How to do it...

The `grep` command is the magic Unix utility for searching in text. It accepts regular expressions, and can produce output in various formats. Additionally, it has numerous interesting options. Let's see how to use them:

1. To search for lines of text that contain the given pattern:

   ```
   $ grep pattern filename
   this is the line containing pattern
   ```

   Or:

   ```
   $ grep "pattern" filename
   this is the line containing pattern
   ```

2. We can also read from `stdin` as follows:

   ```
   $ echo -e "this is a word\nnext line" | grep word
   this is a word
   ```

3. Perform a search in multiple files by using a single `grep` invocation, as follows:

   ```
   $ grep "match_text" file1 file2 file3 ...
   ```

4. We can highlight the word in the line by using the `--color` option as follows:

   ```
   $ grep word filename --color=auto
   this is the line containing word
   ```

5. Usually, the `grep` command only interprets some of the special characters in `match_text`. To use the full set of regular expressions as input arguments, the `-E` option should be added, which means an extended regular expression. Or, we can use an extended regular expression enabled `grep` command, `egrep`. For example:

   ```
   $ grep -E "[a-z]+" filename
   ```

   Or:

   ```
   $ egrep "[a-z]+" filename
   ```

6. In order to output only the matching portion of a text in a file, use the `-o` option as follows:

   ```
   $ echo this is a line. | egrep -o "[a-z]+\."
   line.
   ```

7. In order to print all of the lines, except the line containing `match_pattern`, use:

   ```
   $ grep -v match_pattern file
   ```

   The `-v` option added to `grep` inverts the match results.

8. Count the number of lines in which a matching string or regex match appears in a file or text, as follows:

```
$ grep -c "text" filename
10
```

It should be noted that `-c` counts only the number of matching lines, not the number of times a match is made. For example:

```
$ echo -e "1 2 3 4\nhello\n5 6" | egrep  -c "[0-9]"
2
```

Even though there are six matching items, it prints `2`, since there are only two matching lines. Multiple matches in a single line are counted only once.

9. To count the number of matching items in a file, use the following trick:

```
$ echo -e "1 2 3 4\nhello\n5 6" | egrep -o "[0-9]" | wc -l
6
```

10. Print the line number of the match string as follows:

```
$ cat sample1.txt
gnu is not unix
linux is fun
bash is art
$ cat sample2.txt
planetlinux

$ grep linux -n sample1.txt
2:linux is fun
```

or

```
$ cat sample1.txt | grep linux -n
```

If multiple files are used, it will also print the filename with the result as follows:

```
$ grep linux -n sample1.txt sample2.txt
sample1.txt:2:linux is fun
sample2.txt:2:planetlinux
```

11. Print the character or byte offset at which a pattern matches, as follows:

```
$ echo gnu is not unix | grep -b -o "not"
7:not
```