Exercise 2

09-17-2023

Kohl Johnson

# Part 1 Files

**Step 3.C Screenshot**

A screenshot of a computer

Description automatically generated

**Step 3.D Response:** The output within the method determineHealth() was displayed because either SuperHero object called it. At the end of the while loop, when one of the SuperHero objects is defeated, the loop ends and displays whichever SuperHero object is victorious.

# Part 2 Files

**Step 3.C Screenshot**

A screenshot of a computer

Description automatically generated

**Step 3.D Response:** The output of "...with a power of #" was called because the Bomb and Gun classes are a subclass of Weapon, which contains the method called fireWeapon(int power). From weapon1 (bomb), the fireWeapon method was called with power of 10. With the weapon2 (gun), the fireWeapon method was called with power of 5.

**Step 4.C Screenshot**

A screenshot of a computer

Description automatically generated

**Step 4.D Response:** The output "In Bomb.fireWeapon()..." was displayed because the method was called from weapon1 (bomb) class. While there is another method with the same name in the Superclass, the subclasses method overrides it and is used instead.

**Step 5.D Response:** The output "In overloaded Gun.fireWeapon()" displayed because the method was called from weapon2, but had no parameters given. Lastly, "In Weapon.fireWeapon()..." was displayed because the method was called with super.fireWeapon(5) from the overloaded fireWeapon method in the Gun class.

**Step 6.D ScreenshotScreens screenshot of a computer

Description automatically generated**

**Step 6.F Bomb Class Screenshot**

**A screenshot of a computer

Description automatically generated**

**Step 6.F Gun Class Screenshot**

**A screenshot of a computer

Description automatically generated**

**Step 6.F Weapon Class Screenshot**

**A screenshot of a computer

Description automatically generated**

**Step 7.A Screenshot**

**A screenshot of a computer

Description automatically generated**

**Step 7.A Response:** There was an error because the class must be an abstract class to use abstract methods.

**Step 7.B Screenshot**

**A screenshot of a computer

Description automatically generated**

**Step 7.B Response:** There was an error because the final method defined in the Superclass cannot be overridden from the Gun and Bomb classes.

**Step 7.C Screenshot**

**A screenshot of a computer

Description automatically generated**

**Step 7.C Response:** The error occurred because abstract methods do not define a body.

# Part 3 Files

**Step 2.H Screenshot**

**A screenshot of a computer

Description automatically generated**

**Step 2.I Response:** The comparisons between Person1 and Person2 are expected. They are not equal because of the firstName. Person3 and Person1 should have been from what I understand... so why it said not equal I have yet to figure out.

**Step 5.A Screenshot**

**A screenshot of a computer

Description automatically generated**

**Step 5.B Response:** The "not identical using ==" was displayed because Person1 does not equal Person2. The "not identical using equals()" was displayed because Person1's first name is not the same as Person2's first name. The "these persons are identical using equals()" was displayed because Person3 is a copy of Person1.

**Step 5.C Response:** a toString() method can help with my milestone project with printing out a list format of all the products within the store, shopping cart, and with the admin application. Equals() will be helpful comparing products to one another, especially when it comes to restocking the inventory manager.

**Step 5.D Response:** The @Override annotation shows that the child class method overrides the parent classes method. Its good practice to use this annotation to keep track of which methods are actually being used and therefore giving you a greater idea of what is actually going on behind the scenes.

# Part 4 Files

**Setting Breakpoints Screenshot**

**A screenshot of a computer

Description automatically generated**

**Inspecting Variables Screenshot**

**A screenshot of a computer

Description automatically generated**

**Stepping Tasks Screenshot**

**A screenshot of a computer

Description automatically generated**

**Inspecting Call Stack Screenshot**

**A screenshot of a computer

Description automatically generated**