STORAGE MONITORING IN LINUX

AIM : Create a Python script to verify disk space output from Linux

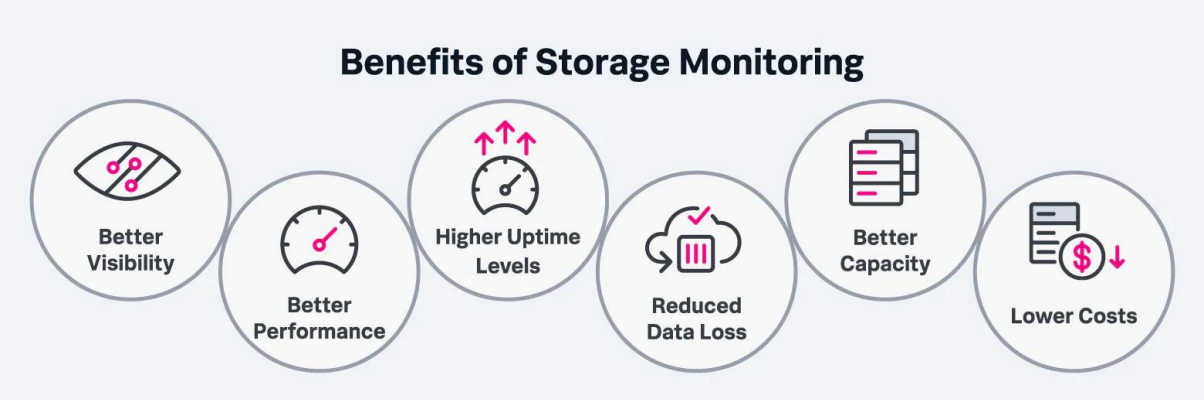
APPLICATIONS :

- LINUX(UBUNTU)
- PYCHARM
- PUTTY

WHAT IS STORAGE MONITORING?

Storage monitoring, or storage performance monitoring, is the practice of tracking the performance, availability and overall health of physical and virtual storage devices. Data storage (provided by vendors such as Dell, EMC, Microsoft as well as countless others) is the backbone of all types of computing endeavors, whether you're working on a spreadsheet, checking your email or playing a video game on your Xbox over the internet. In fact, there is virtually no mainstream computing operation that does not require access to some form of storage.

WHAT ARE THE BENEFITS OF STORAGE MONITORING?



some of the primary advantages of Storage Monitoring

- Better Visibility
- Better Performance
- Higher Uptime Levels
- Reduce Data Lose

- Better Capacity
- Lower Cost

PYTHON_SCRIPT :

```
import os
from psutil import disk_usage
import sys
import logging
import subprocess

THRESHOLD_PERCENTAGE = 90



# Defining logger
def make_logger():
    log = logging.getLogger(_name_)
    log.setLevel(logging.INFO)
    formatter = logging.Formatter('%(filename)s - %(asctime)s - %(levelname)s - %(message)s')
    handler = logging.StreamHandler(sys.stdout)
    handler.setFormatter(formatter)
    log.addHandler(handler)
    return log



def display_disk_usage():
    disk = disk_usage(os.path.realpath('/'))

    logger.info("Total disk space: %d GiB" % (disk.total // (2 ** 30)))
    logger.info("Used disk space: %d GiB" % (disk.used // (2 ** 30)))
    logger.info("Free disk space: %d GiB" % (disk.free // (2 ** 30)))
```

```
    logger.info("Used disk percentage: %d" % disk.percent + '%')


    return disk.percent


def disk_usage_threshold(in_use_disk_percentage):
    if in_use_disk_percentage < THRESHOLD_PERCENTAGE:
        return 'Disk usage is under the threshold level'
    logger.warning(f'\nWarning your disk usage above the threshold it is {in_use_disk_percentage}% full')
    ask_user_for_action = input('If you wish to delete logs and local postgres data print "yes"\n')
    if ask_user_for_action == 'yes':
        clean_disk_usage()
        logger.info('clean disk usage')
    else:
        logger.info(f'Be careful your disk usage is {in_use_disk_percentage}%')



def clean_disk_usage():
    # Stop docker and PM2
    logger.info('\nStopping Dockers and PM2 processes\n')
    subprocess.run('sudo sh /home/pi/Desktop/speedboatBox/scripts/stop_speedboatbox.sh', shell=True, capture_output=True,
                   check=True)


    logger.info('\nDeleting log.out directory\'s content ...\n')
    subprocess.run('sudo truncate -s 0 /home/pi/Desktop/log.out', shell=True, capture_output=True, check=True)


    logger.info('\nDeleting local postgres data-base ...\n')
    try:
```

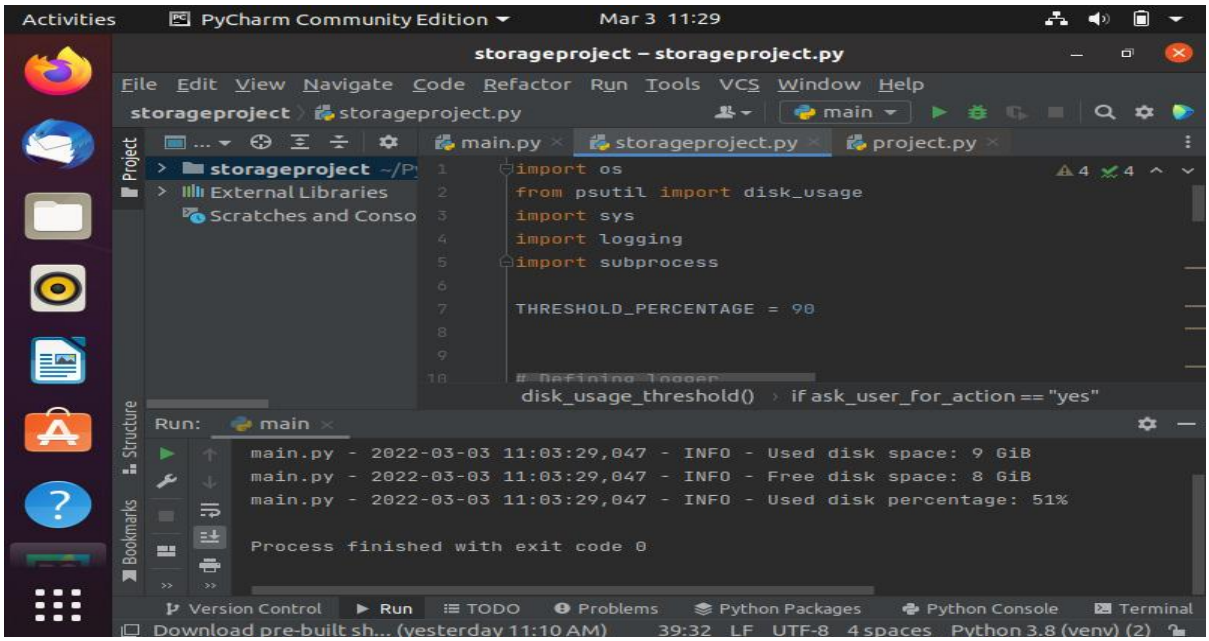
```
        subprocess.run('sudo sh
/home/pi/Desktop/speedboatBox/postgresql_service/delete_db.sh', shell=True,
                       capture_output=True)
    except subprocess.CalledProcessError as e:
        logger.info(e)


    display_disk_usage()


    # Start dockers and PM2 processes
    logger.info('\nStarting dockers and PM2 processes ...\n')
    subprocess.run('sudo sh
/home/pi/Desktop/speedboat/scripts/start_speedboatbox.sh', shell=True,
capture_output=True,
                   check=True)



if _name_ == '_main_':
    logger = make_logger()
    in_use_disk_percentage = display_disk_usage()
    disk_usage_threshold(in_use_disk_percentage)
```



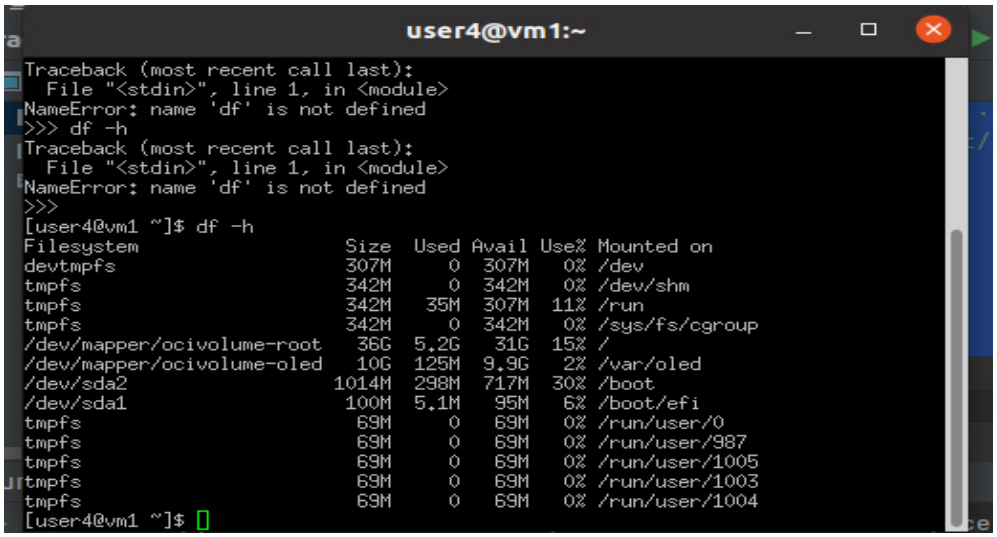
OUTPUT :

Used disk space = 9 GIB

Free disk space = 8 GIB

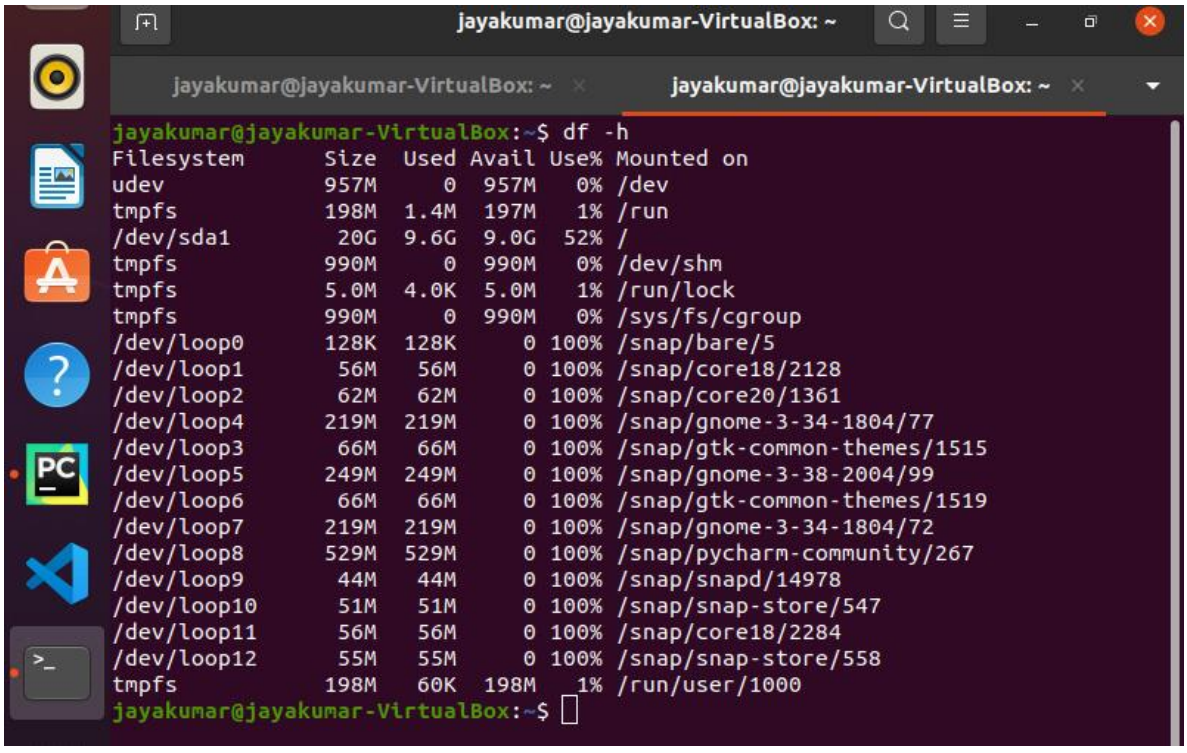
Used disk space = 51%

PUTTY :

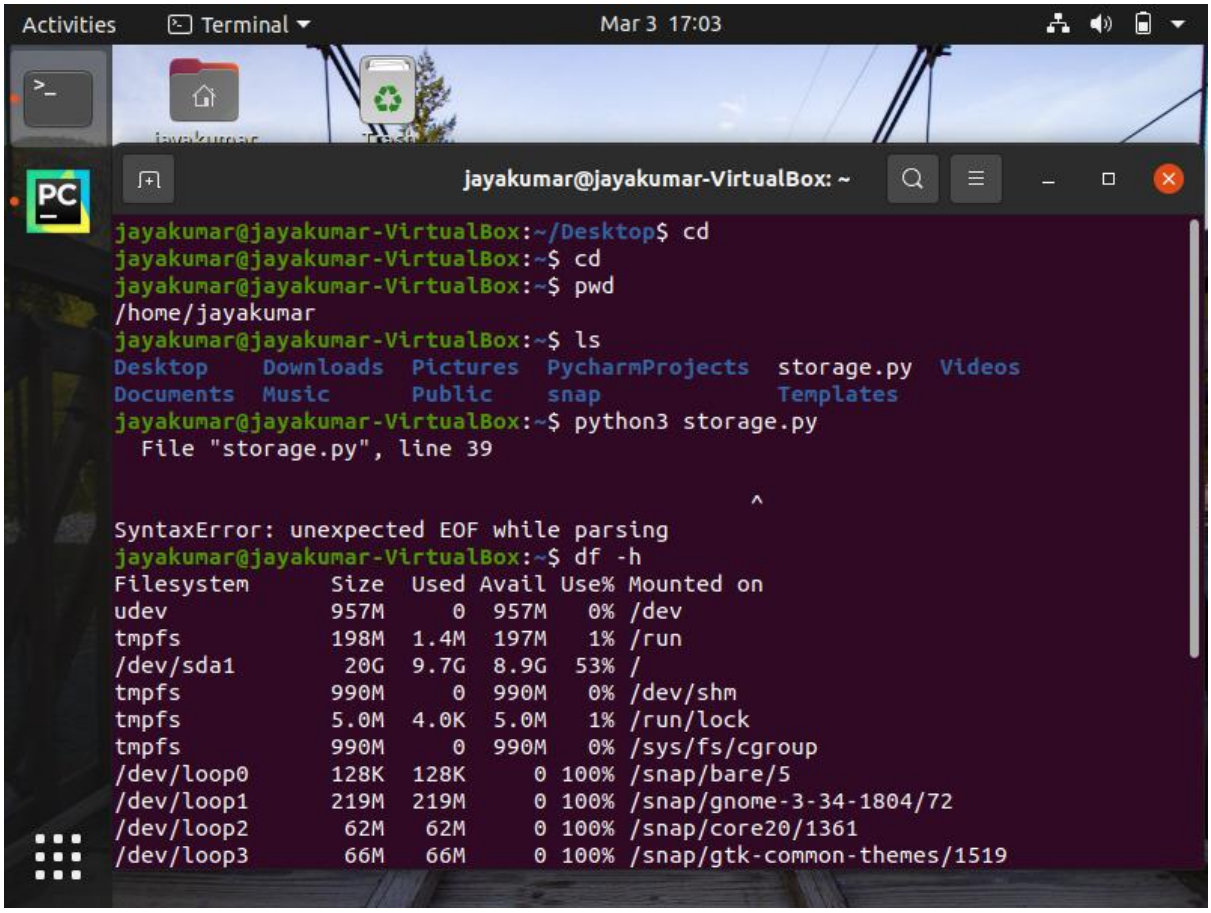


LINUX OUTPUT :

Command : df -h



Command : python3 storage.py



RESOURCE :

- https://youtu.be/asnGiNwo4RU
- www.nagios.com/solutions/storage-monitoring/
- https://www.w3schools.com/python/python_modules.asp
- https://youtu.be/KEIvXwUm8iE