S&NA

System Administrator:

a system administrator is a person responsible for the configuration, management and reliable operations of computer system, especially multi users computers such as servers.

System administration refers to the management of one or more hardware and software systems.

Components of SA

=> Installing and config servers
=> Maintain Servers and Servers based activities
=> Troubleshooting Servers related problems
=> Work as a support and
=> solves issues related to user, applications
=> Project management
=> System Monitoring
=> Back up & recovery
=> Manage Access Control

AUSB2

Command Line

SHELL:

Shell is a UNIX Term. It is a program that allows user to interact and give commands to the OS.
SHELL is a command language interpreter that executes commands.

↳ BASH

stands for Bourne Again SHELL.

BASH is a Unix Shell that runs programs in a command line interpreter.
BASH has many built-in commands and consists of combinations of built-in commands

It has ability to launch programs and to control the programs that are launched by it.

JOB Control:

When a job is started, it has taken over the terminal.
It can issues control codes.
Once the program is done, it gives full control back to BASH.

Commands:

Consider a user 'yang' logged into the system

=> wants to launch Firefox

command

[yang@server-7 $ firefox &

user id — ready to accept command — program to run — command to run program

=> to stop the running job and so control returned to the Bash

Simply press

=> Ctrl + Z

=> to identify no of jobs running

=> [user id-7 $ jobs

output [1]+ Running firefox &

The output shows

job name (Firefox)

job number ([1])

Job is in 2 states

either running or stopped

=> To bring job to ~~background~~ foreground i.e give it back the control of terminal

=> [user id-7 $ fg [number]

To send job to background and to give control back to the terminal

[userid -]$ bg [number]

Enviroment Variable:

is a variable that effect the way running process will behave on a computer.

Priting all variable

[userid -]$ printev

Setting Enviromnt Variable

[userid -]$ variable = value

e.g

[userid -]$ FOO = BAR

=> To export variable for other program

[userid -7$ export FOO]

E.V

=> We can use combination is commands like

[userid -7$ export FOO="You are Welcome"]

=> To print value of E.V

[userid-7$ printenv FOO]

=> To unset E.V

[userid -7$ unset (variable name)]

eg unset FOO

Pipes

Pipes are mech through which output of one program can be sent as input of other program.

The vertical bar (|) reps pipes.

In Linux: command in linux pipes executed concurrently

In windows: commands runs in order and intermediate result in stored files.

Redirection

Redirection is the mechanism of taking output of a program and have it automatically send to a file.

3 classes
- ↳ output to a file
- ↳ append to a file
- ↳ send to a file as I/P

(i) To collect output of a program to a file

we use (>) greater than symbol to send output to a file

```
userid -7$ ls > /temp/directory_listing
```

- ls → output of program
- > → command symbol
- /temp/directory_listing → file in which o/p is to be sent

(ii) To append some output or text/code to the end of a directory

we use (>>) double greater than sign

[userid -]$ echo "Direc list" >> /temp/directory_listing

(iii) To use output of a file as input to some program

we use (<) sign that followed by file path name

[userid -]$ grep "root" < /etc/passwd

This command will send output from password directory to grep program

Command-Line Shortcuts

Enviromental Variable as a parameter

BASH allows to use E.V as parameter we use $ befor E.V while passing it as a parmeter to a program

e.g

passing E.V FOO as a parameter
we write
$FOO
so the value of FOO will be passed.

Multiple Commands

We can write multiple command on a same line by seperating them with a semicolon (;)

e.g

```
Tuserid -7$ FOO="100" ; printev FOO
```

Backtick

Backticks are represented by (`)

Anything inside the backticks is treated as a command to be executed.

Backtick is not a quotation sign.
It has a very special meaning.
Everything you write in the backticks is executed first before the main command and the result of command inside back ticks is used as parameter of outer main command

eg

[user id -]$ kill `cat /var/run/named/named.pid`

this returns process #

[user id -]$ kill 2533