

# Les chaînes de caractères

**Une chaîne de caractères (de type String) est une suite ordonnée de caractères (de type char): Exemple:**

"abcdefghijklmnoprstuvwxyz" est une chaîne de caractères (entourée de guillemets "...")

**Rappel:**

- Un caractère est stocké dans une variable de type char.

**Exemple:** char car='e'; (entouré de '...') - les caractères sont classés par le code ASCII

Déclaration d'une chaîne de caractères

**String nom variable;**

Déclaration

```
String chaine;
```

## La table ASCII

|    |    |    |    |    |     |     |     |     |     |     |     |       |     |    |
|----|----|----|----|----|-----|-----|-----|-----|-----|-----|-----|-------|-----|----|
| 0  | 32 | 64 | 8  | 96 | '   | 128 | ç   | 160 | á   | 192 | l   | 224   | ó   |    |
| 1  | 33 | !  | 65 | â  | 97  | a   | 129 | ü   | 161 | í   | 193 | ł     | 225 | b  |
| 2  | 34 | "  | 66 | B  | 98  | b   | 130 | é   | 162 | ó   | 194 | T     | 226 | ô  |
| 3  | 35 | #  | 67 | C  | 99  | c   | 131 | â   | 163 | ú   | 195 | t     | 227 | ò  |
| 4  | 36 | \$ | 68 | D  | 100 | d   | 132 | ä   | 164 | ñ   | 196 | -     | 228 | õ  |
| 5  | 37 | %  | 69 | E  | 101 | e   | 133 | à   | 165 | N   | 197 | +/-   | 229 | õ  |
| 6  | 38 | &  | 70 | F  | 102 | f   | 134 | å   | 166 | ¤   | 198 | ã     | 230 | p  |
| 7  | 39 | ,  | 71 | G  | 103 | g   | 135 | ç   | 167 | ¤   | 199 | Ã     | 231 | p  |
| 8  | 40 | (  | 72 | H  | 104 | h   | 136 | ê   | 168 | ¸   | 200 | ll    | 232 | þ  |
| 9  | 41 | )  | 73 | I  | 105 | i   | 137 | ë   | 169 | ¤   | 201 | II    | 233 | ú  |
| 10 | 42 | *  | 74 | J  | 106 | j   | 138 | è   | 170 | ¬   | 202 | ll    | 234 | û  |
| 11 | 43 | +  | 75 | K  | 107 | k   | 139 | ï   | 171 | ½   | 203 | ¶     | 235 | ù  |
| 12 | 44 | ,  | 76 | L  | 108 | l   | 140 | î   | 172 | ¾   | 204 |       | 236 | ý  |
| 13 | 45 | -  | 77 | M  | 109 | m   | 141 | ì   | 173 | •   | 205 | =     | 237 | ÿ  |
| 14 | 46 | .  | 78 | N  | 110 | n   | 142 | ‰   | 174 | «   | 206 |       | 238 | -  |
| 15 | 47 | /  | 79 | O  | 111 | o   | 143 | ã   | 175 | »   | 207 | ¤     | 239 | ·  |
| 16 | 48 | 0  | 80 | P  | 112 | p   | 144 | É   | 176 | ••• | 208 | δ     | 240 | -  |
| 17 | 49 | 1  | 81 | Q  | 113 | q   | 145 | æ   | 177 | ••• | 209 | ð     | 241 | ±  |
| 18 | 50 | 2  | 82 | R  | 114 | r   | 146 | æ   | 178 | ••• | 210 | Ê     | 242 | =  |
| 19 | 51 | 3  | 83 | S  | 115 | s   | 147 | ô   | 179 | —   | 211 | É     | 243 | %  |
| 20 | 52 | 4  | 84 | T  | 116 | t   | 148 | ö   | 180 | —   | 212 | È     | 244 | ¶  |
| 21 | 53 | 5  | 85 | U  | 117 | u   | 149 | ò   | 181 | Á   | 213 | I     | 245 | §  |
| 22 | 54 | 6  | 86 | V  | 118 | v   | 150 | û   | 182 | Â   | 214 | Í     | 246 | ÷  |
| 23 | 55 | 7  | 87 | W  | 119 | w   | 151 | ù   | 183 | À   | 215 | Î     | 247 | -  |
| 24 | 56 | 8  | 88 | X  | 120 | x   | 152 | ÿ   | 184 | ¤   | 216 | Ï     | 248 | •  |
| 25 | 57 | 9  | 89 | Y  | 121 | y   | 153 | ö   | 185 | —   | 217 | J     | 249 | .. |
| 26 | 58 | :  | 90 | Z  | 122 | z   | 154 | ü   | 186 | —   | 218 | Gamma | 250 | .  |
| 27 | 59 | ;  | 91 | [  | 123 | {   | 155 | ø   | 187 | —   | 219 | ■     | 251 | I  |
| 28 | 60 | <  | 92 | \  | 124 | I   | 156 | £   | 188 | —   | 220 | ■     | 252 | z  |
| 29 | 61 | =  | 93 | ]  | 125 | }   | 157 | Ø   | 189 | ¢   | 221 | —     | 253 | z  |
| 30 | 62 | >  | 94 | ^  | 126 | ~   | 158 | ×   | 190 | ¥   | 222 | —     | 254 | ■  |
| 31 | 63 | ?  | 95 | _  | 127 | Δ   | 159 | f   | 191 | —   | 223 | ■     | 255 | ■  |



# Les chaînes de caractères

**Une chaîne de caractères (de type String) est une suite ordonnée de caractères (de type char): Exemple:**

"abcdefghijklmnopqrstuvwxyz" est une chaîne de caractères (entourée de guillemets "...")

## Déclaration d'une chaîne de caractères

**String nom variable;**

## Déclaration

```
String chaine;
```

## Déclaration d'une chaîne vide

```
String chaine="";      // Deux guillements consécutifs  
                      // (sans espace)
```

# Les chaînes de caractères

## Affectation

```
variable="texte";
```

## Affectation

```
String chaine1, chaine2, chaine3;  
chaine1="Bonjour";  
chaine2="";           // Chaine vide  
chaine3=chaine1;
```

## Affichage d'une chaîne

```
String chaine1;  
chaine1="Bonjour";  
System.out.println(chaine1);
```

# Les chaînes de caractères

nombre de caractères d'une chaîne (ou longueur)

variable.length()

## Longueur d'une chaine

```
int longueur;
String chaine;
chaine="Bonjour";
longueur=chaine.length(); //Nombre de caractères
System.out.println("Il_y_a "+longueur+" caractères dans la chaine");
```

# Les chaînes de caractères

i-ième caractère d'une chaîne

variable.charAt(i-1) (de type char)

variable.charAt(0) est le premier caractère de la chaîne variable  
variable.charAt(1) est le second caractère de la chaîne variable

i-ième caractère d'une chaîne

```
char c;  
String chaine;  
chaine="Bonjour";  
c=chaine.charAt(3);  
System.out.println("le_4-ième_caractère_est"+c);
```

# Les chaînes de caractères

## Juxtaposition de deux chaînes (concaténation)

chaine1+chaine2

## Juxtaposition de deux chaînes

```
String chaine1, chaine2, chaine3, chaine4;  
chaine1="Bonjour";  
chaine2="jourBon";  
chaine3=chaine1+chaine2;  
System.out.println(chaine3); // Affiche Bonjour jourBon  
chaine4=chaine2+chaine1;  
System.out.println(chaine4); // Affiche jourBonBonjour
```

# Les chaînes de caractères

## Test d'égalité de deux chaînes

chaine1.equals(chaine2) renvoie true si les contenus sont égaux, et false sinon

## Test d'égalité de deux chaînes

```
String chaine1, chaine2;  
chaine1="Bonjour";  
chaine2="Bonjour";  
// A NE PAS FAIRE  
if(chaine1==chaine2)  
    System.out.println("les_adresses_sont_égales");  
// A FAIRE  
if(chaine1.equals(chaine2))  
    System.out.println("les_contenus_des_chaines_sont_égaux");
```

# Les chaînes de caractères

## Les méthodes

### Les méthodes de la classe String

- **endsWith()** : Recherche si le mot se termine par le ou les caractères passés en paramètre
- **startsWith()** : Recherche si le mot commence par le ou les caractères passés en paramètre
- **charAt()** : Recherche le caractère placé à la position spécifiée en paramètre : 0...length()-1
- **indexOf()** : Localise un caractère ou une sous-chaine dans un mot, à partir du début du mot. -1 si la sous-chaine n'est pas trouvée
- **LastIndexOf()** : ... à partir de la fin du mot
- **substring()** : Extrait une sous-chaine d'un mot
- **compareTo()** : compare deux mots et retourne : 0 si les deux mots sont identiques; negative si mot1<mot2; positive si mot1>mot2
- **equals()** : true si mot1=mot2; false si mot1≠mot2
- **equalsIgnoreCase()** : sans différencier les majuscules des minuscules
- **regionMatches()** : portions identiques
- **toLowerCase()** : Transforme en minuscules
- **toUpperCase()** : Transforme en majuscule
- **concat()** : concatène
- **replace()** : remplace le caractère donné en premier argument par le caractère donné en deuxième argument
- **length()** : calcule la longueur de la chaîne

# Les chaînes de caractères

## Suppression des 'a' dans une chaîne

```
String chaine1, chaine2;
int i, longueur;
longueur=chaine1.length();
chaine2="";
for(i=0; i<=longueur-1; i++)
    if(chaine1.charAt(i) !='a')
        chaine2=chaine2+chaine1.charAt(i);

System.out.println(chaine2);
```

# Les chaînes de caractères

## Remplacement des 'a' par des 'o' dans une chaîne

```
String chaine1,chaine2;
int i, longueur;
longueur=chaine1.length();
chaine2="";
for(i=0;i<=longueur-1;i++)
    if(chaine1.charAt(i)!='a')
        chaine2=chaine2+chaine1.charAt(i);
    else
        chaine2=chaine2+'o';
System.out.println(chaine2);
```