Chapter 8 Worksheet

(1) What will happen when we add this actor to the world?

```
import greenfoot.*;  // (World, Actor, GreenfootImage, Greenfoot and MouseInfo)

public class BlowUpActor extends Actor
{
    private String imageName;

    public BlowUpActor(String image) {
        imageName = image;
        setImage(new GreenfootImage(imageName));
    }
}
```

We want the BlowUpActor's image to grow 30 times and then disappear. We are going to create an array of 30 images in different sizes and as act is being called, we are going to show each image until we reach the 30th. When we get to the 30th, the original image will disappear and we will insert a new Square objects into the world.

(2) Do you understand all the parts of this code? If not, now is the time to ask.

```
public class BlowUpActor extends Actor
{
    private String imageName;
    private int imageDisplayed;

    public BlowUpActor(String image) {
        imageName = image;
        initializeImages();
        setImage(new GreenfootImage(imageName));
        imageDisplayed = 0;
    }

    private void initializeImages() {
    }

    public void act() {
       grow();
    }

    private void grow() {
       imageDisplayed = imageDisplayed + 1;
    }
}
```

We will work on the last part of what the BlowUpActor does first – disappearing after displaying the images and then adding new Square objects into the world.

```
private void grow() {
   imageDisplayed = imageDisplayed + 1;
}
```

(3) Write the condition inside the if-statement that will determine when the last image has been displayed:

```
private void grow() {
   imageDisplayed = imageDisplayed + 1;

   if(                                    ) {
   }
}
```

(4) Write the line of code inside the body of the if-statement to remove the BlowUpActor from the world.

```
private void grow() {
   imageDisplayed = imageDisplayed + 1;

   if(imageDisplayed == 30) {




   }
}
```

(5) Write a loop that will execute 10 times. You can leave the body of the loop empty for now.

```
private void grow() {
   imageDisplayed = imageDisplayed + 1;

   if(imageDisplayed == 30) {
      getWorld().removeObject(this);






   }
}
```

(6) Fill in the code that will assign the variables x and y random values for the x and y placement of the Square object.

```
private void grow() {
   imageDisplayed = imageDisplayed + 1;
   if(imageDisplayed == 30) {
      getWorld().removeObject(this);

      for(int count = 0; count < 10; count = count + 1) {


         int x =


         int y =

      }
   }
}
```

(7) Write the code that will create a Square object and assign it to the variable s.

```
private void grow() {
   imageDisplayed = imageDisplayed + 1;
   if(imageDisplayed == 30) {
      getWorld().removeObject(this);
      for(int count = 0; count < 10; count = count + 1) {
         int x = Greenfoot.getRandomNumber(getWorld().getWidth());
         int y = Greenfoot.getRandomNumber(getWorld().getHeight());


         Square s =
      }
   }
}
```

(8) Write the code that will place the Square s into the world at location (x,y).

```
private void grow() {
   imageDisplayed = imageDisplayed + 1;
   if(imageDisplayed == 30) {
      getWorld().removeObject(this);
      for(int count = 0; count < 10; count = count + 1) {
         int x = Greenfoot.getRandomNumber(getWorld().getWidth());
         int y = Greenfoot.getRandomNumber(getWorld().getHeight());
         Square s = new Square(Greenfoot.getRandomNumber(200) + 6);




      }
   }
}
```

(9) Write the code that declares an instance variable whose type is an array of GreenfootImages. Its name should be images.

(10) Write the code that declares an instance variable whose type is a number. This instance variable should be declared to be a constant. That is, its value will never change. Its value should be 30. Its name should be NUMBER_OF_IMAGES.

(11) Write the code that creates an array of GreenfootImages of size NUMBER_OF_IMAGES and assigns it to the variable named images.

```
private void initializeImages() {




}
```

(12) Write the code that creates a variable named baseImage whose type is a GreenfootImage and whose value is set to be a new GreenfootImage created from the file named imageName.

```
private void initializeImages() {
    images = new GreenfootImage[NUMBER_OF_IMAGES];




}
```

Chapter 8 Worksheet

(13) Write the code for a loop that executes NUMBER_OF_IMAGES times. Leave the body of the loop empty for now.

```
private void initializeImages() {
    images = new GreenfootImage[NUMBER_OF_IMAGES];
    GreenfootImage baseImage = new GreenfootImage(imageName);







}
```

(14) Write the code that assigns each element of the array images a new GreenfootImage based off of the baseImage

```
private void initializeImages() {
    images = new GreenfootImage[NUMBER_OF_IMAGES];
    GreenfootImage baseImage = new GreenfootImage(imageName);

    for(int count = 0; count < NUMBER_OF_IMAGES; count = count + 1) {






    }
}
```

(15) Write the code to scale each of the GreenfootImages in the array to be size x size.

```
private void initializeImages() {
    images = new GreenfootImage[NUMBER_OF_IMAGES];
    GreenfootImage baseImage = new GreenfootImage(imageName);

    for(int count = 0; count < NUMBER_OF_IMAGES; count = count + 1) {
      images[count] = new GreenfootImage(baseImage);

      int size = (count + 1) * (baseImage.getWidth()/NUMBER_OF_IMAGES);







    }
}
```