14.6 Tokenizing `Strings`

When you read a sentence, your mind breaks it into **tokens**—individual words and punctuation marks that convey meaning to you. Compilers also perform tokenization. They break up statements into individual pieces like keywords, identifiers, operators and other programming-language elements. We now study class `String`'s `split` method, which breaks a `String` into its component tokens. Tokens are separated from one another by **delimiters**, typically white-space characters such as space, tab, newline and carriage return. Other characters can also be used as delimiters to separate tokens. The application in Fig. 14.18 demonstrates `String`'s `split` method.

When the user presses the *Enter* key, the input sentence is stored in variable `sentence`. Line 14 invokes `String` method `split` with the `String` argument `" "`, which returns an array of `String`s. The space character in the argument `String` is the delimiter that method `split` uses to locate the tokens in the `String`. As you'll learn in the next section, the argument to method split can be a regular expression for more complex tokenizing. Lines 15–16 display the length of the array `tokens`—i.e., the number of tokens in `sentence`. Lines 18–20 output each token on a separate line.

```
1  // Fig. 14.18: TokenTest.java
```

```
2  // Tokenizing with String method split
3  import java.util.Scanner;
4
5  public class TokenTest {
6     // execute application
7     public static void main(String[] args) {
8        // get sentence
9        Scanner scanner = new Scanner(System.in);
10       System.out.println("Enter a sentence and pr
11       String sentence = scanner.nextLine();
12
13       // process user sentence
14       String[] tokens = sentence.split(" ");
15       System.out.printf("Number of elements: %d%n
16          tokens.length);
17
18       for (String token : token {
19          System.out.println(token);
20       }
21    }
22 }
```

```
Enter a sentence and press Enter
This is a sentence with seven tokens
Number of elements: 7
The tokens are:
This
is
a
sentence
with
seven
tokens
```

Fig. 14.18

Tokenizing with `String` method `split`.