

7 Linux Grep OR, Grep AND, Grep NOT Operator Examples

 thegeekstuff.com/2011/10/grep-or-and-not-operators

10/21/2011

by Ramesh Natarajan on October 21, 2011

Question: Can you explain how to use OR, AND and NOT operators in Unix grep command with some examples?

Answer: In grep, we have options equivalent to OR and NOT operators. There is no grep AND operator. But, you can simulate AND using patterns. The examples mentioned below will help you to understand how to use OR, AND and NOT in Linux grep command.

The following employee.txt file is used in the following examples.

```
$ cat employee.txt
100 Thomas Manager Sales $5,000
200 Jason Developer Technology $5,500
300 Raj Sysadmin Technology $7,000
400 Nisha Manager Marketing $9,500
500 Randy Manager Sales $6,000
```

You already knew that grep is extremely powerful based on these [grep command examples](#).

Grep OR Operator

Use any one of the following 4 methods for grep OR. I prefer method number 3 mentioned below for grep OR operator.

1. Grep OR Using \|

If you use the grep command without any option, you need to use \| to separate multiple patterns for the or condition.

```
grep 'pattern1\|pattern2' filename
```

For example, grep either Tech or Sales from the employee.txt file. Without the back slash in front of the pipe, the following will not work.

```
$ grep 'Tech\|Sales' employee.txt
100 Thomas Manager Sales $5,000
200 Jason Developer Technology $5,500
300 Raj Sysadmin Technology $7,000
500 Randy Manager Sales $6,000
```

2. Grep OR Using -E

grep -E option is for extended regexp. If you use the grep command with -E option, you just need to use | to separate multiple patterns for the or condition.

```
grep -E 'pattern1|pattern2' filename
```

For example, grep either Tech or Sales from the employee.txt file. Just use the | to separate multiple OR patterns.

```
$ grep -E 'Tech|Sales' employee.txt
100 Thomas Manager Sales $5,000
200 Jason Developer Technology $5,500
300 Raj Sysadmin Technology $7,000
500 Randy Manager Sales $6,000
```

3. Grep OR Using egrep

egrep is exactly same as 'grep -E'. So, use egrep (without any option) and separate multiple patterns for the or condition.

```
egrep 'pattern1|pattern2' filename
```

For example, grep either Tech or Sales from the employee.txt file. Just use the | to separate multiple OR patterns.

```
$ egrep 'Tech|Sales' employee.txt
100 Thomas Manager Sales $5,000
200 Jason Developer Technology $5,500
300 Raj Sysadmin Technology $7,000
500 Randy Manager Sales $6,000
```

4. Grep OR Using grep -e

Using grep -e option you can pass only one parameter. Use multiple -e option in a single command to use multiple patterns for the or condition.

```
grep -e pattern1 -e pattern2 filename
```

For example, grep either Tech or Sales from the employee.txt file. Use multiple -e option with grep for the multiple OR patterns.

```
$ grep -e Tech -e Sales employee.txt
100 Thomas Manager Sales $5,000
200 Jason Developer Technology $5,500
300 Raj Sysadmin Technology $7,000
500 Randy Manager Sales $6,000
```

Grep AND

5. Grep AND using -E 'pattern1.*pattern2'

There is no AND operator in grep. But, you can simulate AND using grep -E option.

```
grep -E 'pattern1.*pattern2' filename
grep -E 'pattern1.*pattern2|pattern2.*pattern1' filename
```

The following example will grep all the lines that contain both "Dev" and "Tech" in it (in the same order).

```
$ grep -E 'Dev.*Tech' employee.txt
200 Jason Developer Technology $5,500
```

The following example will grep all the lines that contain both "Manager" and "Sales" in it (in any order).

```
$ grep -E 'Manager.*Sales|Sales.*Manager' employee.txt
```

Note: Using [regular expressions in grep](#) is very powerful if you know how to use it effectively.

6. Grep AND using Multiple grep command

You can also use multiple grep command separated by pipe to simulate AND scenario.

```
grep -E 'pattern1' filename | grep -E 'pattern2'
```

The following example will grep all the lines that contain both “Manager” and “Sales” in the same line.

```
$ grep Manager employee.txt | grep Sales
100 Thomas Manager Sales $5,000
500 Randy Manager Sales $6,000
```

Grep NOT

7. Grep NOT using grep -v

Using grep -v you can simulate the NOT conditions. -v option is for invert match. i.e It matches all the lines except the given pattern.

```
grep -v 'pattern1' filename
```

For example, display all the lines except those that contains the keyword “Sales”.

```
$ grep -v Sales employee.txt
200 Jason Developer Technology $5,500
300 Raj Sysadmin Technology $7,000
400 Nisha Manager Marketing $9,500
```

You can also combine NOT with other operator to get some powerful combinations.

For example, the following will display either Manager or Developer (bot ignore Sales).

```
$ egrep 'Manager|Developer' employee.txt | grep -v Sales
200 Jason Developer Technology $5,500
400 Nisha Manager Marketing $9,500
```

> [Add your comment](#)

If you enjoyed this article, you might also like..