

java.util.Scanner

CSCI 136: Spring 2022

The Scanner

A `Scanner` object can parse user input entered on the console or from a file. A `Scanner` breaks its input into separate tokens (which are typically separated by white space), and then returns them one at a time. The scanner provides methods to convert the tokens into values of different types. For example, this code reads two numbers from the console and prints their sum:

```
Scanner in = new Scanner(System.in);
int i = in.nextInt();
int j = in.nextInt();
System.out.println(i+j);
```

`Scanner` also provides methods to test whether there is any input left, and if there is, what type of token appears next. This functionality is provided through methods like `hasNextInt`, `hasNextDouble`. For example, the following code reads integers and adds them up until there is no more input or a non-numeric token is encountered:

```
Scanner in = new Scanner(System.in);
int sum = 0;
while (in.hasNextInt()) {
    sum += in.nextInt();
}
```

Creating Scanners

Whenever using `Scanner`, be sure to include the following import:

```
import java.util.Scanner;
```

One can create a `Scanner` in one of two ways:

1. To read from the console, use the following:

```
Scanner input = new Scanner(System.in);
```

2. To read from a file, use the following:

```
Scanner input = new Scanner(new FileStream("filename.txt"));
```

Scanner Methods

Method	Computes
<code>nextBoolean()</code>	reads and converts next token to a boolean value
<code>nextInt()</code>	reads and converts next token to a integer value
<code>nextLong()</code>	reads and converts next token to a long value
<code>nextDouble()</code>	reads and converts next token to a double value
<code>next()</code>	reads next token and returns it as a <code>String</code>
<code>nextLine()</code>	reads until the next new line and returns a <code>String</code>
<code>hasNextBoolean()</code>	returns true iff the next token is either "true" or "false"
<code>hasNextInt()</code>	returns true iff the next token is an integer
<code>hasNextLong()</code>	returns true iff the next token is a long
<code>hasNextDouble()</code>	returns true iff the next token is a real number
<code>hasNext()</code>	returns true iff there is at least one more token of input
<code>hasNextLine()</code>	returns true iff there is another line of input