



**MTC @ GT**

# **WORKSHOP**

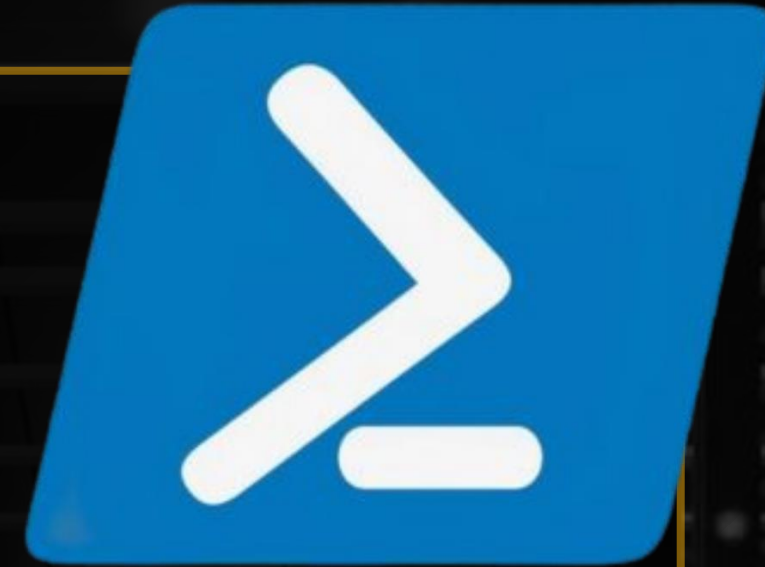
# **INTRO TO SHELLS**

---

# WHAT IS A SHELL?

INTRO TO SHELLS

```
root@mtc:/home> man shell
```



Basic interface for a computer



Run scripts

Change system state

IPC

Process  
Management

Work with files

Run syscalls



# THERE ARE MULTIPLE SHELLS?

## INTRO TO SHELLS

There are a lot of shells out there. You should know about these:

**1. sh** (Original bourne shell)

1. Default shell on older/small Unix systems

**2. bash** (Bourne Again SHell)

1. By far most popular shell today

**3. csh** (C Shell)

- Competitor to sh/bash – syntax inspired by the C Programming language
- Modern variant is tcsh
- DO NOT USE
  1. No one uses it anymore
  2. Terrible documentation/online support

**4. Zsh** (Z shell)

1. Default shell on newer Macs (licensing issues w/ bash)

**5. PowerShell** – Modern shell developed by Microsoft, only major shell that is truly cross-platform





# SHELL SYNTAX – VARIABLES

## INTRO TO SHELLS



```
1 name="Alice"           # Mind the spaces
2 echo $name
3 echo ${#name}          # 5
4
5 # Substrings
6 name="Alice"
7 echo ${name:0:2}       # "Al"
8
9 # Array variables
10 fruits=("apple" "banana" "orange")
11 echo ${fruits[0]}      # First element
12 echo ${fruits[@]}      # All elements
13 echo ${#fruits[@]}     # Array length
```





# SHELL SYNTAX – CTL FLOW

## INTRO TO SHELLS



```
1 # Conditionals
2 name="Alice"
3 if [ "$name" = "Alice" ];
4 then
5     echo "Hello Alice"
6 elif [ "$name" = "Bob" ];
7 then
8     echo "Hello Bob"
9 else
10    echo "Hello stranger"
11 fi
```



```
1 # Loops
2 for i in {1..5}; do
3     echo "Number: $i"
4 done
5
6 counter=1
7 while [ $counter -le 5 ];
8 do
9     echo "Count: $counter"
10    counter=$((counter +
11    1))
12 done
```





# SHELL SYNTAX - CTD

## INTRO TO SHELLS



```
1 # Command substitution -- $(command)
2 res=$(./fib 12)
3 res=`./fib 12` # Legacy syntax, not recommended
4
5 # Pipe operator ("|") feeds result into another command
6 ls | grep "*.txt"
7 # You can chain pipes
8 ls | grep "*.txt" | wc -l # Outputs num text files in (.)
```





# SHELL SYNTAX - CTD

## INTRO TO SHELLS



```
1 # Redirection
2 echo 2 + 3 > my_file # Overwrite my_file
3 echo print\('hello'\) >> my_file # Append to end of my_file
4
5 python3 < my_file # Redirect my_file to stdin of python3
6
7 echo hi && lol &> my_file # &> only redirects errors
```





# SHELL SYNTAX - LOGIC

## INTRO TO SHELLS



```
1 # Logic
2 echo $((1 + 2)) # Math done in $(( ... ))
3 echo $((8 ** 2)) # Exponents
4
5 # Watch the spaces
6 if [ -d . ]; then echo this is a dir; fi
7 if [ -f my_file ]; then echo this is a file; fi
8 if [ -e my_file ]; then echo file or dir found; fi
9 if [ ! -e nope ]; then echo no file found; fi
10 if [ 1 -eq 1 ]; then echo 1 = 1; fi # -gt, -lt too
11
12 typoe && echo 1st command worked!
13 typo || echo 1st command failed!
```





# SPECIAL VARIABLES

## INTRO TO SHELLS

<b>\$0</b>	Name of current script
<b>\$#</b>	# of arguments passed to the script
<b>\$*</b>	Script args as a string
<b>@</b>	Script args as an array
<b>\$1-\$9</b>	Args 1 thru 9
<b>\$?</b>	Status of the last command / most recently executed process
<b>\$_</b>	PID (Process ID) of the last background command
<b>\$\$</b>	Gets the PID of the current shell
<b>\$-</b>	Current set of options in current shell
<b>_</b>	Output of last command



# TOOLS YOU SHOULD KNOW

## INTRO TO SHELLS

- |                    |                          |                        |
|--------------------|--------------------------|------------------------|
| 1. <b>cat</b>      | 17. <b>ssh</b>           | 33. <b>kill</b>        |
| 2. <b>cd</b>       | 18. <b>scp</b>           | 34. <b>systemctl</b>   |
| 3. <b>ls</b>       | 19. <b>python3</b>       | 35. <b>sudo</b>        |
| 4. <b>rm/rmdir</b> | 20. <b>gcc</b>           | 36. <b>su</b>          |
| 5. <b>touch</b>    | 21. <b>lscpu</b>         | 37. <b>chmod/chown</b> |
| 6. <b>mv/cp</b>    | 22. <b>uname</b>         | 38. <b>clear</b>       |
| 7. <b>ln</b>       | 23. <b>top</b>           | 39. <b>history</b>     |
| 8. <b>source</b>   | 24. <b>ps</b>            | 40. <b>which</b>       |
| 9. <b>env</b>      | 25. <b>alias</b>         | 41. <b>export</b>      |
| 10. <b>ip</b>      | 26. <b>exit</b>          | 42. <b>true/false</b>  |
| 11. <b>echo</b>    | 27. <b>nc/nmap</b>       | 43. <b>less</b>        |
| 12. <b>tar</b>     | 28. <b>head/tail</b>     |                        |
| 13. <b>man</b>     | 29. <b>diff</b>          |                        |
| 14. <b>pwd</b>     | 30. <b>vim/nano/code</b> |                        |
| 15. <b>apt</b>     | 31. <b>grep</b>          |                        |
| 16. <b>tree</b>    | 32. <b>find</b>          |                        |





# TIPS & TRICKS

## INTRO TO SHELLS

- Autocomplete
- History search
- RC Files
  - **.bashrc**
  - **.profile**
  - **.bash\_aliases / .aliases**
- Everything in Linux is a file!!



**YOUR TURN!**

**[github.com/Saad-Mufti/intro-to-shells](https://github.com/Saad-Mufti/intro-to-shells)**





# FEEDBACK

