Matthew Guthrie

6/18/17

A5 Questions

**1. Open Picture.java and look for the method getPixels2D. Is it there?**

The method getPixels2D is not in Picture.java.

**2. Open SimplePicture.java and look for the method getPixels2D. Is it there?**

The method getPixels2D is in SimplePicture.java.

**3. Does the following code compile? DigitalPicture p = new DigitalPicture();**

No, that code does not compile. DigitalPicture is an interface, so DigitalPicture objects cannot be created.

**4. Assuming that a no-argument constructor exists for SimplePicture, would the following code compile? DigitalPicture p = new SimplePicture();**

This code would compile. SimplePicture is not abstract or an interface, and it implements DigitialPicture. This means that SimplePicture objects can be created, and the variable they are assigned to can be of type DigitalPicture.

**5. Assuming that a no-argument constructor exists for Picture, does the following code compile? DigitalPicture p = new Picture();**

Yes, this code compiles. Picture extends SimplePicture. Since SimplePicture implements DigitalPicture, all of it’s subclasses, including Picture, can be assigned to variables of type DigitalPicture.

**6. Assuming that a no-argument constructor exists for Picture, does the following code compile? SimplePicture p = new Picture();**

Yes, this code compiles. Since Picture extends SimplePicture, a Picture object can be assigned to a SimplePicture variable.

**7. Assuming that a no-argument constructor exists for SimplePicture, does the following code compile? Picture p = new SimplePicture();**

This code will not compile. SimplePicture is a superclass of Picture, meaning that SimplePicture objects cannot be assigned to Picture variables.

A7 Questions

**1. How many times would the body of this nested for loop execute?**

**for (int row = 7; row < 17; row++) for (int col = 6; col < 15; col++)**

The body of this loop will execute (17-7)\*(15-6) times, or 90 times.

**2. How many times would the body of this nested for loop execute?**

**for (int row = 5; row <= 11; row++) for (int col = 3; col <= 18; col++)**

The body of this loop will execute (11+1-5)\*(18+1-3) times, or 119 times