

Strings

	1st	2nd	3rd	4th			4th to Last	3rd to Last	2nd to Last	Last	END	<pre>str.substring(0,3) → "abc" str.substring(1,3) → "bc"</pre>
String str	a	b	c	d	.	.	w	x	y	z		<pre>int len = str.length(); str.substring(len-2) → "yz" str.substring(len-4, len-1) → "wxy"</pre>
Index (position)	0	1	2	3	.	.	len - 4	len - 3	len - 2	len - 1	str.length()	

EVEN	1st	2nd	3rd	4th	5th	6th	7th	8th	END	<pre>int mid = str.length()/2; for example 8/2 = 4</pre>
String str	a	b	c	d	e	f	g	h	str.length()	<p>For even-length strings, there is NO MIDDLE CHARACTER. The character at index mid will be the first character in the right-half of the string.</p>
Index (position)	0	1	2	3 mid-1	4 mid	5 mid+1	6	7		<p>str.substring(mid-1, mid+1) gives the middle 2 characters.</p>

ODD	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	END	<pre>int mid = str.length()/2; e.g. 9/2 = 4 (integer division)</pre>
String str	a	b	c	d	e	f	g	h	i	str.length()	<p>For odd-length strings, mid contains the value of the index of the middle character.</p>
Index (position)	0	1	2	3 mid-1	4 mid	5 mid+1	6 mid+2	7	8		<p>str.substring(mid-1, mid+2) gives the middle 3 characters.</p>

String Methods

String objects call these methods using **DOT** (.) syntax, e.g.:
String + **DOT** + **Method**: **word.length()** **“abcde”.substring(0,1)**

Method	Input Parameters	Description	Return Type
length()	No Parameters	Returns the number of characters in the String.	int (NUMBER)
substring (int <i>startIdx</i> , int <i>stopIdx</i>)	<i>startIdx</i> : Index of the 1 st character in the new string <i>endIdx</i> : Stopping index; the character at this index is NOT included; that is, the last character in the new string is at index <i>endIdx-1</i> .	Returns a new string containing (<i>stopIdx-startIdx</i>) # of characters. The 1 st character of the new String is the character at position <i>stopIdx</i> . Characters are added up to, but not including the character at index <i>stopIdx</i> .	String (TEXT)
substring (<i>startIdx</i>)	<i>startIdx</i> : Index of the 1 st character in the new string	Returns a new string containing (<i>stopIdxlength()+1</i>) # of characters. The new String includes all characters in the original string starting with the character at <i>startIdx</i> and continuing to the end.	String (TEXT)
equals (String <i>strTest</i>)	<i>strTest</i> : The String used to compare the String object to.	Returns true if strTest matches the object String “abc” matches “abc” “abc” does NOT match “Abc”	Boolean (true/false)
equalsIgnoreCase (String <i>strTest</i>)	<i>strTest</i> : The String used to compare the String object to. Matches strings regardless of uppercase or lowercase letters	Returns true if strTest matches the object String regardless of uppercase or lowercase, i.e. treats uppercase and lowercase letters as if they are the same. “abc” matches “abc” “abc” matches “ABC”	Boolean (true/false)
endsWith (String <i>suffix</i>)	<i>suffix</i> : The String you want to match at the end of the String object.	Returns true if the String object ends with suffix .	Boolean (true/false)
startsWith (String <i>prefix</i>)	<i>prefix</i> : Does the object String start with prefix ?	Returns true if the String object begins with prefix .	Boolean (true/false)
startsWith (String <i>prefix</i> , int <i>offsetIdx</i>)	<i>prefix</i> : Imagine that this creates a substring starting at position <i>offsetIdx</i> . Does that substring start with prefix ?	Returns true if the imagined substring begins with prefix .	Boolean (true/false)
isEmpty ()	No Parameters	Returns true if the String object has no characters. This is the same thing as str.length() == 0 ;	Boolean (true/false)