**السلاسل النصية :**

**تستخدم السلاسل النصية لتخزين النصوص.**

**السلسلة النصية تحوي مجموعة من المحارف محاطة بعلامتي اقتباس**

**يدعى رقم الخانة index حيث أول حرف في السلسلة يكون index=0 والحرف الثاني يكون index=1 وهكذا حتى أخر حرف في السلسلة**

**مثال:**

String greeting = "Hello";

**يمكن استخدام المعامل + بين السلاسل لدمج سلسلتين**

**مثال:**

String firstName = "John";

String lastName = "Doe";

System.out.println(firstName + " " + lastName);

**يمكن أيضا استخدام** concat() :

String firstName = "John ";

String lastName = "Doe";

System.out.println(firstName.concat(lastName));

**السلسلة النصية هي عبارة عن صف جاهز**

**في جافا يحوي مجموعة من الدوال التي يمكن استدعاءها لتنفيذ مجموعة من العمليات على السلسلة النصية**

**يمكن الحصول على طول السلسلة باستخدام الطريقة الاتية:**

String txt = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";

System.out.println("The length of the txt string is: " + txt.length())

**يوجد عدة دوال متاحة مثال على ذلك :**

**دالة ترجع السلسلة النصية التي استدعتها بحروف صغيرة او كبيرة**

String txt = "Hello World";

System.out.println(txt.toUpperCase()); // Outputs "HELLO WORLD"

System.out.println(txt.toLowerCase()); // Outputs "hello world"

**دالة للبحث عن محرف معين في سلسلة نصية** indexOf()**:**

**ترجع هذه السلسلة أول index للحرف الذي نبحث عنه ابتداء**

**من أول السلسلة.**

**مثال:**

String txt = "Please locate where 'locate' occurs!";

System.out.println(txt.indexOf("locate")); // Outputs 7

**بما ان السلسلة النصية يجب أن توضع ضمن علامتي اقتباس لن يتمكن المترجم من فهم هذا وسيصدر خطأ:**

String txt = "We are the so-called "Vikings" from the north.";

**ولحل هذه المشكلة نستخدمbackslash escape character**.

|  |  |  |
| --- | --- | --- |
| **Description** | **Result** | **Escape character** |
| Single quote | ' | \' |
| Double quote | **"** | \" |
| Backslash | \ | \\ |

**تقوم**\"**بإضافة علامة اقتباس مزدوجة إلى السلسلة النصية**

مثال :

String txt = "We are the so-called \"Vikings\" from the north.";

[\\out](file:///\\out) put We are the so-called "Vikings" from the north.

**تقوم \' بإضافة علامة اقتباس مفردة إلى السلسلة النصية** :

مثال :

String txt = "It\'s alright.";

[\\out](file:///\\out) put it's alright .

**تقوم \\ بإضافة \ إلى السلسلة النصية :**

String txt = "The character \\ is called backslash.";

[\\out](file:///\\out) put The character \ is called backslash.

**بعض محارف الهروب في جافا :**

|  |  |
| --- | --- |
| **Result** | **Code** |
| New Line | \n |
| Carriage Return | \r |
| Tab | \t |
| Backspace | \b |
| Form Feed | \f |

**بعض الدوال المستخدمة :**

|  |  |  |
| --- | --- | --- |
| **اسم الدالة** | **الشكل** | **الشرح** |
| **charAt** | **public char charAt(int index)** | ترجع الحرف الموجود على الـ **index** الذي نمرره لها مكان الباراميتر index في الـ String الذي قام باستدعائها. |
| **lastIndexOf** | **public int lastIndexOf(String str)** | تبحث في الـ String الذي قام باستدعائها عن آخر **index** يوجد ابتداء من عنده نفس النص الذي نمرره لها مكان الباراميتر str و ترجعه. |
| **contains** | **public boolean contains(CharSequence cs)** | نرجع true في حال كان الـ String الذي قام باستدعائها يحتوي على نفس قيمة النص الذي نمرره لها مكان الباراميتر cs ككائن من الكلاس Char Sequence. غير ذلك ترجع false. |
| **substring** | **public public String substring(int startIndex, int endIndex)** | ترجع String عبارة عن جزء (**substring**) من الـ String الذي قام باستدعائها. |
| **replace** | **public String replace(char oldSequence, char newSequence)** | عند استدعائها نمرر لها قيمتين عبارة عن char أو CharSequence. تبحث في الـ String الذي قام باستدعائها عن القيمة الأولى التي نمررها لها و تبدلها بالقيمة الثانية التي نمررها لها. |
| **equalsIgnoreCase** | **public boolean equalsIgnoreCase(String str)** | تقارن قيمة الـ String الذي قام باستدعائها مع قيمة الـ String الذي نمرره لها مكان الباراميتر str. ترجع true في حال كانت جميع أحرفهم متطابقة و لا يهمها إذا كانت الأحرف كبير أو صغيرة, غير ذلك ترجع false. |
| **compareTo** | **public int compareTo(String anotherString)** | تقارن قيمة الـ String الذي قام باستدعائها مع قيمة الـ String الذي نمرره لها مكان الباراميتر anotherString. ترجع **0** في حال كانت جميع أحرفهم متساوية. و ترجع قيمة أكبر أو أصغر من **0** تمثل الفارق بين كود الـ **ASCII** الخاص بأول حرف مختلف تم إيجاده بينهما. |

**اختبر نفسك :**

اكمل الجزء المفقود واسند له القيمة hello

\_\_\_\_\_\_\_ greeting = \_\_\_\_\_\_\_;