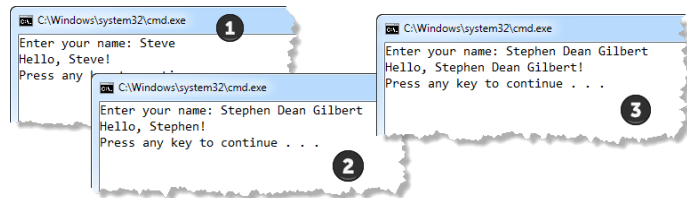


String Input and Output

You may use `>>` and `<<` to read and write `string` objects, like this:

```
1 | cout << "Enter your name: ";
2 | string name;
3 | cin >> name;
4 | cout << "Hello, " << name << "!" << endl;
```

This version of the program reads a `string` input by the user into the variable `name` and then includes `name` as part of the greeting, as shown in the screenshots below:



1. If the user enters only a first name, then all goes as you'd expect.
2. However, the user enters a **full name** instead of just the first, only the first is read.

Even though the program contains no code to split the name apart, it somehow still uses **only** the first name when it prints its greeting.

Why? Because `>>` **stops reading** as soon as it sees the first **whitespace character**. A whitespace character is any character that appears as blank space on the screen, and includes the tab and newline characters.

3. To read **an entire line of text**, use the `string` function `getline()` like this, in place of line 3:

```
3 | getline(cin, name);
```

This **reads an entire line** from `cin` into the variable `name`. When run, the program allows you to display the full name of the user instead of just the first name.



This course content is offered under a [CC Attribution Non-Commercial](#) license. Content in this course can be considered under this license unless otherwise noted.