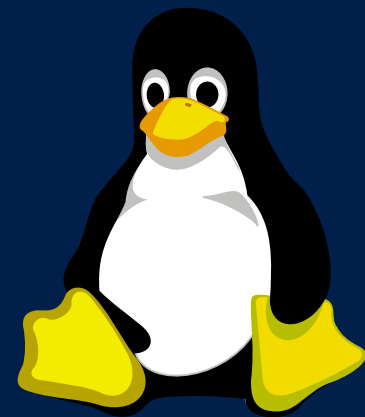




ALRIGHT!

# LINUX TUTORIAL



# AWK SCRIPTING

# Syntax

awk options 'pattern {action}' file\_name

echo "Hello" | awk options 'pattern {action}'

options:

-F field separator

-v var=value

-f file

**How *AWK* works?**

**test.txt**

**This is line one**

**This is line two**

**Row 1**      **This is line one**

**Row 2**      **This is line two**

## Fields

	F1	F2	F3	F4
--	----	----	----	----

Row 1	This	is	line	one
-------	------	----	------	-----

Row 2	This	is	line	two
-------	------	----	------	-----



# Terms used in AWK

**NR** - No. of record/row

**NF** - No. of fields

**\$0** - Print everything

**\$1, \$2** - Field no.

# Examples

- **Print only a given column**
- **Print last column**
- **Search a word**
- **Print only a given line no. (let's say line 5)**
- **Print row or line no. at start of each line**

- **Print range of lines (line3 to 6)**
- **Get line no. of empty lines**
- **Search multiple words**
- **Ignore case while searching**
- **How to check if a given char is present in column**

# Working with CSV or different delimiter

- How to work with CSV file
- Print data of employees whose salary>50k
- What if a file is having multiple delimiter

# Use Cases Where AWK can be useful

- How to only get Status of service
- How to get list of files
- How to read logs in range of time
- Get files modified in Oct

# Useful Functions

- How to replace a word?
- Length of line/field
- Index/position of a word in a given line
- Print values in upper or lower letter

# AWK Scripting Concepts

awk

**'BEGIN{start\_action}**

**pattern/condition {action}**

**END{stop\_action}'**

file\_name



# Other Programming Concepts we can use

- Variables
- If-Else
- Array
- For Loop
- While loop
- Functions

- **How to find total/sum of salary**
- **How to find average salary**
- **How to print no. of lines**
- **How to ignore headers/first row to count  
no. of users**

# Using Conditions

- How to get length of longest line?
- Print High for salary >50k else LOW
- Print Total salary paid in Loan  
department

# Using AWK condition/pattern in a file

- Add your conditions and pattern in a file let's say file.awk
- `awk -f file.awk <file.txt>`

# Using AWK as a script

```
#!/bin/awk -f
```

```
BEGIN{start_action}
```

```
    pattern/condition {action}
```

```
END{stop_action}
```

- Print only given feild/column
  - **awk '{print \$2}' file**
- Print last row or last column
  - **awk '{print \$NR}' file**
  - **awk '{print \$NF}' file**
- Search for a word in file
  - **awk '/word/ {print \$2}' file**

- Print only given line no.
  - `awk 'NR==5 {print $0}' file`
- Print line with each row
  - `awk '{NR print $0}' file`
- Search for multiple words in file
  - `awk '/word1|word2/ {print $2}' file`

- Read from CSV file and print a given column
  - `awk -F, '{print $1}' file`
- Print range of lines from a file
  - `awk 'NR==3, NR==6 print $0' file`
- Get the empty lines
  - `awk 'NF==0 {print NR}' file`



- Print data of employees, salary > 50000
  - `awk -F, '$NF>50000' file`
- P

# Functions

- Replacing a word in a file
  - `awk -F, '{gsub("from", "to") print $0} file'`
- Print index/position of a given word
  - `echo "Hi Paul" | awk '{print index($0, "Paul")}'`
- Get length of a word/row
  - `echo "Hi Paul" | awk '{print length($2)}'`

- Split the words in a row in an array
  - `awk -F, '{split($0, arr, ","); print arr[1]} file'`
- Print

# AWK as Script

awk ‘

**BEGIN{start\_action}**

pattern/condition {action}

**END{end\_action}**

‘

file\_name

**awk** -f script.awk **file\_name**

- Find sum/total salary
  - `awk -F, '{sum+=$NF} END{print sum}' file`
- Find average salary
  - `awk -F, '{sum+=$NF} END{print sum/NR}' file`