```
Damasio Eli Gerena's Notes for quiz

#singe line comments!
'''Multi-line
name    Damasio Eli Gerena IV
date    may 5th 2014
class   dpw 2014 may
'''
#print "hello world!"

#variables
first_name='eli '
last_name='de man'
year_born=2014-1987
#year_born+= 1 # -= *= += /+ for assignment tonight!!!! mathmatic  assignment
operators !!!!!!!! must have a defined variable before use
#print year_born

#conditionals
'''
if something > something:
    print "something"
elif something > something:
    print or do something
elif something > something else:
    print run something()
else:
    print can default something if others aren't true

'''
#arrays
students=["eli","e7li","e6li","el4i","eli3","eli2",]
students.append("arturo")
students[0].split()

#dictionaries -associative arrays
#name_of_dict={'key':value}
class_info={"name": a-variable,'something': 9,"something else":"a string"}

#loops
# for each loop
for s in students:
    pass
#for loop with a count          V-can be left blank and it will assume +1)
# for i in range (start, end ,inc/dec)
for i in range(0,10):
    pass
# also with random numbers
import random

#for loop with a count
# for i in range (start, end ,inc/dec)
for i in range(0,10):
    #print random.randrange(1,20)
    pass

while this is ><= that:


#def
def calc_area(h,w):

    area=h*w
    print area

calc_area(300,200)

#format string method
```

```
user_name="guts"
message='''
Welcome home {user_name}!
'''

message=message.format(**locals())
#print message

#getting information from the user
first_name=raw_input('number:')
print str(int(first_name)+2)+",is tiring."


day 4

linking using a anchor tag
<a href="?vars=value">My link</a>

so if get ?var == a value

use a dict or array to use a loop to show the different tags.
str=""
for c in cities:
        str+="<a href=' "+c.name+" '>'+c.name+'</a>'

notes
 for classes inheritance
first make your class

class Classname(object):
        def __init__(self):#need that!
                # properties 4 example
                self.age=0

        #then any methods
        def do(self,Whateverneededtodo):

---class inheritance--
class One(object): # is the first class

class Two(One): # uses its internals to instantiate all that was in One into Two
        def __init__(self):
                One.__init__(self)
                super(One,self).__init__()

An abstract class is one that doesn't get instantiated. it is soley used to create other
classes.

'''---- for changing pages using the url on form gets.
mainHandler
def get(self):
        #can also use try:
        if self.request.GET['form']=="deli"

if not self.request.GET:
        var=Page()
        var.content
elif self.request.GET["form"}=='deli'
        deli_form=FormPage()
        #show deli form --- which would be a instance of class form
--------
polymorphism---
rewrite the function.
just write the overwriting function with the same name

-----
get and setters example
@property
```

```
    def sound(self):
        return self.__a value or the whole of it.

    @sound.setter
    def sound(self,value):
        self.__value = value



attribute looks like
        req_variable=value # is accessible wwihtin a function

        #variable
        self.variable # since its accessible within a class

        self.__property=value #acess9ble by other classes
        V
        V
        V
@property
def property (self)
        return self.__property

____
Building html you will self.response.write() at some tim.
remember what html is made of.
html header body foot and closing tags. Cut up the html and make them variables so as to put
them together while adding your content via local formatting with variables.

____
#note on xml all xml elements and tags are made up so you must view it to see it.
to use xml for a db or api to import use- from xml.dom import minidom

to use json  for a db or api
to import use- import json

insure your importing any and all classes within your class or library file.
to do so use- from 'File' import *(all)
also you will need urllib2

key for using db or api  make sure in:

Controller-
instantiate the view
                vvvv
setup the api basic page
        if there are url variables
        set the request and its params
        create the url + params set previously
 will have sorting of what page your on and what view ail be displayed to the user.

Model-
        get api info
                using a urllib2.request(url)
                build the opener var using the url lib again
                then create a container for said info. using the opener.open(request var set
earlier)

        parse it!!!!
                this is where xml and json will differ
                xmlVar= minidom.parse(data)

        look at whats in the xml
                find and get what we want by creating a Arr var and using the (preview parse
var) to then getElementByTagName.
        from there loop through to indiviualy
The model will take in and build objects or arrays of objects for you to set to variables for
use.
```

VIEW-
	each page will be a function within the view that is .syntax called on the instance of the view
	each page function will be built specific lay for its content.
	The controller will call for the specific View when it is needed.
	each page will need to go through the data and display as necessary. Loop it.loop it good.

MVC notes
day 6

- controller exchng' info with model and vie
- manages changes and creations of models and views
- governs the application9 or portions of an application)
How
	view (tells controller, that the user wants for something to happen

	controller checks there access and credentials, then tells model ghead and do what you need to do what they want.

	model says : okay got it its done here you go controller.

	controller then collects the data and gives it back to view

	view gets it then shows it to the user.

when using a send request in mvc
 a note on nodes and there targeting.
'''
nodes:
    <tag>
        <x>
            <tag></tag>
    in the above you would have to use firsChild to understand where it is.

    if a tag has no end tag and is self closing you will not need a specific such as firstChild you may still need to use [] as if it was an array
'''

so for your understanding of MVC
the controller is the main handler in your main py file. it will tell the model to use ITS Class's methods to get information. So the method will get what the controller tells it to. After the method gets it it will return that information in the format it sends it as. The controller will then take that information and use it by sending it to the view which is a class that can be in the same file commonly referred to as a lib. the VIEW will then get that data from the controller and use it in its formatting the HTML that it will send to the controller to then be written to the user.

To shorten it up. On load of your page. The controller will response.write to the user a default html. this will come from the view. On a RESPONSE from the html a call will be made for information. The controller will tell the Model to use a class to get this information via a url(plus its parameters) that is built then requested, it builds the opener to open it, and then load it to a object. The Model will then add any specific perimeters sent by the user needed for the obj.

Aggregate and composites-

Aggregates are elements or data that continue on existing after the page or main code container disappear. They are brought on separately from the page.

my shirt is a aggregate.

Composites are parts of a greater pieces or components. These components live as long as they

are attached. A class that uses a super is a composite of its super. A element on a page is composite of the html page. No page no element.