CSE8A Lecture 4

- IN TERM EXAM 1 THIS FRIDAY
  - Cover everything through last Monday's class
  - Similar to PI, reading questions. End of chapter questions are good to study.
  - Through Lecture 3 – Slide #7 (Chapter 3 – no explicit questions about "this reference")
  - Attend discussion section today Wed: 2-3pm (Center 216)
    Wed: 3-4pm (Center 109)
    Wed: 4-5pm (Center 109)
- PSA2 published soon – watch weebly web page; due next Monday
  - Find a partner for PSA2… pixelObjst on the Piazza forum
- Interviews for PSA1 deadline - midnight tonight
- Turnin: turn in as many times as you want, but not after deadline
- PSAs: 24 hour early turn-in bonus starts PSA2. Interview always 20% of PSA grade.

Lab this week

- More experience with “methods” in a Picture class (makeRed)
- A “tester” application
  - Opens a Picture
  - Shows the Picture
  - Makes the Picture red (by calling a method)
  - Shows the (modified, red) Picture

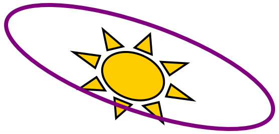
Open Lab Hours

- Many, in B250 (NOT B240)
- Procedure
  - Arrive in Lab, work.
  - When you have a question, look on the board for the tutor queue.
  - Add your name and location
  - Tutor will find you

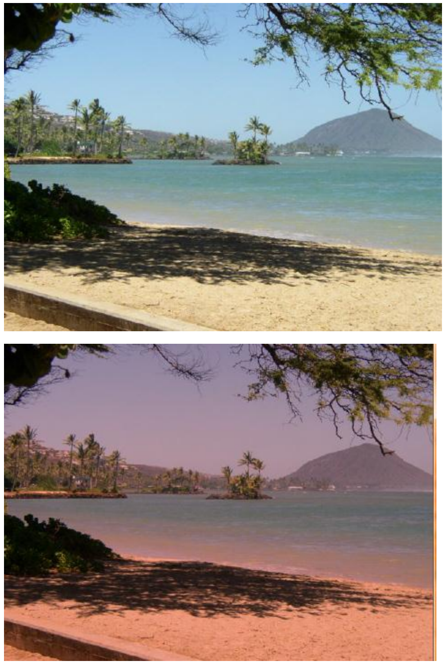
CLICKERS OUT!

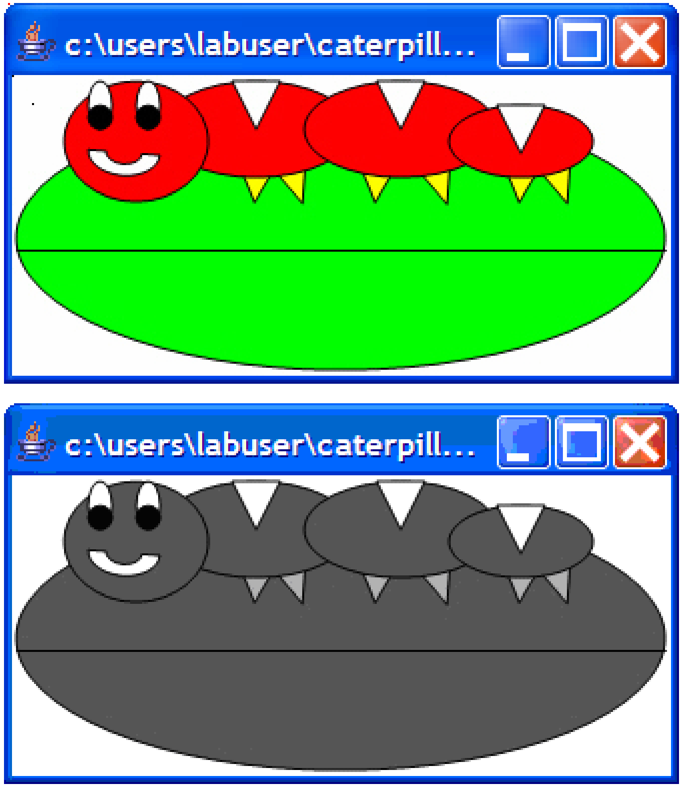
Today in CSE 8A

- Thinking *loopily* **`for`** a **`while`**



- Memory models
- Red, green and blue, oh my!

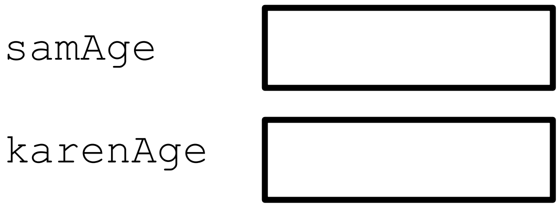




CS Concept: Primitives vs. Objects

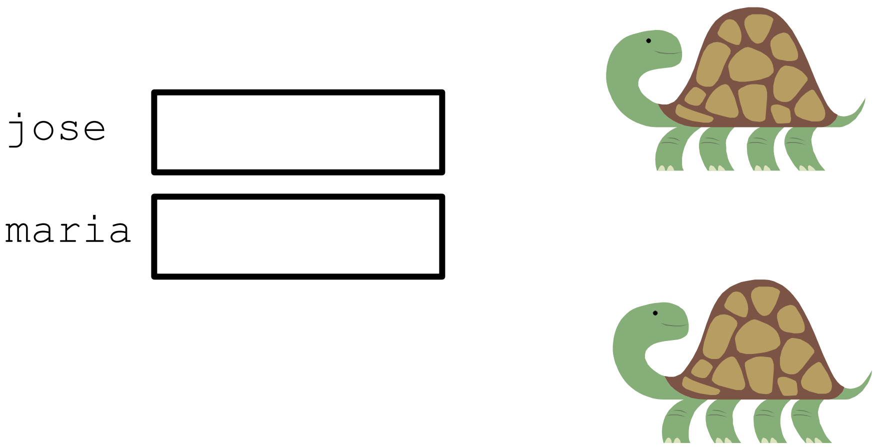
Primitives

```
int samAge   = 20;
int karenAge = 25;
samAge = karenAge;
```



Objects

```
World world1 = new World(200,100);
Turtle maria = new Turtle( 25, 25, world1);
Turtle jose  = new Turtle(100, 50, world1);
jose = maria;
```



1) Solo: (45 sec)
2) Discuss: (1 min)
3) Group: (30 sec)

CS Concept: for each loops

```
Pixel[] pixelArray = this.getPixels();
int value = 0;
for (Pixel pixelObj: pixelArray)
{
  value = pixelObj.getGreen();
  pixelObj.setBlue(value);
}
```

What does this code do?

A. Decreases the blue component of a picture

B. Sets the green component of each pixel to be the same as the blue component

C. Sets the blue component of each pixel to be the same as the green component

D. Loops over all pixels in pixelArray. For each pixelObj calls getGreen and stores that into value. Then sets value into blue.

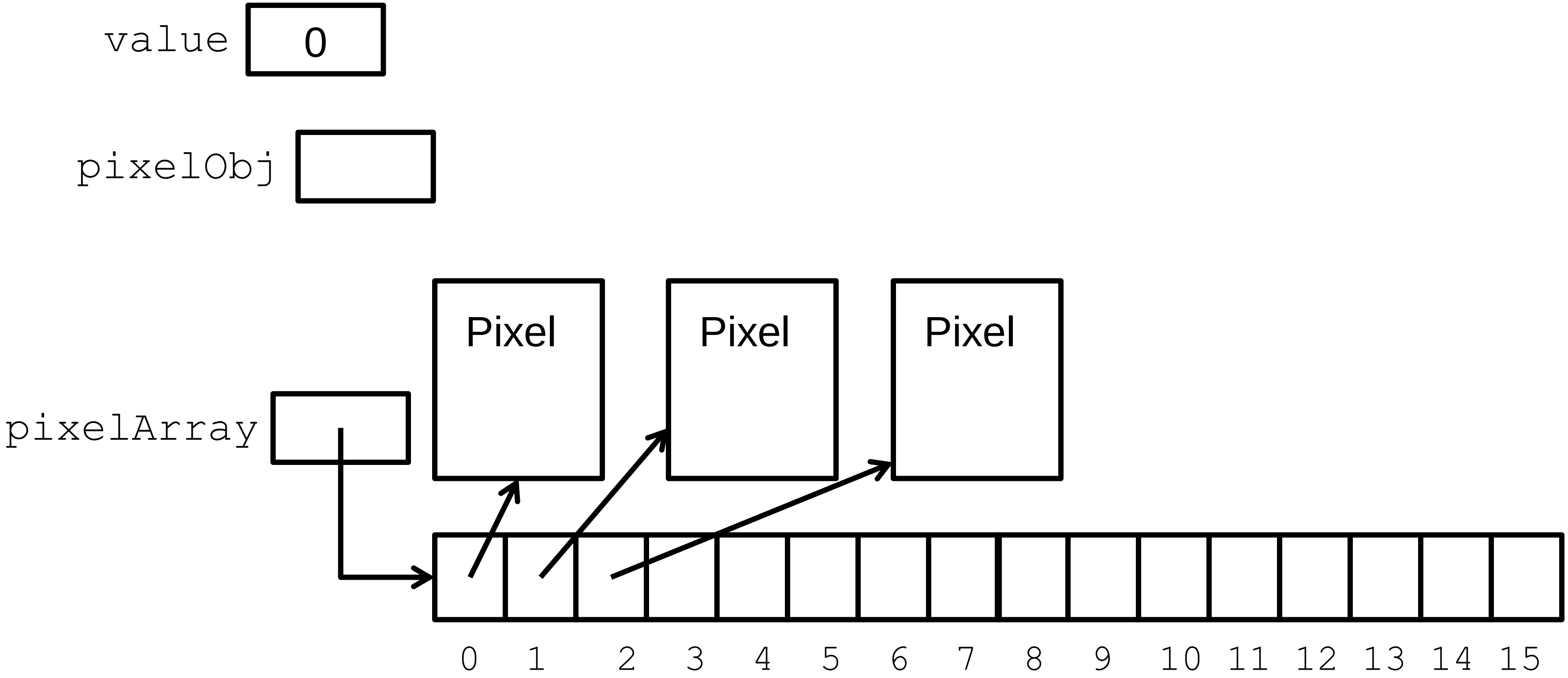
E. None of the above.

Deconstructing for each loops

```
Pixel[] pixelArray = this.getPixels();
int value = 0;
Pixel pixelObj = //first pixel in picture
value = pixelObj.getGreen();
pixelObj.setBlue(value);
pixelObj =       //next pixel in picture
value = pixelObj.getGreen();
pixelObj.setBlue(value);
pixelObj =       //next pixel in picture
value = pixelObj.getGreen();
pixelObj.setBlue(value);
pixelObj =       //next pixel in picture
value = pixelObj.getGreen();
pixelObj.setBlue(value);
pixelObj =       //next pixel in picture
value = pixelObj.getGreen();
pixelObj.setBlue(value);
pixelObj =       //next pixel in picture
value = pixelObj.getGreen();
pixelObj.setBlue(value);
pixelObj =       //next pixel in picture
ON AND ON THROUGH ALL PIXELS IN PICTURE
```

```
Pixel[] pixelArray = this.getPixels();
int value = 0;
for (Pixel pixelObj: pixelArray)
{
  value = pixelObj.getGreen();
  pixelObj.setBlue(value);
}
```

Memory models!



Summary: Rules of Memory Model Drawing

- Two kinds of variables in Java memory:
  - **Primitive type variables and class type object reference variables:**
    - **primitive type: "boxes" with actual values in them**
    - **Class/object: "boxes" have an arrow in them pointing to an object, or are null**
  - **Objects themselves: class/object type ACTUAL DATA is stored here**
- When you see a declaration (like ***`type varName;`***):
  - **Write down the name next to a box**
  - **Put value (primitive type) or pointing arrow (class type) when assigned**
- When you see instantiation (creation) of an actual object (including arrays)
  - **Draw larger box (or multi-boxes if array) and fill in values**
- Execution of other kinds of statements
  - **If assignment is made, change value in box on left side of the assignment**
  - **If you need to evaluate an expression, use values currently in the boxes**

1) Solo: (30 sec)
2) Discuss: (1 min)
3) Group vote: (30 sec)

Types Practice

- Consider the following code:

```
int value = 0;
Pixel[] pixArray = this.getPixels();
int index = 0;
Pixel p1 = pixArray[0];
value = p1.getRed();
```

What are the types of: `pixArray`, `p1`, `value`?

A. Pixel, Pixel, int
B. Pixel array, Pixel, int
C. Pixel array, Pixel, double
D. Pixel, double, int

1) Solo: (2 min)
2) Discuss: (2 min)

Memory Model Practice

- Draw the memory model for the following code

```
int value = 0;
Pixel[] pixArray = this.getPixels();
int index = 0;
Pixel p1 = pixArray[0];
value = p1.getRed();
```

How does it change with:

```
value = (int) (value * .5);
P1.setRed(value);
```

1) Solo ONLY: (45 sec)

Clicker Quickie:

- In the `decreaseRed` method in the book we see

```
Pixel[] pixelArray = this.getPixels();
int value = 0;
for (Pixel pixelObj: pixelArray)
{
  value = pixelObj.getRed();
  value = (int) (value * .5);
  pixelObj.setRed(value);
}
```

What happens if you remove the `(int)`?

A) `value = value * .5` would cause an error
B) `pixelObj.setRed(value)` would cause an error
C) The red channel would be set to 0 for all Pixels
D) Nothing would change

1) Solo: (60 sec)
2) Discuss: (2min)
3) Group: (30 sec)

CS Concept: while loops

How many times is
each section of code executed?

```
Pixel[] pixelArray = this.getPixels();
int value = 0;
Pixel p  = null;
int index = 0;
while (index < pixelArray.length)
{
  value = pixelArray[index].getRed();
  value = (int) (value * 0.5);
  pixelArray[index].setRed(value);
  index = index + 1;
}
```

What does this code do?*

```
Pixel[] pixelArray = this.getPixels();
int value = 0;
int index = 0;
while (index < pixelArray.length)
{
  value = pixelArray[index].getRed();
  value = (int) (value * 0.5);
  pixelArray[index].setRed(value);
  index = index + 1;
}
```

A. Removes all red from the picture

B. Changes ½ of the red pixels to not be red

C. Reduces the red component of ½ of the pixels

D. Reduces the red component of each pixel to ½ of its original value

E. Sets the red component of each pixel to 0.5

* This is something you WILL be asked to do on an exam/quiz.
  You should also be able to draw the memory model!

Exam Practice (1)

- Which picture represents what is shown when the following code is executed (assume filenames describe the pictures they represent).

```
String fName1 = “caterpillar.jpg”;
String fName2 = “sunset.jpg”;
Picture p1 = new Picutre(fName1);
Picture p2 = new Picture(fName2);
p1 = p2;
p1.show();
```

A. A sunset
B. A caterpillar
C. Neither of these

Exam Practice (1a)

- Which picture represents what is shown when the following code is executed (assume filenames describe the pictures they represent).

```
String fName1 = "catepillar.jpg";
String fName2 = "sunset.jpg";
Picture p1 = new Picture(fName1);
Picture p2 = new Picture(fName2);
p1 = p2;
Pixel[] pixArray = p1.getPixels();
for  (Pixel pix: pixArray)
{
    pix.setColor( Color.black );
}
p2.show();
```

A. A sunset

B. A caterpillar

C. A picture that's all black

D. None of these

Exam Practice (2)

- Complete the following code with a for-loop that removes the blue compixelObjnent from all Pixels in the picture

```
Pixel[] pixelArray = this.getPixels();
int value = 0;
```

Exam Practice (3)

- Complete the following method to set the green value in a picture to a specific value. You must use a while-loop to do this.

```
public void setGreen(___________________________)
{
   Pixel[] pixelArray = this.getPixels();
   int value = 0;
   int index =            ;









}
```

1) Solo: (20 sec)
2) Discuss: (1 min)
3) Group: (20sec)

Chapter 2 review!

- What is stored in the variable **name** after this line of code is executed?

```
String name = “Shari";
```

A. “Shari“

B. Shari

C. The address in memory where “Shari" is stored

D. new String(“Shari");

Hint: Are Strings, objects or primitives?

Just a reminder:
Important things are in the book

- If I didn’t talk about it in class, I thought it was “easier” to understand
  - Not that it wasn’t important!
- Suggestions:
  - Office hours
  - STUDY GROUPS: or reading groups
    - Say, let’s read until 4.2.1 then explain to each other, ask each other questions.
    - “So – I think this is important. What this code is doing is… “

1) Solo
2) Discuss

```
Page 89:
Picture p = new Picture(FileChooser.pickAFile());
p.getPixel(10,100).setColor(Color.black);
```

```
Page 99:
p.repaint();
```

TODO

- Work on PSA2
- Study for Exam#1
- No reading quiz for Friday