

- 5) a) In the Programming window now delete the **import** hsa.Console; statement
 - b) Run the program again and write what happened
-

- 6) a) Select OK
 - b) Select from the menu bar Edit and then Undo Typing (Ctrl+Z)
import hsa.Console; should reappear
 - c) Run the program again (there should be no errors)
 - 7) a) change the line **static** Console c; to **static** Console screen;
 - b) change the line c = **new** Console(); to screen = **new** Console();
 - c) change the line c.println("My name is Sue"); to screen.println("My name is Sue");
 - d) Run the program and write what happened
-

- 8) a) Under the menu item File select Close (Ctrl+W)
- b) When asked to Save changes to PrintName.java? select **No**
- c) When asked to Exit "Ready"? select OK

Summary:

println() is a *method* (like a function) that performs some action which is to show something on an output screen. It must be associated (connected) to what Java calls an *object* by a period actually called a *dot operator*. The object is either c or lastly screen. If using our lab book to output you must have the import statement of **import** hsa.Console; . An object can only be used only if we know what *Class* it is from. The class for c (or screen) is the Console class. The Console class is written by the author of our lab book and creates an output window screen designed by him. (More about Classes and objects later.)

Answer the following

- 1) The variable screen or c is called a(n) what? 1) _____
- 2) The word println() is called a(n) what? 2) _____
- 3) The word Console is called a(n) what? 3) _____
- 4) The class that allows us to use screen.println() or c.println() is what? 4) _____

Turn in this sheet to be graded!

Unit 1 Lab 2 - White Space & Import

class hour _____

Name _____
Pts: 10

page 1 of 2

I. White Space

- 1) a) If not logged on to computer go through the first 4 steps in I. from Lab 1
b) Under File from the *Ready to Program* window select Open...
 - 2) Get to your H drive and select PrintName.java (get help if confused)
 - 3) Run the program (there should be no errors)
 - 4) a) In the Programming window delete the last three clauses that start with `//`
and also delete the `//` in each line (Note: do not change what is front of the `//`)
b) Run the program again and write what happened (there should be no errors)
-

- 5) a) Starting with the **import** `java.awt.*;` statement put all lines including braces all in the same (long) line without deleting anything
b) Run the program again and write what happened (there should be no errors)
-

*Java does not care if the program is written in one line, many lines, or indented lines with braces lining up above and below each other. Another way of saying that is that white spaces are ignored by the computer except to separate words. A good programming formats their program to make it easier for a human to read and debug.

- 6) a) Delete the space between class and PrintName in the public etc. line (not the very top line).
b) Run the program again and write what happened
-

- 7) Click OK
- 8) Under the menu item Edit select Undo Typing (Ctrl+Z)
- 9) a) Under the menu item File select Close (Ctrl+W)
b) When asked to Save changes to PrintName.java? select **No**
c) When asked to Exit "Ready"? select OK

II. Import

- 1) a) If not logged on to computer go through the first 4 steps in I. from Lab 1
b) Under File from the *Ready to Program* window select Open...
 - 2) Get to your H drive and select PrintName.java (get help if confused)
 - 3) Run the program (there should be no errors)
 - 4) a) In the Programming window delete the **import** `java.awt.*;` statement
b) Run the program again and write what happened
-

*The import statement **import** `java.awt.*;` is actually not needed in our programs until we start using graphics. (It was just included in the original template in case any graphics were going to be used.)

OVER