1、读程序，写结果：

class A {

public String Show(D obj) { return ("A and D"); }

public String Show(A obj) { return ("A and A"); }

}

class B extends A {

public String Show(B obj) { return ("B and B"); }

public String Show(A obj) { return ("B and A"); }

}

class C extends B {

public String Show(C obj) { return ("C and C"); }

public String Show(B obj) { return ("C and B"); }

}

class D extends B {

public String Show(D obj) { return ("D and D"); }

public String Show(B obj) { return ("D and B"); }

}

public class mainTest {

public static void main(String args[]){

A a1 = new A();

A a2 = new B();

B b = new B();

C c = new C();

D d = new D();

System.out.println(a1.Show(b));

System.out.println(a1.Show(c));

System.out.println(a1.Show(d));

System.out.println(a2.Show(b));

System.out.println(a2.Show(c));

**System.out.println(a2.Show(d));**

System.out.println(b.Show(b));

System.out.println(b.Show(c));

**System.out.println(b.Show(d));**

}}

2、读程序，写结果：

**class Base {**

**private String name = "base";**

**public Base() {**

tellName();}

**public void tellName() {**

System.*out.println("Base tell name: " + name);* }

}

**public class Dervied extends Base {**

**private String name = "dervied";**

**public Dervied() {**

tellName();}

**public void tellName() {**

System.*out.println("Dervied tell name: " + name);*}

**public static void main(String[] args){**

**new Dervied();**

}}

3、练习3：Example2：生成动物

1. ）循环通过标准输入端输入需要生成的动物，当遇到结束标志，则结束程序运行。
2. ）每次生成动物，通过标准输出端显示动物的信息。
3. ）动物的信息包括：目前所有动物的总数，当前这一类动物的总数。
4. ）整个程序结构用工厂模式设计，保证将来动物园有新的动物加入时，程序可扩展。