Unit 9 → Inheritance

Inheritance

- EXAMPLE→ Person class
    - Instance variables of name and birthday (“has a” relationship)
    - A student is a person
- “has a” relationships help determine an instance variable
- “is a” relationships help determine class
- Superclasses
    - Extends parent class
    - Able to use methods from superclass in subclass without having to redeclare it
    - Syntax
        - `public class` SubClass `extends` SuperClass `{}`
    - `super()` must always come first in a constructor

Writing Constructers for the Subclass

- `super()`
    - Must *always* come first an a constructor
    - Passes parameters from a superclass to a subclass
    - Used to make a call to a superclass
        - If superclass doesn’t have anything, don’t include
- Subclasses *do not* inherit a constructor from the superclass
    - Subclass *must* have its own constructor
    - If there isn’t one present, Java will have a default one to be used

Overriding Methods

- Allows subclass to redfine a method instead of using a superclasses version
- Used when superclass and subclass has the same signature `public` class
- Override vs Overload
    - Overriding
        - In subclass and superclass
        - Same name, *same* parameters
        - `@Override` used to override a method
            - Not necessary, but good practice
            - Helps with debugging
            - Make code more readable
            -
    - Overload
        - Same class
        - Same name, *different* parameters

`super` Keyword

- `super.`*`method`*`()` goes to the superclass and uses the method called there
    - Used tp c;ass a superclass's method with correct parameters
- `super` is similar to `this`
    - `super` refers to classes
    - `this` returns to objects

Referencing

- Polymorphism is the capability of a method to do different things depending on the object it's acting on
    - Acts differently among classes
    - Poly=many & morph=forms
    - Take on different forms depending on implementation
- *SuperClass* `nameOfObject= new` *SubClass*`();`
- `ArrayList`s can have different subclasses of the same superclass