**Unit1\_ Exercise\_Output**

**Introduction to Objects/Basic Coding**

**Character Strings**

Every character string is an \_\_\_\_\_\_\_\_\_\_\_\_\_\_ in Java, defined by the \_\_\_\_\_\_\_\_\_\_\_\_\_\_ class**.** Every string literal, delimited by double quotation marks, represents a String object**.** The *string \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ operator* (+) is used to append one string to the end of another.It can also be used to append a number to a string.A string literal \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ be broken across two lines in a program.

System.out.println ("We present the following examples to clarify "

+ "string concatenation:");

System.out.println ();

// A string can contain numeric digits

System.out.println ("Letters in the Hawaiian alphabet: 12");

// A numeric value can be concatenated to a string

System.out.println ("Speed of ketchup: " + 40 + " km per year");

Output:

We present the following examples to clarify string concatenation

Letters in the Hawaiian alphabet: 12

Speed of ketchup: 40 km per year

The plus operator (+) is also used for arithmetic \_\_functions\_\_\_\_. The function that the + operator performs depends on the type of the information on which it operates. If both operands are strings, or if one is a string and one is a number, it performs \_\_String\_\_\_\_concatenation. If both operands are numeric, it \_\_adds\_\_ them. The + operator is evaluated left to right. Parentheses can be used to force the operation order.

System.out.println ("24 and 45 concatenated: " + 24 + 45);

System.out.println ("24 and 45 added: " + (24 + 45));

System.out.println (24 + 45 + " = 24 and 45 added");

Output:

24 and 45 concatenated: 2445

24 and 45 added: 69

69 = 24 and 45 added

**println vs print**

System.out.println prints the character string and returns the \_\_cursor\_\_\_ to the next line. System.out.print prints the character string and leaves the cursor on the \_\_\_\_same\_\_\_\_ line after the last character printed. Write the output from the following statements:

System.out.println (“APCS is fun!”); Output: APCS is fun!

System.out.print (“Three…”); Three…Two…One…

System.out.print (“Two…”); Liftoff!

System.out.print (“One…”);

System.out.println (“Liftoff!”);

**Escape Sequences**

Some characters have special meaning in Java. A double quote represents the beginning or ending of a string of characters. If you want to print out a double quote, you need to use a \_\_\_\\_\_ character (called an escape sequence) to tell the compiler the next character will be treated in a special way.

System.out.println ("I said \"Hello\" to you.");

Output:

I said “Hello” to you

System.out.println ("Roses are red,\n\tViolets are blue,\n" +

"Sugar is sweet,\n\tBut I have \"commitment issues\",\n\t" +

Meaning

newline

double quote

single quote

backslash

tab (not on AP)

Escape Sequence

\n

\"

\'

\\

\t

"So I'd rather just be friends.");

Output:

**Roses are red, Violets are blue, Sugar is sweet,**

**But I have “Commitment issues”, So I’d rather just be friends**

What System.out.println statement would you use to print out the following character sequence?

\\//”\\\///”\\//

System.out.println(\\\\////\”\\\\\\//////\”\\\\////)