

2.4 awk 命令_物联网 / 嵌入式工程师 - 慕课网

“ 慕课网慕课教程 2.4 awk 命令涵盖海量编程基础技术教程，以图文图表的形式，把晦涩难懂的编程专业用语，以通俗易懂的方式呈现给用户。

AWK 语言的基本功能是在文件或者字符串中基于指定规则浏览和抽取信息。awk 抽取信息后，才能对其他文本操作。它是一个强大的文本分析工具。简单来说 awk 就是把文件逐行的读入，以空格为默认分隔符将每行切片，切开的部分再进行各种分析处理。之所以叫 AWK 是因为其取了三位创始人 Alfred Aho, Peter Weinberger, 和 Brian Kernighan 的 Family Name 的首字符。

注：sed 命令常用于一整行的处理。而 awk 更倾向于把一行分为多个“字段”然后进行处理。

- awk 的适用场景

- 超大文件处理；
- 输出格式化的文本报表；
- 执行算数运算；
- 执行字符串操作等。

- 语法

- 格式

- `awk [options] 'pattern {action}' filename`

options : 可选参数

-F : 指明输入时用到的字段分隔符，默认分隔符为空格或tab键

-v (var=VALUE) : 自定义变量

pattern : 匹配规则

action : 某些计算操作/格式化数据/流控制语句

filename: 文件名

示例：

```
awk -F ":" '{print $1}' /etc/passwd
```

```
awk '条件1 {动作1} 条件2 {动作2} ... ' 文件名
```

条件 (pattern) :

一般使用关系表达式作为条件，条件符合执行对应的动作。

`x > 10` 判断变量x是否大于10

`x >= 10` 大于等于10

`x <= 10` 小于等于

动作 (Action):

- 示例用法

- 新建一个 student.txt, 内容如下：

ID	NAME	PHP	Linux	MySQL	Average
1	Liming	82	95	86	87.66
2	Sc	74	96	87	85.66
3	Gao	99	83	93	91.66

- 测试命令

- 输出文件内容

```
[linux#linux] awk '{print}' student.txt
```

ID	NAME	PHP	Linux	MySQL	Average
1	Liming	82	95	86	87.66
2	Sc	74	96	87	85.66
3	Gao	99	83	93	91.66

- 输出第 2 列内容

- ```
[linux#linux] awk '{print$2}' student.txt
```

  
NAME  
Liming  
Sc  
Gao

- 格式输出

- ```
[linux#linux] awk '{print $2 $6}' student.txt
```

 //不调整格式输出, 连在一起
NAMEAverage
Liming87.66
Sc85.66
Gao91.66

```
[linux#linux] awk '{print $2"\t"$6}' student.txt
```

NAME	Average
Liming	87.66
Sc	85.66
Gao	91.66

- awk 输出磁盘信息

- ```
[linux#linux] df -Th | grep tmpfs | awk '{print $1"\t"$5}'
```

  
udev 1.9G  
tmpfs 388M  
tmpfs 2.0G  
tmpfs 5.0M  
tmpfs 2.0G  
tmpfs 391M  
tmpfs 391M

- 场景设计: 当我们发现我们的 / dev/sda1 磁盘空间占用率的数字

- ```
[linux#linux] df -Th | grep "dev/sda1" | awk '{print $6}' | cut -d "%" -f 61
```

- awk BEGIN 关键字

- 介绍: 默认情况下, awk 会从输入中读取一行文本, 然后针对该行的数据执行程序脚本, 但有时可能需要在处理数据前运行一些脚本命令, 这就需要使用 BEGIN 关键字。

- 格式

- ```
BEGIN{commands}
```

  
功能: 在执行awk命令前, 先执行BEGIN对应的动作

- 示例用法

- ```
[linux#linux] awk '{print $2 "\t" $5}' student.txt
```


NAME MySQL
Liming 86
Sc 87
Gao 93


```
[linux#linux] awk 'BEGIN{print "test start!";} {print $2 "\t" $5}' student.txt
```


test start!
NAME MySQL
Liming 86
Sc 87
Gao 93

- awk END 关键字

和 BEGIN 关键字相对应, END 关键字允许我们指定一些脚本命令, awk 会在读完数据后执行它们, 例如:

```
[linux#linux] cat /etc/passwd | tail -1 | awk -F ":" 'BEGIN {print "The data3 File Contents: "} {pr
```


The data3 File Contents:
x
End of File

练习 1:

获得 df -Th 命令中 / dev/sda1 显示的总磁盘的大小。

全文完

本文由 简悦 SimpRead 优化，用以提升阅读体验

使用了 全新的简悦词法分析引擎 beta，点击查看详细说明

