**Finding Stuff**

|  |  |
| --- | --- |
| Recursively search a file hierarchy for the files that have a specified characteristic:  find [starting directories] [characteristics] | Search the **contents** of one or more files for a textual pattern:  grep regularExpression FILES |

**find command: [starting directories]**

|  |  |
| --- | --- |
| (blank) | Start at the current directory |
| . | Start at the current directory |
| path/to/startingDirectory | Relative to current directory |
| /path/to/startingDirectory | Relative to root directory |

**find command: [characteristics]**

|  |  |
| --- | --- |
| -name *pattern* | File globbing – file name matches the pattern |
| -iname *pattern* | Like **-name**, but the match is case insensitive. |
| -regex *pattern* | File name matches regular expression *pattern*. |
| -iregex *pattern* | Like **-regex**, but the match is case insensitive. |
| -lname *pattern* | File is a symbolic link whose contents match shell *pattern*. |
| -ilname *pattern* | Like **-lname**, but the match is case insensitive |
| -user *uname* | File is owned by user *uname* |
| -group *gname* | File belongs to group *gname* |
| -size *n*[cwbkMG] | File uses *n* units of space, rounding up |
| -executable | Matches files which are executable and directories which are searchable |
| -perm *mode* | File's permission bits are exactly *mode* (octal or symbolic) |
| -empty | File is empty and is either a regular file or a directory. |
| -amin *n* | File was last accessed *n* minutes ago. |
| -anewer *file* | File was last accessed more recently than *file* was modified. |
| -atime *n* | File was last accessed *n*\*24 hours ago. |
| -cmin *n* | File's status was last changed *n* minutes ago. |
| -cnewer *file* | File's status was last changed more recently than *file* was modified. |
| -ctime *n* | File's status was last changed *n*\*24 hours ago. |
| -mmin *n* | File's data was last modified *n* minutes ago. |
| -mtime *n* | File's data was last modified *n*\*24 hours ago. |
| -newer *file* | File was modified more recently than *file*. |
| -used *n* | File was last accessed *n* days after its status was last changed |
| -maxdepth n | Only go this deep into the file hierarchy (1 means current folder only) |
| -type t | d means directory, f means regular file |
|  |  |
| *expr1* -a *expr2* | logical AND combination of two characteristics (default if missing) |
| *expr1* -o *expr2* | logical OR combination of two characteristics |
| -not *expr* | logical negation of a characteristic |

If the set of characteristics result in TRUE for a specific file, its name is output to stdout.

File globbing for filename patterns must be enclosed in quotes to protect them from shell substitutions.

Combine characteristics using parentheses to override precedence, but () must be escaped:  
 \( … \)

To perform operations on the files that are found by find, pipe the find output to xargs.

**xargs command:**

xargs cmd will execute the specified cmd on each line coming from stdin.

Notes:

* The find command has *actions* that can be performed, but don't use them. Use xargs.
* Verify that you are accessing the correct files before executing actions on the files using xargs.

**grep command: "global regular expressions print"**

|  |  |
| --- | --- |
| grep –option pattern filename  egrep –option pattern filename  fgrep –option pattern filename  pgrep –option pattern filename | searches for a matching pattern (regular expression)  grep –E (supports extended regular expressions)  grep –F (fast grep that does **not** use regular expressions)  searches process ID's; equivalent to **ps -aux | grep process name** |

Sends each line of the input that matches the specified pattern to stdout.

If no filename(s) are specified, the filenames are read from stdin.

Matching Options:

|  |  |
| --- | --- |
| -i | Ignore case |
| -v | Invert the sense of matching, to select non-matching lines. |
| -w | Select only those lines containing matches that form whole words. |
| -a | Process a binary file as if it was text |

Output Options:

|  |  |
| --- | --- |
| -c | Print a count of matching lines for each input file. |
| -n | Prefix each line of output with the 1-based line number within its input file. |
| -C num | Print num lines of leading and trailing context. |
| -o | Output only the matching characters (not the entire line that contained the pattern). |

To search multiple files from a file hierarchy, pipe find ouput into grep using xargs:

find path/to/search | xargs grep "pattern"