

JC2002 Java 编程 - 实践 3（第 3 天）

本实践的目的是让你熟悉面向对象编程概念，特别是类继承和方法重载，并在 Java 程序中定义和使用类和方法方面获得一些练习。

1. 在文件 "**TestDoor.java**" 中定义**门类**，其中包含一个布尔型受保护实例变量 **isOpen**，初始化为 **false**。此外，定义两个公共方法 **openDoor()** 和 **closeDoor()**，其功能如下：

- 调用 **openDoor()** 且 **isOpen** 为 **false** 时，打印文本 "门已打开"、
将 **isOpen** 设为 **true**，然后返回。
- 调用 **openDoor()** 且 **isOpen** 为 **true** 时，打印文本 "门已经打开！" 并返回。
- 调用 **closeDoor()** 时，如果 **isOpen** 为 **false**，则打印文本 "门已关闭！" 并返回。
- 调用 **closeDoor()** 且 **isOpen** 为 **true** 时，打印文本 "门已关闭"，并设置 **isOpen** 为 **false**，然后返回。

2. 在同一文件 "**TestDoor.java**" 中定义带有 **main()** 方法的公共类 **TestDoor**，该方法可创建一个 **Door** 实例并循环获取用户反馈。在循环中，可以通过以下选项读取用户输入的整数：

1: 开门

2: 关门

3: 退出

您可以使用 **switch...case** 结构来实现菜单选项的不同功能。如果用户输入 1，则调用门实例的 **openDoor()** 方法。如果用户输入 2，则调用门实例的 **closeDoor()** 方法。如果用户输入 3，则退出循环并结束程序。如果用户输入其他内容（默认），则打印 "无效选项！" 并继续循环。

在门关闭和打开时使用不同选项测试程序，以确保其正常工作。

3. 定义扩展（继承）类 **Door** 的 **DoorWithLock** 类，包括初始化为 **false** 的新布尔类

型实例变量 **isLocked**。您仍然可以使用相同的文件 "**TestDoor.java**"。重载方法 **openDoor()**，如果 **isOpen** 为 **false** 而 **isLocked** 为 **true**，该方法将打印文本 "门已锁定，无法打开！"并返回。否则，**openDoor()** 的功能应与 **Door** 类中的相同。

此外，还要实现两个新方法 **lockDoor()** 和 **unlockDoor()**，其功能如下：

- 调用 **lockDoor()** 且 **isOpen** 为 true 时，打印文本 "打开的门无法锁定！"并返回。
 - 调用 **lockDoor()** 时，如果 **isOpen** 为 false 且 **isLocked** 为 true，则打印文本 "门已经锁上了！"然后返回。
 - 调用 **lockDoor()** 时，如果 **isOpen** 为 false 且 **isLocked** 为 false，则打印文本 "门已上锁"，将 **isLocked** 改为 true，然后返回。
 - 调用 **unlockDoor()** 时，如果 **isLocked** 为 false，则打印文本 "门未锁！"并返回。
 - 当调用 **unlockDoor()** 且 **isLocked** 为 true 时，打印文本 "门已解锁"，将 **isLocked** 改为 false 并返回。
4. 修改 **TestDoor** 类中的方法 **main()** 以测试 **DoorWithLock** 类，除了打开和关闭外，还增加了锁定和解锁门的菜单选项（即调用 **lockDoor()** 和 **unlockDoor()** 方法）。菜单选项如下

- 1: 开门
- 2: 关门
- 3: 锁门
- 4: 开锁
- 5: 退出

运行程序并在不同情况下进行测试，例如在门打开时尝试锁门，以确保 **DoorWithLock** 类中的方法正常工作。

5. 定义 **DoorWithCodeLock** 类，该类扩展（继承）了 **DoorWithLock** 类，包括初始化为 0 的新 int 类型实例变量**代码**。

覆盖方法 **lockDoor()**，这样当门关闭并解锁时，该方法会从键盘上获取一个整数输入值，并将该值保存在实例变量**代码**中，然后将 **isLocked** 改为 true 并返回。此外，重载方法 **unlockDoor()**，以便在门被锁定时，从键盘输入一个整数，并与**代码**值进行比较。如果用户的输入与**代码**相符，门就会被打开，但如果用户的输入与**代码**不符，就会打印出 "代码无效！"的文本，门就会继续锁着。

6. 修改 **TestDoor** 的 **main()** 方法，以实例化 **DoorWithCodeLock**，而不是 **DoorWithLock**，并运行程序以测试密码锁是否正常工作。