How it works...

The regular expressions are really easy to design part-by-part. In the e-mail regex, we all know that an e-mail address takes the form `name@domain.some_2-4_letter`. Here, the same is written in the regex language as follows:

```
[A-Za-z0-9.]+@[A-Za-z0-9.]+\.[a-zA-Z]{2,4}
```

`[A-Za-z0-9.]+` means that some combination of characters in the `[]` block should appear one or more times (that is the meaning of +), before a literal `@` character appears. Then, `[A-Za-z0-9.]` also should appear one or more times (`+`). The pattern `\.` means that a literal period should appear, and finally, the last part should be 2 to 4 alphabetic characters.

The case of an HTTP URL is similar to an e-mail, but we don't need the `name@` match part of the e-mail regex.

```
http://[a-zA-Z0-9.]+\.[a-zA-Z]{2,3}
```

See also

- The *Using sed to perform string replacement* recipe in this chapter explains the `sed` command
- The *Using regular expressions* recipe in this chapter explains how to use regular expressions

Removing a sentence in a file containing a word

Removing a sentence containing a word is a simple task when a correct regular expression is identified. This is just an exercise on solving similar problems.

Getting ready

`sed` is the best utility for making substitutions. Hence, let's use `sed` to replace the matched sentence with a blank.

How to do it...

Let's create a file with some text to carry out the substitutions. For example:

```
$ cat sentence.txt
Linux refers to the family of Unix-like computer operating systems
that use the Linux kernel. Linux can be installed on a wide variety of
computer hardware, ranging from mobile phones, tablet computers and video
game consoles, to mainframes and supercomputers. Linux is predominantly
known for its use in servers.
```

We will remove the sentence containing the words `mobile phones`. Use the following `sed` expression for this task:

```
$ sed 's/ [^.]*mobile phones[^.]*\.//g' sentence.txt
Linux refers to the family of Unix-like computer operating systems
that use the Linux kernel. Linux is predominantly known for its use in
servers.
```

This recipe assumes that no sentence spans more than one line, for example, a sentence should always begin and end on the same line in the text.

How it works...

Let's evaluate the `sed` regex `'s/ [^.]*mobile phones[^.]*\.//g'`. It has the format `'s/substitution_pattern/replacement_string/g`. It replaces every occurrence of `substitution_pattern` with the replacement string.

Here, the substitution pattern is the regex for a sentence. Every sentence is delimited by ".", and the first character is a space. Therefore, we need to match the text that is in the format `"space" some text MATCH_STRING some text "dot"`. A sentence may contain any characters except a "dot", which is the delimiter. Hence, we have used `[^.]`. `[^.]*`matches a combination of any characters except the dot. In between, the text match string "`mobile phones`" is placed. Every match sentence is replaced by `//` (nothing).

See also

- The *Using sed to perform text replacement* recipe in this chapter explains the `sed` command
- The *Using regular expressions* recipe in this chapter explains how to use regular expressions