**#python path**

C:\Users\yaydog\AppData\Local\Programs\Python\Python38-32\python.exe

**#installing 3rd party pyperclip module**

**List of 3rd party important modules**

cd C:\Users\yaydog\AppData\Local\Programs\Python\Python38-32\Scripts

pip.exe install pyperclip

import pyperclip

**(centos)**

Find installed: pip list

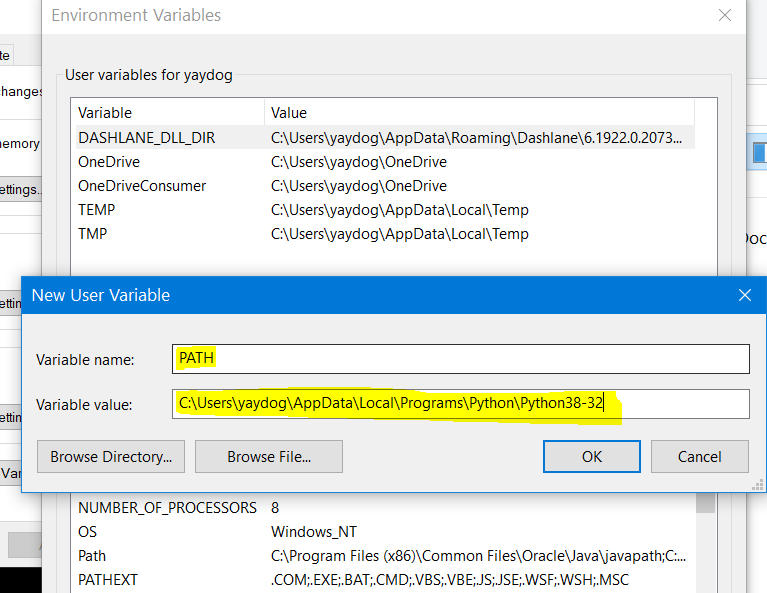
**(testing code line by line editor)**

pip install notebook

cmd> jupyter notebook

<http://localhost:8888>

**#adding python to environment variable**



**#viewing installed modules**

help('modules')

**#running py script from batch file**

"C:\Users\yaydog\AppData\Local\Programs\Python\Python38-32\python.exe" "C:\Users\yaydog\Desktop\test.py" %\*

**#running py script from bash file**

/usr/bin/python /root/testing.py

**#running bash inside python**

**(without screen output)**

#!/usr/bin/env pyhon

import subprocess

bashCommand = "yum install -y httpd"

output = subprocess.check\_output(['bash','-c', bashCommand])

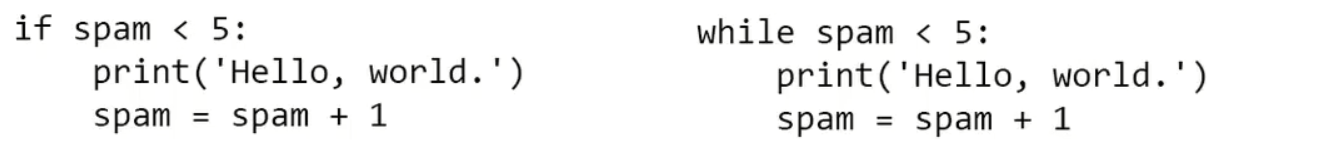
subprocess.call(cmd, shell=True)

**(with screen output)**

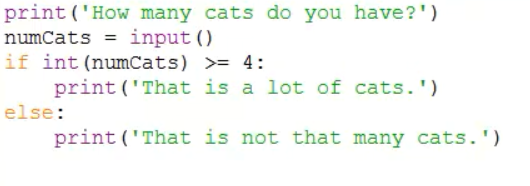
#!/usr/bin/env pyhon

import os

os.system('/root/test.sh')



**#if-else syntax**



**#taking input**

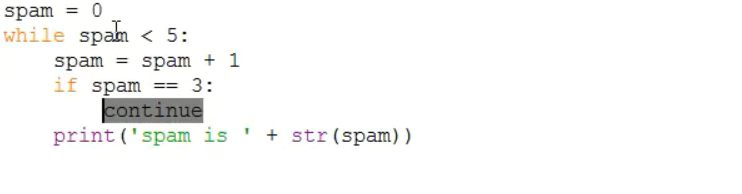
var=input()

**#break statement(causes execution to immediately leave loop)**

if var == variable:

break

**#continue statement (it returns to while loop again)**



**#for loop**

for var in range(8,10,1):

print(var)

**8**

**9**

(use increment 1, start from 8 and stop at 10)

**#built-in function and modules**

import module1,module2,module3

module.functionname(args1,args2)

import module1 import \*

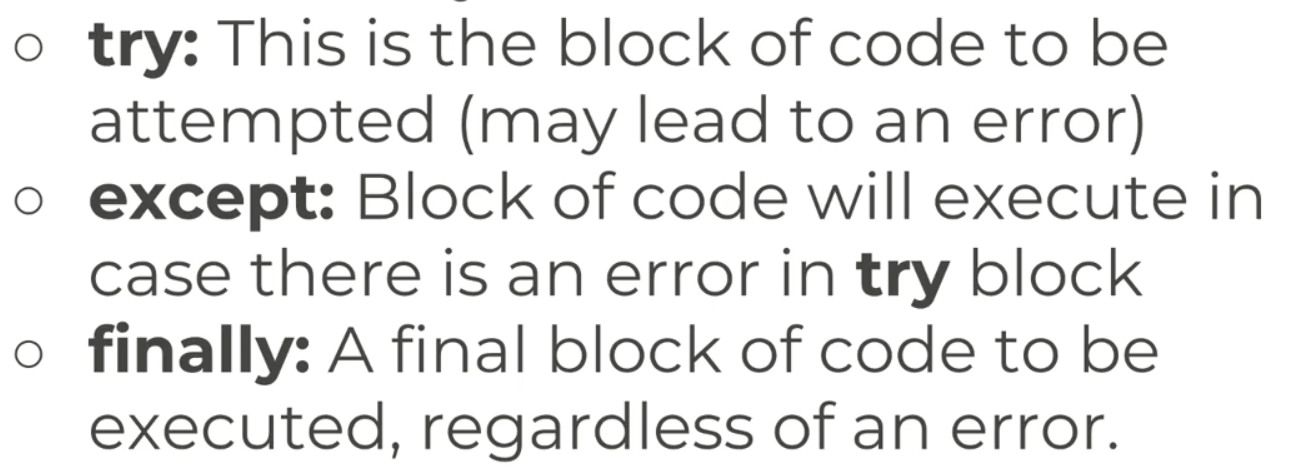
functionname(args1,args2)

**#termination of program at any line with sys.exit**

import sys

sys.exit()

**#error handling with specific error code**



Error types: <https://www.tutorialsteacher.com/python/error-types-in-python>

def func(n1,n2):

result=n1/n2

try:

func(3,1)

except ZeroDivisionError:

print('bad calculation')

except OSError:

print('bad calculation')

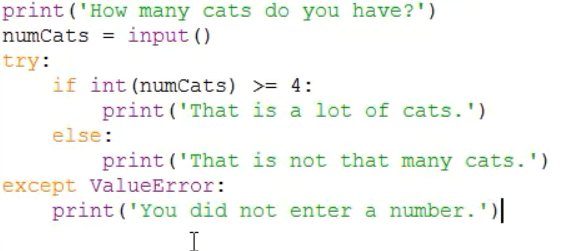
else:

print('correct calculation')

finally:

print('i will print anyway')

**#input validation**



**#code verification and evaluation**

pip install pylint

pylint code.py

**#lists**

var= ['cat','rat','bat']

var[0]

'cat'

var[-1]

'bat'

print(var[0])

cat

var.index('bat') #returns index value of list member

2

var.**append**('newanimal') #adds new member at the end of list

var

['cat', 'rat', 'bat', 'newanimal']

var.**insert**(2,'newanimal') #adds new member in any position in list

var

['cat', 'rat', 'newanimal', 'bat']

var.**remove**('newanimal') #removes member of list

var

['cat', 'rat', 'bat']

var.**sort**() #sorts members alphabetically

var

['bat', 'cat', 'rat']

var.sort(reverse=True) #sorts members alphabetically in reverse order

var

['rat', 'cat', 'bat']

var[0]=10 #changing index values

var

[10, 'rat', 'bat']

var[:2]

[10, 'rat']

del var[0] #deletion of first index of list

var

['rat', 'bat']

len(var) #finding how many indexes have in list

2

'rat' in var #check if given variable in list

True

**#for loop with list**

var= ['cat','rat','bat']

for i in var:

print(i)

cat

rat

bat

**#displaying indexes of list**

list(range(0,100,2))

**#copying list**

import copy

var= ['cat','rat','bat']

var2=copy.deepcopy(var)

print(var2)

['cat', 'rat', 'bat']

**#custom function**

def testfunc(param):

param.append('finalelement')

var=[1,2,3]

testfunc(var)

print(var)

[1, 2, 3, 'finalelement']

**#file path**

print(r'c:\test')

c:\test

**#current filepath**

import os

var=os.getcwd()

print(var)

**#changing filepath**

os.chdir('/home') #linux

os.chdir('C:\\') #windows

**#check if file exists**

>>> os.path.exists('/root/testing.py')

True

>>> os.path.exists('C:\\Users\\yaydog\\Desktop\\cover short.txt')

True

>>> os.path.exists('C:\\Users\\yaydog\\Desktop\\cover short.txt')

**#check path is folder or file**

os.path.isfile

os.path.isdir

**#get file size**

var= 'C:\\Users\\yaydog\\Desktop\\cover short.txt'

os.path.getsize(var)

**#shows content of folder**

os.listdir('C:\\Users\\yaydog\\Desktop\\')

**#create folder**

os.makedirs('C:\\test')

os.makedirs('/root/newfolder')

**#reading a file**

file= open('C:\\Users\\yaydog\\Desktop\\cover short.txt', 'r' )

var1=file.read()

var2=file.readlines()

print(var)

file.close()

**#writing to a file (overwrite by deleting existing content)**

file= open('C:\\Users\\yaydog\\Desktop\\cover short.txt', 'w' )

file.write('endofline2\n')

file.close()

**#writing to a file (add content without deleting original file)**

file= open('C:\\Users\\yaydog\\Desktop\\cover short.txt', 'a' )

file.write('\nendofline3')

file.close()

**#copying-moving files**

import shutil

shutil.copy( 'C:\\Users\\yaydog\\Desktop\\cover short.txt', 'C:\\test\\copyfile.txt')

shutil.move( 'C:\\Users\\yaydog\\Desktop\\cover short.txt', 'C:\\test\\')

**#copying folder**

shutil.copytree( 'C:\\Users\\yaydog\\Desktop\\test', 'C:\\test\\test\_backup')

**#deleting file**

import os

os.unlink('C:\\Users\\yaydog\\Desktop\\file.txt')

**#deleting folder and its contents**

import shutil

shutil.rmtree('C:\\test')

import send2trash

send2trash.send2trash('C:\\test\\file.txt')

**#creating folder tree (display folders and contents)**

**import** os  
**for** foldername,subfolders,filenames **in** os.walk(**'E:\Corendon-15.01.2020'**):  
 print(foldername)  
 print(filenames)

**#downloading files from internet**

cd C:\Users\yaydog\AppData\Local\Programs\Python\Python38-32\Scripts

pip.exe install requests

pip.exe install beautifulsoup4

import requests

import os

import bs4

url = 'https://www.python.org/ftp/python/3.8.2/Python-3.8.2.tar.xz'

os.chdir('C:\\')

r = requests.get(url, allow\_redirects=True)

open('Python-3.8.2.tar.xz', 'wb').write(r.content)

**#getting status code**

res=requests.get('https://www.python.org/ftp/python/3.8.2/Python-3.8.2.tar.xz')

res.status\_code

200: ok

res.raise\_for\_status() # if it does not return value, everything is ok

**#READING FROM EXCEL**

(windows) pip install openpyxl

(linux)

import openpyxl

import os

wb= openpyxl.load\_workbook('C:\\Users\\yaydog\\Desktop\\test.xlsx')

*#finding sheet names*

print(wb.sheetnames)

*#finding value of cell*

sh = wb["Sheet1"]

print(sh["A1"].value)

*#finding all values of a columns*

first\_column = sh['A']

for x in range(len(first\_column)):

print(first\_column[x].value)

*#Determine total number of rows*

print(sh.max\_row)

#Determine total number of columns

print(sh.max\_column)

**#CREATING EXCEL**

import openpyxl

import os

os.chdir('C:\\Users\\yaydog\\Desktop')

wb= openpyxl.Workbook()

sh = wb.create\_sheet("Mysheet",0)

sh['A1']=55

sh['A2']='Hello'

wb.save("sample.xlsx")

**#MANIPULATING DOCX**

pip install python-docx

import os

from docx import Document

from docx.shared import Inches

from docx.shared import Pt

document = Document()

document.add\_heading('Document Title', 0)

p1=document.add\_paragraph()

p1content='SELAMINALEYKUM'

runner = p1.add\_run(p1content)

runner.bold = True

runner.italic = True

style = document.styles['Normal']

font = style.font

font.name = 'Arial'

font.size = Pt(16)

p1.style = document.styles['Normal']

document.add\_page\_break()

os.chdir('C:\\Users\\yaydog\\Desktop')

document.save('simple.docx')

**#TAKING SCREENSHOT**

import pyautogui

import time

time.sleep(20)

pyautogui.screenshot('C:\\Users\\yaydog\\Desktop\screenshot.jpeg')

**#COUNTER MODULE**

pip install counter

import os

os.chdir('C:\\Users\\yaydog\\Desktop\\')

from collections import Counter

string1='stringtesting'

#print(Counter(string1)) #counter string

#print(Counter(string1).most\_common(2)) #most used letter in string

string2='stringtest coming up'

splitted=string2.split() #string split

#print(splitted)

#print(Counter(splitted)) #counter of splitted string

list1 = ['x','y','z','x','x','x','y', 'z']

#print(Counter(list1)) # counter list

#print(sum(Counter(list1).values())) # count total elements in list

#print(list(Counter(list1))) #list unique elements

**#DAY AND TIME OPERATIONS**

import datetime

t=datetime.time(23,30,30) #assigning date to variable

print(t)

today=datetime.date.today()

print(today) #today's date full

print(today.day) #today's day

print(today.month) #today's date full

print(today.year) #today's date full

new=today.replace(year=1990) #manipulating date

print(new)

(getting date in custom format)

import datetime

#a=day

#b=month

#d=day of month

# convert datetime to custom format

format = "%a %b %d %H:%M:%S:%Y"

today = datetime.datetime.today()

print(today.strftime(format))

**#FINDIND TIME ELAPSED DURING CODE WORK**

import timeit

import random

def test():

return random.randint(10, 100)

starttime = timeit.default\_timer()

test()

print("The time difference is :", timeit.default\_timer() - starttime)

**#PSUTİL**

pip install psutil

import psutil

psutil.users() : show connected user info

[suser(name='root', terminal='tty1', host='', started=1585672704.0, pid=793), suser(name='root', terminal='pts/0', host='192.168.1.9', started=1585726720.0, pid=42467)]

psutil.pids() : shows all processes and get process info

>>> psutil.Process(64524)

psutil.Process(pid=64524, name='python', started='17:50:02')

(psutil.Process(64524)).name() : 'python'

(psutil.Process(64524)).exe() : '/usr/bin/python2.7'

(psutil.Process(64524)).status() : running

(psutil.Process(64524)).parent()

psutil.Process(pid=42467, name='bash', started='10:39:10')

(psutil.Process(64524)).username() : root

(psutil.Process(64524)).open\_files()

.suspend()

.resume()

.p.terminate()

.p.kill()

psutil.net\_io\_counters(pernic=True) : network information

psutil.net\_connections()

psutil.disk\_partitions()

psutil.disk\_usage('/')

psutil.virtual\_memory()

**#PWD MODULE**

pwd.getpwnam("root")

pwd.struct\_passwd(pw\_name='root', pw\_passwd='x', pw\_uid=0, pw\_gid=0, pw\_gecos='root', pw\_dir='/root', pw\_shell='/bin/bash')

pwd.getpwuid(0)

**#GLOB MODULE**

finds all the pathnames matching a specified pattern (similar to find in bash)

glob.glob('/etc/sysconfig/network-scripts/ifcfg\*')

['/etc/sysconfig/network-scripts/ifcfg-lo', '/etc/sysconfig/network-scripts/ifcfg-ens33']

**#SAVING OUTPUT OF COMMAND TO FILE**

import subprocess

import sys

try:

cmd = 'cat /etc/hosts'

print('running command:', cmd)

output = subprocess.check\_output(cmd, shell=True, stderr=subprocess.STDOUT)

except subprocess.CalledProcessError as error:

sys.exit('error: {error}')

else:

print('success!')

with open('hosts.txt', 'w') as f:

f.write(output)