Basics of Java Programming
- Strings and Printing

CSC 1051 – Algorithms and Data Structures I

Dr. Mary-Angela Papalaskari
Department of Computing Sciences
Villanova University

Course website:

www.csc.villanova.edu/~map/1051/

Lab 1:

- Learn about jGrasp - the programming environment that we will be using in this class
  - Compile [icon] …. and run [icon] a java program

- Understand the relationship between a Java class name and the name of the .java file where the class is defined

- Practice using basic Java output statements and adding comments

- Learn about variables, string literals, concatenation. *E.g.*,

```
System.out.println ("Howdy " + name);
System.out.println ("The answer is " + x);
System.out.print ("Counting... up: " + (count + 1));
System.out.println ("  ... and\n ... down: " + (count - 1));
```

- Explore Java syntax
- Experience some errors!

Character Strings

- A *string literal* is represented by putting double quotes around the text

- Examples:

```
"This is a string literal."
"123 Main Street"
"X"
```

Character Strings

- A *string literal* is represented by putting double quotes around the text

- Examples:

```
"This is a string literal."
"123 Main Street"
"X"
```

spaces matter in here!

The println Method

- In the `Lincoln` program we invoked the `println` method to print a character string

- The `System.out` object represents a destination (the monitor screen) to which we can send output

```
System.out.println ("Whatever you are, be a good one.");
```

object   method name   information provided to the method (parameters)

The print Method

- In the `Lincoln` program we invoked the `println` method to print a character string

- The `System.out` object represents a destination (the monitor screen) to which we can send output

- `print` is similar to the `println` except that it does not advance to the next line

```
System.out.print ("Whatever you are, be a good one.");
```

object   method name   information provided to the method (parameters)

String Concatenation

- The *string concatenation operator* (+) is used to append one string to the end of another

```
"And one more " + "thing"
```

Hands on:

- Use MyQuote.java as a starting point (program from Lab 1), focus on this part of the code:

```
System.out.println ("Howdy " + name);
System.out.println ("The answer is " + x);
System.out.print ("Counting... up: " + (count + 1));
System.out.println ("  ... and\n ... down: " + (count - 1));
```

- Try the following:
  1) What if you remove the parentheses around (count + 1)?
  2) What happens if we try this way of breaking a line:
```
System.out.print ("Counting...
          up: " + (count + 1));
```
  3) How can we get all this output to print all in one line?

- Other examples (textbook): Countdown.java Facts.java

Escape Sequences

- What if we wanted to print the quote character?

- Let's try something like this…

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

- An *escape sequence* is a series of characters that represents a special character

- An escape sequence begins with a backslash character (\)

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