

## next\* Methods

The following methods each read some input from the console and consume that input, advancing the cursor to the position directly after the text that was read in. Remember that we say a method “blocks” if it suspends the program until it receives input.

- `next()` - skips whitespace and returns the next word as a `String`. Blocks until the user enters a word. Leaves the cursor positioned immediately after the word that was read.
- `nextInt()` - like `next()`, but interprets the word as a decimal integer. Crashes if the word cannot be interpreted as a decimal integer.
- `nextDouble()` - like `next()`, but interprets the word as a decimal number. Crashes if the word cannot be interpreted as a decimal number.
- `nextLine()` - returns all characters up to the next newline (`'\n'`) character entered. Blocks until the user enters a newline character. Leaves the cursor positioned immediately after the newline character.

## hasNext\* Methods

`nextInt()` and `nextDouble()` have the ability to crash your program. To protect your program from these crashes, you can use the following methods to “peek ahead” into the input and find out in advance whether calling `nextInt()` and `nextDouble()` will cause a crash. Note that the methods below do not advance the cursor, so calling them twice in a row will always yield the same result. Also note that if there is no input available (i.e. the user hasn't typed anything yet), both of these methods will block until the user enters a word of input.

- `hasNextInt()` - Returns `true` if and only if the next call to `nextInt()` will not cause a crash.
- `hasNextDouble()` - Returns `true` if and only if the next call to `nextDouble()` will not cause a crash.

## Rules of Thumb

1. Never call `nextInt()` or `nextDouble()` without first calling `hasNextInt()` or `hasNextDouble()` to check if the operation is safe. Only call the `next*` method if the corresponding `hasNext*` method returned true.
2. Every time you call `hasNextInt()` or `hasNextDouble()`, make sure to call one of the `next*` methods (which might be just `next()`, since this method cannot crash) before you make another `hasNext*` call. This makes sure that you consume the next word of input before you use a `hasNext*` method to peek ahead again.