find text strings recursively in Linux and UNIX - find grep in sub-directories command examples

Author : admin

GNU Grep is equipped with a special option "**-r**" to grep recursively. **Looking for string in a file in a sub-directories tree with the -r option is a piece of cake**. You just do:

```
grep -r 'string' /directory/
```

or if you want to *search recursively non-case sensitive for text*

```
grep -ri 'string' .
```

Another *classic GNU grep use (I use almost daily) is whether you want to match all files containing (case insensitive) string among all files*:

```
grep -rli 'string' directory-name
```

Now if you want to *grep whether a string is contained in a file or group of files in directory recursively* on some other **UNIX** like **HP-UX** or **Sun OS / Solaris** where there is no GNU grep installed by default here is how to it:

```
find /directory -exec grep 'searched string' {} dev/null ;
```

Note that **this approach to look for files containing string on UNIX is very slow**Thus on not too archaic UNIX systems for some better search performance it is better to use **xargs**;

```
find . | xargs grep searched-string
```

A small note to open here is by using **xargs** there might be weird results when run on filesystems with filenames starting with **"-"**.

Thus comes the **classical (ultimate) way to grep for files containing string** with ***find + grep***, e.g.

```
find / -exec grep grepped-string {} dev/null ;
```

Another way to search a string recursively in files is by using UNIX OS '*' (star) expression:

```
grep pattern * */* */*/* 2>/dev/null
```

Talking about ***recursive directory text search in UNIX***, should mention another good GNU GREP alternative ACK - check it on betterthangrep.com :) . **Ack** is perfect for programmers who have to **dig through large directory trees of code for certain variables, functions, objects etc.**