Python script for verifying disk space in Linux

Welcome! To the documentation of python script for verifying disk space in Linux.

objective of the python script:

verifying the disk space from Linux and alert . And list the file system which are beyond the threshold limit of 90%

steps:

- **system information**
- **disk space information**
- **alert about disk space**
- **listing the filesystems which are beyond threshold**

Built in modules:

built in modules are pre defines functions. Each built in module contains resources for certain system specific functionalities

system Information:

This step involves getting the information about the system that we are running the script using built in modules of python.

Modules used:

- **os**
- **platform**

os module:

The os module in python provides functions for interacting with the operating system. OS module provides portable way of using operating system dependant functionality

Platform module:

The Platform module is used to get information about the platform on which the program is being executed

1. import platform:

This module already exists in the python library and does not require any installation using pip. It can be imported using the following syntax.

Syntax: import platform

Program for displaying system information:

```
#display system information
print ("system information:")

#import module
import platform

#display operating system
print ('operating system:',platform.system())
#display system information
print ('system host:',platform.node())
#display operating system version
print('os version:',platform.version())
#display machine type
print ('machine type:',platform.machine())
#display processor type
print ('Processor type:',platform.processor())
```

platform.system():This function displays the operating system name.
Platform.node():This function displays the network/host name.
platform.version():This function displays the os version name.
platform.machine():This function displays the machine type name.
platform.processor():This function displays the processor name.

Output of the programe:

```
system information:
operating system: Linux
system host: sugumar-VirtualBox
os version: #33~20.04.1-Ubuntu SMP Mon Feb 7 14:25:10 UTC 2022
machine type: x86_64
Processor type: x86_64
```

Disk space information:

This step involves getting the information about the disk space information.

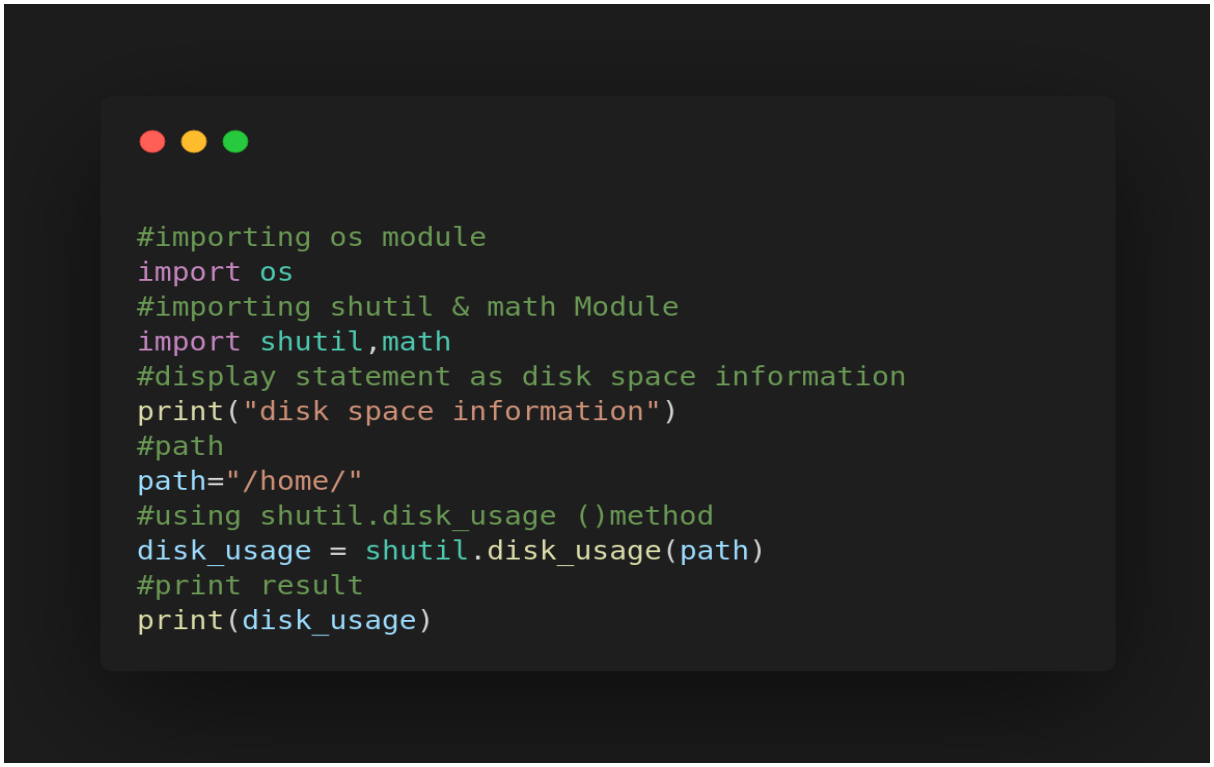
Modules used:

- **os**
- **shutil**

shutil Module:

This Shutil module in python offers several functions to deal with operations on files and Their collection.

Program for disk space information:



```
#importing os module
import os
#importing shutil & math Module
import shutil,math
#display statement as disk space information
print("disk space information")
#path
path="/home/"
#using shutil.disk_usage ()method
disk_usage = shutil.disk_usage(path)
#print result
print(disk_usage)
```

shutil.disk_usage () :
This method tells the disk usage statistics about the given path as a named tuple with the attributes total=Total memory, used=used memory, free=free memory.

syntax= shutil.disk_usage(path)

Output:

```
disk space information
usage(total=21198217216, used=10879488000, free=9218301952)
```

Alert about disk space:

This step involves giving alert about the disk space.

```
#printing datas from disk_usage tuple
a=used= disk_usage[1]
b=total=disk_usage[0]
c=free=disk_usage[2]
#printing used disk%
print('used disk :',int((a/b)*100),'%')
#printing available disk percentage
print ('available disk :',int((c/b)*100),'%')
#printing alerting statement
if ((a/b)*100)>=90:
    print("disk running space is low")
else:
    print ("disk running space is safe" )
```

((used/total)*100) :used disk percentage
((free/total)*100):available disk percentage.

If else condition:

if statement is used to print the statement that if condition is true.
Else condition is used to print the statement that if condition is false.

Syntax: if(condition):
print(statement)
else:
print(statement)

Output:

```
used disk : 51 %
available disk : 43 %
disk running space is safe
```

Listing the filesystem with disk usage beyond 90%:

This step involves listing the file system which are beyond the threshold limit of 90%

1.listing the file system with disk usage

2. listing the file systems with disk usage beyond 90%.

module used :

- **subprocess**

subprocess module:

it allows to Linux/Unix commands directly on python, subprocess module allows you to spawn new processes ,connect to their input/output/error pipes and obtain their return codes.

Listing the file system with disk usage:

```python
#imprt subprcess
import subprocess
#subprocess and shell commands
df = subprocess.run(['df', '-h'], stdout=subprocess.PIPE)
output = df.stdout.decode()
#listing the filesystem
print(output)
```

Output:

```
Filesystem      Size  Used Avail Use% Mounted on
udev            811M     0  811M   0% /dev
tmpfs           169M  1.4M  168M   1% /run
/dev/sda1        20G   11G  8.6G  55% /
tmpfs           844M   56M  788M   7% /dev/shm
tmpfs           5.0M  4.0K  5.0M   1% /run/lock
tmpfs           844M     0  844M   0% /sys/fs/cgroup
/dev/loop0      128K  128K     0 100% /snap/bare/5
/dev/loop1      213M  213M     0 100% /snap/code/88
/dev/loop2      111M  111M     0 100% /snap/core/12725
/dev/loop3       56M   56M     0 100% /snap/core18/2128
/dev/loop4       56M   56M     0 100% /snap/core18/2284
/dev/loop5       62M   62M     0 100% /snap/core20/1361
/dev/loop7      219M  219M     0 100% /snap/gnome-3-34-1804/77
/dev/loop6      219M  219M     0 100% /snap/gnome-3-34-1804/72
/dev/loop8      249M  249M     0 100% /snap/gnome-3-38-2004/99
/dev/loop10      66M   66M     0 100% /snap/gtk-common-themes/1519
/dev/loop13      44M   44M     0 100% /snap/snapd/14978
/dev/loop11      51M   51M     0 100% /snap/snap-store/547
/dev/loop12      55M   55M     0 100% /snap/snap-store/558
/dev/loop9       66M   66M     0 100% /snap/gtk-common-themes/1515
tmpfs           169M   40K  169M   1% /run/user/1000
/dev/sr0         59M   59M     0 100% /media/sugumar/VBox_GAs_6.1.32
```

subprocess.run:

The subprocess. Run Method takes a list of arguments.when the method called , it executes the command and return the completed process.

[‘df -h’]:

df commands shows the information about all mounted filesystems with disk space. To display information about disk drives in human readable format -h flag is used.

Stdout:

stdout is used to display output directly to the screen console

decode():

this method used to convert from one encoding scheme,in which argument is encoded to the desired encoding scheme.

2.listing the files which are over threshold value:

```
import subprocess
#subprocess and shell commands
df = subprocess.run(['df', '-h'], stdout=subprocess.PIPE)
output = df.stdout.decode()
#listing the filesystem
print(output)

usage=dict()
threshold =int(input("enter threshold value"))
for line in output.strip().split('\n')[0:]:
#getting the lines with actual data
    fields=line.split()#cbreak line into fields
    usage[fields[5]]=fields[4]#mapping usage with mount

usage.pop("Mounted")#removing header from the dictionary

file_systems = {k:v for k,v in usage.items() if int(v.strip('%'
))>threshold}
print("files systems beyond threshold limit:",threshold)
print(file_systems)
```

usage=dict():

to make dictionary with output values.

usage[fields[5]]=fields[4]:

To make a dictionary with mount as key and used% as a value

v.strip(“%”):

used for removing percentage from the values

Threshold is a limit value.

For entering threshold value 90 we get output as:

```
enter threshold value90
files systems beyond threshold limit: 90
{'/snap/bare/5': '100%', '/snap/code/88': '100%', '/snap/core18/2128': '100%', '/snap/core18/2284': '100%', '/snap/core/12725': '100%', '/snap/core20/1361': '100%', '/snap/gnome-3-34-1804/72': '100%', '/snap/gnome-3-34-1804/77': '100%', '/snap/gnome-3-38-2004/99': '100%', '/snap/gtk-common-themes/1515': '100%', '/snap/gtk-common-themes/1519': '100%', '/snap/snap-store/547': '100%', '/snap/snap-store/558': '100%', '/snap/snapd/14978': '100%', '/media/sugumar/VBox_GAs_6.1.32': '100%'}
```

if we enter threshold value as 50 we get output like:

```
enter threshold value50
files systems beyond threshold limit: 50
{'/': '55%', '/snap/bare/5': '100%', '/snap/code/88': '100%', '/snap/core18/2128': '100%', '/snap/core18/2284': '100%', '/snap/core/12725': '100%', '/snap/core20/1361': '100%', '/snap/gnome-3-34-1804/72': '100%', '/snap/gnome-3-34-1804/77': '100%', '/snap/gnome-3-38-2004/99': '100%', '/snap/gtk-common-themes/1515': '100%', '/snap/gtk-common-themes/1519': '100%', '/snap/snap-store/547': '100%', '/snap/snap-store/558': '100%', '/snap/snapd/14978': '100%', '/media/sugumar/VBox_GAs_6.1.32': '100%'}
```

conclusion:

Python script for verifying disk space output from Linux and listing the file systems which are 90% or above used and giving alert created and tested successfully..

Thankyou!