breaktime.py

```
## Import time functions for sleep and current time functions

import webbrowser
## import to open url in webbrowser

import urllib2
## import to validate url input can be opened

dataInputs = {}

def getIntegerValue(varName, promptText):
  while True:
  ## Loop until integer input is received
    try:
    ## Attempt to get valid input
      userInput = int(raw_input(promptText))
    except ValueError:
    ## Inform input is invalid and keep looping
      print("Sorry, that was not understood please use integer as input.")
      continue
    else:
      ## got integer input so break loop
      dataInputs[varName] = userInput
      break

def getUrlInput(varName, promptText):

  ## Attempt to prevent: HTTPError: HTTP Error 403: Forbidden errors
  opener = urllib2.build_opener()
  opener.addheaders = [('User-agent', 'Mozilla/5.0')]
  urllib2.install_opener(opener)
  ## see -> http://stackoverflow.com/a/14471612/1815624


  while True:
  ## Loop until valid url input is received
    userInput = raw_input(promptText)
    ## Attempt to get valid input
    try:
    ## Check url can be opened
        urllib2.urlopen(userInput)
    except:
    ## Inform input is invalid and keep looping
        print ("Your input was not accessible web address: ")
        print (userInput)
        print("Please try again.")
    else:
      ## got url input so break loop
```

```
getIntegerValue("total_breaks","Enter Total Number of breaks: ")
getIntegerValue("break_interval", "Enter total number of minutes between breaks: ")
getIntegerValue("break_duration", "Enter total number of minutes for break duration:
")
getUrlInput("link", "Enter Url to open: ")
## call to functions passing variables
## url used in lesson-> http://www.youtube.com/watch?v=YXw16RzMofo
## viewpure is worth a mention -> http://viewpure.com/sjEbloAvI2c

break_count = 0
## Start break count at 0

print("The break counter started on "+time.ctime())
## Inform start time

while break_count < dataInputs["total_breaks"]:
## Loop from 0 until user desired total breaks are met.

  time.sleep(dataInputs["break_interval"] * 60)
  ## Program is to sleep while you work until break time
  ## break_interval * 60 to convert seconds to minutes

  print("This break has started on "+time.ctime())
  ## Inform user the break starting time stamp

  webbrowser.open(dataInputs["link"])
  ## Open desired url in browser

  time.sleep(dataInputs["break_duration"] * 60)
  ## Sleep until end of break duration
  ## break_duration * 60 to convert seconds to minutes

  print("This break has ended on "+time.ctime())
  ## Inform user the break ending time stamp

  break_count = break_count + 1
  ## add 1 to break_count and loop if total_breaks have not been met


print("This concludes all breaks.")
```