



String

Strings are immutable in Java

String str = "Hello World";

Syntax	Method	Use-case
String str = "Hello"; System.out.println(str.length());	length()	Returns the number of characters in the string
String str = "World"; System.out.println(str.charAt(1)); // 'o'	charAt(int index)	Access character at a specific position
String str = "Programmer"; System.out.println(str.substring(0, 7)); // "Program"	substring(int start, int end)	Extracts part of the string
String str = "I love Java"; System.out.println(str.contains("Java")); // true	contains(CharSequence s)	Checks if a sequence is present in the string
String str = "Banana"; System.out.println(str.indexOf("a")); // 1	indexOf(String s)	Finds the index of a character or substring
String str = "Hello"; System.out.println(str.equals("hello")); // false	equals("Hello");	Compares content (case-sensitive)
String str = "Hello"; System.out.println(str.equalsIgnoreCase("HELLO")); // true	equalsIgnoreCase()	Case-insensitive comparison
String str = "JAVA"; System.out.println(str.toLowerCase()); // "java"	toLowerCase()	Converts entire string to lowercase
String str = "java"; System.out.println(str.toUpperCase()); // "JAVA"	toUpperCase()	Converts entire string to uppercase
String str = " Hello "; System.out.println(str.trim()); // "Hello"	trim()	Removes leading and trailing spaces
String str = "Banana"; System.out.println(str.replace("a", "o")); // "Bonono"	replace(char old, char new)	Replace characters or substrings
String str = "Learn Java Fast"; String[] words = str.split(" "); System.out.println(words[0]);	str.split(" ")	Splits string into array based on delimiter
String str = ""; System.out.println(str.isEmpty()); // true	isEmpty()	Checks if string is empty (length == 0)
String str = " "; System.out.println(str.isBlank()); // true	isBlank()	Checks if string is empty or only whitespace
String str = "Hello World"; System.out.println(str.startsWith("He")); // true	startsWith(String prefix)	Checks if string begins with a prefix
String str = "Hello World"; System.out.println(str.endsWith("ld")); // true	endsWith(String suffix)	Checks if string ends with a suffix

String Builder

It's mutable (faster for modification)

StringBuilder str = new StringBuilder ("Hello World");

Common methods in both

charAt(), substring(), length(), indexOf(), lastIndexOf(),
toString(), equals()

Syntax	Method	Use-case
StringBuilder sb = new StringBuilder("Hello "); sb.append("World"); System.out.println(sb); // Hello World	append()	Adds text at the end of the current StringBuilder
StringBuilder sb = new StringBuilder("Hello World"); sb.insert(6, "Java "); System.out.println(sb); // Hello Java World	insert()	Inserts text at a specific position
StringBuilder sb = new StringBuilder("Hello World"); sb.replace(0, 5, "Hi"); System.out.println(sb); // Hi World	replace()	Replaces a part of the string with new text
StringBuilder sb = new StringBuilder("Hello World"); sb.delete(5, 11); System.out.println(sb); // Hello	delete()	Deletes a portion of the string
StringBuilder sb = new StringBuilder("Hello!"); sb.deleteCharAt(5); System.out.println(sb); // Hello	deleteCharAt(int index)	Deletes a character at a specific index
StringBuilder sb = new StringBuilder("Java"); sb.reverse(); System.out.println(sb); // avaJ	reverse()	Reverses the character sequence
StringBuilder sb = new StringBuilder("Java"); sb.setCharAt(0, 'K'); System.out.println(sb); // Kava	setCharAt()	Replaces a single character at a given index
StringBuilder sb = new StringBuilder(); System.out.println(sb.capacity()); // 16 (default)	capacity()	Returns current capacity (not length) of the buffer
StringBuilder sb = new StringBuilder(); sb.ensureCapacity(50); System.out.println(sb.capacity()); // ≥50	ensureCapacity()	Ensures minimum capacity to avoid frequent resizing

Important Differences

Modifying methods like append, insert, replace, delete exist only in StringBuilder, not in String.

String uses methods like concat(), but StringBuilder uses append() instead.

equals() behaves differently:

String: compares content.

StringBuilder: compares object reference (not content), so use sb.toString().equals("text").

