March 6 class.
Homework 2: will be posted this week. There will be no extension for this homework.
Fundamentals will be covered next week.

Today we will look at getting something back from the arduino board:

Open serial monitor which is where arduino can interact with the user.

```
Void setup(){
	Serial.begin(9600); // built in library already programmed in arduino
}

Void loop(){
	Serial.println(“Arduino For everyone”);
}
```

The above code just prints Arduino for everyone constantly in the serial monitor.

```
Int number;
Void setup(){
	Serial.begin(9600); // built in library already programmed in arduino
}

Void loop(){
	number=0;
	Serial.flush();
	while(Serial.available()==0){//check if serial is empty

	}
	while(Serial.available()>1){//check if we entered something in serial
		number=Serial.read()- ‘0’; //convert to actual number

	}
	Serial.print(“You entered: “);
	Serial.println(number);
	Serial.print(number);
	Serial.print(“multiplied by two is: “);
	number= number *2 ;
	Serial.println(“number”);


}
```

This will output the correct data for the number you entered plus the same thing for the “new line” character.

This only works for single digits. If you enter 29 , arduino picks up only the first digit. We need to modify the code to get double digit numbers to work.

In order to get it to work for double digits and remove the new line problem
We need to store the global variable number as a long instead of an int.
And add a check to see if the input it -38 , if it is just continue and the new line will be skipped.
Changed code to work with numbers with multiple digits:

```
long number;
Int a;
Void setup(){
        Serial.begin(9600); // built in library already programmed in arduino
}

Void loop(){
        number=0;
        Serial.flush();
        while(Serial.available()==0){//check if serial is empty

        }
        while(Serial.available()>0){//check if we entered something in serial
                a=Serial.read() -'0';
                if(a==-38) continue;
                number= number+a;
                delay(5);

        }
        Serial.print("You entered: ");
        Serial.println(number);
        Serial.print(number);
        Serial.print("multiplied by two is: ");
        number= number *2 ;
        Serial.println("number");


}
```

The delay is necessary to make the code work.

set up DHT11:
We created one function to read the temperatures
another function to display the data
Inside the loop function we just call both functions followed by a delay.

The code in class did not work because we need to install a library that is built for this sensor which we will do next class.

Yellow wire: connected to 5v
Red wire: connected to ground
Blue wire: connected to pin 0

To use the motor switch:
#include <Servo.h>
The header file is already installed in the ide so to use it we just need to include the header file

Use servo to work with motorized objects that do not continuously spin like a wheel.

Next class we will get the sensor to work correctly.