**PART-1**

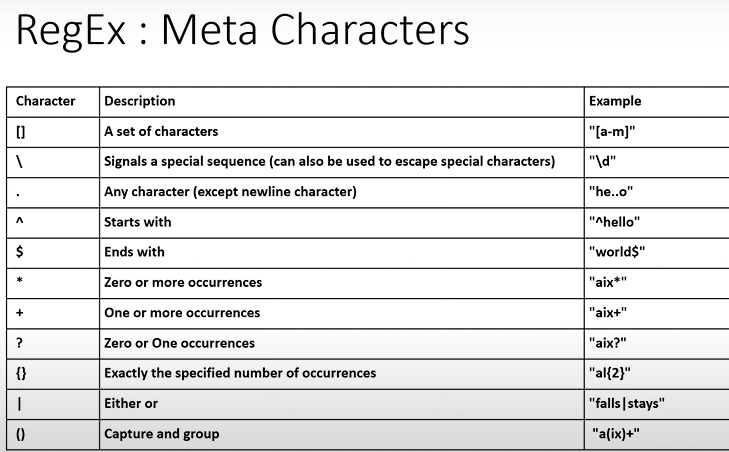
* Linux --> virtual machine
* Echo “Hello world”
* Pwd --> Present working directory
* Cd --> change directory
* **Absolute path --> full path:-**
  + **Cd /home/ec2-user/content/**
* **Relative Path --> short path:-**
  + **Cd home**
* Ls --> list directories
* Ls -l --> long listing
* Ls -a --> all hidden files
* Ls -l -a or ls -la or ls -al (long listing files and hidden files)
* Ls -R --> current list and sub folders list --> R-Recursive
* Ls -r --> **reverse order** while sorting --> opposite to ls -l
* Ls -t --> sort by modification time, newest first --> time sorting
* Also we can use multipule like ( ls -Rt)
* ls /mnt/e/"Susmitha 2025"/ --> list based on file location
* / + tab button on keybord --> to see all Linux paths --> linux file system
* Mnt --> mount
* Clear --> clear the screen
* Cd --> change directory --> linux home path
* Cd .. --> parent directory
* Cd ~ --> home directory
* Cd - (previous directory or last jump)
* Touch file.txt --> create a new empty file
* Touch file --> we can create without file extension
* Touch file.txt --> existing file --> touches the file --> modified current date and time.
* File filename.txt --> we can see type of content ex:- ascii or empty
* Cat filename.txt --> we can see the file data or watching content inside file.
* History --> history of the past commands(top to bottom)
* Motivation:- When you truly understand something, time disappears—you get fully absorbed. Even if you try to forget it later, you won’t be able to. Commands may fade, but deep understanding sticks with you forever. So focus on understanding, not just memorizing.
* Less file.txt --> interactive mode opens like editor --> less is used for large file --> read or parse
* q --> quits the editor
* g --> moves to the Beginning of the text file
* G --> moves to the end of the text file
* /a --> you can search for specific text like ‘a’ inside the text document.
* h / H --> Help
* Up arrow and down arrow to see previous and future commands
* Cp (copy from one folder to another folder)
* Mkdir newfolder -->creates new folder
* Cp filename.txt newfolder --> it copies to the newfolder
* Cp \*.txt newfolder or cp \*.txt newfolder/--> copy .txt all files
* Rm file.txt --> delete the filename or remove filename
* Rm file.txt myfile.txt --> multifile remove
* Rm -r newfolder --> r for recursive directory --> delete the folder or directory
* Rmdir newfolder --> remove the empty folder only
* Rm -r newfolder or rmdir newfolder --> both are same to delete the folder
* Cp \*.txt newfolder/folder --> multifolder
* Mv kkk.txt ktm.txt --> rename the file name or folder name
* Mv hello hellow.txt --> also rename the extension
* Mv \*.txt newfolder --> move files to newfolder --> \*.txt(whiled card)
* Mv /home/ec2-user/content/myfile.txt /home/ec2-user/content/myfile1.txt
* Echo “susmitha is a good girl” > file.txt
* . dot--> current directory
* Cp -i myfile.txt foldername --> I means input promt --> without -I it will overwrite the file
* Mv -b folder1 folder2 --> b means backup -->The backup file ends with ~
* Mkdir -p folder/doler1
* find path -name filename.txt
* Find path -type d -name myfolder --> optional type
* Man find --> Interactive mode like less --> about find --> man means manual knowing find details.
* Whatis find
* Whatis cat
* Whatis ls -->command stands for what is what --> short help
* Help echo
* Ls --help
* Alias --> how many alias are having
* Alias name=”ls -la” --> set short commands
* Unalias name --> undo alias
* Unalias -a --> remore all shortcuts alias
* Exit or logout --> to exit terminal

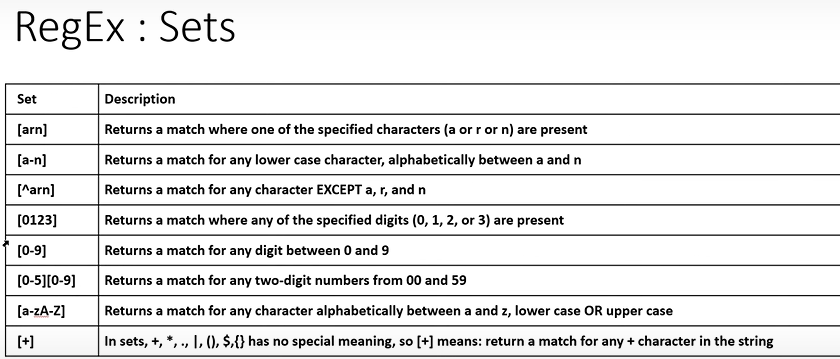
**PART-2 --> Text manipulation commands**

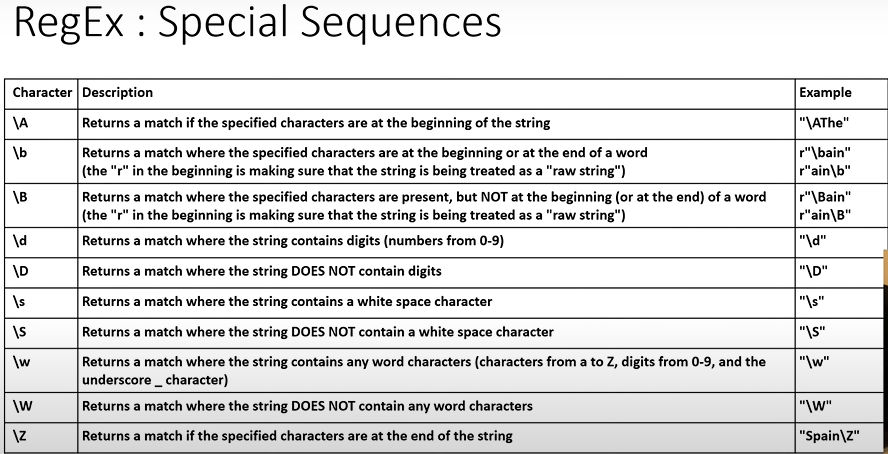
* Echo “hello world” --> like a print statement
* Echo “hello susmitha” > peanut.txt --> to write/replace content in file
* Echo “new line” >> peanut.txt --> (>> for append content to the file) --> existing content + newline.
* Cat < peanut.txt > sample.txt --> (‘<’ for input & ‘>’ for output) --> standard input and standard output.
* Ls /fake/dir>out.txt --> no such file or directory and created empty file
* Pwd >out.txt --> created new file having pwd
* Standard input(<) --> denote 0 , Standard output(>) --> denote 1 , Error --> denote 2
* Echo “good morning” 1> susmitha.txt
* Ls /fake/dir 2> out.txt --> standard error message to write/redirect in file.
* Ls /fake/dir 2> /dev/null --> any message goes to emptyness.
* Etc folder --> host specific configuration files --> etc in linux directory
* Ls -la /etc | less
* Ls | tee new2.txt
* Cat >amg.txt --> cat is input and write in amg.txt
* Ctrl+D --> save and exit from cat
* Printenv --> variable and value
* Export name=”mitran”
* Echo “Hello $name , how are you ?”
* Unset name
* Sudo su - root --> sudo switch user ubantu to root user
* exit --> from root user to ubantu user

**REGULAR EXPRESSION:-**

* ^[a-zA-Z0-9]+[@][a-zA-Z]+[.][a-zA-z.]+$ -->Email pattern
* Cat > sample.txt
* Grep “Mitran” sample.txt--> grep means search/find
* Grep -i “Mitran” sample.txt--> -i means both uppercase letters and lowercase letters
* Grep -I “mitran” \*.txt --> search in files or folder not subfolders
* Grep -iw “mitran” sample.txt --> -w means whole word
* Grep -iwn “mitran” sample.txt --> -n means displays with number.
* Grep “[A-Z]\*” sample.txt --> capital letters search
* Sed s/@/U/g sample.txt --> s-substitute/@-find/U-replace/g-global
* Awk ‘{print $1}’ employee.txt --> $1 means - 1st column - column filtering
* Awk ‘/sales/ {print $1, $4}’ employee.txt -->sales means row filtering
* Cat /var/log/syslog --> very very large logs
* Tail /var/log/syslog --> displays last 10 lines --> default 10
* Tail -n 3 /var/log/syslog --> displays last 3 lines
* Head /var/log/syslog --> displays first 10 lines --> default 10
* Head -n 3 /var/log/syslog --> displays first 3 lines
* Tail -f filename --> to watch live logs
* In a file have 100 lines , display 30-45 lines --> interview question
* Head -n 45 /var/log/syslog | tail -n 16 --> first 45 lines after last 16 lines output 30-45
* **Vi or vim test.txt --> linux text editor**
* Command line mode by default replace --> up arrow , down arrow
* I --> insert any data
* Esc --> to come outside
* :wq --> w for save and q for quit and enter
* dd --> to delete the line
* :%d --> % means everything, d - for delete.
* :2d --> 2 means 2 lines delete based on cursor where we mension
* :q! --> for quit without save
* Sort new.txt --> sorting assending order
* Sort -r new.txt --> -r means reverse --> sorting descending order
* Ls -l | sort --> sorting files
* Ls -l | sort -r --> reverse order
* Uniq new2.txt --> uniq words
* Uniq -c new2.txt --> count of uniq words
* Uniq -d new2.txt --> displaying duplicate values
* Sort new2.txt
* Sort new2.txt | uniq
* Sort new2.txt | uniq -c
* Wc new2.txt --> world count --> output 14-no of lines, 14-no of words 71-no of byte or characters-->
* Wc -lcw new2.txt --> l-lines, c-count of letters, w -words…

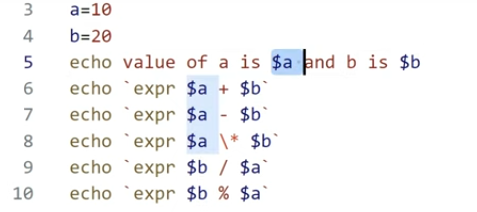


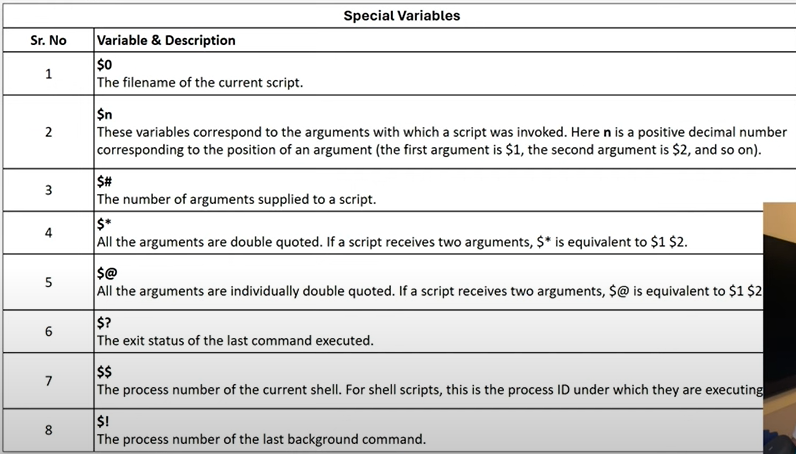


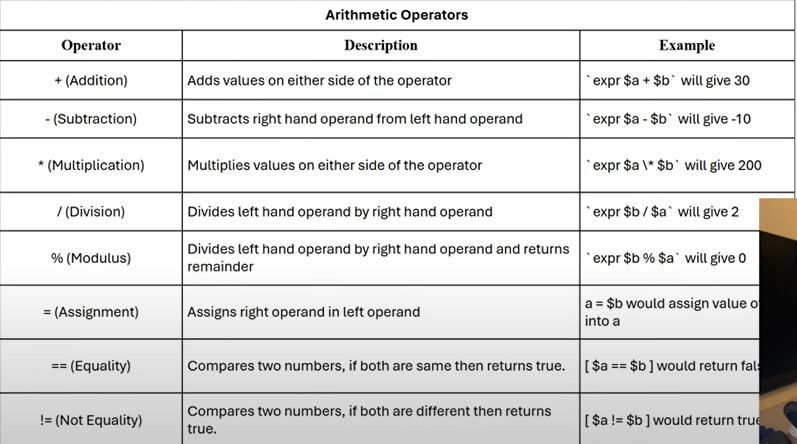


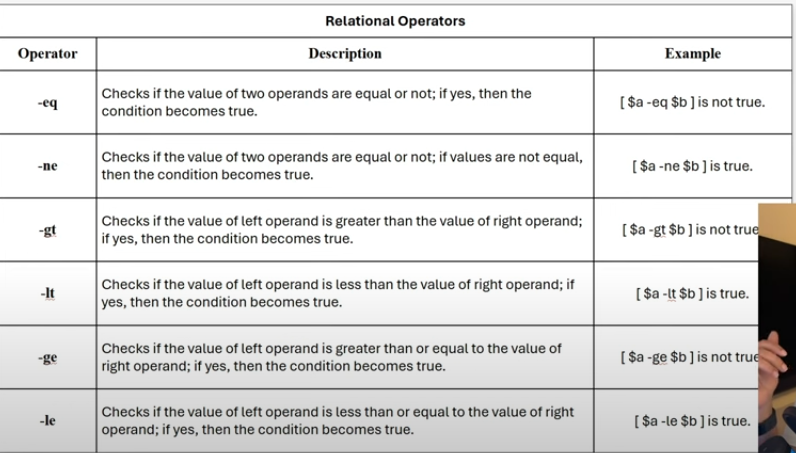
**PART-3 --> Introduction to Shell Scripting**

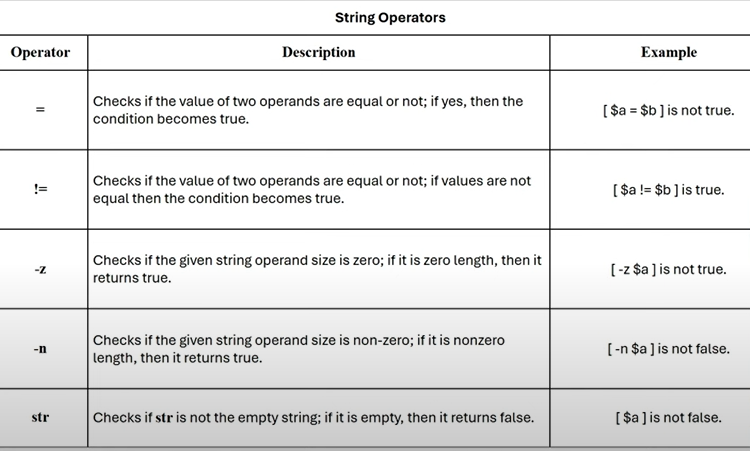
* Code . --> in ubantu code dot ,it will autopen vscode --> ubantu interconnected to vscode through WSL.
* Touch main.sh --> create new file
* Cat /etc/shells --> availabe shells display
* #! /bin/bash --> #! means shebang
* Chmod +x main.sh -->+ means add , x means executable permissions.
* Sh main.sh or ./main.sh --> to run shell script
* Printenv --> environment variables
* Export age=26 --> Env variable --> local check ,env check
* Echo $HOME --> shell variable

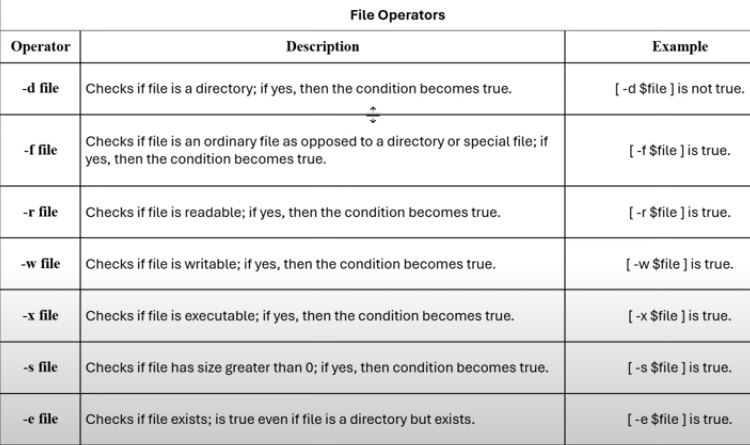






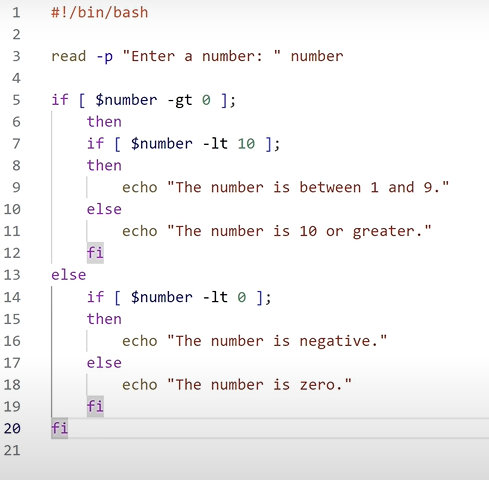




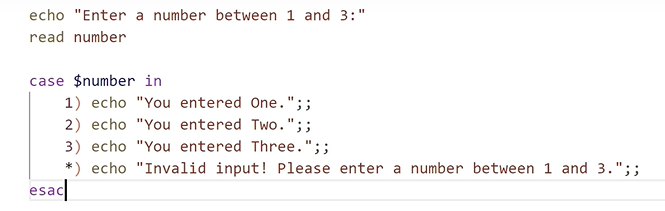


**PART-4 --> Shell Scripting with control statements.**

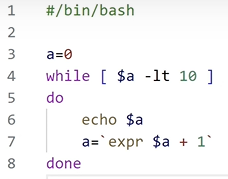
* Nested if else



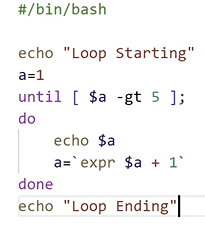
* Switch case



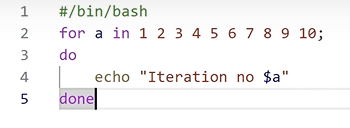
* While loop --> If condition is true



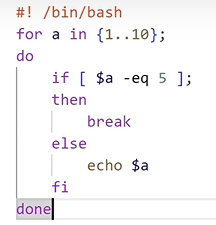
* Until Loop --> oppsite to while loop --> if condition is false



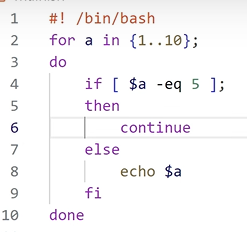
* For range --> for x in {1..50..2};



* Break

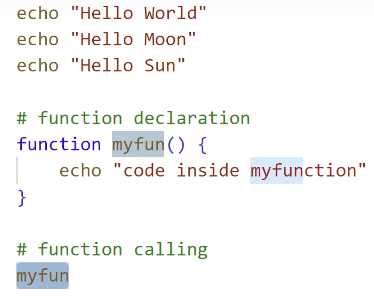


* Continue

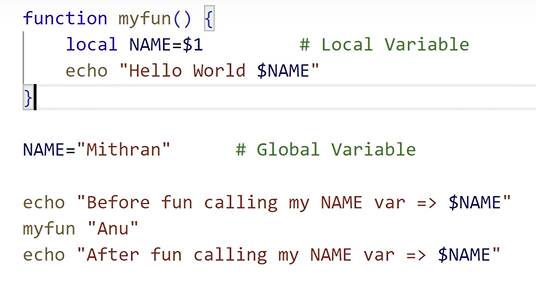


**PART-5 --> Shell Scripting with Functions, variables and bugging.**

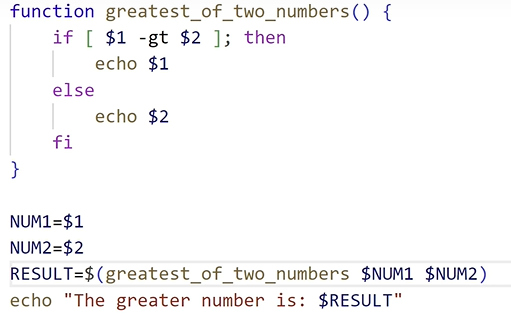
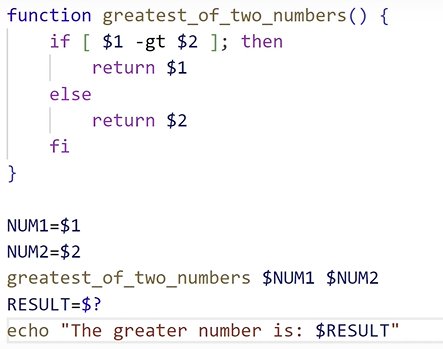
* Function



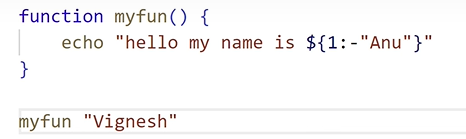
* Variable



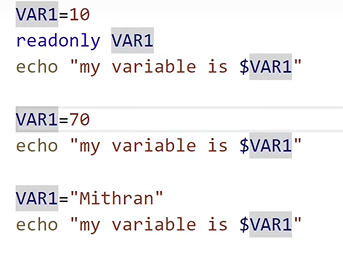
* Function with return and echo.



* Function with default argument.



* Read on variable



* Debugging mode --> bash -x main.sh

