

Today

- Released [PA6](#)
- Submit [ICA](#) to Gradescope
- Chapter 5
 - Booleans

CS 101

Boolean Expressions

Chapter 5

Built-in Boolean Variables

Name one of the two built-in Boolean Variables we've learned last class

Boolean Variable

- We can define boolean variables

```
boolean skyIsBlue = true;  
  
boolean UofARocks = true;  
  
boolean ASURocks = false;  
  
if (UofARocks) {  
    // ...  
}
```

Boolean Operators

- There are several ways to make a boolean expression
 - Just write the name of a boolean variable (for example: **keyPressed** or **mousePressed**)
 - Compare two non-boolean values (numbers or characters) with comparison operators

< > <= >= == !=
 - Use Boolean logical operators

Boolean operator: Not (!)

- Another unary operator: **not** (!)
- Put this to the left of any boolean expression
- Turns value from **true** to **false**, or vice-versa
- Use parentheses to force the order when needed
- **&&** , **||** and **!** can also all be used with parentheses to force the order of operation, just like with math operators

Boolean Operators: AND (&&) and OR (||)

- We have the Boolean operators *and* and *or*:

&&

||

- These operators expect a value to the left and right to operate on
- However, the left and right-hand sides must be booleans!

$expr_1 \ \&\& \ expr_2$

- These compare boolean values

Boolean Operators

- **&&** (logical and)
 - **X && Y** is **true** if both **X** and **Y** are true
 - If one is **true** or neither are **true**, it makes the whole expression **false**
- **||** (logical or)
 - **X || Y** is **true** if both **X** and **Y** are **true**, or if one of them is **true**
 - If neither are **true**, it makes the whole expression **false**

&&

||

		X	
		True	False
Y	True	True	False
	False	False	False

		X	
		True	False
Y	True	True	True
	False	True	False

Truth tables for the boolean operators && and ||

Example

What will be drawn when this code runs?

- A. 1 rect, 1 ellipse, 1 triangle
- B. 1 rect, 1 ellipse
- C. 1 rect, 1 triangle
- D. 1 ellipse
- E. 1 triangle

```
boolean a = true;

boolean b = true;

boolean c = false;

if (c || b) {
    rect(100, 100, 50, 50);
}

if (a && false) {
    ellipse(200, 200, 70, 70);
} else if (c) {
    triangle(50,50, 100,100, 200,200);
}
```

Example

Which combination of values should we set so we draw an ellipse?

- A. a, b, c = true
- B. a, b, c = false
- C. a, c = true, b = false
- D. Any combination
- E. It's impossible

```
boolean a = _____ ;  
  
boolean b = _____ ;  
  
boolean c = _____ ;  
  
if (c || b) {  
    rect(100, 100, 50, 50);  
}  
  
if (a && false) {  
    ellipse(200, 200, 70, 70);  
} else if (c) {  
    triangle(50,50, 100,100, 200,200);  
}
```

Boolean Operators

- The left and right-hand side of a boolean operator can be another boolean expression, not just a **true**, **false**, or a boolean variable
- For example

```
boolean skyIsBlue = true;  
boolean UofARocks = true;  
boolean ASURocks = false;  
  
if (UofARocks || ASURocks && UofARocks ) {  
    rect(100, 100, 50, 50);  
}
```

Question 1

What will be drawn
when this code runs?

Type this in and see
if it works as you
expected.

```
void setup(){  
  size(200,200);  
}  
  
void draw(){  
  if (mousePressed || (keyPressed && key == 'a') ){  
    ellipse(mouseX, mouseY, 30, 30);  
  } else {  
    rect(mouseX-15, mouseX-15, 30, 30);  
  }  
}
```

Question 1

What is the difference?
Describe what is each
of these programs
doing.

```
void draw(){  
    if (keyPressed && key == 'a') {  
        ellipse(mouseX, mouseY, 30, 30);  
    }  
}
```

```
void draw(){  
    if (keyPressed || key == 'a') {  
        ellipse(mouseX, mouseY, 30, 30);  
    }  
}
```

```
void draw(){  
    if (key == 'a') {  
        ellipse(mouseX, mouseY, 30, 30);  
    }  
}
```

Boolean Operators instead of using nested ifs

```
void draw() {  
    .  
    .  
    .  
    if (xCord < 200) {  
        if (yCord < 200) {  
            rect(xCoord, yCoord, 50, 75);  
        }  
    }  
}
```

```
void draw() {  
    .  
    .  
    .  
    if ( (xCord < 200) && (yCord < 200) ){  
        rect(xCoord, yCoord, 50, 75);  
    }  
}
```

Question 2. What will the value of each variable be?

```
boolean iAmCool = true || false
```

```
boolean youAreCool = (false && false) || true
```

```
boolean weAreCool = iAmCool && youAreCool
```

```
boolean theyAreCool = (! weAreCool) || false
```

iAmCool = true

youAreCool = true

weAreCool = true

theyAreCool = false

```
theyAreCool = (! true) || false  
              = false || false  
              = false
```

Example

What happens when
this code runs?

Why?

```
void draw(){  
    if (!keyPressed && key == 'a'){  
        ellipse(mouseX, mouseY, 30, 30);  
    } else if (!mousePressed && !keyPressed){  
        rect(mouseX-15, mouseY-15, 30, 30);  
    }  
}
```

Question 3. What will the value of each variable be?

```
boolean a = true || true && true
```

```
boolean b = false && false || false
```

```
boolean c = a || (! b)
```

```
boolean d = b && c && a
```

SOLUTION FROM A STUDENT

```
a = true
```

```
b = false
```

```
c = true
```

```
d = false
```

Question 4. Using Boolean Expressions

- Download [website_buttons.pde](#) from the lectures page on the class website
- How can we reduce the number of if-statements needed in the program?
- (Focus on the if statements related to the Amazon button.)

SOLUTION FROM A STUDENT

In this case, you can use boolean operators in order to reduce the amount of "if" statements in the code. For the case of the Amazon button you can use if (mouseY>25 && mouseY<75 && mouseX>50 && mouseX<250)

```
if ((mouseY < 75) && (mouseY > 25) && (mouseX > 50)
&& (mouseX < 250)) {
    link("http://www.amazon.com");
    rect(50, 25, 200, 50);
}
```

Question 5 cont...

Draw a circle on a 700 x 700 canvas. The user should be able to move the circle left and right using the 'j' and 'l' keys.

The y position of the circle will remain 150, but the x position will change based on whether keyPressed is true and whether the 'j' or 'l' key is pressed.

1. Define a global variable named XCoord for the x-coordinate and use it to draw the circle.

SOLUTION FROM A STUDENT

```
int xCoord = 350; //global variable

void setup() {
    size(700,700);
}

void draw() {
    background(255,255,255);
    ellipse(xCoord,150,200,200);
}
```

Question 5

2. Fill out the code below and place it into the draw function to complete the functionality described in the previous slide

```
if (keyPressed == true) {  
    if (key == _____) { // go left  
        xCoord = xCoord - _____;  
  
    } else if (key == _____) { // go right  
        xCoord = xCoord + _____;  
    }  
    keyPressed = false;  
}
```

// draw the circle using the answer from the previous question