

### variables

```
int x = 2; // typé explicitement
```

```
var p = 5; // type inféré - var générique avec inférence de type
```

```
dynamic z = 8; // la variable peut prendre n'importe quel type
```

```
z = "cool"; // cool
```

```
// si vous n'avez jamais l'intention de changer une variable, utilisez final ou const. Quelque chose comme ceci : final email = "temid@gmail.com"; // Similaire à var mais ne peut pas être réassigné final String email = "temid@gmail.com"; // vous ne pouvez pas changer la valeur
```

```
const qty = 5; // Constante au moment de la compilation
```

### Opérateurs

```
print(2 + 3);
```

```
print(2 - 3);
```

```
print(2 * 3);
```

```
print(5 / 2);
```

```
print(5 ~/ 2);
```

```
print(5 % 2);
```

**Increment** `preIncrement` - Increment a before b gets its value. / `postIncrement` - Increment a AFTER b gets its value.

**Decrement** `b = --a;` : predecrement - Decrement a before b gets its value. / `b = a--;` : postdecrement - Decrement a AFTER b gets its value.

### Types de données

concaténation `String res = str1+ " " +str2;`

codeUnits (liste code UTF-16) `print(st.codeUnits);`

voir si chaîne vide ou non + renvoi V ou F `print(str.isEmpty);`

longueur de chaîne `print("La longueur de la chaîne est : ${str.length}");`

contains() (savoir si chaîne existe) `str1.contains('rem');` // true

vérifier si chaîne commence par les caractères spécifiés `str2.startsWith('Lorem');` // true

verif si chaîne se termine par ... `str3.endsWith('ipsum');` // true

convertir chaîne en min et maj `toLowerCase(), toUpperCase()`

fractionner une chaîne selon le paramètre `split()`

divise, convertit chaque élément de la liste et combine en une nouvelle liste `splitMapJoin()`

position première et dernière correspondances du modèle donné `indexOf(), dernierIndexof()`

supprimer les espaces blancs début et fin `trim()`

remplir les chaînes avec le remplissage donné `padLeft(), padRight()`

remplacer toutes les sous chaînes `replaceAll()`

### Types de données (cont)

supprimer les espaces de début et de fin mais pas les espaces entre deux chaînes `compareTo()`

revoit la sous chaîne de la chaîne qui s'étend de start à end (strat inclusif; end exclusif) `substring(int startIndex, [int endIndex])`

### Variables et constantes

déclarer une variable `var`

spécifier directement le type de données `dynamic`

spécifier directement le type de données `String myVariable1 = "MyText";`

déclarer la variable et affecter une seule fois une valeur à la variable `final`

recupérer le type d'une variable `runtimeType`

convertir un type en String `object.toString()`

interpoler des variables `${variable}`

demander nom d'un utilisateur `stdin.readLineSync()`

demander age d'un utilisateur `stdin.readLineSync()`

déclarer une constante `const` ou `final`

Définir la constante à l'aide du mot-clé final `final const_name;` ou `final data_type const_name`

Définir la constante à l'aide du mot-clé const `const const_name;` ou `const data_type const_name`



By DEVOYEURS

[cheatography.com/devoyeurs/](https://cheatography.com/devoyeurs/)

Not published yet.

Last updated 15th January, 2024.

Page 1 of 1.

Sponsored by **Readable.com**

Measure your website readability!

<https://readable.com>