# ReflectionLog: Build a thermostat

Youdis

```java
import com.phidget22.*;

public class Thermostat {

    public static void main(String[] args) throws Exception{
        //Create
        TemperatureSensor temperatureSensor = new TemperatureSensor();
        DigitalInput greenButton = new DigitalInput();
        DigitalOutput greenLED = new DigitalOutput();
        DigitalInput redButton = new DigitalInput();
        DigitalOutput redLED = new DigitalOutput();

        //Address
        redButton.setHubPort(0);
        redButton.setIsHubPortDevice(true);
        redLED.setHubPort(1);
        redLED.setIsHubPortDevice(true);
        greenButton.setHubPort(5);
        greenButton.setIsHubPortDevice(true);
        greenLED.setHubPort(4);
        greenLED.setIsHubPortDevice(true);
```

Importing the phidget library and creating 5 objects 1 temperature sensor, 2 digital inputs, and 2 digital outputs. Each of the digital sensor objects will then be assigned connected to the ports of their corresponding physical item.

```java
        //Open
        temperatureSensor.open(1000);
        greenButton.open(1000);
        greenLED.open(1000);
        redButton.open(1000);
        redLED.open(1000);


        double setTemperature = 21;
        Boolean buttonPressed = true;
        double currentTemperature;
        int loopCounter = 50;
```

Then opening the connection for the 5 objects to their corresponding physical items. Then 4 variables will be initialized, one for the setTemperature which will equal 21 at the start of the program. The buttonPressed debounce variable to make sure the user doesn't hold down a button to repeatedly increase or decrease the setTemperature. The currentTemperature variable which will store a new temperature ever time the loop is reiterated. Finally, the loop counter which will keep track of when to print the messages(every 10 seconds) is set to 50 right away to print the message for the first time right away rather than waiting 10 seconds.

```
35
36          while(true) {
37              loopCounter++;
38              currentTemperature = temperatureSensor.getTemperature();
39              if(greenButton.getState() && !buttonPressed) {
40                  setTemperature++;
41                  buttonPressed = true;
42                  } else if(redButton.getState() && !buttonPressed){
43                      setTemperature--;
44                      buttonPressed = true;
45              }else if((!greenButton.getState() && !redButton.getState()) && buttonPressed) {
46                  buttonPressed = false;
47              }
```

Adding one to the loop counter so it counts the loop. Then store the temperature the temperature sensor records into the current temperature variable. Then will check if either the green or red button is pressed and the buttonPressed is false. If those conditions are met then either the setTemperature will increase one or decrease one depending on which colour button is pressed. If those conditions aren't met then if both buttons aren't being pressed and the buttonPressed Variable equals true then the buttonPressed variable turns false again so the first conditions can be met again in a future iteration of the loop.

```
            //if condition to know when 10 seconds pass to print messages
        if (loopCounter >= 50) {
            System.out.println("Current temperature is: " + currentTemperature + "°C");
            System.out.println("Set temperature is: " + setTemperature + "°C");
            loopCounter = 0;
        }
```

If loopCounter is more than or equal to 50 then the messages will display. Since every loop waits 200ms, 50 * 200ms equals 10000ms which is ten seconds so the messages will be displayed every 10 seconds. Then the loop counter will be set back to 0 so it can start counting up to 10 seconds again.

```
            if (Math.abs(currentTemperature - setTemperature) <= 2) {
                greenLED.setState(true);
                redLED.setState(false);
            }
            else {
                redLED.setState(true);
                greenLED.setState(false);
            }
            Thread.sleep(200);
        }
    }
```

If the absolute value of the current temperature minus the set temperature is less than or equal to 2 then the green LED will turn on and red turn off to indicate the temperature is within a range of the temperature the user set. If their difference is more than 2 then the red LED will turn on and the green turn off to let the user know their set temperature is not within the range for the current temperature. Then the executing thread will sleep for 200ms, therefore pausing the loop for 200ms.