

Table of Contents

Introduction	0
笔试/面试题库	1
java选择题-简单不结果	1.1
java选择题-简单带结果	1.2

GOOD LUCK

GOOD LUCK

Java基础面试题

1. 下面哪些是Thread类的方法 ()

1. start()
 2. run()
 3. exit()
 4. getPriority()
-

2. 下面关于java.lang.Exception类的说法正确的是 ()

1. 继承自Throwable
 2. Serializable
-

3. 下面程序的运行结果是 ()

```
String str1 = "hello";
String str2 = "he" + new String("llo");
System.out.println(str1 == str2);
```

1. true
 2. false
-

4. 下列说法正确的有 ()

1. class中的constructor不可省略
 2. constructor必须与class同名，但方法不能与class同名
 3. constructor在一个对象被new时执行
1. 一个class只能定义一个constructor
-

5.下面程序的运行结果

```
public static void main(String args[]) {  
    Thread t = new Thread() {  
  
        public void run() {  
            pong();  
        }  
    };  
  
    t.run();  
    System.out.print("ping");  
  
}  
  
static void pong() {  
  
    System.out.print("pong");  
  
}
```

-
- 1. pingpong
 - 2. pongping
 - 3. pingpong和pongping都有可能
 - 4. 都不输出
-

6.GC线程是否为守护线程? ()

答案：是

7.volatile关键字是否能保证线程安全? ()

8.下列说法正确的是 ()

- 1. LinkedList继承自List
- 2. AbstractSet继承自Set
- 3. HashSet继承自AbstractSet

4. WeakMap继承自HashMap
-

9. 数字0.6332的数据类型是 ()

1. float
 2. double
 3. Float
 4. Double
-

10. 下面哪个流类属于面向字符的输入流()

1. BufferedWriter
 2. FileInputStream
 3. ObjectInputStream
 4. InputStreamReader
-

11. 不通过构造函数也能创建对象吗 ()

1. 是
 2. 否
-

12. 下面哪些是对称加密算法 ()

1. DES
 2. AES
 3. DSA
 4. RSA
-

13. 接口和抽象类描述正确的有

1. 抽象类没有构造函数
 2. 接口没有构造函数
 3. 抽象类不允许多继承
-

4. 接口中的方法可以有方法体
-

14. 在Java中，关于HashMap类的描述，以下选项错误的是

1. HashMap使用键/值得形式保存数据
 2. HashMap 能够保证其中元素的顺序
 3. HashMap允许将null用作键
 4. HashMap允许将null用作值
-

15. 在Java中可用于创建链表数据结构的对象。

1. LinkedList
 2. ArrayList
 3. Collection
 4. HashMap
-

16. 下列哪些情况可以终止当前线程的运行？

1. 抛出一个例外时。
 2. 当该线程调用sleep()方法时。
 3. 当创建一个新线程时。
 4. 当一个优先级高的线程进入就绪状态时。
-

17. 栈是一种()

1. 存取受限的线性结构
 2. 存取不受限的线性结构
 3. 存取受限的非线性结构
 4. 存取不受限的非线性结构
-

18. 接口是Java面向对象的实现机制之一，以下说法正确的是

1. Java支持多重继承，一个类可以实现多个接口；
 2. Java只支持单重继承，一个类可以实现多个接口；
 3. Java只支持单重继承，一个类只可以实现一个接口；
 4. Java支持多重继承，但一个类只可以实现一个接口。
-

19.在Servlet处理请求的方式为。(选择1项)

1. 以进程的方式
 2. 以程序的方式
 3. 以线程的方式
 4. 以响应的方式
-

20.Delete和 truncate 都可以用来删除表内容，一下描述正确的是？

- 1.Truncate 不需要 rollbacksegment
 2. Delete 需要rollbacksegment
 - 3.Truncate在 提交commit之前仍可回滚
 4. Truncate还可以删除表结构
-

21. 在Java中，一个类可同时定义许多同名的方法，这些方法的形式参数个数、类型或顺序各不相同，传回的值也可以不相同。这种面向对象程序的特性称为

1. 隐藏
 2. 覆盖
 3. 重载
 4. Java不支持此特性
-

22. 如何放掉一个指定占据的内存空间，请选择一个正确答案：

1. 调用free()方法。
2. 代用system.gc()方法。
3. 赋值给该项对象的引用为null.
4. 程序员无法明确强制垃圾回收器运行。

Java基础面试题

下面哪些是Thread类的方法 ()

1. start()
2. run()
3. exit()
4. getPriority()

答案：1、2、4

下面关于java.lang.Exception类的说法正确的是 ()

1. 继承自Throwable
2. Serializable

答案：1

解析：Java异常的基类为java.lang.Throwable, java.lang.Error和java.lang.Exception继承Throwable, RuntimeException和其它的Exception等继承Exception, 具体的RuntimeException继承RuntimeException。 扩展：错误和异常的区别(Error vs Exception)

1) java.lang.Error: Throwable的子类，用于标记严重错误。合理的应用程序不应该去try/catch这种错误。绝大多数的错误都是非正常的，就根本不该出现的。

java.lang.Exception: Throwable的子类，用于指示一种合理的程序想去catch的条件。即它仅仅是一种程序运行条件，而非严重错误，并且鼓励用户程序去catch它。

2) Error和RuntimeException 及其子类都是未检查的异常 (unchecked exceptions) , 而所有其他的Exception类都是检查了的异常 (checked exceptions) . checked exceptions: 通常是从一个可以恢复的程序中抛出来的，并且最好能够从这种异常中使用程序恢复。比如FileNotFoundException, ParseException等。检查了的异常发生在编译阶段，必须要使用try...catch (或者throws) 否则编译不通过。 unchecked exceptions: 通常是因为一切正常的话本不该发生的异常，但是的确发生了。发生在运行期，具有不确定性，主要是由于程序的逻辑问题所引起的。比如ArrayIndexOutOfBoundsException, ClassCastException等。从语言本身的角度讲，程序不该去catch这类异常，虽然能够从诸如RuntimeException这样的异常中catch并恢复，但是并不鼓励终端程序员这么做，因为完

全没必要。因为这类错误本身就是bug，应该被修复，出现此类错误时程序就应该立即停止执行。因此，面对Errors和unchecked exceptions应该让程序自动终止执行，程序员不该做诸如try/catch这样的事情，而是应该查明原因，修改代码逻辑。

RuntimeException： RuntimeException体系包括错误的类型转换、数组越界访问和试图访问空指针等等。

处理RuntimeException的原则是：如果出现 RuntimeException，那么一定是程序员的错误。例如，可以通过检查数组下标和数组边界来避免数组越界访问异常。其他（IOException等等）checked异常一般是外部错误，例如试图从文件尾后读取数据等，这并不是程序本身错误，而是在应用环境中出现的外部错误。

下面程序的运行结果是 ()

```
String str1 = "hello";
String str2 = "he" + new String("llo");
System.out.println(str1 == str2);
```

1. true
2. false

答案：2

解析：因为str2中的llo是新申请的内存块，而==判断的是对象的地址而非值，所以不一样。如果是String str2 = str1，那么就是true了。

4.下列说法正确的有 ()

1. class中的constructor不可省略
2. constructor必须与class同名，但方法不能与class同名
3. constructor在一个对象被new时执行
 1. 一个class只能定义一个constructor

答案：3 解析：这里可能会有误区，其实普通的类方法是可以和类名同名的，和构造方法唯一的区分就是，构造方法没有返回值。

5.下面程序的运行结果

```
public static void main(String args[]) {  
  
    Thread t = new Thread() {  
  
        public void run() {  
            pong();  
        }  
    };  
  
    t.run();  
    System.out.print("ping");  
  
}  
  
static void pong() {  
  
    System.out.print("pong");  
  
}
```

1. pingpong
2. pongping
3. pingpong和pongpong都有可能
4. 都不输出

答案：2

解析：这里考的是Thread类中start()和run()方法的区别了。start()用来启动一个线程，当调用start方法后，系统才会开启一个新的线程，进而调用run()方法来执行任务，而单独的调用run()就跟调用普通方法是一样的，已经失去线程的特性了。因此在启动一个线程的时候一定要使用start()而不是run()。

6.GC线程是否为守护线程？（）

答案：是

解析：线程分为守护线程和非守护线程（即用户线程）。只要当前JVM实例中尚存在任何一个非守护线程没有结束，守护线程就全部工作；只有当最后一个非守护线程结束时，守护线程随着JVM一同结束工作。守护线程最典型的应用就是GC（垃圾回收器）

7. volatile关键字是否能保证线程安全？（）

答案：不能

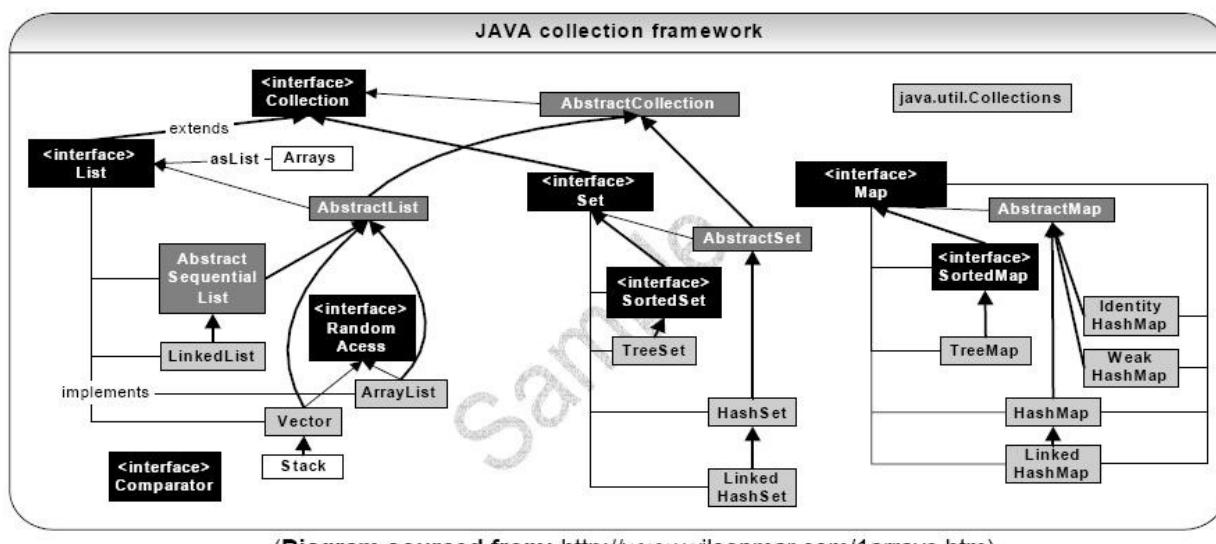
解析：volatile关键字用在多线程同步中，可保证读取的可见性，JVM只是保证从主内存加载到线程工作内存的值是最新的读取值，而非cache中。但多个线程对 volatile的写操作，无法保证线程安全。例如假如线程1，线程2 在进行read,load 操作中，发现主内存中 count的值都是5，那么都会加载这个最新的值，在线程1堆count进行修改之后，会write到主内存中，主内存中的count变量就会变为6；线程2由于已经进行read,load操作，在进行运算之后，也会更新主内存count的变量值为6；导致两个线程及时用volatile关键字修改之后，还是会存在并发的情况。

8.下列说法正确的是（）

1. LinkedList继承自List
2. AbstractSet继承自Set
3. HashSet继承自AbstractSet
4. WeakMap继承自HashMap

答案：1、3

解析：下面是一张下载的Java中的集合类型的继承关系图，一目了然。



9. 数字0.6332的数据类型是（）

1. float
2. double
3. Float
4. Double

答案：2

解析：默认为double型，如果为float型需要加上f显示说明，即0.6332f

10.下面哪个流类属于面向字符的输入流()

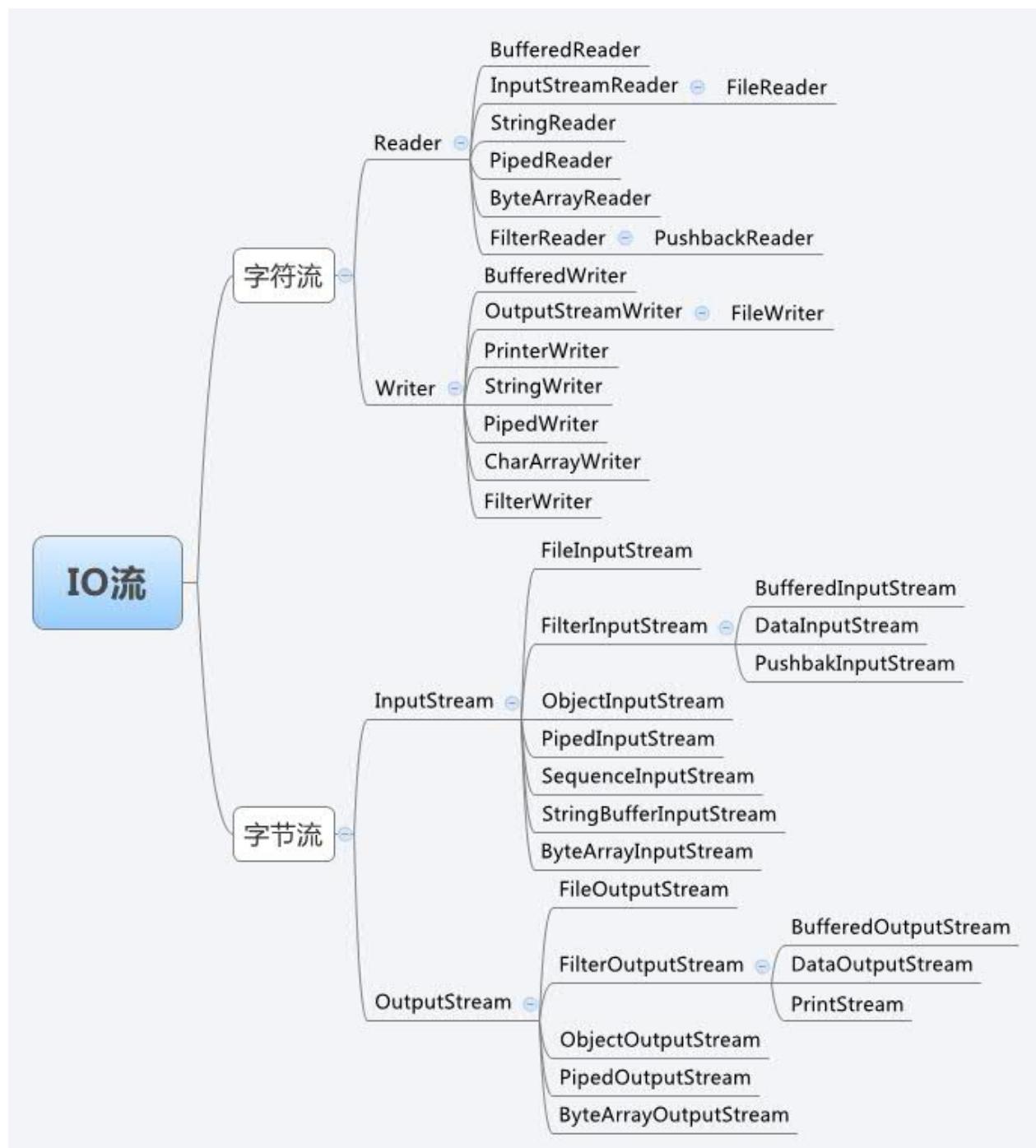
1. BufferedWriter
2. FileInputStream
3. ObjectInputStream
4. InputStreamReader

答案：4

解析：Java的IO操作中有面向字节(Byte)和面向字符(Character)两种方式。面向字节的操作为以8位为单位对二进制的数据进行操作，对数据不进行转换，这些类都是InputStream和OutputStream的子类。面向字符的操作为以字符为单位对数据进行操作，在读的时候将二进制数据转为字符，在写的时候将字符转为二进制数据，这些类都是Reader和Writer的子类。

总结：以InputStream（输入）/OutputStream（输出）为后缀的是字节流；以Reader（输入）/Writer（输出）为后缀的是字符流。

扩展：Java流类图结构，一目了然，解决大部分选择题：



11. 不通过构造函数也能创建对象吗 ()

1. 是
2. 否

答案：1

解析：Java创建对象的几种方式（重要）：

(1) 用new语句创建对象，这是最常见的创建对象的方法。 (2) 运用反射手段,调用java.lang.Class或者java.lang.reflect.Constructor类的newInstance()实例方法。 (3) 调用对象的clone()方法。 (4) 运用反序列化手段，调用java.io.ObjectInputStream对象的readObject()方法。

(1)和(2)都会明确的显式的调用构造函数； (3)是在内存上对已有对象的影印，所以不会调用构造函数； (4)是从文件中还原类的对象，也不会调用构造函数。

12.下面哪些是对称加密算法 ()

1. DES
2. AES
3. DSA
4. RSA

答案：1、2

解析：常用的对称加密算法有：DES、3DES、RC2、RC4、AES

常用的非对称加密算法有：RSA、DSA、ECC

使用单向散列函数的加密算法：MD5、SHA

13.接口和抽象类描述正确的有

1. 抽象类没有构造函数
2. 接口没有构造函数
3. 抽象类不允许多继承
4. 接口中的方法可以有方法体

答案： (2,3) (两项)

14. 在Java中，关于HashMap类的描述，以下选项错误的是

1. HashMap使用键/值得形式保存数据
2. HashMap 能够保证其中元素的顺序
3. HashMap允许将null用作键
4. HashMap允许将null用作值

答案:2

15. 在Java中可用于创建链表数据结构的对象。

1. LinkedList
2. ArrayList
3. Collection
4. HashMap

答案：1

16.下列哪些情况可以终止当前线程的运行?

1. 抛出一个例外时。
2. 当该线程调用sleep()方法时。
3. 当创建一个新线程时。
4. 当一个优先级高的线程进入就绪状态时。

答案：1、2、4

17.栈是一种()

1. 存取受限的线性结构
2. 存取不受限的线性结构
3. 存取受限的非线性结构
4. 存取不受限的非线性结构

答案：1 栈 (stack) 在计算机科学中是限定仅在表尾进行插入或删除操作的线性表。

18. 接口是Java面向对象的实现机制之一，以下说法正确的是

1. Java支持多重继承，一个类可以实现多个接口；
2. Java只支持单重继承，一个类可以实现多个接口；

3. Java只支持单重继承，一个类只可以实现一个接口；
4. Java支持多重继承，但一个类只可以实现一个接口。

答案：2 Java支持单重继承，一个类只能继承自另外的一个类，但是一个类可以实现多个接口。

19.在Servlet处理请求的方式为。(选择1项)

1. 以进程的方式
2. 以程序的方式
3. 以线程的方式
4. 以响应的方式

答案：3

Servlet采用多线程来处理多个请求同时访问，Servlet容器维护了一个线程池来服务请求。

20.Delete和 truncate 都可以用来删除表内容，一下描述正确的是？

- 1.Truncate 不需要 rollbacksegment
2. Delete 需要rollbacksegment
- 3.Truncate在 提交commit之前仍可回滚
4. Truncate还可以删除表结构

答案：1、2 TRUNCATE TABLE 语句:删除表中所有的数据并且释放表的存储空间，可以使用 DELETE 语句删除数据，DELETE产生rollback，如果删除大数据量的表速度会很慢，同时会占用很多的rollback segments .truncate 是DDL操作，不产生rollback，速度快一些

20. 在Java中，一个类可同时定义许多同名的方法，这些方法的形式参数个数、类型或顺序各不相同，传回的值也可以不相同。这种面向对象程序的特性称为

1. 隐藏
2. 覆盖
3. 重载
4. Java不支持此特性

答案：重载

21. 如何放掉一个指定占据的内存空间，请选择一个正确答案：

1. 调用free()方法。
2. 代用system.gc()方法。
3. 赋值给该项对象的引用为null.
4. 程序员无法明确强制垃圾回收器运行。

答案： 4