Assessment

LATEST SUBMISSION GRADE

100%

1. Polymorphism refers to _____ . **1 / 1 point**

- ○ The ability to cast a reference to a different type.
- ○ The ability to have multiple inheritance
- ◉ The ability for many classes to be invoked through a common interface
- ○ The ability for a class to implement poly (multiple) interfaces

✓ **Correct**
Correct.

2. What is the output of this code? **1 / 1 point**

```
publicclass Test {
publicstaticclass A {
publicint getA() { return 5; }
  }
publicstaticclass B extends A {
publicint getA() { return 10; }
  }
publicstaticvoid main(String[] args) {
    A a1 = new A(), a2 = new B();
    B b1 = new B();
    System.out.printf("A1: %d, A2: %d, B1: %d\n", a1.getA(), a2.getA(), b1.getA());
  }
}
```

- A1: 10, A2: 10, B1: 10
- A1: 5, A2: 10, B1: 10
- A1: 5, A2: 5, B1: 5
- A1: 5, A2: 5, B1: 10

✓ **Correct**
Correct.

3. What is the output of this code? **1 / 1 point**

```
publicclass Test {
publicstaticclass A {
publicint getA() { return 5; }
  }
publicstaticclass B extends A {
publicint getA() { return 10; }
    public int getB() { return 15; }
  }
publicstaticvoid main(String[] args) {
  A a1 = new A();
  B b1 = new B();
  System.out.printf("AA: %d, AB: %d, BA: %d, BB: %d\n", a1.getA(), a1.getB(), b1.getA(), b1.getB());
 }
}
```

- AA: 10, BA: 10, BB: 15
- It doesn't compile.
- AA: 5, BA: 5, BB: 10
- AA: 5, BA: 10, BB: 15

✓ **Correct**
Correct.

4. What is the output of this code?

1 / 1 point

```
publicclass Test {
publicstaticclass A {
publicint getA() { return 5; }
 }
publicstaticclass B extends A {
publicint getA() { return 10; }
   public int getB() { return 15; }
 }
publicstaticvoid main(String[] args) {
   A a1 = new A();
   B b1 = new B();
   System.out.printf("AA: %d, BA: %d, BB: %d\n", a1.getA(), b1.getA(), b1.getB());
 }
}
```

- ○ It doesn't compile.
- ○ AA: 5, BA: 5, BB: 10
- ○ AA: 10, BA: 10, BB: 15
- ◉ AA: 5, BA: 10, BB: 15

✓ **Correct**
Correct.

5. When is polymorphism the most important?

1 / 1 point

- ◉ When we need to process a whole group of similar objects.
- ○ When dealing with a single object.

✓ **Correct**
Right.

6. _____ is a powerful technique for grouping objects by inheritance or the implementation of common interfaces. **1 / 1 point**

- ○ Method
- ○ References
- ◉ Polymorphism

✓ **Correct**
Correct.

7. Which of the following codes will follow these steps in order when executing? **1 / 1 point**

1. Java creates a new entry in the reference table.

2. Java enters the type for the new entry as **Car**.

3. Java enters the name of the new entry as **myCar**.

4. Java sets up a block of memory to store the attributes of a **SportsCar** and enters the address of this memory in the attribute block field of the new entry.

5. Java sets up a block of memory to store the methods of a **SportsCar** and enters the address of this memory in the method block field of the new entry.

- ○ myCar SportsCar = new Car ( ) ;
- ◉ Car myCar = new SportsCar ( ) ;
- ○ new Car = myCar SportsCar ( ) ;

✓ **Correct**
That's right!

8. In creating object references, if the method you are calling is NOT part of the object or one of its superclasses, what will happen? **1 / 1 point**

- ◉ A compiler error will be generated.
- ○ Java will store the method and add the address in the method block field of the new entry.
- ○ The method will be created in the method block automatically.

✓ **Correct**
Right.

9. You should only refer to classes when necessary, like when using _____ . 1 / 1 point

- ○ interfaces
- ○ members
- ◉ new

✓ **Correct**
Correct.

10. True or false: References, including members, local variables, and parameters, should be defined in terms of interfaces. 1 / 1 point

- ◉ True
- ○ False

✓ **Correct**
Right!