

(`char`)

C - Les caractères (`char`)

Le type `char` en C sert à représenter un caractère unique. Lorsque l'on veut stocker plusieurs caractères à la suite, comme un mot ou une phrase, on utilise un tableau de caractères, noté `char[]`.

5TTR

6TTR

4GMS

 Découverte

Les caractères (`char`)

Les caractères permettent de stocker une lettre ou un symbole ASCII. Ils sont souvent utilisés pour gérer des messages ou des commandes.

Exemple : Communication avec un robot via Bluetooth.

```
1 | char commandeRecue = 'A'; // Commande reçue : avancer
```

En C, on utilise les **apostrophes** (*single quotes*) pour les caractères.

En réalité, un `char` contient le code ASCII du caractère:

```
1 | char code = 'A';  
2 | printf("%c", code); // --> Affiche A  
3 | printf("%d", code); // --> Affiche 65
```

Les chaînes de caractères (`char[]`)

Les chaînes de caractères sont utilisées pour manipuler des mots ou des messages. Contrairement à d'autres langages comme Python, C ne possède pas de type "*string*". Les chaînes de caractères sont stockées sous forme de tableau de caractères (`char[]`).

Il faut utiliser des guillemets (*double quotes*) pour définir une *string*:

```
1 | char prenom[] = "Marinette";  
2 | char messageLCD[] = "Robot prêt"; // Message statique
```

⚠ Important Les chaînes de caractères sont TOUJOURS terminées par le caractère `\0`. C l'ajoute automatiquement quand tu declares une chaîne avec des `"`. Cela aura de l'importance plus tard quand tu réserveras toi-même l'espace mémoire pour y stocker ta chaîne: il faut réserver un espace suffisamment grand que pour y stocker aussi le `\0`.

D'ailleurs cela peut se voir en affichant la taille d'une chaîne:

```
1 | char chaine[] = "azerty";  
2 | printf("%d", sizeof(chaine)); // Affiche 7
```