

Cours Java Mardi

Exercice

Méthode Main est le point d'entrée du programme

Système input

```
Scanner sc = Scanner(System.in); //input
```

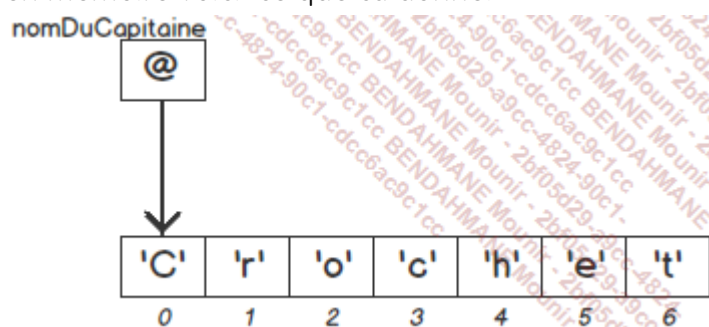
next Double attend un nombre à virgule

Les Chaînes de caractères

Les chaînes de caractères sont des objets en Java, elles sont donc soumises aux règles de la programmation orientée objet.

```
String chaine = "une chaîne de caractères";
```

en mémoire voici ce que ça donne:



En général on a tendance à considérer les String comme des types primitifs, mais ce n'est pas le cas. En réalité, lorsqu'on crée une string on crée une instance de la classe `String` qui contient un tableau de char.

```
String chaine = new String("une chaîne de caractères");
```

Après sa création, une chaîne de caractères ne peut plus être modifiée. On dit qu'elle est `immutable`.

Après sa création, une chaîne de caractères ne peut plus être modifiée. On dit qu'elle est `immutable`.

La classe String nous met à disposition de nombreux outils pour manipuler les chaînes de caractères.

```
String chaine = "une chaine de caracteres";
chaine.length();
// retourne la taille de la chaine chaine.charAt(0);
// retourne le caractere a l'index 0 chaine.substring(0, 4);
// retourne la sous chaine de caracteres de l'index 0 a 4 chaine.indexOf("chaine");
// retourne l'index de la premiere occurrence de "chaine" chaine.replace("chaine",
"string");
// remplace la premiere occurrence de "chaine" par "string" chaine.toUpperCase();
// retourne la chaine en majuscule chaine.toLowerCase(); // retourne la chaine en
minuscule chaine.trim();
// supprime les espaces en debut et fin de chaine chaine.split(" ");
// retourne un tableau de String en decoupant la chaine a chaque espace
chaine.equals("une chaine de caracteres");
// retourne true si les deux chaines sont identiques chaine.equalsIgnoreCase("UNE
CHAINE DE CARACTERES");
// retourne true si les deux chaines sont identiques en ignorant la casse
chaine.startsWith("une");
// retourne true si la chaine commence par "une" chaine.endsWith("res");
// retourne true si la chaine se termine par "res"
```

Comparer les chaînes de caractères

En java il est impossible de comparer deux chaines de caracteres avec l'opérateur ==

```
String chaine1 = "une chaine de caracteres";

String chaine2 = "une chaine de caracteres";

chaine1 == chaine2; // retourne false
```

Pour comparer deux chaînes de caractères il faut utiliser la méthode `equals`

```
String chaine1 = "une chaine de caracteres"; String chaine2 = "une chaine de
caracteres"; chaine1.equals(chaine2); // retourne true
```

Cette fonction compare les chaînes de caractères caractère par caractère.

La méthode `format` permet de formater une chaîne de caractères en utilisant des marqueurs de position.

```
boolean estVrai = true; int entier = 10;
double decimal = 10.5;
String chaine = "une chaine de caracteres";
```

```
System.out.println("un booleen : %b, un entier : %d, un decimal : %f, une chaine : %s",  
estVrai, entier, decimal, chaine);
```

ou encore

```
boolean b=true;
```

```
String s="chaine"; int i=56; double d=5.5;
```

```
System.out.println(String.format("boolean : %b %n" + "chaine de caractères : %s %n"  
+ "entier : %d %n" + "entier en hexadécimal : %x %n" + "entier en octal : %o %n" +  
"décimal : %f %n" + "décimal précis au dixième : %.1f %n" + "décimal au format  
scientifique : %e %n", b,s,i,i,i,d,d,d));
```