

Unit Two Lecture Four

Topic 1: The String method length()

In Java, strings are enclosed in quotation marks, which are not themselves part of the string. Note that, unlike numbers, strings are objects. (You can tell that String is a class name because it starts with a capital letter.) The number of characters in a String is called the length of that String.

For example:

```
String a = "Agent ";
String b = "007";

a += b;           // This is called concatenation of strings

System.out.println(a);

int n = a.length();
System.out.println("Length of string = " + n);
```

If this code were executed, the output would be...

```
Agent 007
Length of string = 9
```

In the last line of code, you are actually concatenating two strings. The value of n is automatically cast as a string.

Topic 2: The String method substring(x,y)

This method allows the user to extract part of a string, starting at position x and containing all characters up to but not including position y.

For example:

```
String x = "Computer";
String y = x.substring(3,6);
System.out.println(y);
```

If this code were executed, the output would be...

```
put
```

Remember, the C is in position 0!
Notice, the 6 - 3 would give you the length of String y.

Topic 3: Reading data from text files

Obtaining data from a text file is not as easy in Java as it is in other languages. Remember, Java was created to be used on the internet where most data would be entered from the keyboard. For this reason, reading data from text files will not be tested on the AP exam. However, having the knowledge to read data from a text file will make many of our projects a lot less painful.

In Java, there are a number of ways to retrieve data from text files. The simplest mechanism for reading text is to use the Scanner class, the same class we used for console input.

Suppose I would like to read the first line of text from a file (U2A3.txt) on the Z drive in the JavaData folder and return the substring starting at position 5 and going through position 12. The code would look like this...

```
import java.io.*;
import java.util.Scanner;

public String getString()
{
    Scanner in;

    try    // error handling structure to make sure the file exists - Mandatory!!
    {
        in = new Scanner(new File("Z:\\Students\\WHSPROG\\JavaData\\U2A3.txt"));
    }
    catch (IOException e)
    {
        throw new RuntimeException(e.toString());
    }

    String inputLine = in.nextLine()           // reads first line of file

    String myString = inputLine.substring(5, 13); // determines substring

    return myString;                          // returns substring to calling block
}
```

The error handling structure (catching & throwing exceptions) will be covered later in this course.

Topic #4: When *public static void main(String[] args)* can't be used

The keyword `static` indicates that the main method will not operate on an object. This means that the main method can't be used to call a method from within the driver class and it can't contain global variables (instance fields). If this is the case, have the main method call a default constructor and this default constructor would contain the code you would have placed in main.

Assignment U2A3: The Easter Sunday Problem