pyduino_website.py

```
from flask import Flask, render_template,request, redirect, url_for
from pyduino import *
import time

app = Flask(__name__)

# initialize connection to Arduino
# if your arduino was running on a serial port other than '/dev/ttyACM0/'
# declare: a = Arduino(serial_port='/dev/ttyXXXX')
a = Arduino()
time.sleep(3)

# declare the pins we're using
LED_PIN = 3
ANALOG_PIN = 0

# initialize the digital pin as output
a.set_pin_mode(LED_PIN,'O')

print 'Arduino initialized'



# we are able to make 2 different requests on our webpage
# GET = we just type in the url
# POST = some sort of form submission like a button
@app.route('/', methods = ['POST','GET'])
def hello_world():
```

```
# variables for template page (templates/index.html)
author = "Kyle"

# if we make a post request on the webpage aka press button then do stuff
if request.method == 'POST':

   # if we press the turn on button
   if request.form['submit'] == 'Turn On':
      print 'TURN ON'

      # turn on LED on arduino
      a.digital_write(LED_PIN,1)

   # if we press the turn off button
   elif request.form['submit'] == 'Turn Off':
      print 'TURN OFF'

      # turn off LED on arduino
      a.digital_write(LED_PIN,0)

   else:
      pass

# read in analog value from photoresistor
readval = a.analog_read(ANALOG_PIN)

# the default page to display will be our template with our template variables
return render_template('index.html', author=author, value=100*(readval/1023.))
```

```
# unsecure API urls
@app.route('/turnon', methods=['GET'] )
def turn_on():
    # turn on LED on arduino
    a.digital_write(LED_PIN,1)
    return redirect( url_for('hello_world') )


@app.route('/turnoff', methods=['GET'] )
def turn_off():
    # turn off LED on arduino
    a.digital_write(LED_PIN,0)
    return redirect( url_for('hello_world') )



if __name__ == "__main__":

    # lets launch our webpage!
    # do 0.0.0.0 so that we can log into this webpage
    # using another computer on the same network later
    app.run(host='0.0.0.0')
```