

1. 定义一个父类 **Animal**，有 **eat** 和 **sleep** 方法  
定义一个 **Dog** 子类，覆盖 **eat** 方法，并且有自己特有的 **shout** 方法  
定义一个 **Cat** 子类，覆盖 **eat** 方法，并且有自己特有的 **climbTree** 方法  
定义一个 **Bird** 子类，覆盖 **eat** 方法，并且有自己特有的 **fly** 方法  
定义一个 **Fish** 子类，覆盖 **eat** 方法，并且有自己特有的 **swim** 方法  
要求：
  - ① 定义一个 **Animal** 数组，遍历此数组，统计数组中 **Dog** 的个数
  - ② 遍历 **Animal** 数组，调用 **fly** 方法
  - ③ 定义一个函数，函数名为 **getAnimal**（获取动物）  
如果金钱>50 则返回 **Fish** 对象  
如果金钱>200 则返回 **Bird** 对象  
如果金钱>3000 则返回 **Cat** 对象  
如果金钱>5000 则返回 **Dog** 对象  
并在主函数中调用此函数
2. 定义一个父类 **Person**，有 **play** 和 **sleep** 方法  
定义一个子类 **Teacher**，覆盖 **play** 方法，并定义 **teach** 方法  
定义一个子类 **Student**，覆盖 **play** 方法，并定义 **study** 方法  
定义一个子类 **Worker**，覆盖 **play** 方法，并定义 **work** 方法  
定义一个子类 **Driver**（司机），覆盖 **play** 方法，并定义 **drive**（驾驶）方法  
要求：
  - ① 定义一个 **Person** 数组，遍历此数组，调用 **play** 方法
  - ② 遍历此数组，统计数组中学生个数
  - ③ 定义一个函数，接收一个 **Person** 类型的参数  
如果这个参数是 **Teacher**，则调用 **teach** 方法  
如果这个参数是 **Student**，则调用 **teach** 方法  
如果这个参数是 **Worker**，则调用 **teach** 方法  
如果这个参数是 **Driver**，则调用 **drive** 方法

3.（多态）有如下代码

```
class Super{
    public void m(){
        foo();
    }
    public void foo(){
        System.out.println("foo() in Super");
    }
}
class Sub extends Super{
    public void foo(){
        System.out.println("foo() in Sub");
    }
}
public class TestSuperSub{
```

```

        public static void main(String args[]){
            Super s = new Sub();
            s.m();
        }
    }

```

选择正确答案

- A. 程序编译不通过
- B. 编译通过，输出 `foo() in Super`
- C. 编译通过，输出 `foo() in Sub`

4. 如何判断引用中指向的对象是否是某一类型

5. Role 类

方法 `play () ; //打印 Role play();`

Dancer 类继承于 Role

方法 `play () ; //打印 Dancer play();`

Singer 类继承于 Role

方法 `play () ; //打印 Singer play();`

1) 定义上面三个类

2) 定义一个 Role 类型的数组，遍历此数组，调用 `paly` 方法