

* echo man

- ① -n
⇒ do not output the trailing newline.
- ② -e
⇒ enables interpretation of backslash escapes
- ③ -E
⇒ disables interpretation of backslash escapes (default)

3/1/20

*

⇒ write nano on terminal to create shell script.

- on writing it will open text editor.
- then write program.

★ Simple program

```
echo "welcome to shell"  
echo "Hello world"
```

- press **ctrl + O** to save file
it will ask to name of file
- press **ctrl + X** to close file
- write **sh<filename>.sh** to run shell script

★ Program to assign value

→ nano

```
num=100
```

```
echo "value is :$num"
```

After steps

- **ctrl + O**
- **p2.sh**
- **ctrl + X**
- **sh p2.sh**

★ If condition program

syntax:

```
if [ condition ]
then
    statement / command
else
    statement / command
fi
```

Ⓐ Comparison operator

- gt → greater than
- ge → greater than equal
- eq → equal
- lt → less than
- le → less than equal

Ⓑ Comparison program

nano

x=5

y=10

```
if [ $x -gt $y ]
```

then

echo "x is gr"

else

echo "y is gr"

fi

④ else if condition

Syntax:

if [- condition -]
then

 stmt / command

elif [- condition -] then
 stmt / command

else

 stmt / command

fi

6/1/20

Page No.:

Date: 11

- To compare numeric numbers we use relational operators in []
- To compare string character wise relational operators in []

IMP * redirection

- Take input and store it in screen.
 - Take output and store it in a file.
 - Instead of writing it we can show it as a output.
 - error redirection
- stores output of error in a file
- redirection is used when the output of a command or a script is needed to be sent to a file instead of showing on output staring screen

- 3 types of redirection: Field descriptor

- ① Input redirection [std_{in}] [FD₀]
- ② Output redirection [std_{out}] [FD₁]
- ③ Error redirection [std_{err}] [FD₂]

① Input redirection

↳ Syntax: command name < content/
statement/
command

Eg: ls < echo

① cat < &m

display &m on terminal

② Output redirection

↳ Syntax: command > filename

Eg: ls -a > list

output of this is stored in list
file.

• agar existing file ma nesct command
khele to overwrite thse.

ette append kren pole

ls -a > file

③ Error redirection

↳ command $\boxed{2>}$ filename
↳ due to FD2

Eg: ls -b 2> errorlist

→ Field Descriptor

- Every file in unix is considered as a file and every file has a field descriptor.
- Whenever a script/command is executed 3 files are open which are FDO, FDI & FD2.
- After execution the output is sent to the field descriptor of the output device.

- The field descriptor is responsible to displaying the output on the screen.

→ Error redirection

- Error redirection is used to redirect error to a file while executing shell scripts/ commands to avoid clutter on the screen or to analyse the errors later on

* Pipes

• Syntax:

command1 | command2 | command3

Eg: ls -a | cat > list

→ Pipes are used as a buffer or block of data with two field descriptors where one field descriptor is for reading

and the other field descriptor
is for writing.

→ Pipes are used to run two commands or more than two commands consecutively

* while loop

Syntax

while [condition]

do

statements / command

done

Eg: n=1

echo n

while [\$n -lt 10]

do

echo \$n

'expr \$n + 1' || \$((n+1))

done