1、static的作用，static修饰方法和变量分别表示什么？

1） a、隐藏

b、static保持变量内容的持久

c、默认初始化为0

2）方法是静态方法，变量是静态变量

2、string类可不可以被继承？

不可以

3、一般用哪个类装字符串？string buffer和stringbuilder的区别？

1）String

2) String 没有预留任何缓冲空间

StringBuffer 底层大量使用了synchronized修饰符

同一时间只允许一个线程进行操作

效率相对较低 但是不会出现并发错误

StringBuilder 同一时间允许多个线程同时进行操作

效率较高 但是可能出现并发错误

4、hashCode方法可不可以覆盖？怎么用？

可以, public int hashCode() {

int hash = hashCode;

if (hash == 0) {

if (count == 0) {

return 0;

}

final int end = count + offset;

final char[] chars = value;

for (int i = offset; i < end; ++i) {

hash = 31\*hash + chars[i];

}

hashCode = hash;

}

return hash;

}

5、java的基本数据类型？char类型可不可以装汉字？

1)byte short int long float double char boolean

2)可以

6、线程怎么创建？线程池有哪些？

extends Thread（） 覆盖run（）

implements Runable（） 覆盖run（）

implements Callable<X>（） 覆盖call（）

\*:弥补了原本run()返回类型定义为void

方法签名后有throws Exception

java.util.concurrent.\*;

ExecutorService es = Executors.newFixedThreadPool(3);

. .newCachedThreadPool();

. .newSingleThreadExecutor();

7、hashMap和hash table的区别？他们的接口是什么？

1.同步特性不同

HashMap 同一时间允许多个线程同时进行操作 效率较高 但是可能出现并发错误

Hashtable 同一时间只允许一个线程进行操作 效率较低 但是不会出现并发错误

\*:JDK5.0开始 Collections.synchronizedMap(hashMap);

2.对于null的处理不同

HashMap 无论主键还是值 都是可以存放null的 只不过主键要求唯一

所以只能存放一个null

Hashtable 无论主键还是值 都不允许存放null 会出现异常NullPointerException

3.底层实现的细节不同

HashMap 默认分为16个小组 分组组数可以指定 但是最终结果一定是2的n次方

正因为分组组数都是2的n次方 所以可以使用 &(分组组数-1)来计算散列

Hashtable 默认分为11个小组 分组组数完全可以随意指定

正因为如此 它不能使用&(分组组数-1) 而只能使用%分组组数

4.出现的版本不同

HashMap since JDK1.2

Hashtable since JDK1.0

8、判断一般用什么方法？哪个方法效率高？switch case的参数类型？

==和equals ，== String

9、monggoDB用过吗？redis用过吗？redis可以存在txt文件中么？？

没有 用过 ，可以

10、MyBatis和Hibernate和JDBC的区别？mybatis怎么用，怎么搭建？

a、Hibernate自动生成SQL语句，并且自动执行SQL语句，并且自动返回对应的结果。不灵活，走自动生成，效率最低。

b、MyBATIS SQL语句需要自己写，SQL开发工作量大了。移植性差。但是灵活。

c、JDBC所有的SQL语句都必须自己写，自动手动执行，自动手动返回结果，最快，冗余，重复代码多

11、JS中怎么选取一个id、一个class元素？

document.getElementById(IDname)

document.getElementsByClassName('ClassName')

12、在页面中用js和Jquery哪个多一些？

jquery多一些

13、JS中read方法的作用？

14、JSP页面怎么提高加载效率？

15、StringMVC的注解有哪些？request mapping注解是干什么用的？

16、bean的作用域？singleton和prototype的作用和区别

17、maven工程怎么打包、怎么删除、怎么编译、怎么安装、怎么测试？maven工程怎么避免test包在主程序之前运行？

18、final的作用，final修饰的类、方法、变量表示什么？

19、集合的接口有哪些？

collection

20、Error和Exception的区别？他们的父接口是什么？

Error（错误）表示系统级的错误和程序不必处理的异常，是java运行环境中的内部错误或者硬件问题

Exception（违例）表示需要捕捉或者需要程序进行处理的异常，它处理的是因为程序设计的瑕疵而引起的问题或者在外的输入等引起的一般性问题，是程序必须处理的

21、Hibernate和Mybatis区别？

22、Hibernate:get和load区别？

23、Hibernate三种状态？

24、Hibernate怎么处理大数据？

答：Query接口中 scroll()方法，可以前后项遍历

25、mybatis反向生成，1.集成插件 2.代码方式

26、mybatis动态标签,例如 <if test=""></if>

27、struts2和springmvc区别？

28、struts2配置文件中<package name="" namespace=""> name和namespace有什么不同

29、struts2工作原理

30、struts2模型驱动 实现ModelDriven 或者可以在 <input name="user.id">

31、Webservice（soap协议） CXF是http+xml请求 还自带一个rest rest就是http请求 还有个协议叫restful restful有单独的框架

32、redis了解一下

33、oracle和mysql区别？ 两个数据库的分页 两个数据库的读写分离 数据库怎么水平分表和垂直分表 水平分表是重点

4种事物特点 存储过程、函数、触发器的创建方式和区别

34、spring七种事物、4种隔离机制（spring使用mysql默认隔离机制是什么；oracle默认隔离机制是什么。）

35、tomcat怎么改端口号和编码；tomcat怎么配置jndi；tomcat内存溢出怎么配置参数；tomcat怎么配置启动时间（也叫超时）

36、java设计模式

37、了解json spring有个json包 有个类ObjectMapper

38、数据库编译异常有哪些

答：1.数据库创建表列的时侯编译出错

2.SQL语句编译出错

39、数据文件、日志文件、临时文件、控制文件

40、数据库的隔离级别、数据库脏读 幻读

41、redis 的应用场景（什么时候用） 在redis中如何实现session共享

42、你对现在IT行业中时时出现的新技术的看法

答：新技术层出不穷，是IT这个领域的常态。几乎每天都会诞生一种，据说会改变世界的革命性技术。对于程序员来说，终生学习也注定是常态。 适者生存

43、了解过中间件？

44、九大内置对象，作用

45、用java写一个等腰三角形

0

1 1

2 2 2

3 3 3 3

46.数据库优化方案 至少写出5个

答：创建索引了 sql语句优化 数据量大了分表（水平、垂直）、分布式读写分离

47、 一道类似的题，以下是举例子

国家（country） 性别（sex） 人口（population）

中国 1 340

中国 2 260

美国 1 45

美国 2 55

加拿大 1 51

加拿大 2 49

英国 1 40

英国 2 60

按照国家和性别进行分组，得出结果如下

国家 男 女

中国 340 260

美国 45 55

加拿大 51 49

英国 40 60

答：SELECT country,

SUM( CASE WHEN sex = '1' THEN population ELSE 0 END), --男性人口

SUM( CASE WHEN sex = '2' THEN population ELSE 0 END) --女性人口

FROM Table\_A GROUP BY country;

48、说说oracle里的高级函数 也就是 分析函数

答：1.row\_number over()

2.sum over()

3.start with connect by

4.SYS\_CONNECT\_BY\_PATH()

49、oracle和mysql执行计划

50、UTF-8一些编码的概念

51、线程池与多线程的区别

答：一个线程的周期分为：创建、运行、销毁三个阶段。处理一个任务时，首先创建一个任务线程，然后执行任务，完了，销毁线程。

而线程处于运行状态的时候，才是真的在处理我们交给它的任务，这个阶段才是有效运行时间。所以，

我们希望花在创建和销毁线程的资源越少越好。如果不销毁线程，而这个线程又不能被其他的任务调用，那么就会出现资源的浪费。为了提高效率，

减少创建和销毁线程带来时间和空间上的浪费，出现了线程池技术。这种技术是在开始就创建一定量的线程，批量处理一类任务，

等待任务的到来。任务执行完毕后，线程又可以执行其他的任务。

等不再需要线程的时候，就销毁。这样就省去了频繁创建和销毁线程的麻烦。

51、oracle如何强制索引

答：1）一般来说强制索引都是采用/\*+INDEX(表名,索引名字)\*/ 的方法,但是这种方法有个缺点，如果索引的名字改变了，就会导致程序重新改动，大大增加维护成本。

其实索引提示还可以使用列的方法进行,语法是/\*+INDEX(表名,(索引列的列表))\*/,这种方法即使索引的名字变了，也不会导致程序重新改动，除非索引列的顺序变化了。

2）地址：http://blog.chinaunix.net/uid-22948773-id-3126680.html

53、时间复杂度

54、快速排序的原理

答：http://www.cnblogs.com/hjy9420/p/5032309.html

55、什么是单点登录？

单点登录全称Single Sign On（以下简称SSO），是指在多系统应用群中登录一个系统，便可在其他所有系统中得到授权而无需再次登录，包括单点登录与单点注销两部分

http://blog.csdn.net/paranoia\_zk/article/details/53836106

56、redis持久化

rdb：redis database 、 aof ：append only file

57、java中用单例模式有什么好处

java 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。 使用Singleton的好处还在于可以节省内存，因为它限制了实例的个数，有利于Java垃圾回收（garbage collection）。

单例模式主要有3个特点，：

1、单例类确保自己只有一个实例。

2、单例类必须自己创建自己的实例。

3、单例类必须为其他对象提供唯一的实例。

其实无非是三个方面的作用：

第一、控制资源的使用，通过线程同步来控制资源的并发访问；

第二、控制实例产生的数量，达到节约资源的目的。

第三、作为通信媒介使用，也就是数据共享，

58、Spring框架的作用

Spring框架主要用于与其他技术(struts,hibernate等)进行整合,

可将应用程序中的Bean组件实现低耦合关联.最终可以提高系统扩展

和维护性.

将来我们利用Spring框架管理系统的各个组件(Action,Service,DAO)

.采用Spring的IOC和AOP机制实现各组件的关联.从而实现了低耦合调用.增强了系统可维护性和扩展性.

59、Struts2、spring2、hibernate3在SSH中各起什么作用

简单的说：

struts 控制用的

hibernate 操作数据库的

spring用解耦的

详细的说：

STRUTS 在 SSH 框架中起控制的作用 , 其核心是 Controller, 即 ActionServlet,

而 ActionServlet 的核心就是 Struts-confi g.xml. 主要控制逻辑关系的处理 .

hibernate 是数据持久化层 , 是一种新的对象、关系的映射工具 , 提供了从 Java

类到数据表的映射，也提供了数据查询和恢复等机制 , 大大减少数据访问的复杂度。把对数据库的直接操作 , 转换为对持久对象的操作 .

SPRING 是一个轻量级的控制反转 (IoC) 和面向切面 (AOP) 的容器框架 , 面向接口的编程 ,

由容器控制程序之间的（依赖）关系，而非传统实现中，由程序代码直接操控。这也就是所谓 “ 控制反转 ” 的概念所在：

（依赖）控制权由应用代码中转到了外部容器，控制权的转移，是所谓反转。依赖注入，即组件之间的依赖关系由容器在运行期决定，

形象的来说，即 由容器动态的将某种依赖关系注入到组件之中

起到的主要作用是解耦

Struts 、 spring 、 Hibernate 在各层的作用

1 ） struts 负责 web 层 .

ActionFormBean 接收网页中表单提交的数据，然后通过 Action 进行处理，再 Forward 到对应的网页。

在 struts-config.xml 中定义 <action-mapping>, ActionServlet 会加载。

2 ） spring 负责业务层管理，即 Service （或 Manager).

1 ． service 为 action 提供统计的调用接口，封装持久层的 DAO.

2 ．可以写一些自己的业务方法。

3 ．统一的 javabean 管理方法

4 ．声明式事务管理

5. 集成 Hiberante

3 ） Hiberante ，负责持久化层，完成数据库的 crud 操作

hibernate 为持久层，提供 OR/Mapping 。

它有一组 .hbm.xml 文件和 POJO, 是跟数据库中的表相对应的。然后定义 DAO ，

这些是跟数据库打交道的类，它们会使用 PO 。

在 struts+spring+hibernate 的系统中，

对象的调用流程是： jsp-> Action － > Service ->DAO ->Hibernate 。

数据的流向是 ActionFormBean 接受用户的数据， Action 将数据从 ActionFromBean 中取出，

封装成 VO 或 PO,

再调用业务层的 Bean 类，完成各种业务处理后再 forward 。而业务层 Bean 收到这个 PO 对象之后，

会调用 DAO 接口方法，进行持久化操作。

60.struts2的工作原理

在web容器中注册FilterDispatcher，容器启动时，初始化FilterDispatcher，加载default.properties文件，查看当前工程负责处理的请求类型以及配置文件，按照配置文件的设置顺序以及名字加载struts配置文件

客户端发送请求，FilterDispatcher过滤到，查看请求类型与defaultproperteies文件中指定的拦截请求类型是否匹配，如果匹配成功，表示该请求可以被处理；将请求转发给ActionProxy，该代理对象负责创建一个ActionInvocation实例（调度者）。调度者对请求进行解析，根据解析的结果字符串，在配置文件中的package包中查找与其匹配的Action节点（name）；实例化action节点class属性指定的业务逻辑Action；然后查看action周围是否有action引用，如果有，按照引用的顺序先进后出的原则执行拦截器，执行完毕之后，执行action的execute方法，该业务方法返回一个结果字符串，根据返回字符串在配置文件中匹配对应的视图对象；带着这个视图对象返回拦截器执行拦截器未执行完毕的代码段。

mybatis中#｛｝是占位符，用于参数传递。${}用于SQL拼接。

HashMap如何调整性能和空间的取舍?

构造方法的两个参数:int 分组组数 , float 加载因子

在加载因子不变的情况下 如果我们让分组组数变大

则效率更高 -> 但是需要更多的空间

在分组组数不变的情况下 如果让加载因子变大

则更加节省空间 -> 但是效率更低

\*:我们在已经知道要存放的元素个数的前提下 应该让

分组组数\*加载因子 > 元素总量

避免添加过程当中触发扩容操作