Humna Sultan
Mr. Hendricks
AP CS A
13 November 2020

11/9 - 11/13 Work

10.1: An Introduction to Inheritance

- Inheritance is a mechanism for extending existing classes by adding methods and fields.
- The more general class is called a superclass. The more specialized class that inherits from the superclass is called the subclass.
- Every class extends the Object class either directly or indirectly.
- Inheriting from a class differs from implementing an interface: The subclass inherits behavior and state from the superclass.
- Inheritance allows for code reuse.
- When defining a subclass, you specify added instance fields, added methods, and changed or overridden methods.

10.2: Inheritance Hierarchies

- Sets of classes can form complex inheritance hierarchies.
- The superclass contains features and behaviors that are common to all of the classes that you are creating
- The subclasses support all methods from the superclass, but their implementations may be modified to match the specialized purposes of the subclasses.
- Subclasses are free to introduce additional methods.

10.3: Inheriting Instance Fields and Methods

- You can override methods (use same parameter list) from the superclass
- You can inherit methods from the superclass (you automatically inherit it)
- You can define new methods that don't exist in the superclass.
  - The new method can only be applied to the subclass objects.
- A subclass has no access to private fields of its superclass.
- Use the super keyword to call a method of the superclass.

10.4: Subclass Construction

- To call the superclass constructor, you use the super keyword in the first statement of the subclass constructor.
- If a subclass constructor does not call the superclass constructor, the superclass is constructed with its default constructor (that is, the constructor that has no parameters).
  - However, if all constructors of the superclass require parameters, then the compiler reports an error.

- When the keyword super is followed by a parenthesis, it indicates a call to the superclass constructor.
- If super is followed by a period and a method name, on the other hand, it indicates a call to a superclass method, as you saw in the preceding section.

10.5: Converting Between Subclass and Superclass Types

- Subclass references can be converted to superclass references.
- Conversion of references is different from a numerical conversion, such as a conversion from an integer to a floating-point number.
- The instanceof operator tests whether an object belongs to a particular type.

10.6: Polymorphism

- In Java, the type of a variable does not completely determine the type of the object to which it refers.
- In Java, method calls are always determined by the type of the actual object, not the type of the object reference.
- You should make calculations manually and ensure that the polymorphic methods work correctly.
- Polymorphism means "many forms", and it occurs when we have many classes that are related to each other by inheritance.

10.7: Access Control

- Java has four levels of controlling access to fields, methods, and classes:
    - Public access
    - Private access
    - Protected access
    - Package access
- A field or method that is not declared as public, private, or protected can be accessed by all classes in the same package, which is usually not desirable.
- Classes and interfaces can have public or package access.
    - Classes that are generally useful should have public access.
    - Classes that are used for implementation reasons should have package access.

10.8: Object: The Cosmic Superclass

- Define the toString method to yield a string that describes the object state.
- Define the equals method to test whether two objects have equal state.
- When redefining the method, you are not allowed to change the object signature.
- Note that you must use equals to compare object fields, but use == to compare number fields.
- The clone method makes a new object with the same state as an existing object.

10.9: Using Inheritance to Customize Frames

- Define a JFrame subclass for a complex frame.
- The frame class makes it easier to organize the code that constructs the user-interface elements.
- You can include helper methods to make the initialization code less complex

10.10: Processing Text Input

- Most graphical programs collect text input through text fields.
- The JTextField class provides a text field. When you construct a text field, you need to supply the width—the approximate number of characters that you expect the user to type.
- Use JTextField components to provide space for user input. Place a JLabel next to each text field.

10.11: Text Areas

- A text field holds a single line of text. To display multiple lines of text, use the JTextArea class.
- Use a JTextArea to show multiple lines of text.
- You can add scroll bars to any component with a JScrollPane.