

二级Java考试辅导教程：3.2Java语言的面向对象特性[1] PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/138/2021_2022__E4_BA_8C_E7_BA_A7Java_c97_138787.htm 3.2 Java语言的面向对象特性

3.2.1 类 类是java中的一种重要的复合数据类型，是组成java程序的基本要素。它封装了一类对象的状态和方法，是这一类对象的原形。一个类的实现包括两个部分：类声明和类体。

1.类声明： [public][abstract|final] class className [extends superclassName] [implements interfaceNameList] {...} 其中，修饰符public,abstract,final 说明了类的属性，className为类名，superclassName为类的父类的名字，interfaceNameList为类所实现的接口列表。

2.类体 类体定义如下： class className { [public | protected | private] [static] [final] [transient] [volatile] type variableName. //成员变量 [public | protected | private] [static] [final | abstract] [native] [synchronized] returnType

methodName([paramList]) [throws exceptionList] {statements} //成员方法 }

3.成员变量 成员变量的声明方式如下： [public | protected | private] [static] [final] [transient] [volatile] type variableName. //成员变量 其中， static: 静态变量（类变量）；相对于实例变量 final: 常量 transient: 暂时性变量，用于对象存档 volatile: 贡献变量，用于并发线程的共享来源

： www.examda.com 4.成员方法 方法的实现包括两部分内容：方法声明和方法体。 [public | protected | private] [static] [final | abstract] [native] [synchronized] returnType

methodName([paramList]) [throws exceptionList] //方法声明 {statements} //方法体 方法声明中的限定词的含义： static: 类方

法，可通过类名直接调用 abstract: 抽象方法，没有方法体
final: 方法不能被重写 native: 集成其它语言的代码
synchronized: 控制多个并发线程的访问 方法声明 方法声明包括方法名、返回类型和外部参数。其中参数的类型可以是简单数据类型，也可以是复合数据类型（又称引用数据类型）。对于简单数据类型来说，java实现的是值传递，方法接收参数的值，但不能改变这些参数的值。如果要改变参数的值，则用引用数据类型，因为引用数据类型传递给方法的是数据在内存中的地址，方法中对数据的操作可以改变数据的值。 100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com