



Java 核心技术(高阶)

第一章 Java语法糖

第五节 语法糖(4)

华东师范大学 陈良育



接口的默认方法(1)

- Java最初的设计中，接口的方法都是**没有实现的、公开的**
- Java 8推出接口的**默认方法/静态方法(都带实现的)**，为Lambda表达式提供支持

```
public interface Animal {  
    public void move();  
}
```

```
public interface NewAnimal {  
    public default void move()  
    {  
        System.out.println("I can move.");  
    }  
}
```



接口的默认方法(2)

- 接口的默认方法

- 以`default`关键字标注，其他的定义和普通函数一样
- 规则1: 默认方法不能重写Object中的方法
- 规则2: 实现类可以继承/重写父接口的默认方法
- 规则3: 接口可以继承/重写父接口的默认方法
- 规则4: 当父类和父接口都有(同名同参数)默认方法，子类继承父类的默认方法，这样可以兼容JDK7及以前的代码
- 规则5: 子类实现了2个接口(均有同名同参数的默认方法)，那么编译失败，必须在子类中重写这个default方法



接口的静态方法

- Java 8接口的静态方法 (带实现的)

- 该静态方法属于本接口的，不属于子类/子接口
- 子类(子接口)没有继承该静态方法，只能通过所在的接口名来调用

```
public interface StaticAnimal {  
    public static void move()  
    {  
        System.out.println("I can move");  
    }  
}
```

```
public interface StaticLandAnimal extends StaticAnimal {  
    //也继承不到StaticAnimal的move方法  
}
```

```
public class StaticSwan implements StaticAnimal {  
    public static void main(String[] args) {  
        StaticAnimal.move();  
        StaticLandAnimal.move(); //error  
        new StaticSwan().move(); //error  
    }  
}
```



接口的私有方法

- Java 9接口的私有方法 (带实现的)
 - 解决多个默认方法/静态方法的内容重复问题
 - 私有方法属于本接口，只在本接口内使用，不属于子类/子接口
 - 子类(子接口)没有继承该私有方法，也无法调用
 - 静态私有方法可以被静态/默认方法调用，非静态私有方法被默认方法调用



接口 vs 抽象类(1)

- 相同点(截止至Java 12以前, 接口和抽象类对比)
 - 都是抽象的, 都不能被实例化, 即不能被new
 - 都可以有实现方法
 - 都可以不需要继承者实现所有的方法



接口 vs 抽象类(2)

- 不同点(截止至Java 12以前，接口和抽象类对比)
 - 抽象类最多只能继承一个，接口可以实现多个
 - 接口的变量默认是public static final，且必须有初值，子类不能修改；而抽象类的变量默认是default，子类可以继承修改
 - 接口没有构造函数，抽象类有构造函数
 - 接口没有main函数，抽象类可以有main函数
 - 接口有public/default/private 的方法，抽象类有public/private/protected/ 不写关键字的(default)的方法

总结



- 接口的方法(8/9): 建议少用, 如有Lambda表达式需求, 可使用
 - 默认方法: default标注, 可以为子类/子接口所继承/改写
 - 静态方法: static标注, 只属于本接口, 不属于子类/子接口, 子类/子接口可访问
 - 私有方法: private标注, 只属于本接口, 不属于子类/子接口



谢谢!