

2. 类与对象的关系：类是对象的模板，对象是类的实例，类只能通过对象才可以使用。
3. 类的组成：属性、方法。
4. 对象的产生格式：类名称 对象名称 = new 类名称()。
5. 如果一个对象没有被实例化而直接使用，则使用时会出现空指向异常。
6. 类属于引用数据类型，进行引用传递时，传递的是堆内存的使用权。
7. 类的封装性：通过 `private` 关键字进行修饰，被封装的属性不能被外部直接调用，而只能通过 `setter` 或 `getter` 方法完成。只要是属性，就必须全部封装。
8. 构造方法可以为类中的属性初始化，构造方法与类名称相同，无返回值类型声明。如果在类中没有明确的定义出构造方法，则会自动生成一个无参的什么都不做的构造方法。在一个类中的构造方法可以重载，但是每个类都至少有一个构造方法。
9. `String` 类在 Java 中较为特殊，`String` 可以使用直接赋值的方式，也可以通过构造方法进行实例化，前者只产生一个实例化对象，而且此实例化对象可以重用，后者将产生两个实例化对象，其中一个垃圾空间，在 `String` 中比较内容使用 `equals` 方法，而“==”比较的只是两个字符串的地址值。字符串的内容一旦声明则不可改变。
10. 在 Java 中使用 `this` 关键字可以表示当前的对象，通过 `this.属性` 可以调用本类中的属性，通过 `this.方法()` 可以调用本类中的其他方法，也可以通过 `this()` 的形式调用本类中的构造方法，但是调用时要放在构造方法的首行。
11. 使用 `static` 声明的属性和方法可以由类名称直接调用，`static` 属性是所有对象共享的，所有对象都可以对其进行操作。当一个类中不需要保存属性时，可以考虑将这个类中的方法全部定义为 `static`，这样做可以节约内存空间。
12. 对象数组的使用要分成两个部分：第一步是声明数组，第二步是为数组开辟空间。开辟空间后数组中的每个元素的内容都是 `null`。
13. 内部类是在一个类的内部定义另外的一个类，使用内部类可以方便地访问外部类的私有操作。在方法中声明的内部类要想访问方法的参数，则参数前必须加上“`final`”关键字。
14. 对象比较是一个类内部具备的方法，在对象比较操作中要依次执行：数据是否为 `null`，地址是否相同，属性是否相同等判断。
15. 链表是一种动态对象数组的基本实现，`Link` 类是用户主要操作的工具类，而 `Node` 类将负责节点的关系维护。如果要保存自定义类型，则必须提供对象比较操作，否则 `contains()`、`remove()` 两个方法将无法正常使用。

课后习题

一、填空题

1. 面向对象的三大特征：_____、_____、_____。
2. 类由_____和_____组成。
3. _____运算符的作用是根据对象的类型分配内存空间。当对象拥有内存空间时，会自动调用类中的_____为对象_____。
4. 使用_____修饰的类成员称为私有成员。私有成员只能在_____中使用。

5. 构造方法的名称与_____相同。
6. _____关键字可以让类中的属性和方法对外部不可见。
7. this 关键字可以调用本类中的_____, _____, _____, 调用_____时必须放在_____的首行。
8. 在 Java 中数组排序的方法是_____。

二、选择题

1. 如果希望方法直接通过类名称访问, 在定义时要使用的修饰符是 ()。
 - A. static
 - B. final
 - C. abstract
 - D. this
2. 如果类中没有定义构造方法, 系统会提供一个默认的构造方法。默认构造方法的特点是 ()。
 - A. 无参数有操作
 - B. 有参数无操作
 - C. 既无参数也无任何操作
 - D. 有参数有操作
3. 有一个类 Demo, 对与其构造方法的正确声明是 ()。
 - A. void Demo(int x){...}
 - B. Demo(int x){...}
 - C. Demo Demo(int x){...}
 - D. int Demo(){}
4. 以下关于面向对象概念的描述中, 不正确的一项是 ()。
 - A. 在现实生活中, 对象是指客观世界的实体
 - B. 程序中的对象就是现实生活中的对象
 - C. 在程序中, 对象是通过一种抽象的数据类型来描述的, 这种抽象数据类型称为类 (class)
 - D. 在程序中, 对象是一组变量和相关方法的集合
5. () 不属于面向对象程序设计的基本要素。
 - A. 类
 - B. 对象
 - C. 方法
 - D. 安全
6. 下列程序的执行结果是 ()。

```
public class TestDemo {
    public void fun() {
        static int i = 0;
        i++;
        System.out.println(i);
    }
    public static void main(String args[]) {
        Demo d = new Demo();
        d.fun();
    }
}
```

- A. 编译错误
 - B. 0
 - C. 1
 - D. 运行成功, 但不输出
7. 顺序执行下列程序语句后, 则 b 的值是 ()。

```
String str = "Hello";
String b = str.substring(0, 2);
```

- A. Hello
 - B. hello
 - C. He
 - D. null

三、判断题

1. 没有实例化的对象不能使用。 ()
2. 不可以为类定义多个构造方法。 ()
3. 使用 static 声明的方法可以调用非 static 声明的方法。 ()
4. 非 static 声明的方法可以调用 static 声明的属性或方法。 ()
5. String 对象可以使用==进行内容的比较。 ()
6. 垃圾是指无用的内存空间, 会被垃圾收集机制回收。 ()
7. 构造方法可以有返回值类型的声明。 ()
8. 匿名对象是指使用一次的对象, 使用之后将等待被垃圾回收。 ()
9. 使用 static 定义的内部类就成为外部类。 ()
10. 多个实例化对象之间不会互相影响, 因为保存在不同的内存区域之中。 ()

四、简答题

1. String 类的操作特点。
2. 简述垃圾对象的产生。
3. static 方法如何调用? 非 static 方法如何调用?
4. 类与对象的关系是什么? 如何创建及使用对象?
5. 举例说明子类对象的实例化过程。
6. 简述 this 与 super 关键字的区别。

五、编程题

1. 编写并测试一个代表地址的 Address 类, 地址信息由: 国家, 省份, 城市, 街道, 邮编组成, 并可以返回完整的地址信息。
2. 定义并测试一个代表员工的 Employee 类。员工属性包括“编号”“姓名”“基本薪水”“薪水增长额”; 还包括“计算增长后的工资总额”。的操作方法。
3. 编写程序在将字符串“want you to know one thing”, 统计出字母“n”和字母“o”的出现次数。
4. 设计一个 Dog 类, 有名字、颜色、年龄等属性, 定义构造方法来初始化类的这些属性, 定义方法输出 Dog 信息。编写应用程序使用 Dog 类。
5. 字符串操作。
 - (1) 从字符串“MLDN 中心 Java 技术学习班 20130214”中提取开班日期。
 - (2) 将“MLDN JAVA 高端技术培训”字符串中的“Java”替换为“JAVA EE”。
 - (3) 取出“Java 技术学习班 20130214”中的第八个字符。
 - (4) 清除“Java 技术学习班 20130214”中的所有“0”。
 - (5) 从任意给定的身份证号码中提取此人的出生日期。
6. 编写一个银行账户类, 类的构成包括:
 - 数据成员: 用户的账户名称、用户的账户余额;
 - 方法包括: 开户(设置账户名称及余额), 利用构造方法完成;
 - 查询余额。