

Core Java

9. Classes: Inheritance

Inheritance

- Specializing general classes into more specific ones
- Often known as interface (abstraction) and implementation (concrete)
- All classes in Java inherit from *java.lang.Object*
- It is possible to expose more methods than what you inherit from, not lesser

Syntax

```
[access-specifier] [other-specifier]  
class classname [implements  
interfacename,] [extends classname]  
{  
}
```

Interface

- Interfaces are way to specify object behaviors
- The focus is on methods than variables
- Interfaces can not be instantiated directly
- They can be used to invoke underlying implementation

Rule of Inheritance

- A Superclass pointer may reference a Subclass object
- An Interface can not have objects of it's own, but can point to objects of implementing classes
- The functionality is limited to whatever functionality is exposed by the current object's type
- Methods may be overridden or not defined (abstract classes)

Demonstration

- Compile and Execute a few programs

Questions?