



- | | |
|----|--|
| 1 | La commande Texte |
| 2 | La commande LaTeX |
| 3 | La commande TexteTourné |
| 4 | La commande FractionTexte |
| 5 | La commande TexteMath |
| 6 | La commande NotationScientifique |
| 7 | La commande Ordinal |
| 8 | La commande FractionContinue |
| 9 | La commande Tableau |
| 10 | Les commandes UnicodeEnLettre, LettreEnUnicode, UnicodeEnTexte et TexteEnUnicode |
| 11 | Commandes pour manipuler des chaînes de caractères |

http://url.univ-irem.fr/ft22



QR Code

GeoGebra offre de nombreuses commandes, utilisables en particulier au sein d'un script ou bien en combinaison avec la commande **Séquence** (voir la fiche technique **Listes et matrices**, page 515), pour créer ou manipuler des chaînes de caractères.

Dans les exemples qui suivent, nous vous proposons quelques exemples simples d'utilisation de ces différentes commandes à inscrire directement dans le champ de saisie.

Saisie: `LaTeX["\frac{a}{b}=\frac{" a "}{ " b "}"]`

1 La commande Texte

La commande **Texte** offre plusieurs syntaxes possibles :

- **Texte**[<objet>] permet d'afficher la valeur de l'objet <objet>.

Exemple(s)

- ✎ **Texte**["Bonjour"] affiche... « Bonjour »!
- ✎ Si n désigne une variable numérique, **Texte**[n] affiche la valeur de n .

- Le paramètre <objet> de l'instruction **Texte** peut être formé de plusieurs objets.

Exemple(s)

- Si $f(x) = 2x + 3$ et si $g(x) = 5x - 1$:
- ✎ **Texte**[" $f(x) =$ " + f] affiche $f(x) = 2x + 3$.
 - ✎ **Texte**[" $f(x) =$ " + f + " et $g(x) =$ " + g] affiche $f(x) = 2x + 3$ et $g(x) = 5x - 1$.
(l'opérateur de concaténation « + » peut-être remplacé par un simple espace)

- **Texte**[<objet>, <point>] permet d'afficher la valeur de l'objet <objet> à la position du point <point>.

Exemple(s)

- ✎ Si f désigne une fonction numérique, **Texte**[f , (3, 2)] affiche l'expression de f au point de coordonnées (3; 2).
- ✎ Si a désigne un segment, **Texte**[a , A] attache la longueur du segment au point A.

- **Texte**[<objet>,<booléen pour substitution de variable>] permet d'afficher la valeur de l'objet <objet> avec ou sans remplacement des variables par leurs valeurs numériques.

Exemple(s)

Si $a = 2$, $b = 5$ et si $f(x) = ax + b$:

 **Texte**[f, true] affiche $2x+5$.

 **Texte**[f, false] affiche $ax+b$.

- **Texte**[<objet>,<point>,<booléen pour substitution de variable>] permet d'afficher la valeur de l'objet <objet> à la position du point <point> avec ou sans remplacement des variables par leurs valeurs numériques.

Exemple(s)

Si $a = 3$ et si $b = a^2$:

 **Texte**[b, (5,6), true] affiche 9 au point de coordonnées (5;6).

 **Texte**[b,A, false] affiche a^2 aux coordonnées du point A.

- **Texte**[<objet>,<point>,<booléen pour substitution de variables>,<booléen pour formule LaTeX>] permet d'afficher la valeur de l'objet <objet> à la position du point <point> avec ou sans remplacement des variables par leurs valeurs numériques et avec ou sans mise en forme LaTeX.

Exemple(s)

Si $a = 2$, $b = 5$ et si $f(x) = ax + b$:

 **Texte**[f, (0,0), true, true] affiche $2x + 5$ (en tant que formule LaTeX) au point de coordonnées (0;0).

 **Texte**[f, (0,0), false, true] affiche $ax + b$ (en tant que formule LaTeX) au point de coordonnées (0;0).

2 La commande LaTeX

Si on souhaite produire un texte directement sous forme de formule LaTeX, on peut utiliser la commande **LaTeX** qui dispose de plusieurs syntaxes possibles.

- **LaTeX**[<objet>] retourne la valeur de l'objet <objet> en tant que formule LaTeX.

Exemple(s)

Si $a = 2$, $b = 3$ et $f(x) = ax + b$:

 **LaTeX**[f] affiche $2x + 3$ en tant que formule LaTeX.

 **LaTeX**["\frac{a}{b}=\frac{a}{b}" a "{" b "}"] affiche $\frac{a}{b} = \frac{a}{b}$.

- **LaTeX**[<objet>,<booléen pour substitution de variables>] permet d'afficher la valeur de l'objet <objet> avec ou sans remplacement des variables par leurs valeurs numériques.

Exemple(s)

Si $a = 5$, $b = 7$ et $u = \frac{a}{b}$:

 **LaTeX**[u, true] affiche $\frac{5}{7}$ en tant que formule LaTeX.

 **LaTeX**[u, false] affiche $\frac{a}{b}$.

- **LaTeX**[<objet>,<booléen pour substitution de variables>,<booléen pour afficher le nom de l'objet>] retourne la valeur de l'objet <objet> avec ou sans remplacement des variables par leurs valeurs numériques et avec l'affichage éventuel du nom de l'objet devant sa valeur.

Exemple(s)

Si $a = 2$, $b = 3$ et $f(x) = ax + b$:

 **LaTeX**[f,true,true] affiche $f(x) = 2x + 3$ en tant que formule $\LaTeX$.

 **LaTeX**[f,false,true] affiche $f(x) = ax + b$ en tant que formule $\LaTeX$.

3 La commande **TexteTourné**

La commande **TexteTourné**[<texte>,<angle>] permet d'effectuer une rotation d'angle <angle> de l'objet texte <texte>.

Exemple(s)

 **TexteTourné**["Bonjour",30°] affiche Bonjour .

 **TexteTourné**["Au revoir",-40°] affiche Au revoir .

Remarque :

- La commande **TexteTourné**[<texte>,<angle>] s'appuie en réalité sur la commande $\LaTeX$ `\rotatebox{<angle>}{<texte>}` pour produire l'affichage désiré. Ainsi, **TexteTourné**["Bonjour",30°] est équivalent à **LaTeX**["\rotatebox{45}{Bonjour}"].
- GeoGebra connaît aussi la commande **TexteVertical**[<texte>] qui est un raccourci pour la commande **TexteTourné**[<texte>,90°].
La syntaxe **TexteVertical**[<texte>,<point>] permet de spécifier la position du texte <texte>.

4 La commande **FractionTexte**

La commande **FractionTexte** possède deux syntaxes possibles.

- Si on souhaite afficher un nombre décimal sous forme de fraction irréductible, il faut recourir à la commande **FractionTexte**[<nombre>].

Exemple(s)

Si $a = 0,6$:

 **FractionTexte**[a] affiche $\frac{3}{5}$.

 **LaTeX**["a=" + (**FractionTexte**[a])] affiche $a = \frac{3}{5}$.

- La syntaxe **FractionTexte**[<point>] permet d'afficher les coordonnées d'un point sous forme fractionnaire.

Exemple(s)

Si $A(1.3;5.6)$:

 **FractionTexte**[A] affiche $\left(\frac{3}{5}, \frac{28}{5}\right)$.

La commande **TexteMath** tente d'écrire un nombre décimal sous forme exacte à l'aide de fractions, de racines carrées ou de constantes telles que le nombre π ou encore e (constante d'EULER). Plusieurs syntaxes sont disponibles.

- **TexteMath**[<nombre>] laisse GeoGebra tenter de déterminer une écriture satisfaisante du nombre <nombre> sous la forme d'un multiple d'une racine carrée. Si GeoGebra ne trouve pas de réponse acceptable, le nombre est retourné tel quel.

Exemple(s)

 **TexteMath**[1.73205080756] affiche $\sqrt{3}$.
 **TexteMath**[0.5] affiche $\frac{1}{2}$.
 **TexteMath**[sin(60°)] affiche $\frac{\sqrt{3}}{2}$.
 **TexteMath**[1.20710678118] affiche $\frac{\sqrt{2}+1}{2}$.

- La syntaxe **TexteMath**[<nombre>, <liste>] demande à GeoGebra d'écrire le nombre <nombre> sous la forme d'un multiple d'un ou de plusieurs éléments de la liste <liste>.

Exemple(s)

 **TexteMath**[3.14159265358, {pi}] affiche 3.14159265358000.
 **TexteMath**[3.141592653589, {pi}] affiche π .
 **TexteMath**[11.4247779607, {pi}] affiche $3\pi + 2$.
 **TexteMath**[1.322706261689, {sqrt(3), sqrt(5)}] affiche $\frac{\sqrt{5} + \sqrt{3}}{3}$.
 **TexteMath**[1.693147180559, {ln(2)}] affiche $\ln(2) + 1$.
 **TexteMath**[0.388325975494, {exp(1)}] affiche $\frac{e}{7}$.

- **TexteMath**[<point>] tente d'écrire les coordonnées du point <point> sous forme exacte.

Exemple(s)

 Si le point A a pour coordonnées polaires (2; 45°), **TexteMath**[A] affiche $(\sqrt{2}; \sqrt{2})$.
 **TexteMath**[2*e^(i*30°)] affiche $(\sqrt{3}; 1)$.

6 La commande **NotationScientifique**

Deux syntaxes sont possibles avec la commande **NotationScientifique**.

- **NotationScientifique**[<nombre>] permet d'afficher le nombre <nombre> sous forme scientifique.

Exemple(s)

 **NotationScientifique**[123456] affiche 1.23456×10^5 .
 **NotationScientifique**[0.000037] affiche 3.7×10^{-5} .
 **LaTeX**[**NotationScientifique**[0.01]] affiche 1×10^{-2} en tant que formule $\text{\LaTeX}$.

- **NotationScientifique**[<nombre>, <précision>] retourne le nombre <nombre> sous forme scientifique, arrondi avec le nombre de chiffres significatifs spécifiés par le paramètre <précision>.

Exemple(s)

-  `NotationScientifique[123456,2]` affiche 1.2×10^5 .
-  `NotationScientifique[123456,4]` affiche 1.235×10^5 .

7 La commande Ordinal

La commande `Ordinal[<nombre entier>]` retourne le rang associé au nombre <nombre entier>.

Exemple(s)

-  `Ordinal[1]` affiche 1er.
-  `Ordinal[2]` affiche 2e.
-  Si $a = 5$, `Ordinal[a]` affiche 5e.

8 La commande FractionContinue

La commande `FractionContinue` permet d'écrire un nombre sous forme de fraction continue. La fraction est calculée numériquement avec une précision de 10^{-8} . Plusieurs syntaxes sont possibles.

- `FractionContinue[<nombre>]` permet de créer, en tant que formule $\text{\LaTeX}$, une fraction continue du nombre <nombre>.

Exemple(s)

-  `FractionContinue[2.205]` affiche :
$$2 + \frac{1}{4 + \frac{1}{1 + \frac{1}{7 + \frac{1}{5}}}}$$
-  `FractionContinue[pi]` affiche :
$$3 + \frac{1}{7 + \frac{1}{15 + \frac{1}{1 + \frac{1}{292 + \dots}}}}$$

- `FractionContinue[<nombre>, <niveau>]` permet de créer, en tant que formule $\text{\LaTeX}$, une fraction continue du nombre <nombre> possédant un nombre d'étages inférieur ou égal au paramètre <niveau>.

Exemple(s)

-  `FractionContinue[2.205,3]` affiche :
$$2 + \frac{1}{4 + \frac{1}{1 + \dots}}$$
-  `FractionContinue[pi,3]` affiche :
$$3 + \frac{1}{7 + \frac{1}{15 + \dots}}$$

- `FractionContinue[<nombre>, <niveau>, <booléen>]` permet de créer, en tant que formule $\text{\LaTeX}$, une fraction continue du nombre <nombre> possédant un nombre de niveaux inférieur ou égal au paramètre <niveau> (ce paramètre est optionnel) et avec un affichage en ligne de la fraction si <booléen> vaut `true`

Exemple(s)

-  `FractionContinue[2.205,true]` affiche $[2;4,1,7,5]$.
-  `FractionContinue[pi,3,true]` affiche $[3;7,15,\dots]$.

9 La commande Tableau

Comme son nom l'indique, la commande `Tableau` permet de créer une zone de texte présentée sous la forme d'un tableau.

- **Tableau**[<liste 1>, <liste 2>, ...] permet de créer un tableau à plusieurs lignes. Chaque liste est affichée sur une ligne différente du tableau dont les éléments sont alignés par défaut à gauche de chaque cellule. Puisque les paramètres de la commande **Tableau** sont des listes, elles doivent être écrites entre accolades.

Exemple(s)

 **Tableau**[{"Hello", 10, 20}, {"a", 4}] affiche :

Hello	10	20
a	4	

 Si $a = 1$, $b = 2$ et $c = 3$, **Tableau**[{a,b,c}] affiche :

1	2	3
---	---	---

 **Tableau**[{TexteTourné["Hello", 30°], 10}, {LaTeX["\frac{2}{3}"], 5.4}] affiche :

Hello	10
$\frac{2}{3}$	5.4

- **Tableau**[<liste 1>, <liste 2>, ..., <alignement>] reprend une syntaxe identique à celle étudiée précédemment, mais ajoute le paramètre optionnel <alignement> qui permet de contrôler à la fois l'alignement, l'orientation ainsi que les bordures du tableau. Ce dernier paramètre est un objet texte et doit donc être écrit entre guillemets.

En ce qui concerne l'orientation du texte, le paramètre <alignement> peut prendre les valeurs "h" (par défaut), pour horizontal (les listes sont affichées sur chaque ligne du tableau) ou bien "v", pour vertical (les listes sont en colonnes).

L'alignement du texte peut être spécifié à l'aide des valeurs "l" (*left*) pour un alignement gauche (par défaut), ou "r" (*right*) pour un alignement à droite des cellules ou bien "c" (*center*) pour un alignement au centre.

Les valeurs "h" et "v" d'une part, et "l", "r" et "c" d'autre part, peuvent être combinées entre elles. Ainsi, les différentes valeurs permises pour le paramètre <alignement> sont les suivantes : "h", "v", "l", "r", "c", "hl", "hr", "hc", "vl", "vr" et "vc".

Exemple(s)

 **Tableau**[{1,2}, {3,4}, "v"] affiche :

1	3
2	4

 **Tableau**[{"oui", "et"}, {"non", "ou"}, "vr"] affiche :

oui	non
et	ou

À la suite des valeurs d'orientation et/ou d'alignement du texte, il est possible de préciser les éventuelles bordures ou lignes à faire apparaître dans le tableau.

- La bordure de gauche peut prendre les valeurs {, (, [ou | | ;
- On écrira "_" pour demander le tracé des lignes horizontales, et/ou "|" pour celui des lignes verticales ;
- La bordure de droite peut prendre les valeurs },),] ou | |.

Exemple(s)

 **Tableau**[{1,2}, {3,4}, "vc_|"] affiche :

1	3
2	4

 **Tableau**[{1,2}, {3,4}, "|||"] affiche :

1	2
3	4

 **Tableau**[**{1,2}**,**{3,4}**, "c(_)"] affiche :

$$\left(\begin{array}{cc} 1 & 2 \\ 3 & 4 \end{array} \right)$$

 **Tableau**[**{"3x+2y"**, "**=**", "**14**", **{ "-4x+5y"**, "**=**", "**-1**", "c{"] affiche :

$$\begin{cases} 3x+2y & = & 14 \\ -4x+5y & = & -1 \end{cases}$$

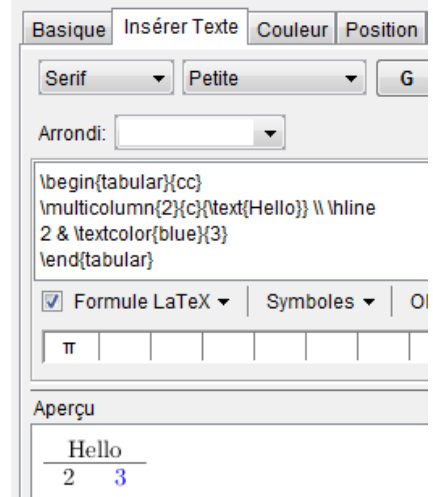
Remarque :

GeoGebra s'appuie sur $\text{\LaTeX}$ et sur son environnement `tabular` pour composer les tableaux avec la commande **Tableau**. Il est ainsi parfaitement autorisé, dans GeoGebra, de composer directement ses tableaux en $\text{\LaTeX}$, ce qui peut procurer davantage de souplesse et permettre des constructions inaccessibles avec la commande **Tableau**.

Par exemple, la fusion de plusieurs cellules devient possible avec $\text{\LaTeX}$ ainsi que la coloration du texte d'une cellule particulière ou encore le tracé d'une seule ligne horizontale dans le tableau.

```
\begin{tabular}{cc}
\multicolumn{2}{c}{\text{Hello}} \\ \hline
2 & \textcolor{blue}{3} \\
\end{tabular}
```

La commande **Tableau** permet cependant d'utiliser des listes dynamiquement construites (par exemple, à l'aide de la commande **Séquence**), ce qu'il ne serait pas possible de faire en $\text{\LaTeX}$.



10 Les commandes **UnicodeEnLettre**, **LettreEnUnicode**, **UnicodeEnTexte** et **TexteEnUnicode**

Unicode est un standard informatique qui permet, entre autres, d'établir une correspondance entre un nombre entier et un caractère.

La commande **UnicodeEnLettre**[<nombre>] renvoie le caractère correspondant au code Unicode <nombre>.

Exemple(s)

 **UnicodeEnLettre**[72] crée le texte H.

 **UnicodeEnLettre**[35] crée le texte #.

La commande **LettreEnUnicode**[<caractère>] renvoie le code Unicode du caractère <caractère>.

Exemple(s)

 **LettreEnUnicode**["A"] crée le nombre 65.

 **LettreEnUnicode**["1"] crée le nombre 49.

La commande **UnicodeEnTexte**[<liste>] associe chaque code Unicode de la liste <liste> avec le caractère correspondant afin de former une chaîne de caractères.

Exemple(s)

-  `UnicodeEnTexte[{66,111,110,106,11,117,114}]` crée le texte Bonjour.
-  `UnicodeEnTexte[{50,120,61,51}]` crée le texte $2x=3$.

La commande `TexteEnUnicode[<texte>]` renvoie la liste des codes Unicode qui correspondent aux caractères de la chaîne <texte>.

Exemple(s)

-  `TexteEnUnicode["IREM"]` retourne la liste {73,82,69,77}.
-  `TexteEnUnicode["x=y"]` retourne la liste {120,61,121}.

11 Commandes pour manipuler des chaînes de caractères

GeoGebra possède également quelques commandes permettant de manipuler les chaînes de caractères.

La commande `Longueur[<texte>]` renvoie le nombre de caractères de l'objet texte <texte>.

Exemple(s)

-  `Longueur["IREM"]` retourne le nombre 4.
-  `Longueur["Quelques mots"]` retourne le nombre 13.

La commande `Position` permet de déterminer la position de la première occurrence d'une chaîne de caractères dans un texte. Cette commande, sensible à la casse, prend deux syntaxes différentes :

- `Position[<chaîne>,<texte>]` renvoie sous forme d'un nombre la position de la première occurrence de la chaîne de caractères <chaîne> dans l'objet texte <texte>. Si la chaîne n'est pas trouvée, la valeur non défini est renvoyée.

Exemple(s)

-  `Position["RE","IREM de REIMS"]` retourne le nombre 2.
-  `Position["maths","Vive les Maths"]` retourne non défini.

- `Position[<chaîne>,<texte>,<départ>]` recherche la chaîne de caractères <chaîne> dans l'objet texte <texte> sans tenir compte des caractères de <texte> situés avant la position <départ>.

Exemple(s)

-  `Position["RE","IREM de REIMS",4]` retourne le nombre 9.
-  `Position["ma","mathématiques",2]` retourne le nombre 6.

La commande `Extraite[<texte>,<début>,<fin>]` retourne la chaîne constituée des caractères de <texte> situés de la position <début> à la position <fin>. On peut omettre le paramètre <fin> si on souhaite extraire les caractères de <texte> jusqu'à la dernière position.

Exemple(s)

-  `Extraite["Le théorème de Pythagore",16]` renvoie l'objet texte Pythagore.
-  `Extraite["Le théorème de Pythagore",4,11]` renvoie l'objet texte théorème.

La commande `Premiers[<texte>,<fin>]` est un raccourci pour `Extraite[<texte>,1,<fin>]`. Si le paramètre <fin> est omis, GeoGebra retourne uniquement le premier caractère de <texte>.

Exemple(s)

 **Premiers**["Bonjour"] renvoie l'objet texte B.

 **Premiers**["Une souris verte",3] renvoie l'objet texte Une.

La commande **Derniers**[<texte>,<nombre>] retourne une chaîne contenant les derniers caractères de <texte>. Le nombre de caractères retournés est fourni par le paramètre <nombre>, qui peut être omis (dans ce cas, GeoGebra renvoie uniquement le dernier caractère de <texte>).

Exemple(s)

 **Derniers**["Au revoir"] renvoie l'objet texte r.

 **Derniers**["Une souris verte",5] renvoie l'objet texte verte.

