

LẬP TRÌNH JAVA

Bài 6: Xử lý chuỗi (String)

Khái niệm

- ❑ Chuỗi là một dãy ký tự liên tục, String là một đối tượng mà biểu diễn một dãy ký tự liên tục. Lớp String được sử dụng để tạo đối tượng String.
- ❑ Chuỗi (String) trong Java cung cấp nhiều khái niệm đa dạng giúp bạn thao tác và xử lý với chuỗi như so sánh, cắt, nối, tìm độ dài, thay thế, tìm chuỗi con.
- ❑ Chuỗi là một đối tượng mà biểu diễn dãy các giá trị char.
- ❑ Một mảng các ký tự làm việc khá giống như chuỗi trong Java
`char[] ch={'j', 'a', 'v', 'a', 'c', 'o', 'b', 'a', 'n'};`
`String s=new String(ch);`
Tương tự như: `String s = "javacoban";`

Cách tạo đối tượng String

❑ String Literal

```
String s = "helloworld";
```

❑ Sử dụng từ khóa new

```
String s = new String("helloworld")
```

Các phương thức của String

- ❑ **String** cung cấp rất nhiều các phương thức để thực hiện các thao tác với chuỗi như: `compare()`, `concat()`, `equals()`, `split()`, `length()`, `replace()`, `compareTo()`, `intern()`, `substring()`, ...

Các phương thức của String

No.	Phương thức	Mô tả
1	char charAt(int index)	Trả về giá trị char cho chỉ số cụ thể.
2	int length()	Trả về độ dài chuỗi.
3	static String format(String format, Object... args)	Trả về chuỗi được format.
4	static String format(Locale l, String format, Object... args)	Trả về chuỗi được format theo vùng miền(Quốc gia).
5	String substring(int beginIndex)	Trả về chuỗi con bắt đầu từ chỉ số index.
6	String substring(int beginIndex, int endIndex)	Trả về chuỗi con từ chỉ số bắt đầu đến chỉ số kết thúc.
7	boolean contains(CharSequence s)	Kiểm tra chuỗi chứa chuỗi không, kết quả trả về là giá trị boolean
8	static String join(CharSequence delimiter, CharSequence... elements)	Trả về chuỗi được nối từ nhiều chuỗi.
9	static String join(CharSequence delimiter, Iterable<? extends CharSequence> elements)	Trả về chuỗi được nối từ nhiều chuỗi.
10	boolean equals(Object another)	Kiểm tra sự bằng nhau của chuỗi với đối tượng.

No.	Phương thức	Mô tả
11	boolean isEmpty()	Kiểm tra chuỗi rỗng.
12	String concat(String str)	Nối chuỗi cụ thể.
13	String replace(char old, char new)	Thay thế tất cả giá trị char cụ thể bằng một giá trị char mới.
14	String replace(CharSequence old, CharSequence new)	Thay thế tất cả các chuỗi bằng một chuỗi mới.
15	static String equalsIgnoreCase(String another)	So sánh chuỗi, không phân biệt chữ hoa hay chữ thường.
16	String[] split(String regex)	Trả về mảng các chuỗi được tách ra theo giá trị regex.
17	String[] split(String regex, int limit)	Trả về mảng các chuỗi được tách ra theo giá trị regex và có giới hạn.
18	String intern()	Trả về chuỗi interned.
19	int indexOf(int ch)	Trả về vị trí của ký tự ch cụ thể.
20	int indexOf(int ch, int fromIndex)	Trả về vị trí của ký tự ch tính từ từ vị trí fromIndex.

No.	Phương thức	Mô tả
21	int indexOf(String substring)	Trả về vị trí của chuỗi con substring.
22	int indexOf(String substring, int fromIndex)	Trả về chuỗi con bắt đầu từ vị trí substring đến vị trí fromIndex.
23	String toLowerCase()	Trả về chuỗi chữ thường.
24	String toLowerCase(Locale l)	Trả về chuỗi chữ thường bằng việc sử dụng locale cụ thể.
25	String toUpperCase()	Trả về chuỗi chữ hoa.
26	String toUpperCase(Locale l)	Trả về chuỗi chữ hoa bằng việc sử dụng locale cụ thể.
27	String trim()	Xóa khoảng trắng ở đầu và cuối của chuỗi.
28	static String valueOf(int value)	Chuyển đổi giá trị kiểu dữ liệu đã cho thành chuỗi.

Bài tập

1. Nhập một chuỗi ký tự. Đếm số từ của chuỗi ký tự đó. Ví dụ " Truong hoc " có 2 từ.
2. Nhập một chuỗi ký tự. Đếm số lần xuất hiện của từng từ trong chuỗi đó. Ví dụ: “ngon ngu lap trinh java la ngon ngu huong doi tuong” => ngon: 2, ngu: 2, lap: 1, trinh: 1, java: 1, la: 1, huong: 1, doi: 1, tuong: 1.
3. Viết chương trình thực hiện nhập một chuỗi họ tên theo cấu trúc: họ...đệm...tên; chuyển chuỗi đó sang biểu diễn theo cấu trúc tên...họ...đệm.
4. Nhập 2 chuỗi ký tự có số ký tự bằng nhau, trộn lẫn 2 chuỗi ký tự đó vào nhau. Ví dụ: “ngon ngu lap trinh” và “lap trinh doi tuong” => “ngon lap ngu trinh lap doi trinh tuong”