

COMP 110-001

String and Console I/O

Yi Hong

May 18, 2015

Announcements

- Labs 0 & 1 are due today by 11:59pm
- Homework 0 grades are posted
- Office hours are fixed: MW 3-4pm, or by appointment

Review of How to Use a Variable

- **Declare** a variable
 - `int age;`
 - A container to hold data
- **Assign** a value to the variable
 - `age = 16;`
 - Puts a value into the container
- **Change** the value of the variable
 - `age = age + 1;`
 - Interpreted from right to left

Review of Last Lecture

- What are the two main data types in Java?
 - Primitive type and class type
- List some of the primitive types
 - int, float, double, char, boolean ...
- Can a **double** variable be assigned to an **int** variable?
- How to do type casting?
 - (Type_Name) Expression

Today

- String
- Console I/O

String

- A Class Type
- Objects of String class can be defined as:
 - `String myString = "UNC is Great!";`
- Each String object consists of
 - A sequence of characters (char)

String	U	N	C		i	s		G	r	e	a	t	!
Indices:	0	1	2	3	4	5	6	7	8	9	10	11	12

- A set of methods that can process the sequence of characters

Some Methods in the Class String

String myString = "UNC is Great!"

U	N	C		i	s		G	r	e	a	t	!
0	1	2	3	4	5	6	7	8	9	10	11	12

```
int strLength = myString.length();
```

```
System.out.println(strLength);
```

13

```
char strFirstLetter = myString.charAt(0);
```

```
System.out.println(strFirstLetter);
```

U

```
String strLowerCase = myString.toLowerCase();
```

```
System.out.println(strLowerCase);
```

unc is great!

```
System.out.println(myString.substring(0, 3));
```

UNC

String

- Check Java API for the whole list of methods in String
- You do not need to memorize them
- But you should know how to use the classes if you have related documents

More methods in the Class String, see p.86

You will learn how to use these methods in Labs on Wednesday

Escape Characters

- How to put quotes in a string

```
System.out.println("How do I put \"quotes \" in my string?");
```

- But what about backslashes?

```
System.out.println("How do I put \\ in my string?");
```

\"	Double quote
\'	Single quote
\\	Backslash
\n	New line
\r	Carriage return
\t	Tab

More Examples

- “abc\\def”
 - abc\def
- “The motto is \nGo for it!”
 - The motto is
Go for it!
- “How’s this”
 - How’s this
 - `char singleQuote = ‘\’;`

String Concatenation

- String name = "May";
- String sentence;
- sentence = "My dog's name is " + name;

My dog's name is May

The Empty String

- A string can have any number of characters, including zero
- The string with zero characters is called the empty string
- The empty string is useful and can be created in many ways including
 - `String str = "";`

Try It Yourself

- Run code in Eclipse
 - See `StringsAndChars.java`, `TestStringMethods.java` on the course website for more details

Console I/O:

Screen Output and Keyboard Input

Screen Output

- We've seen several examples of screen output already
- `System.out.println("This is a string");`
 - `System.out` is an object built in Java SDK
 - `println` is one of the methods available to the `System.out` object

Print Versus Println

```
System.out.print("This is a string");
```

```
System.out.println("This is a string");
```

- What is the difference?

```
System.out.print("This prints ");
```

```
System.out.print("multiple parts ");
```

```
System.out.println("in one line");
```


Screen Output

- Use concatenation operator (+) to join the things you want to display

```
System.out.println("Lucky number = " + 13 +  
    "Secret number = " + number);
```

- Note: no spaces are added. If you want to add a space, add it into the string

```
"Secret number =" → "Secret number = "
```

Keyboard Input

- Java 5.0 ~ 7.0 have reasonable facilities for handling keyboard input
- These facilities are provided by the **Scanner** class in the **java.util** package
 - *A package is a library of classes*

How to Use the Scanner Class

- Include at the beginning of a program
`import java.util.Scanner;`
- Create an object of the Scanner class
`Scanner Scanner_object_name = new Scanner(System.in);`
- Read data using methods of the object
`Scanner_object_name.next();` → String
`Scanner_object_name.nextLine();` → String
`Scanner_object_name.nextInt();` → int
`Scanner_object_name.nextDouble();` → double
`Scanner_object_name.nextBoolean();` → boolean

More About `nextLine()`

- The `nextLine()` method reads the remainder of the current line, even if it is empty
 - E.g.: `int n = keyboard.nextInt();`
`String s1 = keyboard.nextLine();`
`String s2 = keyboard.nextLine();`

Input: 13 is the lucky number,
and which one is
the secret number?
- Make sure to read **GOTCHA** on p.97

Next Class

- Review of Chapter 1 & 2
- Review programs in lectures and labs