The Learning House

427 S 4th Street #300

Louisville KY 40202

Java Object-Oriented Concepts

Lesson 9 - Interfaces

SOFTWARE GUILD

Objectives

- Understand:
  - How to declare and implement Java interfaces
  - Restrictions on Java interfaces
  - How to extend Java interfaces
  - Polymorphism with respect to interfaces

Interfaces

- An interface is a contract
- Defines methods that all classes that adhere to the contract must implement
- For example - the Colorable interface might say “Every class that signs this contract will provide a getColor() and setColor() method.”

Declaring Interfaces

- Almost identical to class declaration
- Use interface keyword instead of class keyword:

```
public interface Colorable {
	public void setColor(String color);
	public String getColor();
}
```

Implementing an Interface

- Classes must declare that they implement an interface:

```
public class Ball implements Colorable {
	// some methods

	public void setColor(String color) {
		// implementation
	}
	public String getColor() {
		// implementation
	}
}
```

Interface Restrictions

- No member fields
- Can define constants
- None of the methods can have implementations

Implementing Multiple Interfaces

- Classes can implement more than one interface
- For example: Colorable and Debuggable
- Your class must implement all of the methods defined for both Interfaces

Extending Interfaces

- Interfaces can be extended in the same way that classes can be extended:

```
public interface Debuggable {
     public void displayStatus(String id);
     public void displayError(String error);
}
public interface DebugLogging extends
Debuggable {
     public void logStatus(String id);
     public void logError(String error);
}
```

Polymorphic Interfaces

- Interfaces are can be be treated polymorphically
- For example, an object that implements DebugLogging can be treated as a DebugLogging type or a Debuggable type

Examples

- RPG Satchel
  - Item/MagicalItem

Assignment

- GameBot