**Directions**: Create a Python module on your Z:\GameDev folder named oop\_ex3.py. Add the following code to the module. Do not forget to test!!!

***Java Directions****: If you are doing this as a Java exercise, you will need to create a class file for each class and test it in a main method. Sample Java main method follows:*

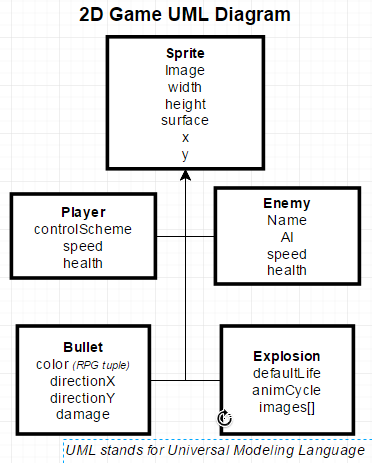
*Public static void main(String[] args) {*

*}*

**Inheritance**: When a class copies all methods and properties from another class.

* **Superclass** – The class from whom the methods and properties are being copied. *It is also called the parent or base class.*
* **Subclass** – The class that is copying (inheriting) the methods and properties from the Superclass. *It is also called the child class.*

**For Example:** In the example below, the Player, Enemy, Bullet, and Explosion classes are subclasses of the Sprite class. When we create an instance of the Player class, all methods and properties from Sprite are included in the Player class. *The same holds true for instances of the Enemy, Bullet and Explosion classes.*



***Note:*** *UML Diagrams are also known as “Object Diagrams”.*

Look here [**For help on crating sub classes whose superclass have variables**](http://interactivepython.org/runestone/static/JavaReview/OOBasics/ooInheritanceAndConstructors.html)

**Sprite Class**

1. Define a class named Sprite. This class should have all the attributes *(instance variables)* defined in the above diagram and several methods:
2. playSound():
3. Getter and setter methods for all attributes *(see below for details)*
   1. Attributes
      1. image *(String - path to image)*
      2. width *(integer)*
      3. height *(integer)*
      4. surface *(String the parent surface of the Sprite i.e. “DISPLAYSURF”)*
      5. x *(integer describing position)*
      6. y *(integer describing position*)
   2. Methods:
      1. playSound() method that **prints** “The Sprite makes a sound”.
   3. Getters and Setters for all instance variables. Example Getter and Setter methods for the image attribute:

def setImage(self, image):

self.image = image

def getImage(self):

return self.image

1. Create a constructor for the Sprite class that:
   1. Accepts a value for each attribute *(instance variable)* described above.
   2. Prints the value of the surface attribute using the getSurface() meth0d.
2. Create an instance of the Sprite class passing values for all attributes defined above into the constructor.
3. Execute the playSound() method

***Continued on next page***

**Player Class *(Subclass of Sprite)***

1. Define a class named Player. This class must be a subclass of the Sprite class.
   1. Attributes:
      1. controlScheme
      2. speed
      3. health
   2. Methods:
      1. getControlScheme()
      2. getSpeed()
      3. getHealth()
      4. setControlScheme()
      5. setSpeed()
      6. setHealth()
      7. move() - Just print “*The Player Moves*” in this method.
      8. playSound() - Just print “*The player makes a sound*” in this method.
2. Create a constructor for the Player class that:
   1. Accepts the values controlScheme, speed, and health as parameters and sets the appropriate attributes.
   2. Prints the values of controlScheme, speed, and health using the appropriate “get” methods.

**Enemy class *(Subclass of Sprite)***

1. Define a class named Enemy. This class that is a subclass of the Sprite class.
   1. Attributes:
      1. name
      2. AI
      3. speed
      4. health
   2. Methods:
      1. Getters and setters for all the attributes above.
2. Create a constructor for the Enemy class that:
   1. Accepts the values name, AI, speed, and health as parameters and sets the appropriate instance variable values.
   2. Prints the values of: name, AI, speed, and health using the appropriate “getter” methods.
3. Create the playSound() method that:
   1. Prints “Enemy says bang”
   2. Calls the playSound() method of the superclass.

***Continued on next page***

**Test your Classes**

It is always a good idea to **test as you go**, but now we want to test the entire set of classes.

1. Create an instance of the Sprite class with the following attributes:
   1. Image = “\images\sprite.png”
   2. Values you select for width and height
   3. Surface = “DISPLAYSURF”
   4. Values you select for x and y
2. Create an instance of the Player class that has a controlScheme = “mouse”, speed = 25, and health = 50 in the constructor.
   1. Modify values inherited from the Sprite class (Superclass). Use the **“setter” methods** defined in the Sprite class to set the values of:
      1. image = “\images\player\_ship.png”
      2. width = 50 and height = 50
      3. Surface = “DISPLAYSURF”
      4. X = 100 and y = 500
   2. Print the values of all properties of the Player instance *(including those inherited from the Sprite class.)*
      1. Optionally: Feel free to create a printValues() method that prints the values of all variables for you, since you repeat this below.
   3. Set the value of the controlScheme instance variable to “keyboard” using the setControlScheme() method.
   4. Set the value of the image instance variable to “screen” using the setImage() method.
   5. Print all instance variable values again (as done in a previous step)
3. Create an instance of the Enemy class that has the values: name = “creeper” AI = “follow-player”, speed = 20, and health = 30 passed into the constructor.
   1. Set the value of the AI instance variable to “run-away”using the setAI() method.
   2. Print the AI value of your Enemy using the getAI() method.
   3. Set the values of speed, x, and y using the appropriate “setter” methods.
   4. Print the value of speed, x and the y using the appropriate getter methods.
4. Call the playSound() and getImage() methods for the following object instances you created above:
   1. Sprite
   2. Play
   3. Enemy