day80 Arduino button, 3/31/22 and 4/1/22

We're going to use the breadboard, a resistor, a pushbutton, some wires, and an LED light. Each team only makes one circuit; work together. You both need to write the program, though.

Please be gentle with the electronic parts: when putting the pushbutton onto the board, don't bend the pins all over the place; we don't have replacement parts for most of these things.

If you ever find a broken part, please give it to me, don't put it back into your bag.

Ideally we have one little baggie for the LEDs and other parts, and one baggie for the resistors. Please keep your parts organized. If you need a new bag, let me know.

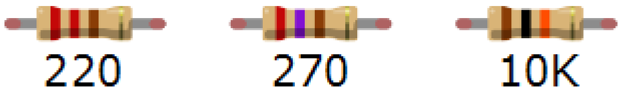
There is a resistor key picture below on step 2. You can tell the different resistors apart by their colored strips. The picture shows you the three kinds we have in our kits.

Partners should work together to build the circuit, then both partners need to write their own program and make it work. You can share one Arduino connected to your breadboard by unplugging it and then plugging it in to your partner's PC. Remember to upload YOUR code after you plug in the Arduino if it had been running on your partner's PC (the last uploaded program stays loaded even when you unplug it and plug it in to a new computer.)

Part 1: Getting Started with Arduino book:

1. Read p. 40-41.

2. Gather the parts listed on p. 41. The color codes for the resistors are here:



3. Build circuit on p. 41. See page 29 for how to correctly insert the LED. (The long leg goes into pin 13 and the short leg goes into GND.)

4. Open the button_starter_file.ino sketch attached to this assignment and emailed to you. Do a Save as… and save the file with your name in the title. We're only making one circuit today but each partner needs to write and show me their own program. Immediately put your name in the top comment where I asked you to.

5. Add two lines of code to your sketch to make the program print the value of the variable "val" to the serial monitor. You need a Serial.begin(9600) line in setup() and you need a Serial.println() call in the loop() function. Look at previous days' assignments to remember what the lines need to be to do this.

6. Type Control-T to make your code pretty.

(continued on next page)

7. Upload the program to your Arduino. Does the switch turn the LED on and off when you press and hold the button? Open the serial monitor. Does the value shown for "val" that you made the program print change when you press the button or release it? If not, double check your circuit (look at the book picture again, make sure your LED light has the long leg in pin 13 and the short leg in GND, get help, don't move on until you get this working. If you are having trouble with your code, type Control-T to help make it pretty (this fixes the indents, etc. so you can better see where things are, count curly brackets, etc.)

8. After you get everything above working, change the code inside the if statement block of code where the light is turned on to do the following: instead of simply turning the light on, use a for loop here to blink the light from 10 milliseconds to 150 milliseconds, going up by 10 each loop. Print the value of the for loop variable. Type Control-T to format your code properly. Look at previous programs to figure out how to do all of this.

9. Call me over to check you off; each person gets checked off separately.

After both of you get checked off, pull all the parts, stow them safely in their baggies, and put away your Arduino boxes for the day.