

Conditional Execution

Selection Options When Coding

Selectively Executing Code in JavaScript

- Your JavaScript programs can set all pixel values in an image, but not selectively
 - Change just the blue pixels



Selectively Executing Code in JavaScript

- Your JavaScript programs can set all pixel values in an image, but not selectively
 - Change just the blue pixels
 - Change just the pixels on image border



New JavaScript Concepts

- We want to selectively alter pixels
 - Depending on pixel (r,g,b) values
 - Depending on pixel location (x,y) values
- Pixel methods to determine values
 - `p.getRed()` , `p.getBlue()` , `p.getGreen()`
 - `p.getX()` , `p.getY()`
- Javascript statements to select code
 - `if (p.getRed() > 100) ...`
 - `if (p.getY() == 0) ...`

Selectively Changing Pixel Values

- Changing blue devil to yellow devil
 - How can we make the yellow “brighter”?

```
var image = new SimpleImage("duke_blue_devil.png");  
for (var p of image.values()) {  
    p.setRed(255);  
    p.setGreen(255);  
}  
print(image);
```



Developing Pseudo Code

- Changing blue devil to yellow devil
 - Select the blue pixels, change just those!

```
if (pixel is blue) {  
    change to yellow  
}
```

Developing Pseudo Code

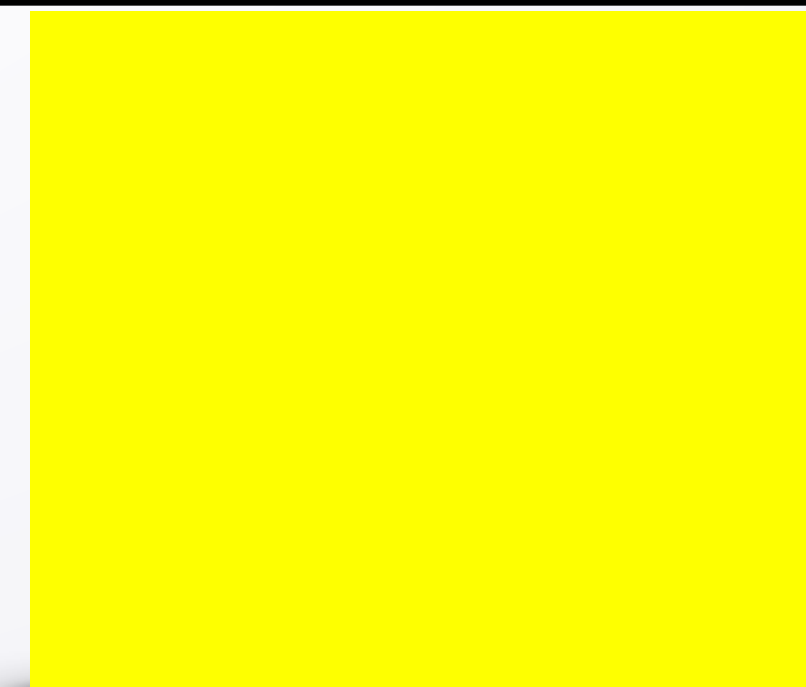
- Changing blue devil to yellow devil
 - Select the blue pixels, change just those!
 - How do we change to yellow?
 - Think about ideas, express in pseudo code, turn ideas into JavaScript
- Develop a program!

```
if (pixel is blue) {  
    pixel.setGreen(255);  
    pixel.setRed(255);  
    pixel.setBlue(0);  
}
```

The First Program May Not Work

- Turning Pseudo Code into JavaScript
 - Why doesn't this program work?

```
var image = new SimpleImage("duke_blue_devil.png");  
for (var p of image.values()) {  
    if (p.getBlue() > 200) {  
        p.setRed(255);  
        p.setGreen(255);  
        p.setBlue(0);  
    }  
}  
print(image);
```



Thinking and Making a Program Work

- What makes a pixel blue?
 - In this image, it's anything but white!

```
var image = new SimpleImage("duke_blue_devil.png");  
for (var p of image.values()) {  
    if (p.getRed() != 255) {  
        p.setRed(255);  
        p.setGreen(255);  
        p.setBlue(0);  
    }  
}  
print(image);
```



Making Changes to a Working Program

- How to make a magenta devil?
 - Same pixels, different change

```
var image = new SimpleImage("duke_blue_devil.png");
for (var p of image.values()) {
    if (p.getRed() != 255) {
        p.setRed(255);
        p.setGreen(0);
        p.setBlue(255);
    }
}
print(image);
```



Expressions for Selective Execution

- We've learned a new JavaScript statement
 - `if (testable condition) { ...}`
- What are the testable conditions?
 - Anything that can be true or false, also called Boolean expression
 - Tests for equality: `==` and `!=`
 - Inequality: `<`, `>`, `<=`, `>=`

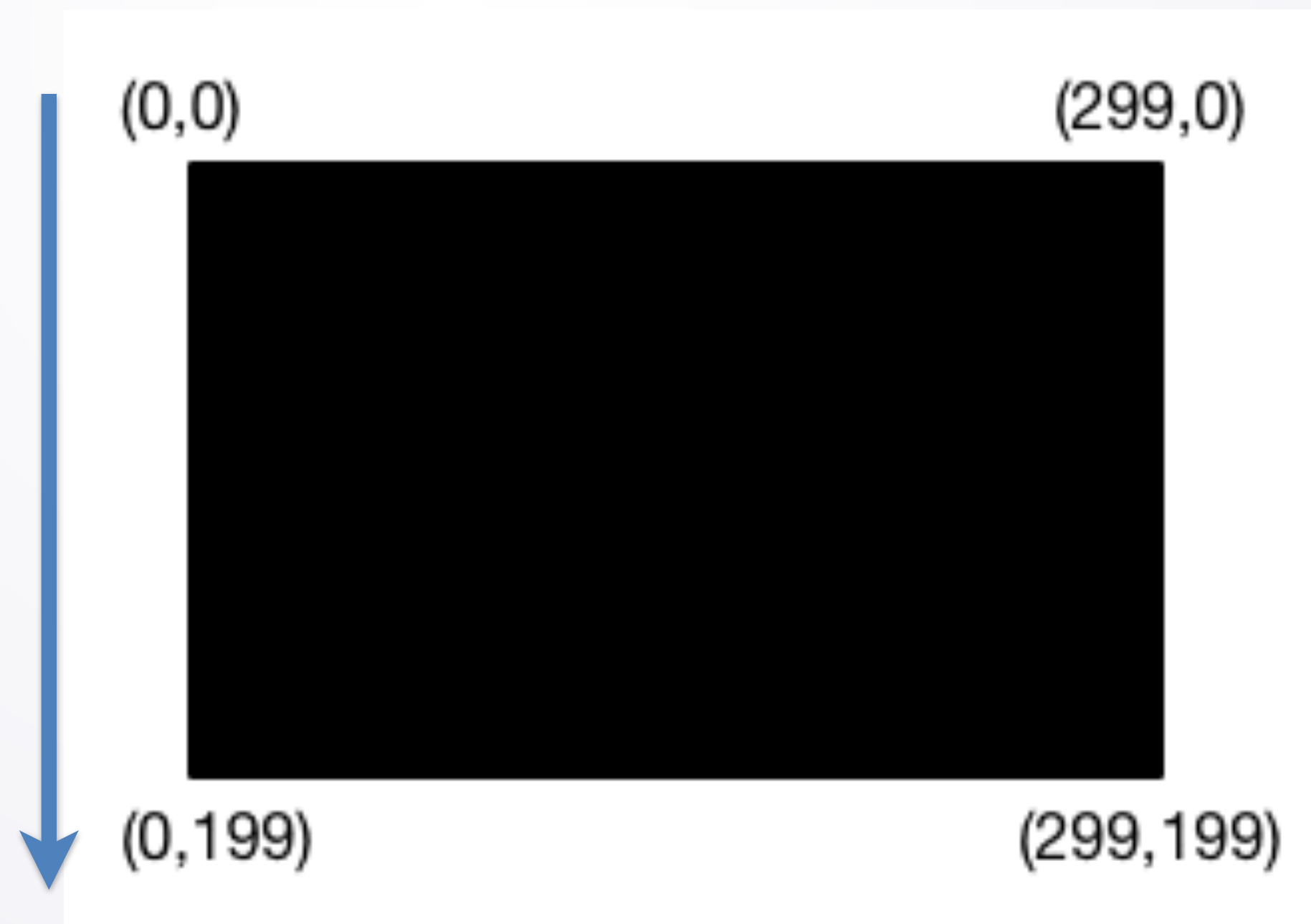
Example: Making Border Black

- Putting a black border around an image
 - Identify the border pixels
 - Change them to black



Finding the Border of an Image

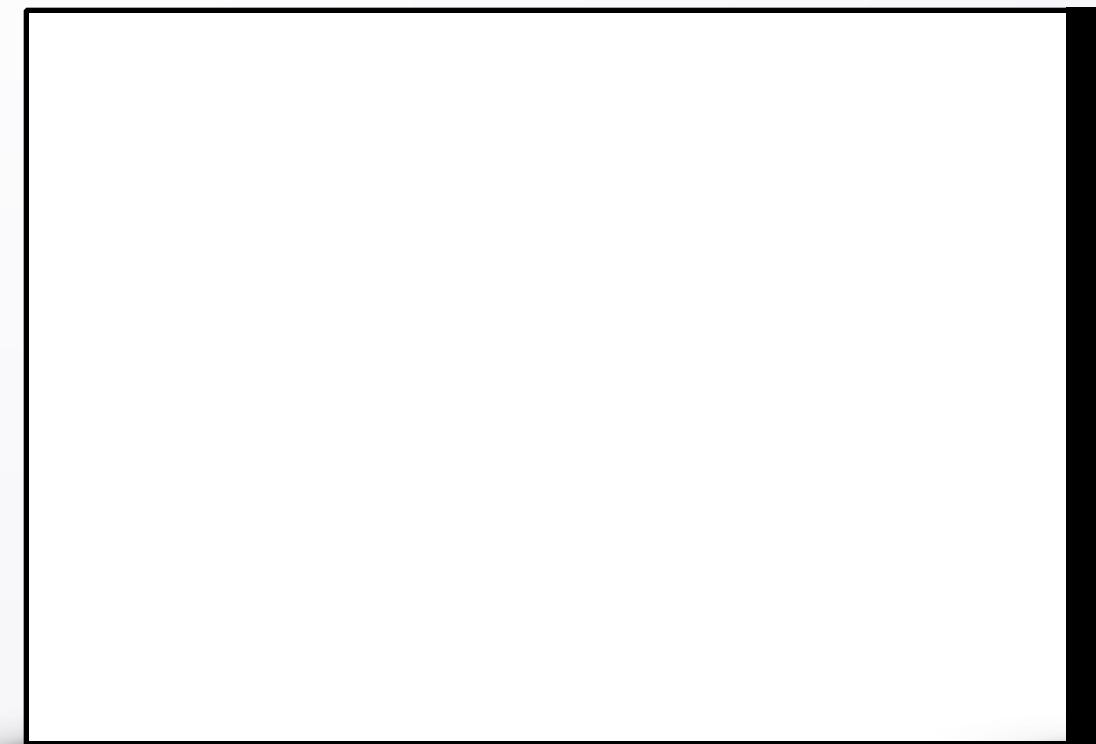
- Every pixel has (x,y) coordinates
 - (0,0) is upper left, (299,199) lower right
 - X increases left to right
 - Y increases top to bottom



Placing a Border on an Image

- Creating an image, placing a border

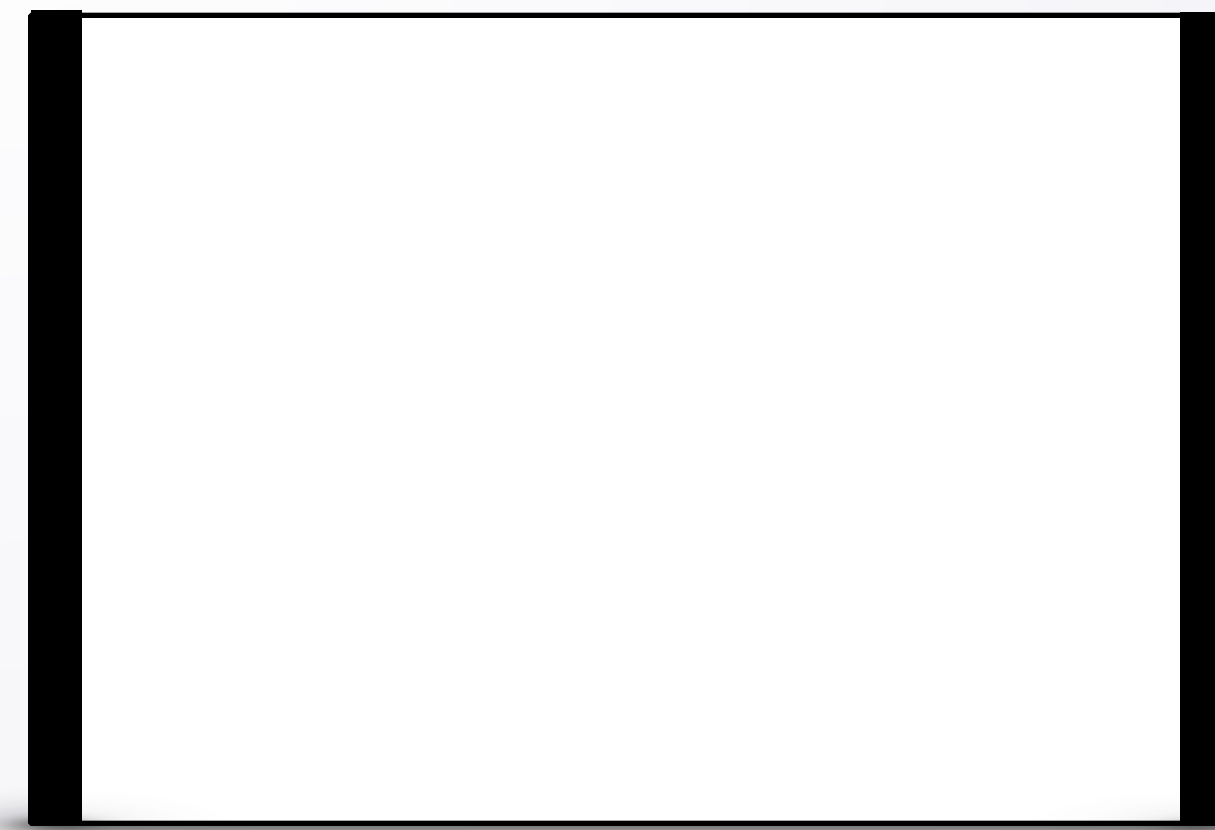
```
var image = new SimpleImage(300,200);  
for (var p of image.values()) {  
    p.setRed(255);  
    p.setGreen(255);  
    p.setBlue(255);  
    if (p.getX() > 289) {  
        p.setGreen(0);  
        p.setBlue(0);  
        p.setRed(0);  
    }  
    // more code here
```



Adding the Left Border

- Select left and right borders with code

```
if (p.getX() > 289) {  
    p.setGreen(0);  
    p.setBlue(0);  
    p.setRed(0);  
}  
if (p.getX() < 10) {  
    p.setGreen(0);  
    p.setBlue(0);  
    p.setRed(0);  
}  
// more code here
```



Drawing the Top and Bottom Borders

- Select top and bottom borders with code


```
if (p.getY() < 10) {  
    p.setGreen(0);  
    p.setBlue(0);  
    p.setRed(0);  
}  
if (p.getY() > 189) {  
    p.setGreen(0);  
    p.setBlue(0);  
    p.setRed(0);  
}
```



Refactoring or Rewriting a Program

- Avoiding duplicated code, this OR that

```
var image = new SimpleImage(300,200);  
print(image);  
for (var p of image.values()) {  
    p.setRed(255);  
    p.setGreen(255);  
    p.setBlue(255);  
    if (p.getX() < 10 || p.getY() < 10 ||  
        p.getX() > 289 || p.getY() > 189) {  
        p.setGreen(0);  
        p.setBlue(0);  
        p.setRed(0);  
    }  
}  
print(image);
```



Summary of New JavaScript Knowledge

- Using **if statements** for selective execution
 - `if (boolean expression) {...}`
 - Boolean expressions: `<`, `<=`, `>`, `>=`, `==`, `!=`
- Logical operators: combine booleans
 - `p || q`: true when p is true **or** q is true (or both)
 - `p && q`: true when both p **and** q are true
- You'll need lots of practice to master these concepts; have fun!