 8 id,name from t_test order by id desc
```

三、请看下面的列表 请你用sql查询写出结果的sql语句

name(姓名)	course(课程)	score
李四	Java	60
李四	c#	60
李四	sqlserver	40
张三	sqlserver	60
张三	Java	60
张三	c#	60

结果:

name	总分
------	----

```
-----
李四                160
张三                180
```

--实现sql

```
select name,sum(score) as 总分 from stu group by name
```

四、将a表的数据 复制到b表请你用sql去实现

答:select * into b from a; --sqlserver

create b table(Id,name) as select Id,name from a;---在oracle数据库的实现

五、用一条 SQL 语句查询出每门课都大于80分的学生姓名

name kecheng fenshu

张三 语文 81

张三 数学 75

李四 语文 76

李四 数学 90

王五 语文 81

王五 数学 100

王五 英语 90

准备数据的 sql 代码:

```
create table score(id int primary key auto_increment,name
varchar(20),subject varchar(20),score int);
```

```
insert into score values
```

```
(null,'张三','语文',81),
```

```
(null,'张三','数学',75),
```

```
(null,'李四','语文',76),
```

```
(null,'李四','数学',90),
```

```
(null,'王五','语文',81),
```

```
(null,'王五','数学',100),
```

```
(null,'王五','英语',90);
```

答案:

A: select distinct name from score where name not in (select distinct name from score where score<=80)

B:select distince name t1 from score where 80< all (select score from score where name=t1);

智力题

【1】假设有一个池塘，里面有无穷多的水。现有2个空水壶，容积分别为5升和6升。问题是如

何只用这2个水壶从池塘里取得3升的水。

【2】周雯的妈妈是豫林水泥厂的化验员。一天，周雯来到化验室做作业。做完后想出去玩。"等等，妈妈还要考你一个题目，"她接着说，"你看这6只做化验用的玻璃杯，前面3只盛满了水，后面3只是空的。你能只移动1只玻璃杯，就使盛满水的杯子和空杯子间隔起来吗?"爱动脑筋的周雯，是学校里有名的"小机灵"，她只想了一会儿就做到了。请你想想看，"小机灵"是怎样做的？

【3】三个小伙子同时爱上了一个姑娘，为了决定他们谁能娶这个姑娘，他们决定用手枪进行一次决斗。小李的命中率是30%，小黄比他好些，命中率是50%，最出色的枪手是小林，他从不失误，命中率是100%。由于这个显而易见的事实，为公平起见，他们决定按这样的顺序：小李先开枪，小黄第二，小林最后。然后这样循环，直到他们只剩下一个人。那么这三个人中谁活下来的机会最大呢？他们都应该采取什么样的策略？

【4】一间囚房里关押着两个犯人。每天监狱都会为这间囚房提供一罐汤，让这两个犯人自己来分。起初，这两个人经常会发生争执，因为他们总是有人认为对方的汤比自己的多。后来他们找到了一个两全其美的办法：一个人分汤，让另一个人先选。于是争端就这么解决了。可是，现在这间囚房里又加进来一个新犯人，现在是三个人来分汤。必须寻找一个新的方法来维持他们之间的和平。该怎么办呢？

按：心理问题，不是逻辑问题

【5】在一张长方形的桌面上放了 n 个一样大小的圆形硬币。这些硬币中可能有一些不完全在桌面内，也可能有一些彼此重叠；当再多放一个硬币而它的圆心在桌面内时，新放的硬币便必定与原先某些硬币重叠。请证明整个桌面可以用 $4n$ 个硬币完全覆盖

【6】一个球、一把长度大约是球的直径 $2/3$ 长度的直尺。你怎样测出球的半径？方法很多，看看谁的比较巧妙

【7】五个大小相同的一元人民币硬币。要求两两相接触，应该怎么摆？

【8】猜牌问题

S 先生、P 先生、Q 先生他们知道桌子的抽屉里有16张扑克牌：红桃 A、Q、4 黑桃 J、8、4、2、7、3 草花 K、Q、5、4、6 方块 A、5。约翰教授从这16张牌中挑出一张牌来，并把这张牌的点数告诉 P 先生，把这张牌的花色告诉 Q 先生。这时，约翰教授问 P 先生和 Q 先生：你们能从已知的点数或花色中推知这张牌是什么牌吗？于是，S 先生听到如下的对话：P 先生：我不知道这张牌。

Q 先生：我知道你不知道这张牌。

P 先生：现在我知道这张牌了。

Q 先生：我也知道了。

听罢以上的对话，S 先生想了一想之后，就正确地推出这张牌是什么牌。

请问：这张牌是什么牌？

【9】一个教授逻辑学的教授，有三个学生，而且三个学生均非常聪明！

一天教授给他们出了一个题，教授在每个人脑门上贴了一张纸条并告诉他们，每个人的纸条上都写了一个正整数，且某两个数的和等于第三个！（每个人可以看见另两个数，但看不见自己的）教授问第一个学生：你能猜出自己的数吗？回答：不能，问第二个，不能，第三个，不能，再问第一个，不能，第二个，不能，第三个：我猜出来了，是144！教授很满意的笑了。请问您能猜出另外两个人的数吗？

【10】某城市发生了一起汽车撞人逃跑事件

该城市只有两种颜色的车,蓝色15% 绿色85%

事发时有一个人 在现场看见了

他指证是蓝车

但是根据专家在现场分析,当时那种条件能看正确的可能性是80%

那么,肇事的车是蓝车的概率到底是多少?

【11】有一人有240公斤水,他想运往干旱地区赚钱。他每次最多携带60公斤,并且每前进一公里须耗水1公斤(均匀耗水)。假设水的价格在出发地为0,以后,与运输路程成正比,(即在10公里处为10元/公斤,在20公里处为20元/公斤.....),又假设他必须安全返回,请问,他最多可赚多少钱?

【12】现在共有100匹马跟100块石头,马分3种,大型马;中型马跟小型马。其中一匹大马一次可以驮3块石头,中型马可以驮2块,而小型马2头可以驮一块石头。问需要多少匹大马,中型马跟小型马?(问题的关键是刚好必须是用完100匹马)

【13】 $1=5$ $2=15$ $3=215$ $4=2145$ 那么 $5=?$

【14】有 $2n$ 个人排队进电影院,票价是50美分。在这 $2n$ 个人当中,其中 n 个人只有50美分,另外 n 个人有1美元(纸票子)。愚蠢的电影院开始卖票时1分钱也没有。

问: 有多少种排队方法 使得 每当一个拥有1美元买票时,电影院都有50美分找钱

注:

1美元=100美分

拥有1美元的人,拥有的是纸币,没法破成2个50美分

【15】一个人花8块钱买了一只鸡,9块钱卖掉了,然后他觉得不划算,花10块钱又买回来了,11块卖给另外一个人。问他赚了多少钱?

【16】有一种体育竞赛共含 M 个项目,有运动员 A, B, C 参加,在每一项目中,第一,第二,第三名分别的 X, Y, Z 分,其中 X, Y, Z 为正整数且 $X > Y > Z$ 。最后 A 得22分, B 与 C 均得9分, B 在百米赛中取得第一。求 M 的值,并问在跳高中谁得第二名。

【17】前提:

- 1 有五栋五种颜色的房子
- 2 每一位房子的主人国籍都不同
- 3 这五个人每人只喝一种饮料,只抽一种牌子的香烟,只养一种宠物
- 4 没有人有相同的宠物,抽相同牌子的香烟,喝相同的饮料

提示:

- 1 英国人住在红房子里
- 2 瑞典人养了一条狗
- 3 丹麦人喝茶
- 4 绿房子在白房子左边
- 5 绿房子主人喝咖啡
- 6 抽PALL MALL烟的人养了一只鸟
- 7 黄房子主人抽DUNHILL烟
- 8 住在中间那间房子的人喝牛奶
- 9 挪威人住第一间房子
- 10 抽混合烟的人住在养猫人的旁边
- 11 养马人住在抽DUNHILL烟的人旁边
- 12 抽BLUE MASTER烟的人喝啤酒
- 13 德国人抽PRINCE烟
- 14 挪威人住在蓝房子旁边
- 15 抽混合烟的人的邻居喝矿泉水

问题是：谁养鱼？？？

【18】5个人来自不同地方，住不同房子，养不同动物，吸不同牌子香烟，喝不同饮料，喜欢不同食物。根据以下线索确定谁是养猫的人。

1. 红房子在蓝房子的右边，白房子的左边（不一定紧邻）
2. 黄房子的主人来自香港，而且他的房子不在最左边。
3. 爱吃比萨的人住在爱喝矿泉水的人的隔壁。
4. 来自北京的人爱喝茅台，住在来自上海的人的隔壁。
5. 吸希尔顿香烟的人住在养马人的右边隔壁。
6. 爱喝啤酒的人也爱吃鸡。
7. 绿房子的人养狗。
8. 爱吃面条的人住在养蛇人的隔壁。
9. 来自天津的人的邻居（紧邻）一个爱吃牛肉，另一个来自成都。
10. 养鱼的人住在最右边的房子里。
11. 吸万宝路香烟的人住在吸希尔顿香烟的人和吸“555”香烟的人的中间（紧邻）
12. 红房子的人爱喝茶。
13. 爱喝葡萄酒的人住在爱吃豆腐的人的右边隔壁。
14. 吸红塔山香烟的人既不住在吸健牌香烟的人的隔壁，也不与来自上海的人相邻。
15. 来自上海的人住在左数第二间房子里。
16. 爱喝矿泉水的人住在最中间的房子里。
17. 爱吃面条的人也爱喝葡萄酒。
18. 吸“555”香烟的人比吸希尔顿香烟的人住的靠右

【19】斗地主附残局

地主手中牌2、K、Q、J、10、9、8、8、6、6、5、5、3、3、3、3、7、7、7、7

长工甲手中牌大王、小王、2、A、K、Q、J、10、Q、J、10、9、8、5、5、4、4

长工乙手中牌2、2、A、A、A、K、K、Q、J、10、9、9、8、6、6、4、4

三家都是明手，互知底牌。要求是：在三家都不打错牌的情况下，地主必须要么输要么赢。

问：哪方会赢？

【20】一楼到十楼的每层电梯门口都放着一颗钻石，钻石大小不一。你乘坐电梯从一楼到十楼，每层楼电梯门都会打开一次，只能拿一次钻石，问怎样才能拿到最大的一颗？

【21】U2合唱团在17分钟内得赶到演唱会场，途中必需跨过一座桥，四个人从桥的同一端出发，你得帮助他们到达另一端，天色很暗，而他们只有一只手电筒。一次同时最多可以有两人一起过桥，而过桥的时候必须持有手电筒，所以就得有人把手电筒带来带去，来回桥两端。手电筒是不能用丢的方式来传递的。四个人的步行速度各不同，若两人同行则以较慢者的速度为准。Bono需花1分钟过桥，Edge需花2分钟过桥，Adam需花5分钟过桥，Larry需花10分钟过桥。他们要如何在17分钟内过桥呢？

【22】一个家庭有两个小孩，其中有一个是女孩，问另一个也是女孩的概率（假定生男生女的概率一样）

【23】为什么下水道的盖子是圆的？

【24】有7克、2克砝码各一个，天平一只，如何只用这些物品三次将140克的盐分成50、90克各一份？

【25】芯片测试：有2k块芯片，已知好芯片比坏芯片多。请设计算法从其中找出一片好芯片，说明你所用的比较次数上限。

其中：好芯片和其它芯片比较时，能正确给出另一块芯片是好还是坏。

坏芯片和其它芯片比较时，会随机的给出好或是坏。

【26】话说有十二个鸡蛋，有一个是坏的（重量与其余鸡蛋不同），现要求用天平称三次，称出哪个鸡蛋是坏的！

【27】100个人回答五道试题，有81人答对第一题，91人答对第二题，85人答对第三题，79人答对第四题，74人答对第五题，答对三道题或三道题以上的人算及格，那么，在这100人中，至少有（ ）人及格。

【28】陈奕迅有首歌叫十年

吕珊有首歌叫3650夜

那现在问,十年可能有多少天?

【29】

1

1 1

2 1

1 2 1 1

1 1 1 2 2 1

下一行是什么？

【30】烧一根不均匀的绳要用一个小时，如何用它来判断半个小时？

烧一根不均匀的绳,从头烧到尾总共需要1个小时。现在有若干条材质相同的绳子,问如何用烧绳的方法来计时一个小时十五分钟呢？（微软的笔试题）

【31】共有三类药，分别重1g,2g,3g，放到若干个瓶子中，现在能确定每个瓶子中只有其中一种药，且每瓶中的药片足够多，能只称一次就知道各个瓶子中都是盛的哪类药吗？

如果有4类药呢？5类呢？N类呢(N可数)？

如果是共有m个瓶子盛着n类药呢(m, n为正整数，药的质量各不相同但各种药的质量已知)？

你能只称一次就知道每瓶的药是什么吗？

注：当然是有代价的，称过的药我们就不用了

【32】假设在桌上有三个密封的盒，一个盒中有2枚银币(1银币=10便士)，一个盒中有2枚镍币(1镍币=5便士)，还有一个盒中有1枚银币和1枚镍币。这些盒子被标上10便士、15便士和20便士，但每个标签都是错误的。允许你从一个盒中拿出1枚硬币放在盒前，看到这枚硬币，你能否说出每个盒内装的东西呢？

【33】有一个大西瓜,用水果刀平整地切,总共切9刀,最多能切成多少份,最少能切成多少份?
主要是过程，结果并不是最重要的

【34】一个巨大的圆形水池，周围布满了老鼠洞。猫追老鼠到水池边，老鼠未来得及进洞就掉入水池里。猫继续沿水池边缘企图捉住老鼠（猫不入水）。已知 $V_{猫}=4V_{鼠}$ 。问老鼠是否有办法摆脱猫的追逐？

【35】有三个桶，两个大的可装8斤的水，一个小的可装3斤的水，现在有16斤水装满了两大桶就是8斤的桶，小桶空着，如何把这16斤水分给4个人，每人4斤。没有其他任何工具，4人自备容器，分出去的水不可再要回来。

【36】从前有一位老钟表匠，为一个教堂装一只大钟。他年老眼花，把长短针装配错了，短针走的速度反而是长针的12倍。装配的时候是上午6点，他把短针指在“6”上，长针指在“12”上。老钟表匠装好就回家去了。人们看这钟一会儿7点，过了不一会儿就8点了，都很奇怪，立刻去找老钟表匠。等老钟表匠赶到，已经是下午7点多钟。他掏出怀表来一对，钟准确无误，疑心人们有意捉弄他，一生气就回去了。这钟还是8点、9点地跑，人们再去找钟表匠。老钟表匠第二天早晨8点多赶来用表一对，仍旧准确无误。请你想一想，老钟表匠第一次对表的时候是7点几分？第二次对表又是8点几分？

【37】今有2匹马、3头牛和4只羊，它们各自的总价都不满10000文钱（古时的货币单位）。如果

2匹马加上1头牛，或者3 头牛加上1只羊，或者4只羊加上1匹马，那么它们各自的总价都正好是10000文钱了。问：马、牛、羊的单价各是多少文钱？

【38】一天，harlan 的 店里来了一位顾客，挑了25元的货，顾客拿出100元，harlan 没零钱找不开，就到隔壁飞白的店里把这100元换成零钱，回来给顾客找了75元零钱。 过一会，飞白来找 harlan，说刚才的是假钱，harlan 马上给飞白换了张真钱，问 harlan 赔了多少钱？

【39】猴子爬绳

这道力学怪题乍看非常简单，可是据说它却使刘易斯·卡罗尔感到困惑。至于这道怪题是否由这位因《爱丽丝漫游奇境记》而闻名的牛津大学数学专家提出来的，那就不清楚了。总之，在一个不走运的时刻，他就下述问题征询人们的意见：

一根绳子穿过无摩擦力的滑轮，在其一端悬挂着一只10磅重的砝码，绳子的另一端有只猴子，同砝码正好取得平衡。当猴子开始向上爬时，砝码将如何动作呢？

"真奇怪，"卡罗尔写道，"许多优秀的数学家给出了截然不同的答案。普赖斯认为砝码将向上升，而且速度越来越快。克利夫顿(还有哈考特)则认为，砝码将以与猴子一样的速度向上升起，然而桑普森却说，砝码将会向下降!"

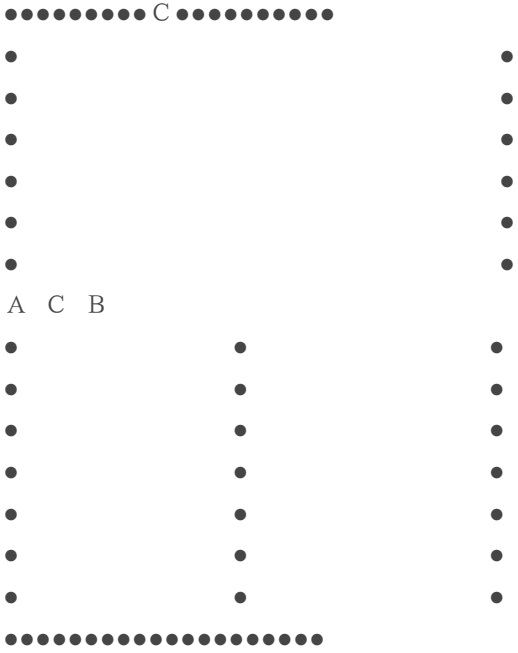
一位杰出的机械工程师说"这不会比苍蝇在绳子上爬更起作用"，而一位科学家却认为"砝码的上升或下降将取决于猴子 吃苹果速度的倒数"，然而还得从中求出猴子尾巴的平方根。严肃地说，这道题目非常有趣，值得认真推敲。它很能说明趣题与力学问题之间的紧密联系。

【40】两个空心球，大小及重量相同，但材料不同。一个是金，一个是铅。空心球表面图有相同颜色的油漆。现在要求在不破坏表面油漆的条件下用简易方法指出哪个是金的，哪个是铅的。

【41】有23枚硬币在桌上，10枚正面朝上。假设别人蒙住你的眼睛，而你的手又摸不出硬币的反正面。让你用最好的方法把这些硬币分成两堆，每堆正面朝上的硬币个数相同。

【42】三个村庄 A、B、C 和三个城镇 A、B、C 坐落在如图所示的环形山内。

由于历史原因，只有同名的村与镇之间才有来往。为方便交通，他们准备修铁路。问题是：如何在这个环形山内修三条铁路连通 A 村与 A 镇，B 村与 B 镇，C 村与 C 镇。而这些铁路相互不能相交。（挖山洞、修立交桥都不算，绝对是平面问题）。想出答案再想想这个题说明什么问题。



【43】屋里三盏灯,屋外三个开关,一个开关仅控制一盏灯,屋外看不到屋里

怎样只进屋一次,就知道哪个开关控制哪盏灯?

四盏呢~

【44】 $2+7-2+7$ 全部有火柴根组成,移动其中任何一根,答案要求为30

说明:因为书写问题作如下解释,2是由横折横三根组成,7是由横折两根组成

【45】5名海盗抢得了窖藏的100块金子,并打算瓜分这些战利品。这是一些讲民主的海盗(当然是他们自己特有的民主),他们的习惯

是按下面的方式进行分配:最厉害的一名海盗提出分配方案,然后所有的海盗(包括提出方案者本人)就此方案进行表决。如果50%或更多的海盗赞同此方案,此方案就获得通过并据此分配战利品。否则提出方案的海盗将被扔到海里,然后下一名最厉害的海盗又重复上述过程。

所有的海盗都乐于看到他们的一位同伙被扔进海里,不过,如果让他们选择的话,他们还是宁可得一笔现金。他们当然也不愿意自己被扔到海里。所有的海盗都是有理性的,而且知道其他的海盗也是有理性的。此外,没有两名海盗是同等厉害的——这些海盗按照完全由上到下的等级排好了座次,并且每个人都清楚自己和其他所有人的等级。这些金块不能再分,也不允许几名海盗共有金块,因为任何海盗都不相信他的同伙会遵守关于共享金块的安排。这是一伙每人都只为自己打算的海盗。

最凶的一名海盗应当提出什么样的分配方案才能使他获得最多的金子呢?

【46】他们中谁的存活机率最大?

5个囚犯,分别按1-5号在装有100颗绿豆的麻袋抓绿豆,规定每人至少抓一颗,而抓得最多和最少的人将被处死,而且,他们之间不能交流,但在抓的时候,可以摸出剩下的豆子数。问他们中谁的存活几率最大?提示:

- 1, 他们都是很聪明的人
- 2, 他们的原则是先求保命,再去多杀人
- 3, 100颗不必都分完
- 4, 若有重复的情况,则也算最大或最小,一并处死

【47】有5只猴子在海边发现一堆桃子,决定第二天来平分.第二天清晨,第一只猴子最早来到,它左分右分分不开,就朝海里扔了一只,恰好可以分成5份,它拿上自己的一份走了.第2,3,4,5只猴子也遇到同样的问题,采用了同样的方法,都是扔掉一只后,恰好可以分成5份.问这堆桃子至少有多少只?

【48】话说某天一艘海盗船被天下砸下来的一头牛给击中了,5个倒霉的家伙只好逃难到一个孤岛,发现岛上孤零零的,幸好有棵椰子树,还有一只猴子!

大家把椰子全部采摘下来放在一起,但是天已经很晚了,所以就睡觉先.

晚上某个家伙悄悄的起床,悄悄的将椰子分成5份,结果发现多一个椰子,顺手就给了幸运的猴子,然后又悄悄的藏了一份,然后把剩下的椰子混在一起放回原处,最后还是悄悄滴回去睡觉了.

过了会儿,另一个家伙也悄悄的起床,悄悄的将剩下的椰子分成5份,结果发现多一个椰子,顺手就又给了幸运的猴子,然后又悄悄滴藏了一份,把剩下的椰子混在一起放回原处,最后还是悄悄滴回去睡觉了.

又过了一会 ...

...

又过了一会 ...

总之5个家伙都起床过,都做了一样的事情

早上大家都起床,各自心怀鬼胎的分椰子了,这个猴子还真不是一般的幸运,因为这次把椰子分成5分后居然还是多一个椰子,只好又给它了.

问题来了,这堆椰子最少有多少个?

【49】小明和小强都是张老师的学生,张老师的生日是 M 月 N 日,

2人 都知道张老师的生日是下列10组中的一天,

张老师把 M 值告诉了小明,把 N 值告诉了小强,

张老师问他们知道他的生日是哪一天吗?

3月4日 3月5日 3月8日

6月4日 6月7日

9月1日 9月5日

12月1日 12月2日 12月8日

小明说:如果我不知道的话,小强肯定也不知道

小强说:本来我也不知道,但是现在我知道了

小明说:哦,那我也知道了

请根据以上对话推断出张老师的生日是哪一天

【50】一逻辑学家误入某部落,被囚于牢狱,酋长欲意放行,他对逻辑学家说:“今有两门,一为自由,一为死亡,你可任意开启一门。现从两个战士中选择一人负责解答你所提的任何一个问题(Y/N),其中一个天性诚实,一人说谎成性,今后生死任你选择。”逻辑学家沉思片刻,即向一战士发问,然后开门从容离去。逻辑学家应如何发问?

【51】说从前啊,有一个富人,他有30个孩子,其中15个是已故的前妻所生,其余15个是继室所生,这后一个妇人很想让她自己所生的最年长的儿子继承财产,于是,有一天,他就向他 说:"亲爱的丈夫啊,你就要老了,我们应该定下来谁将是你的继承人,让我们把我们的30个孩子排成一个圆圈,从他们中的一个数起,每每到10就让那个孩子 站出来,直到最后剩下哪个孩子,哪个孩子就继承你的财产吧!"富人一想,我靠,这个题意相当有内涵了,不错,仿佛很公平,就这么办吧~不过,当剔除过程不断进行下去的时候,这个富人傻眼了,他发现前14个被剔除的孩子都是前妻生的,而且下一个要被剔除的还是前妻生的,富人马上大手一挥,停,现在从这个孩子 倒回去数,继室,就是这个歹毒的后妈一想,倒数就倒数,我15个儿子还斗不过你一个啊~她立即同意了富人的动议,你猜,到底谁做了继承人呢~

【52】“有一牧场,已知养牛27头,6天把草吃尽;养牛23头,9天把草吃尽。如果养牛21头,那么几天能把牧场上的草吃尽呢?并且牧场上的草是不断生长的。”

【53】一个商人骑一头驴要穿越1000公里长的沙漠,去卖3000根胡萝卜。已知驴一次性可驮1000根胡萝卜,但每走一公里又要吃掉一根胡萝卜。问:商人共可卖出多少胡萝卜?

【54】10箱黄金,每箱100块,每块一两

有贪官,把某一箱的每块都磨去一钱

请称一次找到不足量的那个箱子

【55】你让工人为你工作7天,给工人的回报是一根金条。金条平分成相连的7段,你必须在每天结束时都付费,如果只许你两次把金条弄断,你如何给你的工人付费?

【56】有十瓶药,每瓶里都装有100片药(仿佛现在装一百片的少了,都是十片二十片的,不管,咱们就这么来了),其中有八瓶里的药每片重10克,另有两瓶里的药每片重9克。用一个蛮精确的小秤,只称一次,如何找出份量较轻的那两个药瓶?

【57】一个经理有三个女儿,三个女儿的年龄加起来等于13,三个女儿的年龄乘起来等于经理自己的年龄,有一个下属已知道经理的年龄,但仍不能确定经理三个女儿的年龄,这时经理说只有一个女儿的头发是黑的,然后这个下属就知道了经理三个女儿的年龄。请问三个女儿的年龄分别是多少?为什么?

【58】有三个人去住旅馆,住三间房,每一间房\$10元,于是他们一共付给老板\$30,第二天,老板觉得三间房只需要\$25元就够了于是叫小弟退回\$5给三位客人,谁知小弟贪心,只退 回每人

\$1，自己偷偷拿了\$2，这样一来便等于那三位客人每人各花了九元，于是三个人一共花了\$27，再加上小弟独吞了不\$2，总共是\$29。可是当初他们三个人一共付出\$30那么还有\$1呢？

【59】有两位盲人，他们都各自买了两对黑袜和两对白袜，八对袜了的布质、大小完全相同，而每对袜了都有一张商标纸连着。两位盲人不小心将八对袜了混在一起。他们每人怎样才能取回黑袜和白袜各两对呢？

【60】有一辆火车以每小时 15公里的速度离开洛杉矶直奔纽约，另一辆火车以每小时20公里的速度从纽约开往洛杉矶。如果有一只鸟，以30公里每小时的速度和两辆火车同时启动，从洛杉矶出发，碰到另一辆车后返回，依次在两辆火车来回飞行，直到两辆火车相遇，请问，这只小鸟飞行了多远距离？

【61】你有两个罐子，50个红色弹球，50个蓝色弹球，随机选出一个罐子，随机选取出一个弹球放入罐子，怎么给红色弹球最大的选中机会？在你的计划中，得到红球的准确几率是多少？

【62】你有四个装药丸的罐子，每个药丸都有一定的重量，被污染的药丸是没被污染的重量+1.只称量一次，如何判断哪个罐子的药被污染了？

【63】对一批编号为1~100，全部开关朝上(开)的灯进行以下*作：凡是1的倍数反方向拨一次开关；2的倍数反方向又拨一次开关；3的倍数反方向又拨一次开关.....问：最后为关闭状态的灯的编号。

【64】想象你在镜子前，请问，为什么镜子中的影像可以颠倒左右，却不能颠倒上下？

【65】一群人开舞会，每人头上都戴着一顶帽子。帽子只有黑白两种，黑的至少有一顶。每个人都能看到其它人帽子的颜色，却看不到自己的。主持人先让大家看看别人头上戴的是什么帽子，然后关灯，如果有人认为自己戴的是黑帽子，就打自己一个耳光。第一次关灯，没有声音。于是再开灯，大家再看一遍，关灯时仍然鸦雀无声。一直到第三次关灯，才有劈劈啪啪打耳光的声音响起。问有多少人戴着黑帽子？

【66】两个圆环，半径分别是1和2，小圆在大圆内部绕大圆圆周一周，问小圆自身转了几周？如果在大圆的外部，小圆自身转几周呢？

【67】 1元钱一瓶汽水，喝完后两个空瓶换一瓶汽水，问：你有20元钱，最多可以喝到几瓶汽水？

【68】有3顶红帽子，4顶黑帽子，5顶白帽子。让10个人从矮到高站成一队，给他们每个人头上戴一顶帽子。每个人都看不见自己戴的帽子的颜色，却只能看见站在前面那些人的帽子颜色。（所以最后一个人可以看见前面9个人头上帽子的颜色，而最前面那个人谁的帽子都看不见。现在从最后那个人开始，问他是不是知道自己戴的帽子颜色，如果他回答说不知道，就继续问他前面那个人。假设最前面那个人一定会知道自己戴的是黑帽子。为什么？

【69】假设排列着100个乒乓球，由两个人轮流拿球装入口袋，能拿到第100个乒乓球的人为胜利者。条件是：每次拿球者至少要拿1个，但最多不能超过5个，问：如果你是最先拿球的人，你该拿几个？以后怎么拿就能保证你能得到第100个乒乓球？

【70】卢姆教授说：“有一次 我目击了两只山羊的一场殊死决斗，结果引出了一个有趣的数学问题。我的一位邻居有一只山羊，重54磅，它已有好几个季度在附近山区称王称霸。后来某个好事之徒引进了一只新的山羊，比它还要重出3磅。开始时，它们相安无事，彼此和谐相处。可是有一天，较轻的那只山羊站在陡峭的山路顶上，向它的竞争对手猛扑过去，那对手站在土丘上迎接挑战，而挑战者显然拥有居高临下的优势。不幸的是，由于猛烈碰撞，两只山羊都一命呜呼了。

现在要讲一讲本题的奇妙之处。对饲养山羊颇有研究，还写过书的乔治·阿伯克龙比说道：“通过反复实验，我发现，动量相当于一个自20英尺高处坠落下来的30磅重物的一次撞击，正好可以打碎山羊的脑壳，致它死命。”如果他说得不错，那么这两只山羊至少要有多大的逼近速度，才能相互撞破脑壳？你能算出来吗？

【71】据说有人给酒肆的老板娘出了一个难题：此人明明知道店里只有两个舀酒的勺子，分别能舀7两和11两酒，却硬要老板娘卖给他2两酒。聪明的老板娘毫不含糊，用这两个勺子在酒缸里舀酒，并倒来倒去，居然量出了2两酒，聪明的你能做到吗？

【72】已知： 每个飞机只有一个油箱， 飞机之间可以相互加油（注意是相互，没有加油机） 一箱油可供一架飞机绕地球飞半圈，问题：为使至少一架飞机绕地球一圈回到起飞时的飞机场，至少需要出动几架飞机？（所有飞机从同一机场起飞，而且必须安全 返回机场，不允许中途降落，中间没有飞机场）

【73】在9个点上画10条直线，要求每条直线上至少有三个点？

【74】一个岔路口分别通向诚实国和说谎国。来了两个人，已知一个是诚实国的，另一个是说谎国的。诚实国永远说实话，说谎国永远说谎话。现在你要去说谎国，但不知道应该走哪条路，需要问这两个人。请问应该怎么问？

【75】在一天的24小时之中，时钟的时针、分针和秒针完全重合在一起的时候有几次？都分别是什么时间？你怎样算出来的？

=====我是傻 X 般的分割线=====



=====我是傻 X 般的分割线=====

答案：

【1】

- 1、先把5升的灌满，倒在6升里，这时6升的壶里有5升水
- 2.再把5升的灌满，用5升的壶把6升的灌满，这时5升的壶里剩4升水
- 3.把6升的水倒掉，再把5升壶里剩余的水倒入6升的壶里，这时6升的壶里有4升水
- 4.把5升壶灌满，倒入6升的壶， $5-2=3$

【2】

把第二个满着的杯子里的水倒到第五个空着的杯子里

【3】

小黄。因为小李是第一个出手的，他要解决的第一个人就会是小林，这样就会保证自己的安全，因为如果小黄被解决，自己理所当然地会成为小林的目标，他也必定会被打死。而小黄如果第一枪不打小林而去打小李，自己肯定会死（他命中较高，会成为接下来的神枪手小林的目标）。他必定去尝试先打死小林。那么30% 50%的几率是80%（第一回合小林的死亡率，但会有一点点偏差，毕竟相加了）。那么第一回合小黄的死亡率是20%多一点点（小林的命中减去自己的死亡率）。假设小林第一回合死了，就轮到小李打小黄了，那么小李的命中就变成了50%多一点点（自己的命中加上小黄的死亡率）。这样就变成了小李小黄对决，第二回合的小李的第一枪命中是50%，小黄也是。可是如果拖下去的话占上风的自然就是小黄

了，可能赢得也自然是小黄了。至于策略我看大家都领悟了吧。

【4】

甲分三碗汤，乙选认为最多和最少的倒回灌里再平分到剩余的两个碗里，让丁先选，其次是甲，最后是乙

【5】

假如先前 N 个中没有重叠且边上的都超出桌子的边上且全都是紧靠着的.那么根据题意就可以有:空隙个数 $Y=3N/2-3$ (自己推算)

每一个空都要一个圆来盖

桌面就一共有圆的数为:

$$Y+N=3N/2-3$$

$$=5N/2-3 \leq 4N \text{ (除 } N=1 \text{ 外)}$$

所以可以用 $4N$ 个硬币完全覆盖.

【6】

用绳子围球一周后测绳长来计算半径（用纸筒套住球来测更准）

借助排水法测体积后计算半径

【7】

要两人才能做到，

先在平面上摆放一枚，再在这枚硬币的正面立着放两枚（这两枚是侧面接触的），这样，这三枚硬币之间形成一个三角形空隙。剩下的两枚在空隙处交叉就行了，注意这两枚同样是平躺着，但可能需要翘起一定的角度。

【8】

方块5

【9】

经过第一轮，说明任何两个数都是不同的。第二轮，前两个人没有猜出，说明任何一个数都不是其它数的两倍。现在有了以下几个条件：1.每个数大于0 2.两两不等 3.任意一个数不是其他数的两倍。每个数字可能是另两个之和或之差，第三个人能猜出144，必然根据前面三个条件排除了其中的一种可能。假设：是两个数之差，即 $x-y=144$ 。这时1（ $x, y>0$ ）和2（ $x \neq y$ ）都满足，所以要否定 $x+y$ 必然要使3不满足，即 $x+y=2y$ ，解得 $x=y$ ，不成立（不然第一轮就可猜出），所以不是两数之差。因此是两数之和，即 $x+y=144$ 。同理，这时1, 2都满足，必然要使3不满足，即 $x-y=2y$ ，两方程联立，可得 $x=108, y=36$ 。

这两轮猜的顺序其实分别为这样：第一轮（一号，二号），第二轮（三号，一号，二号）。这样分大家在每轮结束时获得的信息是相同的（即前面的三个条件）。

那么就假设我们是 C，来看看 C 是怎么做出来的：C 看到的是 A 的36和 B 的108，因为条件，两个数的和是第三个，那么自己要么是72要么是144（猜到这个是因为72的话，108就是36和72的和，144的话就是108和36的和。这样子这句话看不懂的举手）：

假设自己（C）是72的话，那么 B 在第二回合的时候就可以看出来，下面是如果 C 是72，B 的思路：这种情况下，B 看到的的就是 A 的36和 C 的72，那么他就可以猜自己，是36或者是108（猜到这个是因为36的话，36加36等于72，108的话就是36和108的和）：

如果假设自己（B）头上是36，那么，C 在第一回合的时候就可以看出来，下面是如果 B 是36，C 的思路：这种情况下，C 看到的的就是 A 的36和 B 的36，那么他就可以猜自己，是72或者是0（这个不再解释了）：

如果假设自己（C）头上是0，那么，A 在第一回合的时候就可以看出来，下面是如果 C 是0，A 的思路：这种情况下，A 看到的的就是 B 的36和 C 的0，那么他就可以猜自己，是36或者是36（这个不再解释了），那他可以一口报出自己头上的36。（然后是逆推逆推逆推），现在 A 在第一回合

没报出自己的36, C (在 B 的想象中) 就可以知道自己头上不是0, 如果其他和 B 的想法一样 (指 B 头上是36), 那么 C 在第一回合就可以报出自己的72。现在 C 在第一回合没报出自己的36, B (在 C 的想象中) 就可以知道自己头上不是36, 如果其他和 C 的想法一样 (指 C 头上是72), 那么 B 在第二回合就可以报出自己的108。现在 B 在第二回合没报出自己的108, C 就可以知道自己头上不是72, 那么 C 头上的唯一可能就是144了。

【10】

$15\% \times 80\% / (85\% \times 20\% + 15\% \times 80\%)$

【11】

$f(x) = (60 - 2x) \times x$, 当 $x = 15$ 时, 有最大值450。

1820元设是 X 公里处赚最多钱。问题就成是求一个一元二次方程的最大值, 求得是在15公里处赚钱最多, 450元。一共240公斤.....

【12】

6种结果

大、中、小: $(2 \setminus 30 \setminus 68)(5 \setminus 25 \setminus 70)(8 \setminus 20 \setminus 72)(11 \setminus 15 \setminus 74)(14 \setminus 10 \setminus 76)(17 \setminus 5 \setminus 78)$

【13】

因为 $1=5$, 所以 $5=1$

【14】

本题可用递归算法, 但时间复杂度为 2^n 的 n 次方, 也可以用动态规划法, 时间复杂度为 n 的平方, 实现起来相对要简单得多, 但最方便的就是直接运用公式: 排队的种数 $= (2n)! / [n!(n-1)!]$ 。

如果不考虑电影院能否找钱, 那么一共有 $(2n)! / [n!n!]$ 种排队方法 (即从 $2n$ 个人中取出 n 个人的组合数), 对于每一种排队方法, 如果他会导致电影院无法找钱, 则称为不合格的, 这种的排队方法有 $(2n)! / [(n-1)!(n-1)!]$ (从 $2n$ 个人中取出 $n-1$ 个人的组合数) 种, 所以合格的排队种数就是 $(2n)! / [n!n!] - (2n)! / [(n-1)!(n-1)!] = (2n)! / [n!(n-1)!]$ 。至于为什么不合格数是 $(2n)! / [(n-1)!(n-1)!]$, 说起来太复杂, 这里就不讲了。

【15】

2元

【16】

M=5 C 得第二名

因为 ABC 三人得分共 40 分, 三名得分都为正整数且不等, 所以前三名得分最少为 6 分, $40 = 5 \times 8 = 4 \times 10 = 2 \times 20 = 1 \times 20$, 不难得出项目数只能是 5. 即 $M = 5$.

A 得分为 22 分, 共 5 项, 所以每项第一名得分只能是 5, 故 A 应得 4 个第一名一个第二名. $22 = 5 \times 4 + 2$, 第二名得 2 分, 又 B 百米得第一, $9 = 5 + 1 + 1 + 1 + 1$ 所以跳高中只有 C 得第二名

B 的 5 项共 9 分, 其中百米第一 5 分, 其它 4 项全是 1 分, $9 = 5 + 1 + 1 + 1 + 1$. 即 B 除百米第一外全是第三, 跳高第二必定是 C 所得

【17】

房子 黄 蓝 红 绿 白

国籍 挪威 丹麦 英国 德国 瑞士

饮料 矿泉水 茶 牛奶 咖啡 啤酒

宠物 猫 马 鸟 鱼 狗

香烟 DUNHILL 混合烟 PALLMALL PRINCE BLUE MASTER

【18】

1	2	3	4	5
蓝房子	绿	黄	红	白
北京人	上海	香港	天津	成都
茅台酒	葡萄	矿泉水	茶	啤酒
豆腐	面条	牛肉	比萨	鸡
健牌	希尔顿	万宝路	555	红塔山
马	狗	蛇	猫	鱼

【19】

A 家先打：55

B 家如果打：TT 的话。

C 家随便他吃不吃..

A 家都不跟。（反正 B 家跟 C 家哪家有吃55的话，都不跟。除非 A 家88可以出就跟）

如果刚才是 B 家吃的话，就 B 家出牌：你看。B 家最多也出44然后 C 家吃他66。如果他是出两个99那地主也不跟！；如果 B 家出单的话。地主还有一个2可以压！（反正 B 家跟 C 家肯定会打对子的！）

照刚才那样。A 家牌下面应该剩：2 K Q J T 9 7777 66 3333

B 家：大王 小王 2 A K Q Q J J 9 8 55

C 家：22 AAA K Q J T 99 8 44

A 家吃完88后。B 家吃 JJ（反正无论如何。都会打单的。）要是打单的话。A 家就用2压。B 家双王不可能会压吧。（即使压了也没事。）

A 家用2压完后就打：K Q J T 9

B 家如果用双王吃的话。那等他出牌的时候。马上用3333吃他。如果 B 家没吃的话。C 家会吃：A K Q J T

然后 A 家可以用3333压下 A K Q J T 如果 B 家用双王吃的话。那正合我意了哈。！A 家反正只剩下7777 66了等他打什么.. 都用7777吃他。最后打66

【20】

先拿下第一楼的钻石，然后在每一楼把手中的钻石与那一楼的钻石相比较，如果那一楼的钻石比手中的钻石大的话那就把手中的钻石换成那一层的钻石。

（因为“只能拿一次”是在外文翻译过来的，所以是总共只能拿一次，还是每层只能拿一次？无法知道。但如果这个和“在稻田一直走，不能回头，请你检出最大的一个稻穗”这样的题目一样的话，那么上面的就是正确答案！）