
常用工具类作业

一、 填空题

1. 在 Java 中每个 Java 基本类型在 `java.lang` 包中都在一个相应的包装类，把基本类型数据转换为对象，其中包装类 `Integer` 是_____的直接子类。
2. 包装类 `Integer` 的静态方法可以将字符串类型的数字"123"转换成基本整型变量 `n`，其实现语句是：_____。
3. 在 Java 中使用 `java.lang` 包中的_____类来创建一个字符串对象，它代表一个字符序列可变的字符串，可以通过相应的方法改变这个字符串对象的字符序列。
4. `StringBuilder` 类是 `StringBuffer` 类的替代类，两者的共同点是都是可变长度字符串，其中线程安全的类是_____。
5. `DateFormat` 类可以实现字符串和日期类型之间的格式转换，其中将日期类型转换为指定的字符串格式的方法名是_____。
6. 使用 `Math.random()` 返回带正号的 `double` 值，该值大于等于 0.0 且小于 1.0。使用该函数生成[30,60]之间的随机整数的语句是_____。
7. JDK1.5 后提供了_____关键字，用以定义枚举类。枚举类是一种特殊的类，可以有自己的属性、方法和构造方法。
8. `File` 对象调用方法_____创建一个目录，不包括所有必需但不存在的父目录，当且仅当已创建目录时，返回 `true`；否则返回 `false`。
9. 将字符串"123"转换成基本数据类型。
10. `String` 类的 `trim()` 方法作用是_____。
11. `"hamburger".substring(4, 8)` 返回的结果是_____。
12. `String s = "a"+"b"+"c"` 创建_____个对象。
13. `System.currentTimeMillis()` 表示_____。

二、 选择题

1. 以下选项中关于 `int` 和 `Integer` 的说法错误的是（ ）。(选择二项)
 - A. `int` 是基本数据类型，`Integer` 是 `int` 的包装类，是引用数据类型
 - B. `int` 的默认值是 0，`Integer` 的默认值也是 0
 - C. `Integer` 可以封装了属性和方法提供更多的功能
 - D. `Integer i=5;`该语句在 JDK1.5 之后可以正确执行，使用了自动拆箱功能
2. 分析如下 Java 代码，该程序编译后的运行结果是（ ）。(选择一项)

```
public static void main(String[] args) {  
    String str=null;  
    str.concat("abc");  
    str.concat("def");  
    System.out.println(str);  
}
```

 - A. `null`
 - B. `abcdef`
 - C. 编译错误

D. 运行时出现 `NullPointerException` 异常

3. 以下关于 `String` 类的代码的执行结果是 ()。(选择一项)

```
public class Test2 {  
    public static void main(String args[]) {  
        String s1 = new String("ccjr");  
        String s2 = new String("ccjr");  
        if (s1 == s2)           System.out.println("s1 == s2");  
        if (s1.equals(s2))      System.out.println("s1.equals(s2)");  
    }  
}
```

- A. `s1 == s2`
- B. `s1.equals(s2)`
- C. `s1 == s2`
`s1.equals(s2)`
- D. 以上都不对

4. 以下关于 `StringBuffer` 类的代码的执行结果是 ()。(选择一项)

```
public class TestStringBuffer {  
    public static void main(String args[]) {  
        StringBuffer a = new StringBuffer("A");  
        StringBuffer b = new StringBuffer("B");  
        mb_operate(a, b);  
        System.out.println(a + "." + b);  
    }  
    static void mb_operate(StringBuffer x, StringBuffer y) {  
        x.append(y);  
        y = x;  
    }  
}
```

- A. A.B
- B. A.A
- C. AB.AB
- D. AB.B

5. 给定如下 `Java` 代码，编译运行的结果是 ()。(选择一项)

```
public static void main(String []args){  
    String s1= new String("pb_java_OOP_T5");  
    String s2 = s1.substring(s1.lastIndexOf("_"));  
    System.out.println("s2="+s2);  
}
```

- A `s2=_java_OOP_T5`
- B. `s2=_OOP_T5`

- C. s2=_T5
D. 编译出错
6. 对于语句 **String s="my name is kitty"**，以下选项中可以从其中截取“kitty”的是（ ）（选择二项）
- A. s.substring(11,16)
B. s.substring(11)
C. s.substring(12,17)
D. s.substring(12,16)
7. 分析下面的Java程序段，编译运行后的输出结果是（ ）。（选择一项）
- ```
public class Test {
 public void changeString(StringBuffer sb) {
 sb.append("stringbuffer2");
 }
 public static void main(String[] args) {
 Test a = new Test();
 StringBuffer sb = new StringBuffer("stringbuffer1");
 a.changeString(sb);
 System.out.println("sb = " + sb);
 }
}
```
- A. sb = stringbuffer2stringbuffer1  
B. sb = stringbuffer1  
C. sb = stringbuffer2  
D. sb = stringbuffer1stringbuffer2
8. 给定如下 Java 代码，编译运行的结果是（ ）。（选择一项）
- ```
public static void main(String[] args) {  
    StringBuffer sbf = new StringBuffer("java");  
    StringBuffer sbf1 = sbf.append(",C#");  
    String sbf2 = sbf + ",C#";  
    System.out.print(sbf.equals(sbf1));  
    System.out.println(sbf2.equals(sbf));  
}
```
- A. true false
B. true true
C. false false
D. false true
9. 分析下面的 Java 程序，编译运行后的输出结果是（ ）。（选择一项）
- ```
public class Example {
 String str = new String("good");
 char[] ch = { 'a', 'b', 'c' };
```

---

```
public static void main(String args[]) {
 Example ex = new Example();
 ex.change(ex.str, ex.ch);
 System.out.print(ex.str + "and");
 System.out.print(ex.ch);
}
public void change(String str, char ch[]) {
 str = "test ok";
 ch[0] = 'g';
}
}
```

- A. goodandabc  
B. goodandgbc  
C. test okandabc  
D. test okandgbc
10. 以下程序片段中可以正常编译的是（ ）。(选择一项)
- A. String s = "Gone with the wind";  
String k = s+t;  
String t = "good";  
B. String s = "Gone with the wind";  
String t;  
t = s[3]+"one";  
C. String s = "Gone with the wind";  
String stanford = s.toUpperCase();  
D. String s = "home directory";  
String t = s - "directory";
11. File类中的（ ）方法可以用来判断文件或目录是否存在。(选择一项)
- A. exist()  
B. exists()  
C. fileExist()  
D. fileExists()
12. 在 Java 中，以下 File 类的方法中（ ）用来判断是否是目录。(选择一项)
- A. isFile()  
B. getFile()  
C. isDirectory()  
D. getPath()
13. 分析下面的 Java 程序，编译运行后的输出结果是（ ）。(选择一项)
- ```
public class Example {  
    String str = new String("good"); //引用数据类型在赋值 和 传值的时候 都是 进行地址传递。
```

```

char[] ch = { 'a', 'b', 'c' };
public static void main(String args[]) {
    Example ex = new Example( );
    ex.change(ex.str, ex.ch);
    System.out.print(ex.str + "and");
    System.out.print(ex.ch);
}
public void change(String str, char ch[]) {
    str = "test ok";
    ch[0] = 'g';
}
}

```

A. goodandabc
 B. goodandgbc
 C. test okandabc
 D. test okandgbc

14. 分析下面代码的结果 ()。(选择一项)

```

public static void main(String args[]) {
    String s = "abc";
    String ss = "abc";
    String s3 = "abc" + "def"; // 此处编译器做了优化!
    String s4 = "abcdef";
    String s5 = ss + "def";
    String s2 = new String("abc");
    System.out.println(s == ss);
    System.out.println(s3 == s4);
    System.out.println(s4 == s5);
    System.out.println(s4.equals(s5));
}

```

A. true true false true
 B. true true true false
 C. true false true true
 D. false true false true

三、判断题

1. 方法 `Integer.parseInt()` 的作用是将一个整数转变成 `String`。()
2. `JK1.5` 后提供了自动装箱和自动拆箱功能, 从而可以实现基本数据类型和对应包装类之间的自动转换, 简化了操作。()
3. 执行语句 `String str="abcedf"; int len=str.length;` 后, 能够得到字符串的长度是 6。()
4. 运算符“==”用于比较引用时, 如果两个引用指向内存同一个对象, 则返回 `true`。()
5. `java.sql.Date` 类和 `java.util.Date` 类的关系是前者是后者的父类, 其中前者没有提供无参

数构造方法，而后者可以提供无参数构造方法来获取当前时间。()

6. 求 x 的 y 次方，其表达式为：`Math.pow(x,y)`。()
7. 一个 `File` 对象可以代表一个文件或目录，它可以获取文件和目录属性，也可以访问文件内容。()
8. 在使用 `File` 类中的 `delete()` 方法时，删除时可能文件不存在，所以我们最好先判断一下是否存在，不然会出现 `NullPointerException` 异常。()
9. `Date d = new Date()` 表示的是当前时间。()
10. 递归可以完全使用迭代来代替。()

四、简答题

1. 自动装箱和自动拆箱
2. `String`、`StringBuffer`、`StringBuilder` 区别与联系。
3. `String str="ccjr";`和 `String str= new String("ccjr");`的区别
4. `java.sql.Date` 和 `java.util.Date` 的联系和区别
5. 为什么要使用包装类，包装类的作用。
6. 递归算法的优点是什么?缺点是什么?

五、编码题

1. 验证键盘输入的用户名不能为空，长度大于6，不能有数字。
提示：使用字符串`String`类的相关方法完成
2. 接收从键盘输入的字符串格式的年龄，分数和入学时间，转换为整数、浮点数、日期类型，并在控制台输出。
提示：使用包装类`Integer`、`Double`和日期转换类`DateFormat`实现
3. 将1990年3月3日通过`Calendar`来表示，并得出这天是该年的第几天?将该日期增加35天，是哪一天?使用代码来说明。

六、可选题

1. 生成10个[10,23)之间的随机整数
提示：分别使用`Math.random()`和`Random`类的`nextDouble()`或`nextInt()`实现
2. 打印某个月份的可视化日历

请输入日期：（年月日，格式：2008-9-20）：
2010-1-1

日	一	二	三	四	五	六
					1*	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

提示：使用 DateFormat、Calendar 类实现功能