

OS \rightarrow एक विशेष प्रोग्राम है।

OS के वायर्सः

1. resource allocation & deallocation.
2. external device management

Q.

OS के core हैं Kernel.

Shell command / shell script use करते kernel use करते।

Linux open source OS.



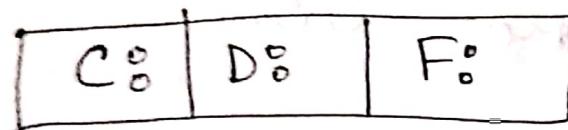
OS 1) win / Linux | Mac OS

2) dual boot | bootable Pendrive

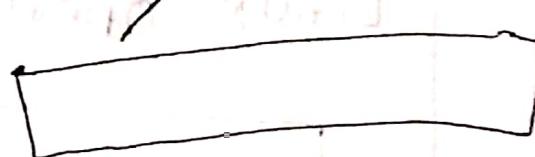
3) VM

VM: 6 একটি OS এর অধিবাস OS এবং
application হিসেবে তার কাজ।
first virtual environment create করে
then অধিবাস মন্তব্য linux install
করো।

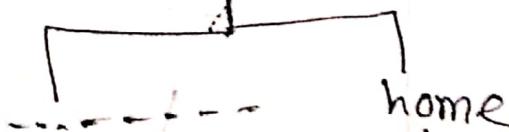
Windows



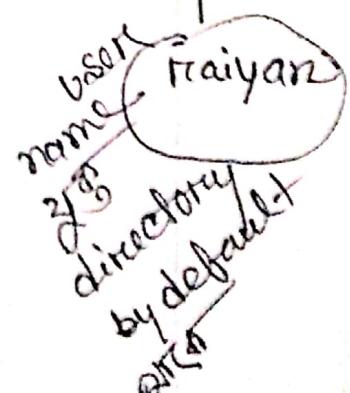
root



root



home



Terminal commands:

whoami → currently root acc দিয়ে
logged in। ১.১৪৩৮৫০৬

pwd - print working directory

by default /home/username

↓ ↓ ↓
root home user directory

faria @ faria: ~ \$

↓
root home /username

cd [এটি ফর্ম হোম ডিকেটরি]
username [এটি ডিকেটরি]

cd [পথ]
change directory: [এটি এবং এখন ডিকেটরি]

[ব্যাপ্ত অসম মত,
file click করে আওয়াজ মত,

Absolute Path	Relative Path
root [মেডে একে রেড -একে রেড]	বাইরি file দ্বা র স্থানের path রেড

cd / → root এ গেন কর

back রয়েছে কাজে double dot (..) নিচের দিকে

/home এ আসো তাহে cd ন ফেলো

হোমে

folder বানাও mkdir [path]

For folder বানাই প্রক্রিয়া ls -a

এবং folder বানাই ls -a

রিমুভ করে folder name

mkdir dir1 dir2 dir3

mkdir -p dir4/dir5/dir6 directory

এখন multiple word মধ্যে directory সৃষ্টি করতে পারবে

মাত্র " " এর মধ্যে নথ করে নিয়ে আসতে হবে

Or mkdir file name1

terminal command save করে রাখতে
ls > output.txt

touch file এর ফার্ম বানাই একটি

prompt এ চেরে

cat name.txt

echo " " → terminal এ print করতে

```
# echo "Hello World 1">> test1.txt  
# echo "Hello World 2">> test2.txt  
# echo "Hello World 3">> test3.txt  
# cat test1.txt test2.txt test3.txt > combine.txt
```

cat combine.txt

Hello world 1

Hello " 2

" " 3

cat command rewrite

মুন্দু সরান্ত করে পুরোটা জোড়া

```
# cat test1.txt test2.txt test3.txt > combine.txt
```

Hello world 1

" " 2

" " 3

" " 1

" " 2

" " 3

মুন্দু file size করে যাওয়ার প্রক্রিয়া scroll

element মেনু হতে জোড়া

less file-path

ଆପୁ ସବୁ କଣ୍ଠେ ପ୍ରେସ୍ତୁତ କାହିଁଲେ file-path

Delete କମ୍ପ୍ୟୁଟରେ କାହିଁଏ:

rm file-name.txt - force file delete

rmdir dir1 → ଖାଲି କିମ୍ବା ନାହିଁ directory
content ଆବଶ୍ୟକ

rm -r dir1 → ଖାଲି କିମ୍ବା
content ଆବଶ୍ୟକ

echo "HelloOOO" > test-2.txt

cat test?.txt

Hello world 1

" " 2

" " 3

cat test*.txt

Hello world 1

Trying HelloOOO

Hello world 2

" " 3

`wc -l test1.txt` → file এর line count
বাবু আবু

Uniq line count আবু

`cat combine.txt | uniq | wc -l`

but ~~it~~ ~~is~~ duplicate line এর মতো আছেন

~~print~~ ~~করে~~ ~~লিখে~~ unique count করতে

পাইথন নামে total line count করতে হবে

১. এখন পর কষ্ট করে `sort` করতে হবে
`sort combine.txt | uniq | wc -l`

↳ 3

move করুন (cut paste)

`mv test1.txt dir`

copy করুন

`cp test1.txt dir`

rename করুন

`mv currentname changed name`

file का permission गृहीत करना

file user group other

chmod 664 test1.txt > after permission
7 → rwx → read execute
6 → rw → write
5 → rx
4 → r
2 → w
ls -l filename → first यादी file
प्रयोग करना for permission
आजू तो test करना है।

self study

Empty file create करना cat >> filename.txt

touch filename

string match करना grep

a string name. filename

head file का 5th line करना head

for second line

head -n 3 temp.txt

শুরুর ৩ লাইন

ব্যক্তি লাইন ছেড়ে দেওয়া

head -n -1 temp.txt

printing all lines except last

1 line

Multiple file এর ক্ষেত্রে

head -n 2 temp.txt temp2.txt

=> temp.txt <=

line 1

line 2

=> temp2.txt <=

line 1

line 2

last শুরুর লাইন print করতে পারিব

tail -n 3 temp.txt

bc এর basic calculator mode

wc file-name → it will give 3 values
line-count word-count string-count file-name
~~for only line-count~~
→ it will give +1 extra for newline

for only line-count
wc -l file-name

for only word-count
wc -w file-name

for only string-count
wc -c file-name

2.11.21

Shell Scripts

collection of command single file
for ex. the file f.sh extension फ़ाइल
save करो, then file को execute करें
output terminal में देता।

File open का command

[फ़ाइल नाम
 $f.sh$]

gedit filename.sh

file open का file का भौतिक command भी है।

1. pwd

2. echo "Hello"

3. ls

Terminal

$f.sh$ → एक फ़ाइल का execute
इसे भौतिक file का execute करना।

यद्यपि permission denied दिखता।

chmod द्वारा change करता।

chmod +rwx f.sh

give everyone all permission

chmod +x f.sh

give everyone execution permission

নিম্নে comment খণ্ড

#!/bin/bash

echo \$0 → bash ক্ষেত্রে

যাইবা bash shell
এখন কোড করবেন।
যাইবা প্রচলিত default
bash নিম্নে কোড করবেন।
তাই লিখি না। আপ্যায়
bash নিম্নে কোড করবেন
লিখতে হবে।

execute

= assignment operation এবং গোল করে
কোন space দেওয়া যাবে না।
অন্য operators এ প্রকৃতি গোল করে space
নিম্ন হবে।

variable

- system variable

(echo \$ USER)
(echo \$ PATH)

→ user এর নাম
ক্ষেত্র।

(~~userpath~~)

→ pwd কি কোড
করবে না
ক্ষেত্র।

- user defined → by default string

↳ a=10 → type declare করবে না।

echo \$ a

→ a এর কোটি variable

অ সুযোগ উপর।

myvar = "Hello"

echo \$myvar → Hello

echo \$"myvar" → "

echo \$myvar → [\$myvar]

echo \\$myvar → [\$myvar]

এই সূচি ক্ষেত্ৰ
য়ে আসে

Input

read var1 var2

variable < value ফিল্টার
করে দেওয়া

echo -n "Enter name" → next line ।
কর্তৃত,

Arithmetic

expr 1 + 1

X=1

x='expr \$x+1'

so arithmetic operation operators normally
কর্তৃত
but * / %

$>: -gt$ shell command
 $>=:$ shell command
Condition shell command
if ($a > b$) if [\$a -gt \$b] ; then

$(\$a -gt \$b)$
Space

elif [\$a -lt \$b] ; then

else

if _else ← fi

b(k)

end

sh

→ (-a)

L-X

||

→ -0

L-X

```
# switch($var)
{   case value1:
    statement1;
    break;
    case value2:
    statement2;
    break;
    default:
    statement3;
}
esac
```

3) condition or operator for false

Yes | Yes) echo "Good Morning";

n^o | N^o) echo "Good Afternoon" ;

Problem:

Input a number
find if it is multiple of 3 or 5

↓ shell command

echo "Enter a Number"

read num

if ['\$num % 3 -eq 0' ; then

'\$num % 5 -eq 0'] ; then

echo "YES"

else

echo "NO"

fi

Terminal

./file-name.sh 20 10 30

command line argument

i.sh → list of command line argument
 echo \$* → no of arguments
 echo \$# → total argument count
 echo \$n → ~~last argument~~
~~last argument~~

Loop

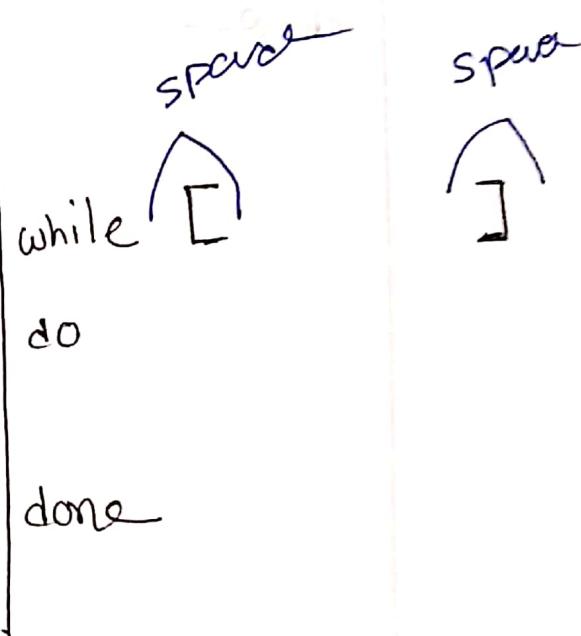
```

for ((i=0; i<=50; i++))  

do  

    echo $i
    
```

done



String Comparison

$s_1 = s_2$

$s_1 \neq s_2$

```
for ((i=1; i <= 100 ; i ++))  
do  
    if [expr $i % 2 -eq 0]; then  
        echo $i  
    fi  
done
```

if slide

or ((++i % 2 == 0 || O = i)) not

not slide

if odds

else slide

slide

if (O == 0) then

slide

else slide

slide

string comparison \neq != sign অর্থাৎ

Until (while এর opposite meaning)

L = D local

Functions

func_name () {
 statement
}

}

declaration & definition

func_name → function calling এর অর্থ
just name কে ফুনকশন হিসেবে

\$* → list of command line arguments

\$# → no of arguments

\$1, .. \$n → nth argument

within a function :

\$* → list of parameters of that function

\$# → number of " " "

Shell script by default set area variable
~~local~~ global

∴ Locally ~~not~~ declare করতেই চাই

Local a=1

প্রোট

variable

global মিথুনের

command line area

./script-name.sh 10 20 30

parameters pass করে script এ মেরুদণ্ড

func_name \$*

func_name 10 20

.sh

```
foo () {  
    echo $1 $3  
}  
  
foo 1 2  
echo $* $#
```

Terminal

```
!./script-name.sh 40 50  
60 70
```

Ans Output: 1
error for \$3 as it is not passed

40 50 60 70 4
 └─────────┘ └─
 \$* \$#

Return কর্যত হলে var এ assign করে ফিল,
function এর মাঝে এসে var প্রিন্ট করে
পরোক্ষ (As variable is global it can be
accessible from anywhere)

Another process

```
f() { echo "123"  
}  
n = "$(f)"
```

multiple echo will
কর্যকর হবে return
ব্যবহার

yes-or-no()

{
echo "Is your name \$??"
echo "Enter yes or no."}

read x

case "\$x" in

y|yes) return 0;;

n|no) return 1;;

esac

function is successful,
executed

function not successful
executed

}

if yes-or-no "\$1" → function
return value?

then

echo "Hi \$1, nice name"

else

echo "Never mind"

fi

return নি বলে কোরি। কর্তৃত কর্ম

variable এ assign করে দিয়ে।

multiple function

f1()

{

echo \$1

}

f2()

{

echo \$2 \$1

}

f1 \$1 \$2

f2 \$3 \$4

Terminal

./script-name.sh 10 20 30 40