Scripting with AWK: A Step further!

Diptesh Kanojia

e-Yantra Team
ERTS Lab, IIT Bombay

IIT Bombay
April 17, 2021

Agenda for Discussion

1. AWK: Why do I need it?
    - Are you happy with SED?
2. Making it ‘AWK’ward!
    - What is AWK?
    - AWK Playfield
    - Example ‘AWK’lings

Motivation

AWK is a text processing langauge which is a step ahead of SED. It overcomes the limitations of SED and is known to be a more advanced tool for manipulating data in a file.

Motivation

AWK is a text processing langauge which is a step ahead of SED. It overcomes the limitations of SED and is known to be a more advanced tool for manipulating data in a file.

AWK is a tool that provides a more programming-like environment allowing you to modify and reorganize data in a file. It also has a *GNU-based sister utility called ‘gawk’*; please do not get confused. They work in a similar fashion.

Motivation

AWK is a text processing langauge which is a step ahead of SED. It overcomes the limitations of SED and is known to be a more advanced tool for manipulating data in a file.

AWK is a tool that provides a more programming-like environment allowing you to modify and reorganize data in a file. It also has a *GNU-based sister utility called ‘gawk’*; please do not get confused. They work in a similar fashion.

It possesses an extensive set of operators and capabilities but we will cover only a few of those which are useful for shell scripting.

What is AWK?

AWK was originally intended to be a data-driven programming language when Alfred **A**ho, Peter **W**einberger, and Brian **K**ernighan wrote it at Bell Labs.

What is AWK?

AWK was originally intended to be a data-driven programming language when Alfred **A**ho, Peter **W**einberger, and Brian **K**ernighan wrote it at Bell Labs.

AWK separates each line passed to it into fields. A field consists of a string of consecutive characters delimited by whitespace, though there are options for changing this.

What is AWK?

AWK was originally intended to be a data-driven programming language when Alfred **A**ho, Peter **W**einberger, and Brian **K**ernighan wrote it at Bell Labs.

AWK separates each line passed to it into fields. A field consists of a string of consecutive characters delimited by whitespace, though there are options for changing this.

AWK parses and operates on each separate field. This makes it ideal for handling structured text files – especially tables – data organized into consistent chunks, such as rows and columns.

Fields and Patterns

awk ‘{print “Welcome to AWK tutorial”}’

echo one two | awk ‘{print $1}’

- $0 for the whole line.
- $1 for the first field.
- $2 for the second field.
- $n for the nth field.

Variables and Options

AWK can have an optional BEGIN{} section of commands that are done before processing any content of the file.

Variables and Options

AWK can have an optional BEGIN{} section of commands that are done before processing any content of the file.

It then, has the main {} section which works on each line of the file.

Variables and Options

AWK can have an optional BEGIN{} section of commands that are done before processing any content of the file.

It then, has the main {} section which works on each line of the file.

Finally there is an optional END{} section of actions that happen after the file reading has finished.

Variables and Options

AWK can have an optional BEGIN{} section of commands that are done before processing any content of the file.

It then, has the main {} section which works on each line of the file.

Finally there is an optional END{} section of actions that happen after the file reading has finished.

Let us try some example AWK commands and see their output for a better understanding.

Example AWK Commands

```
ls -l | awk 'BEGIN {sum=0} {sum=sum+$5} END {print sum}'
ls -l | awk '{for (i=1;i<3;i++) {getline}; print NR,$0}'
awk -F: '{print $1}' /etc/passwd
awk 'NF > 0' sample.txt
awk 'BEGIN { for (i = 1; i <= 7; i++) print int(101 * rand()) }'
```

Practice Eggsamples

Let us go to the terminal and practice these commands!

In this practice, we shall also see how to make AWK scripts and not just use AWK on the command line.

References

- TLDP: AWK
- Linux Command Line and Shell Scripting Bible
- e-Yantra Homepage

Thank You!

Author: Diptesh Kanojia
Contributor: Prashant Sharma

Post your queries at: resources@e-yantra.org