1. Which is not a key concept of Object-Orientation? 1 / 1 point

- ○ Inheritance
- ○ Encapsulation
- ◉ Garbage Collection
- ○ Polymorphism

Correct
Feedback: Although garbage collection is common with O-O runtimes, it is not part of O-O.

2. Which of these best describes Encapsulation? 1 / 1 point

- ○ Wrapping **Tyvek**® around a house
- ○ Providing collection classes to hold bags of data
- ○ Freezing code and data so that it cannot change
- ◉ Hiding code and data behind a contract

Correct
Correct. Implementation hiding.

3. Classes contain two main sections. What are they? 1 / 1 point

- ○ Unique names and addresses
- ○ Goals
- ◉ Attributes (data) and Methods (code)
- ○ Points of view

Correct
Correct.

4. True or false: In a class definition, instance data (attributes) must be declared before (as in the order in which it is declared) it can be referenced in method source code. 1 / 1 point

- ○ True
- ◉ False

Correct
Correct. The order does not matter.

5. A method signature consists of _____ . 1 / 1 point

- ○ The method name, and parameter names
- ○ The access modifier, return type and method name
- ◉ The method name and the ordered types of the parameters
- ○ The return type, method name and parameter names

Correct
Correct.

6. Why did OO programming become popular? 1 / 1 point

- ○ because C++ compilers were having technical difficulties
- ○ because new hardware required objects to work in 64-bit systems
- ◉ to respond to the increased complexity of business programs
- ○ because when OO was created in 1999, people liked it

Correct
The complexity of the average business program has increased exponentially over the last thirty years

7. How many responsibilities should a java class have? 1 / 1 point

- ○ 4
- ○ 3
- ◉ 1
- ○ 2

Correct
Correct. According to the "Single Responsibility Principle" a class should do just one thing.

8. True or False? Attributes describe the actions the class must be able to perform. 1 / 1 point

- ○ True
- ◉ False

Correct
Attributes define the data an object can hold.

9. _____ is when code is scattered accross many different programs that does the same--or nearly the same--thing. 1 / 1 point

- ○ Functional separation
- ◉ Code redundancy
- ○ Encapsulation

Correct
That's tight!

10. The ability to call a common interface on disparate implementations ia called _____. 1 / 1 point

- ○ inheritance
- ◉ polymorphism
- ○ encapsulation

Correct
Right!