Hello! AWK

Awk is a powerful tool in the commandline used for processing the rows and columns of a flat texy file. Awk has built in `string functions` and `associative arrays`. Awk supports most of the operators, conditional blocks, and loops available in C language. You may want to know, what `Awk` stands for? It comes from the surnames of its authors *"Aho, Weinberger, and Kernighan"*. AWK was created at Bell Labs in the 1970s. It is pronounced the same as the name of a bird called `auk`. The GNU implementation of awk is called `gawk`.

This tutorial will give you **just enough** knowledge to read and understand this book, to be a master on AWK, you need to explore relevant literature referenced at end of this book.

The AWK language is a fully data-driven scripting language consisting of a set of `actions` to be taken against streams of textual data - either run directly on files or used as part of a `pipeline` for purposes of extracting or transforming text, such as producing formatted reports.

The very basic syntax of AWK:

```
awk 'BEGIN {start-action} {action} END {stop-action}' filename
```

Note that the actions in the *begin block* are performed before processing the file and the actions in the *end block* are performed after processing the file. The rest of the actions are performed while processing the file!

It can be also written as:

```
awk '/search pattern1/ {Actions}
     /search pattern2/ {Actions}' file
```

In the above AWK syntax:

- search pattern is a regular expression;
- `Actions` are the statement(s) to be performed;
- several patterns and actions are possible in AWK;
- a `file` is an input file; and
- single quotes around program is to avoid shell not to interpret any of its special characters.