COMP I 10/L Lecture 17

Kyle Dewey

Outline

- char type
- Methods on String
 - length
 - charAt
- Two-dimensional arrays

char Type

`a'

(Single quotes denote char)

`a′

(Single quotes denote char)

`b'

`a'

(Single quotes denote char)

`b'

`\n'

-The letter b

`a'

(Single quotes denote char)

`b'

`\n' (newline)

-Newline character (still a single character)

char Type Represents a single character **`**a′ (Single quotes denote char) **`**b' **`**\n' (newline) **`**\t'

char Type Represents a single character **`**a' (Single quotes denote char) **`**b' **`**\n' (newline) **`**\t′ (tab)

-Again, still a single character

Works just like it does with int

Works just like it does with int

'h' + "ello"

Works just like it does with int

Works just like it does with int

"goodby" + 'e'

Works just like it does with int

`h' + "ello"
 "hello"

"goodby" + 'e' "goodbye"

Methods on String

Returns the number of characters in the String

Returns the number of characters in the String

"hello".length()

Returns the number of characters in the String

"hello".length()

5

Returns the number of characters in the String



5

"goodbye".length()

Returns the number of characters in the String



"goodbye".length()

5

Returns the number of characters in the String





Returns the number of characters in the String



Example: StringLength.java

length() vs.length

• length() is defined for Strings

• "foo".length()

- length is defined for arrays
 - (new int[]{1, 2, 3}).length

charAt()

Gets the char at a given 0-indexed position in a String

Gets the char at a given 0-indexed position in a String

"something".charAt(1)

Gets the char at a given 0-indexed position in a String

"something".charAt(1)

`○′

Gets the char at a given 0-indexed position in a String

"something".charAt(1)

`○**′**

"what".charAt(3)

Gets the char at a given 0-indexed position in a String

"something".charAt(1)

`○**′**

"what".charAt(3) 't'

Gets the char at a given 0-indexed position in a String

`○′

"kangaroo".charAt(5)

Gets the char at a given 0-indexed position in a String

`○′

"kangaroo".charAt(5)

'r'

Example: StringCharAt.java

Two-Dimensional Arrays

Recap:Arrays

Recap:Arrays

new int[]{1, 2, 3}

Recap:Arrays

new int[]{1, 2, 3} new String[]{"foo", "bar"}

Recap:Arrays

new int[]{1, 2, 3}

new String[]{"foo", "bar"}

public static int[] baz(int[] input) {

Arrays of Arrays ...but you can also make arrays of arrays.

AKA multi-dimensional arrays.

-Up until this point, you've only been working with single-dimensional arrays

Arrays of Arrays ...but you can also make arrays of arrays. AKA multi-dimensional arrays.

-Up until this point, you've only been working with single-dimensional arrays

Arrays of Arrays ...but you can also make arrays of arrays. AKA multi-dimensional arrays.

public static void blah(String[][] in) {

-Up until this point, you've only been working with single-dimensional arrays

Example Use Case: Spreadsheets

×.	<u>F</u> ile	<u>E</u> dit	<u>¥</u> iew	<u>I</u> nsert	<u>D</u> raw	F <u>o</u> rn	nat	<u>T</u> ools	D <u>a</u> ta	Data <u>b</u> ase	: <u>W</u>	/indow	<u>H</u> elp		
📄 New Workbook 🔲 Table_Mapping.vts															
			Q - 2	x 🗅 🕻		ro 🖌	<u>r</u> 2	ΒI	<u>U</u>					* 🛃 👔	5
Ŧ	Arial		✓ 10	🗐 🚱 🖫	3 1/4 9/	6,00 123	3 G E+	ο -Σ -	+Σ • <i>f</i> *) 🗨 Q		🦻 - 🖉	/ 🗌	05	C C
G16															
	Year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	0 ct	Nov	Dec	Annual	0
1	1990	127.4	128.0	128.7	128.9	129.2	129.9	130.4	131.6	132.7	133.5	133.8	133.8	130.7	
2	1991	134.6	134.8	135.0	135.2	135.6	136.0	136.2	136.6	137.2	137.4	137.8	137.9	136.2	
3	1992	138.1	138.6	139.3	139.5	139.7	140.2	140.5	140.9	141.3	141.8	142.0	141.9	140.3	
4	1993	142.6	143.1	143.6	144.0	144.2	144.4	144.4	144.8	145.1	145.7	145.8	145.8	144.5	
5	1994	146.2	146.7	147.2	147.4	147.5	148.0	148.4	149.0	149.4	149.5	149.7	149.7	148.2	
6	1995	150.3	150.9	151.4	151.9	152.2	152.5	152.5	152.9	153.2	153.7	153.6	153.5	152.4	
7	1996	154.4	154.9	155.7	156.3	156.6	156.7	157.0	157.3	157.8	158.3	158.6	158.6	156.9	
8	1997	159.1	159.6	160.0	160.2	160.1	160.3	160.5	160.8	161.2	161.6	161.5	161.3	160.5	
9	1998	161.6	161.9	162.2	162.5	162.8	163.0	163.2	163.4	163.6	164.0	164.0	163.9	163.0	
10	1999	164.3	164.5	165.0	166.2	166.2	166.2	166.7	167.1	167.9	168.2	168.3	168.3	166.6	
11	2000	168.8	169.8	171.2	171.3	171.5	172.4	172.8	172.8	173.7	174.0	174.1	174.0	172.2	
12	2001	175.1	175.8	176.2	176.9	177.7	178.0	177.5	177.5	178.3	177.7	177.4	176.7	177.1	
13	2002	177.1	177.8	178.8	179.8	179.8	179.9	180.1	180.7	181.0	181.3	181.3	180.9	179.9	
14															
15															
16															
17															
18															

Example: PrintAll2D.java

Example: SetAllToCount.java

Example: Create2DArray.java

Important Points

- Accessing one element: array[row] [column]
- Inner arrays can be of different lengths
- Edits to arrays persist across method calls
 - This is different for non-arrays, which copy
 - More on this difference later in the course