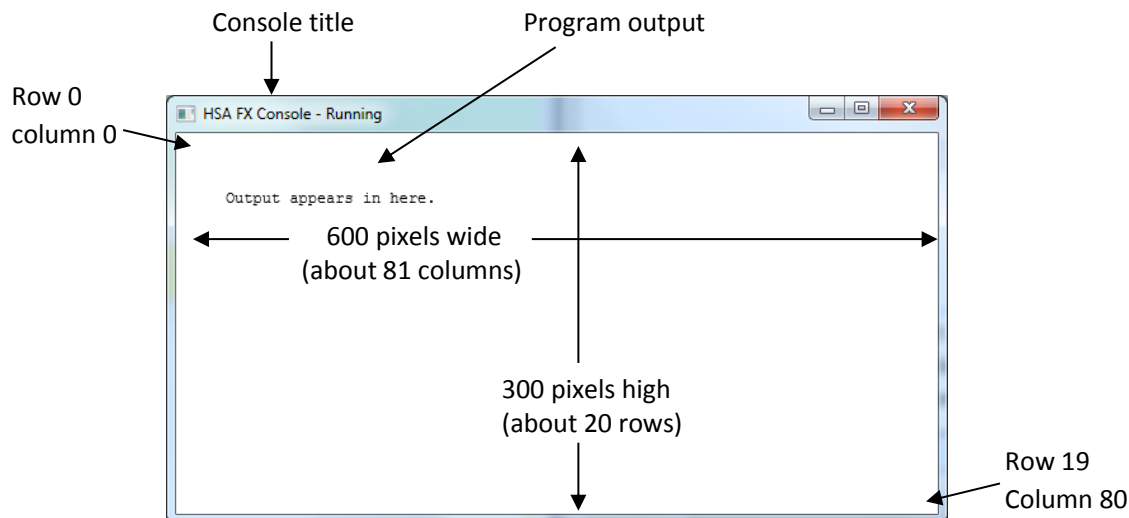


The Ready to Program Console: Simple Text Output

Creating the Console Window

The Console window is the window that pops up when you run a program that uses the **hsafx** package. The console is a Java **object**. The console window contains a text grid that by default holds 20 lines by 81 columns of text. The user can input data directly in the Console window and the output of the program appears there too.



When you are creating a console, you can specify its size, the size of the font it will use, and the title that should appear in the window, like this:

c = new Console (width, height, fontSize, "title", this)

Creates a console named "c" of whatever size you like, depending on what you put for *width*, *height*, *fontSize* (values are in pixels), and title (must be in quotes). You also have to put the word *this* as the last parameter. And don't forget the commas!

Basic Text Output

For basic text output, the console is divided into **rows** and **columns**. The top left corner is row 1 and column 1. When you print something to the screen, where it appears is controlled by an invisible pointer called the **cursor**. The cursor starts in the top left corner (row 1, column 1) and moves to the right and down (increasing columns and rows) depending on how many characters you output.

The row number increases whenever you reach the end of a line, or when you output a **line feed** character. A line feed increases the row number by 1, and sets the column number to 0. When you reach the bottom of the screen, the output starts to scroll.

Text Output Methods

c.clear ()

Clears the entire Console window and sets the cursor to the upper-left corner (*row 1, column 1*).

c.print (something printable)

Prints out whatever you put in the brackets, starting from the current cursor position and moving to the right (and down if you reach the end of the line). Usually, you put a **string** in the brackets, like "hello world" or "12345". You can also print plain numbers without the quotation marks.

c.println()

Outputs a line feed at the current cursor position, which moves the cursor down one line (*row + 1*) and all the way to the left (*column = 0*)

c.println(something printable)

Identical to print method with same arguments, except that a **line feed** is output after the argument. This means that the next time you use c.print or c.println the text will come out on the next line.

c.setFill("color")

Sets the color for text output from *print* and *println* methods. See the separate sheet for a discussion of how to use colors.

c.setBackground("color")

Sets the background color for text output from *print* and *println* methods. See the separate sheet for a discussion of how to use colors.

c.setCursor(row, column)

Sets the cursor position to row *row* and column *col*.

c.close()

Closes the console window.

If you put \n inside a string, it will go down to the next line. If you put \t, it will insert a tab. Try this:

```
c.print("Hello\nWorld.\tbye now");
```

Quick Quiz

What is the difference between these two sequences of commands?

```
c.print("Hello ");
```

```
c.println("Hello ");
```

```
c.print("world. ");
```

```
c.println("world. ");
```

Where will this next line output in each case?

```
c.println("Goodbye.");
```