CS115 Week 3
Class Structure, Comments, Class Attributes (Instance Variables), Object Interaction

BRING A FLASH DRIVE WITH:
Trick the Turtle (with all your own previous updates)

Generic Class Structure
import greenfoot.*;
// makes classes from elsewhere available to our project (scenario)
// everything from greenfoot package
/**
 * A comment describing the class. (not required by the compiler, but necessary for good programming. A /** comment is a java doc comment and is used in the automatically generated API
 */
public class Classname extends Superclassname
{
    Attributes
    Constructors
    Methods
}

Public vs Private
- public vs private for methods – can the method be called by someone outside the class or not?
- public vs private for attributes – can the variable be accessed directly by someone outside the class or not?

Comments
// single line comment
/
*/
* multiple line comment
*
*/
/**
* java Doc comments (right before a class header or * right before a method header, they form the API
*/

Class Attributes (Instance Variables)
- Attributes are the data that describes an object.
- We use variables to represent attributes.
- Variables are named data locations.
  ■ primitive data types like integers, real numbers, characters, or booleans.
  ■ instantiated objects, use the class name as type

Declaring an Instance Variable (and initializing)
accessModifier dataType identifierName;
private int lettuceEaten=0;
// optional initialization, or in the constructor
Assignment Statement
- Assigns an object or value to a variable
- Uses the "=" assignment symbol
- When an object is assigned to a variable, the variable contains a reference (or pointer) to the object. More than one variable can reference an object.

Four Types of Flow of Control
- Sequential Processing
  - Execute instructions in order
- Method Call
  - Jump to code in method, then return
- Selection
  - Choose code to execute based on data value
- Looping or Iteration
  - Repeat operations for multiple data values

Object Interaction
- Object Oriented Programming is all about interacting objects, we have seen a little of this with objects “seeing” other objects and “eating” them.
- An object can also invoke a method of another object

Constructing Objects
- You need to both declare the variable name for the object and construct (instantiate) the object by calling its constructor (same name as the class)
  ```java
  ClassName objectVariableName;
  objectVariableName = new ClassName();
  ```

Constructor
- A constructor of a class is a special kind of method that is needs to be called whenever a new instance (object) is created.
- Constructors are inherited (just like all methods).
- If you do not code a constructor a default constructor with no arguments is created for you.

Object Interaction (continued)
- For objects to interact the classes need to provide a way for the objects to reference each other. See the attributes of Actor class and getWorld() method
- dot notation – needed to call a method of another object (other than this object’s methods)