

Exercises: Inheritance/Polymorphism

For each of the Java programs below, identify whether or not the program is correct by writing **Correct** or **Incorrect**. For a Java program to be **Correct** it must both compile and run without errors. If the program is **Correct**, then write out what would be displayed to the console, if anything. If the program is **Incorrect**, then briefly explain why.

Supporting classes can be found on subsequent pages.

Problem 1

```
1 public class Driver {
2     public static void main( String[] args ) {
3         Dog d1 = new Dog("Fido", 10, "Phil");
4         Snake s42 = new Snake("Snuggles", 7, "Katie");
5         Exotic reference;
6         reference = d1;
7         System.out.println( reference );
8     }
9 }
```

Problem 2

```
1 public class Driver {
2     public static void main( String[] args ) {
3         Dinosaur dino;
4         TRex tina = new TRex("Tina", 26, "Talulah");
5         dino = tina;
6         System.out.println(dino.speak());
7     }
8 }
```

Problem 3

```
1 public class Driver {
2     public static void main( String[] args ) {
3         Pet p;
4         Snake sally = new Snake("Sally", 2, "Suzy");
5         p = sally;
6         System.out.println(p);
7     }
8 }
```

Problem 4

```
1 public class Driver {
2     public static void main( String[] args ) {
3         Dinosaur ex;
4         Tiger ricky = new Tiger("Ricky", 23, "Pi");
5         Brontosaurus booboo = new Brontosaurus("BooBoo", 3, "Yogi");
6         ex = ricky;
7         System.out.println(ex);
8         ex = booboo;
9         System.out.println(ex);
10    }
11 }
```

Problem 5

```
1 public class Driver {
2     public static void main( String[] args ) {
3         Exotic ex;
4         Tiger ricky = new Tiger("Ricky", 23, "Pi");
5         Brontosaurus booboo = new Brontosaurus("BooBoo", 3, "Yogi");
6         ex = ricky;
7         System.out.println(ex);
8         ex = booboo;
9         System.out.println(ex);
10    }
11 }
```

Problem 6

What is the output of the following program?

```
1 public class Driver {
2     public static void main(String[] args) {
3         Cat c1 = new Cat("Betty", 3, "Barb");
4         Dog d1 = new Dog("Rufus", 4, "Ralph");
5         Raptor r1 = new Raptor("Snowball", 10, "Sally");
6         Tiger t1 = new Tiger("Fluffy", 3, "Frank");
7
8         c1.setCutenessFactor(9);
9         r1.setDangerFactor(2);
10
11        System.out.println(c1);
12        System.out.println(t1);
13        System.out.println(d1);
14        System.out.println(r1);
15    }
16 }
```

Problem 7

What is the output of the following program?

```
1 public class Driver {
2     public static void main(String[] args) {
3         Brontosaurus b1 = new Brontosaurus("Barry", 100, "Bart");
4         Snake s1 = new Snake("Sneaky", 2, "Severus");
5         Cat c1 = new Cat("Nellie", 6, "Nell");
6         Dog d1 = new Dog("Chip", 5, "Chris");
7
8         s1.setDangerFactor(10);
9         d1.setCutenessFactor( c1.getCutenessFactor() );
10
11        System.out.println(c1);
12        System.out.println(b1);
13        System.out.println(s1);
14        System.out.println(d1);
15    }
16 }
```

Problem 8

What is the output of the following program?

```
1 public class Driver {
2     public static void main(String[] args) {
3         Cat c1 = new Cat("Princess", 1, "Patti");
4         Cat c2 = new Cat("Issy", 2, "Iris");
5         TRex trex = new TRex("FancyPants", 50, "Frank");
6         Tiger tgr = new Tiger("Tiny", 15, "Terry");
7
8         c1.setCutenessFactor(9);
9         tgr.setDangerFactor(2);
10
11        System.out.println(c1);
12        System.out.println(trex);
13        System.out.println(c2);
14        System.out.println(tgr);
15    }
16 }
```

Problem 9

What is the output of the following program?

```
1 public class Driver {
2     public static void main(String[] args) {
3         Dog d1 = new Dog("Fido", 10, "Phil");
4         Dog d2 = new Dog("Scruffy", 12, "Simon");
5         Snake s42 = new Snake("Snuggles", 7, "Katie");
6         Raptor r2d2 = new Raptor("Sharpie", 88, "Steve");
7
8         d2.setCutenessFactor(2);
9         r2d2.setDangerFactor(10);
10
11        System.out.println(d1);
12        System.out.println(d2);
13        System.out.println(s42);
14        System.out.println(r2d2);
15    }
16 }
```

Problem 10

What is the output of the following program?

```
1 public class Driver {
2     public static void main(String[] args) {
3         Brontosaurus phineas = new Brontosaurus("Phineas", 12, "Mom");
4         Tiger ferb = new Tiger("Ferb", 12, "Mom");
5         Raptor perry = new Raptor("Perry", 6, "Phineas & Ferb");
6         Cat candace = new Cat("Candace", 16, "Mom");
7
8         phineas.setDangerFactor( ferb.getDangerFactor() );
9
10
11        System.out.println(phineas);
12        System.out.println(ferb);
13        System.out.println(perry);
14        System.out.println(candace);
15    }
16 }
```

Supporting Classes

```
1 public class Pet {
2     protected String name;
3     protected int age;
4     protected String owner;
5     protected String type;
6
7     public Pet(String n, int a, String o, String t) {
8         name = n;
9         age = a;
10        owner = o;
11        type = t;
12    }
13
14    public String getOwner() {
15        return owner;
16    }
17
18    public String getName() {
19        return name;
20    }
21
22    public String getType() {
23        return type;
24    }
25
26    public String toString() {
27        return name + " the " + type;
28    }
29
30    public int getAge() {
31        return age;
32    }
33
34    public String speak() {
35        return "I am a " + type + " and I say ";
36    }
37 }
```



```
1 public class Domestic extends Pet {
2     protected int cutenessFactor;
3
4     public Domestic(String n, int a, String o, String t) {
5         super(n, a, o, t);
6         if (t.equals("cat")) {
7             cutenessFactor = 7;
8         } else if (t.equals("dog")) {
9             cutenessFactor = 8;
10        } else {
11            cutenessFactor = 4;
12        }
13    }
14
15    public int getCutenessFactor() {
16        return cutenessFactor;
17    }
18
19    public void setCutenessFactor(int c) {
20        cutenessFactor = c;
21    }
22
23    public String toString() {
24        return "I am " + super.toString()
25            + " and have a cuteness factor of "
26            + cutenessFactor;
27    }
28
29 }
```

```
1 public class Cat extends Domestic {
2     public Cat(String n, int a, String o) {
3         super(n, a, o, "cat");
4     }
5
6     public String speak() {
7         return super.speak() + "meow!";
8     }
9 }
```

```
1 public class Dog extends Domestic {
2     public Dog(String n, int a, String o) {
3         super(n, a, o, "dog");
4     }
5
6     public String speak() {
7         return super.speak() + "bark!";
8     }
9 }
```

```
1 public class Exotic extends Pet {
2     protected int dangerFactor;
3
4     public Exotic(String n, int a, String o, String t) {
5         super(n, a, o, t);
6         if (t.equals("snake")) {
7             dangerFactor = 7;
8         } else if (t.equals("tiger")) {
9             dangerFactor = 8;
10        } else if (t.equals("trex")) {
11            dangerFactor = 10;
12        } else if (t.equals("raptor")) {
13            dangerFactor = 9;
14        } else if (t.equals("brontosaurus")) {
15            dangerFactor = 5;
16        }
17    }
18
19    public int getDangerFactor() {
20        return dangerFactor;
21    }
22
23    public void setDangerFactor(int d) {
24        dangerFactor = d;
25    }
26
27    public String toString() {
28        return "I am " + super.toString()
29            + " and have a DANGER factor of "
30            + dangerFactor;
31    }
32 }
```

```
1 public class Snake extends Exotic {
2     public Snake(String n, int a, String o) {
3         super(n, a, o, "snake");
4     }
5
6     public String speak() {
7         return super.speak() + "sssssssss!";
8     }
9 }
```

```
1 public class Tiger extends Exotic {
2     public Tiger(String n, int a, String o) {
3         super(n, a, o, "tiger");
4     }
5
6     public String speak() {
7         return super.speak() + "rawr!";
8     }
9 }
```

```
1 public class Dinosaur extends Exotic {
2     public Dinosaur(String n, int a, String o, String t) {
3         super(n, a, o, t);
4     }
5
6     public String toString() {
7         return super.toString().toUpperCase();
8     }
9 }
```

```
1 public class Brontosaurus extends Dinosaur {
2     public Brontosaurus(String n, int a, String o) {
3         super(n, a, o, "brontosaurus");
4     }
5
6     public String speak() {
7         return super.speak().toUpperCase() + "munch munch.";
8     }
9 }
```

```
1 public class Raptor extends Dinosaur {
2     public Raptor(String n, int a, String o) {
3         super(n, a, o, "raptor");
4     }
5
6     public String speak() {
7         return super.speak().toUpperCase() + "SCREEECH!";
8     }
9 }
```

```
1 public class TRex extends Dinosaur {
2     public TRex(String n, int a, String o) {
3         super(n, a, o, "trex");
4     }
5
6     public String speak() {
7         return super.speak().toUpperCase() + "RAAAWWRR!";
8     }
9 }
```