# customenvs [fr]

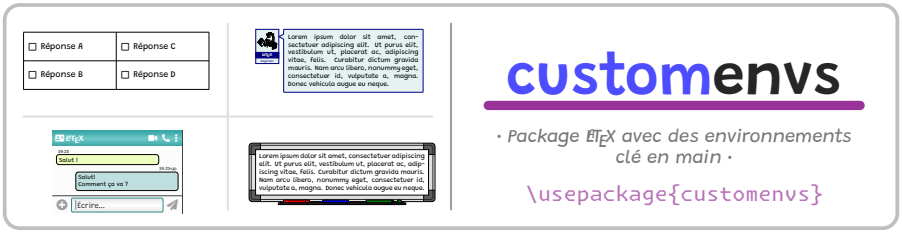
Quelques environnements classiques,
légèrement modifiés, et basés
sur des environnements existants.

Version 0.44c – 2026/11/24

Cédric Pierquet
cpierquet – at – outlook . fr

https://github.com/cpierquet/latex-packages/tree/main/customenvs

https://customenvs.cpierquet.fr




---

## Table des matières

1 Historique . . . . . 4
2 Le package customenvs . . . . . 5
2.1 Idée . . . . . 5
2.2 Chargement . . . . . 5
2.3 Sous-package customenvs-tikzpictos (v0.20d) . . . . . 6
2.4 Sous-package customenvs-exams (v0.1.9) . . . . . 7
2.5 Sous-package customenvs-icons (v0.1.5) . . . . . 8
2.6 Sous-package customenvs-mathpictos (v0.1d) . . . . . 9
2.7 Sous-package customenvs-macros (v0.1.0) . . . . . 9
2.8 Sous-package customenvs-tikzbadge (v0.1a) . . . . . 9
3 Présentation de réponses à un QCM . . . . . 10
3.1 Principe . . . . . 10
3.2 Exemples . . . . . 10
4 Environnement Centrage . . . . . 12
4.1 Principe . . . . . 12
4.2 Exemples . . . . . 12
5 Environnement multi-colonnes . . . . . 14
5.1 Principe . . . . . 14
5.2 Exemples . . . . . 14
6 Énumération avec choix des items, parmi une liste . . . . . 16
6.1 Principe et fonctionnement . . . . . 16
6.2 Exemples . . . . . 16
7 Tableau de compétences . . . . . 18
7.1 Principe et fonctionnement . . . . . 18
7.2 Éléments prédéfinis . . . . . 18

7.3 Exemples . . . 19
8 Crayon de compétences . . . 22
8.1 Principe et fonctionnement . . . 22
8.2 La commande . . . 22
8.3 Exemples . . . 23
9 Roue de compétences . . . 25
9.1 Principe et fonctionnement . . . 25
9.2 La commande et l'environnement . . . 25
9.3 Exemples . . . 26
9.4 « Speed-o-meter » . . . 27
10 Barres de niveau . . . 28
10.1 Principe et fonctionnement . . . 28
10.2 La commande . . . 28
11 PictoCible . . . 29
11.1 Principe et fonctionnement . . . 29
11.2 La commande . . . 29
12 Bandeau de score . . . 31
12.1 Principe et fonctionnement . . . 31
12.2 La commande . . . 31
13 Fenêtre type conversation instantanée . . . 32
13.1 Principe et fonctionnement . . . 32
13.2 L'environnement . . . 32
13.3 Les commandes de création des bulles . . . 32
13.4 Exemples . . . 33
13.5 Style WhatsApp . . . 34
14 Titre d'exercices . . . 37
14.1 Principe et définition du style global . . . 37
14.2 Options de personnalisations . . . 37
14.3 Exemples . . . 39
15 Boîtes divers . . . 42
15.1 Introduction . . . 42
15.2 'Petites' boîtes . . . 42
15.2.1 Numéros encerclés . . . 42
15.2.2 Boîte arrondie, petite boîte type marker . . . 42
15.2.3 Numéro type 'boîte' . . . 43
15.3 Boîtes de 'titres' . . . 44
15.3.1 Titres de sujets d'examens, titres d'exercices . . . 44
15.3.2 Bannière de titre . . . 44
15.3.3 Boîte avec ornements . . . 46
15.3.4 Titre 'bicolore' . . . 46
15.4 Boîtes de présentation . . . 47
15.5 Vignettes . . . 47
15.6 Panneaux autoroutiers . . . 51
15.7 Affichettes de soldes . . . 52
15.8 Boîtes diverses . . . 54
15.8.1 Boîte type message de forum . . . 54
15.8.2 Boîte type 'tableau blanc' . . . 55
15.8.3 Boîte type 'feuille perforée' . . . 56
16 Clavier français . . . 58
16.1 Introduction . . . 58
16.2 Liste des touches . . . 58
16.3 Clés de personnalisations . . . 58
17 Patchs divers . . . 61

17.1 Gérer l’espacement vertical du mode hors-ligne . . . . . 61
17.2 Quelques ajouts à pas-tableur (customenvs-pastableur, 0.1.2) . . . . . 61
17.3 Insertion d’images avec remplissage de l’espace vertical (customenvs-macros) . . . . . 64
17.4 Grille automatique pour les environnements tikzpicture . . . . . 65
17.5 Flèche évasée, en Ti*k*Z . . . . . 67
17.6 Annoter une image . . . . . 69
17.7 Travailler sur des longueurs (customenvs-macros) . . . . . 71

# 1 Historique

v0.44c : Améliorations dans le sous-package customenvs-exams
v0.44b : Améliorations dans le sous-package customenvs-exams
v0.44a : Améliorations dans le sous-package customenvs-exams
v0.43f : Sous-package customenvs-tikzbadge + Correction de bugs mineurs
v0.43e : Mise en conformité pas-tableur
v0.43d : Nouveau pictogramme *tikz*
v0.43c : Améliorations patches pas-tableur
v0.43b : Améliorations patches pas-tableur
v0.43a : Nouveaux pictogrammes *maths*
v0.42f : Nouvelle boîte type *feuille perforée*
v0.42e : Correction d'un *bug* avec fontawesome
v0.42d : Améliorations du code + Clavier français
v0.42c : Nouvelles icones (solarbroken) + nouveaux pictogrammes *maths*
v0.42b : Nouvelles icones (badge AI) + nouveaux pictogrammes *maths*
v0.42a : BoxNumber + sous-package customenvs-mathpictos
v0.41f : Améliorations + nouveaux pictogrammes
v0.41e : Bicolortitle + tcboxboard
v0.41d : Sous-package customenvs-macros
v0.41c : Améliorations dans les sous-packages
v0.41b : Bugfix + pre-compatibilité fa5/6/7 + nouveaux pictogrammes
v0.41a : Option noninlinegraphicx pour garder la compatibilité avec MikTeX + Bugfix
v0.40f : Style WhatsApp pour les fenêtre *conversation*
v0.40e : customenvs-icons v0.1.0
v0.40d : Compat twemojis + Alt 'Exercice' + customenvs-tikzpictos v0.1.4 + customenvs-exams v0.1.0
v0.40c : PictoClippy (customenvs-tikzpictos v0.1.3) + Gestion de longueurs
v0.40b : PictoCalendar (customenvs-tikzpictos v0.1.2) + Améliorations
v0.40a : PictoTraffic (customenvs-tikzpictos v0.1.1) + Améliorations
v0.3.7 : Externalisation des pictogrammes dans un package auxiliaire, customenvs-tikzpictos

---

# 2 Le package customenvs

## 2.1 Idée

L'idée est de proposer des commandes ou environnements classiques avec quelques éléments de personnalisation (via des clés francisées), comme :

- *centrer* avec gestion des espacements autour ;
- écrire en *multi-colonnes* avec gestion des espacements autour ;
- mettre en forme des réponses à des *QCM* ;
- créer une liste avec *choix des items* (de manière aléatoire ou par saisie directe) ;
- créer un tableau de *compétences*.

L'idée globale est de proposer des environnements clé en main, avec personnalisations *explicites*, sans forcément avoir besoin de *se pencher sur le code*, mais il est évident qu'il existe d'autres solutions pour l'utilisateur qui souhaite réellement contrôler son rendu.

Il est ici essentiellement question de gérer les espacements, donc on peut citer comme autres solutions possibles :

- l'utilisation de `\vspace` ou de `\setlength` ;
- le package `spacingtricks`.

## 2.2 Chargement

Le package se charge dans le préambule, via `\usepackage{customenvs}`.
Les packages chargés sont :

- `xstring`, `simplekv`, `listofitems`, `randomlist` et `xintexpr` ;
- `enumitem`, `multicol`, `tabularray`, `xcolor` ;
- `fontawesome5` (en attendant une compatibilité avec la version `6/7`, via les options `[fa6] / [fa7]`!) ;
- `tikz` avec les librairies :
  - `decorations.pathmorphing,positioning,shapes.misc,calc,arrows,arrows.meta`.

À noter que, pour des raisons de compatibilité (ou d'incompatibilité), les packages `enumitem/multicol/tabularray/xcolor/adjustbox/fontawesome5/6/7/inlinegraphicx` peuvent ne pas être chargés par `customenvs` (auxquels cas l'utilisateur devra les avoir chargés pour faire fonctionner certains environnements) via les options :

- ⟨`beamer`⟩ (pour assurer une compatibilité avec beamer) ;
- ⟨`nonenum`⟩ ;
- ⟨`nonmulticol`⟩ ;
- ⟨`nontblr`⟩ ;
- ⟨`nonxcolor`⟩ ;
- ⟨`nonadjustbox`⟩ ;
- ⟨`nontikzbabl`⟩ ;
- ⟨`noninlinegraphicx`⟩ ;
- ⟨`nonfa`⟩ / ⟨`fa6`⟩ / ⟨`fa7`⟩.

```
%chargement avec tous les packages
\usepackage{customenvs}

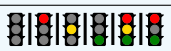
%chargement avec options(s) pour ne pas charger certains packages
\usepackage[option(s)]{customenvs}
```

## 2.3 Sous-package customenvs-tikzpictos (v0.20d)

`customenvs` charge, pour *ses* petits pictogrammes, le package `customenvs-tikzpictos`, qui est fourni avec.
L'idée est de pouvoir charger les petits pictogrammes de manière indépendante à `customenvs`.

```
\tkzpicto%
  [keys]
  <tikz options>
  {type=params}

%type := wifi/network/stars/speedo/bullseye/skills/pill/calendar/clippy/dball
%params := nb/nblevels (sauf bullseye) or dzay/month (calendar) or feeling (clippy)
%key height=len / auto (without depth) / dauto (with depth)
```

| | |
|---|---|
| Wifi |  |
| Wifi (bars) |  |
| Network |  |
| Stars |  |
| Speedometer |  |
| BullsEye |  |
| Battery |  |
| Battery (flip) |  |
| Skills |  |
| Pill |  |
| TrafficLight |  |
| MiniCalendar |  |
| Clippy |  |
| Dball |  |

## 2.4 Sous-package customenvs-exams (v0.1.9)

`customenvs` charge, pour quelques environnements ou commandes type *examens*, le package `customenvs-exams`, qui est fourni avec.
L'idée est de pouvoir charger ces fonctionnalités de manière indépendante à `customenvs`.

```
%réponses à un QCM (cf section suivante si besoin, les clés sont identiques)
\ExamReponsesQCM%
  [%
    EspacesCL={6pt/2pt},%               espaces H/V des cellules
    NbCols=1/2/4,%                      nombre de colonnes
    Filets=TF,%                         bordures
    PoliceLabels={\bfseries},%
    Labels={a.},%                       formattage des labels (box possible)
    EspaceLabels=\kern5pt,%
    Swap=false,%                        échange b<->c
    Largeur=0.99\linewidth,%
    Melange=false,%
    HtAuto=true,%                       hauteur automatique
  ]%
  {réponseA,réponseB,réponseC,réponseD}%
  <options tblr>
```

```
%environnement 'Automatismes'
\begin{ExamAutomatQuest}[options tcbox]{numéro}
...
\end{ExamAutomatQuest}
```

```
%présentation complète 'Automatismes'
\QuestionAutomatismes%
  {numéro}%
  [options tcbox]%
  {Énoncé}%
  [clés QCM]%
  {liste réponses}%
  <options tblr QCM>%
```

## 2.5 Sous-package customenvs-icons (v0.1.5)

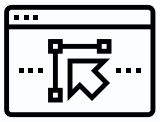
`customenvs` charge, pour *des* petites icônes, le package `customenvs-icons`, qui est fourni avec. L'idée est de pouvoir charger les petits icônes de manière indépendante à `customenvs`.

```
\ceicon%
  [%
    educ=TF,                        %booléen style 'educ'
    design=TF                       %booléen style 'design'
    height=...,                     %(d)auto / height / height+depth
    (d)strut=...                    %choix des boîtes (calculs de dimensions)
  ]%
  <options includegraphics>%
  {nom}
```

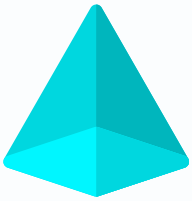
```
{\Huge X\ceicon{brush}X}
```

X X

```
\ceicon[height=2cm,design]{browser-html}
```



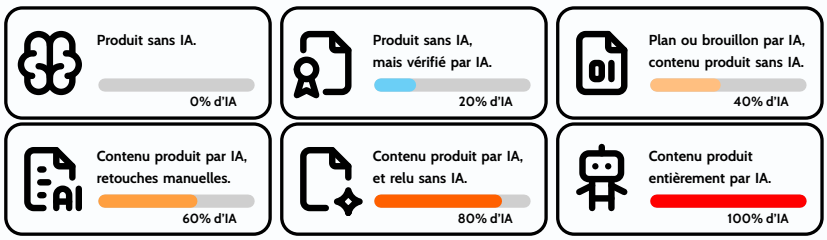
```
\ceicon[height=1in,mathsymb]{pyramid}
```



Il existe également des icônes type *honnêteté IA*, créés grâce à un travail d'Arnaud Durand. (Lien : `https://mathix.org/linux/archives/21431`).

```
\ceiconbadgeia[haut=1.5cm]{0}
\ceiconbadgeia[haut=1.5cm]{20}
\ceiconbadgeia[haut=1.5cm]{40}

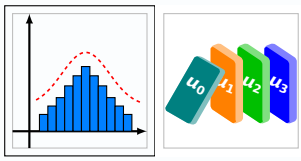
\ceiconbadgeia[haut=1.5cm]{60}
\ceiconbadgeia[haut=1.5cm]{80}
\ceiconbadgeia[haut=1.5cm]{100}
```



## 2.6 Sous-package customenvs-mathpictos (v0.1d)

`customenvs` charge, pour *des* petites icônes, le package `customenvs-mathpictos`, qui est fourni avec. L'idée est de pouvoir charger les petits icônes (réalisées en Ti*k*Z) de manière indépendante à `customenvs`.

```
\cemaththemeicon[height=2cm]{bordures=11}{binomiale}%
\cemaththemeicon[height=2cm]{bordures=01,fond}{recurrence}
```




## 2.7 Sous-package customenvs-macros (v0.1.0)

`customenvs` charge, pour *des* macros diverses et variées, le package `customenvs-macros`, qui est fourni avec. Il s'agit de macros pour :

— insérer une image avec remplissage vertical ;

— travailler sur des longueurs ;

— découper un argument en deux parties.

## 2.8 Sous-package customenvs-tikzbadge (v0.1a)

Il s'agit de macros pour :

— créer un petit badge avec personnalisation(s) ;

— générer une macro pour ses badges personnalisés.

```
\tkzpillbadge[color=green!50!black]{VRAI}\\
\tkzpillbadge[color=red!50!black,icon=x]{FAUX}\\
\tkzpillbadge[color=pink]{\halignmakebox{True}{True,False,Really,Cool}}\\
\tkzpillbadge[color=teal]{\halignmakebox{False}{True,False,Really,Cool}}\\
\tkzpillbadge[color=orange]{\halignmakebox{Really}{True,False,Really,Cool}}\\
\tkzpillbadge[color=olive]{\halignmakebox{Cool}{True,False,Really,Cool}}
```

VRAI
FAUX
True
False
Really
Cool

# 3 Présentation de réponses à un QCM

## 3.1 Principe

L'idée est de proposer une environnement prêt à l'emploi pour présenter, grâce à `tabularray` (et non pas à `multicols`) qui est donc à charger, les réponses à une question type QCM, données en colonnes.

Il est possible de spécifier 2, 3 ou 4 réponses, et dans le cas de 4 réponses il est possible de spécifier 1 ou 2 colonnes.

```
\ReponsesQCM[options]{liste reponses}<options tblr>
```

Les `options` disponibles sont :

- `Largeur` pour spécifier la largeur du tableau, `0.99\linewidth` par défaut ;
- `Filets` pour afficher les filets, `false` par défaut ;
- `EspacesCL` pour les espacements Colonnes/Lignes, sous la forme `col/lign` ou `globale`, et valant `6pt/2pt` par défaut ;
- `NbCols` pour forcer le passage à 2 colonnes dans le cas de 4 réponses, `4` par défaut ;
- `Labels` pour spécifier le formatage des labels, avec `a.` par défaut ;
  - pouvoir valoir `box` pour mettre des cases à cocher ;
  - pouvant faire intervenir `a` pour *numéroter* `a b c d` ;
  - pouvant faire intervenir `A` pour *numéroter* `A B C D` ;
  - pouvant faire intervenir `1` pour *numéroter* `1 2 3 4` ;
- `PoliceLabels` pour la police des labels, `\bfseries` par défaut ;
- `EspaceLabels` pour gérer l'espacement entre le label et la réponse, et valant `\kern5pt` par défaut ;
- `Melange` pour mélanger les réponses, et valant `false` par défaut ;
- `Swap` pour afficher les (4) réponses en mode 2 colonnes sous la forme ACBD ou ABCD, et valant `false` par défaut.

La liste des réponses est à donner sous la forme `answA § answB § ...`
Les options spécifiques, optionnelles et entre `<...>`, sont pour le dernier argument.

## 3.2 Exemples

```
%sortie par defaut
\ReponsesQCM{Réponse A § Réponse B § Réponse C § Réponse D}
```

**a.** Réponse A **b.** Réponse B **c.** Réponse C **d.** Réponse D

```
\ReponsesQCM[Filets]{Réponse A § Réponse B § Réponse C § Réponse D}

\ReponsesQCM[Filets,Melange]{Réponse A1 § Réponse B1 § Réponse C1 § Réponse D1}

\ReponsesQCM[Filets,Melange]{Réponse A2 § Réponse B2 § Réponse C2 § Réponse D2}
```

| | | | |
|---|---|---|---|
| **a.** Réponse A | **b.** Réponse B | **c.** Réponse C | **d.** Réponse D |
| **a.** Réponse D1 | **b.** Réponse A1 | **c.** Réponse C1 | **d.** Réponse B1 |
| **a.** Réponse D2 | **b.** Réponse B2 | **c.** Réponse A2 | **d.** Réponse C2 |

```
\ReponsesQCM[Filets,Labels=(1.),EspaceLabels={~~~}]{Réponse A § Réponse B § Réponse C}
```

| | | |
|---|---|---|
| **(1.)** Réponse A | **(2.)** Réponse B | **(3.)** Réponse C |

```
\ReponsesQCM[Labels={A.},PoliceLabels={\color{red}\bfseries}]%
    {Réponse A § Réponse B § Réponse C § Réponse D}
```

**A.** Réponse A **B.** Réponse B **C.** Réponse C **D.** Réponse D

```
\ReponsesQCM[NbCols=4,Labels={A.},PoliceLabels={\color{red}\bfseries}]%
    {Réponse A § Réponse B § Réponse C § Réponse D}
```

**A.** Réponse A **B.** Réponse B **C.** Réponse C **D.** Réponse D

```
\ReponsesQCM[NbCols=2,Labels={A.},PoliceLabels={\color{red}\bfseries}]%
    {Réponse A § Réponse B § Réponse C § Réponse D}
```

**A.** Réponse A
**B.** Réponse B
**C.** Réponse C
**D.** Réponse D

```
\ReponsesQCM[NbCols=2,Swap,Labels={A.},PoliceLabels={\color{red}\bfseries}]%
    {Réponse A § Réponse B § Réponse C § Réponse D}
```

**A.** Réponse A
**B.** Réponse B
**C.** Réponse C
**D.** Réponse D

```
% la case à cocher est \def\ReponsesQCMbox{\raisebox{-0.2ex}{\faSquare[regular]}}
\ReponsesQCM[Filets,NbCols=2,EspacesCL=6pt/10pt,Labels=box]%
    {Réponse A § Réponse B § Réponse C § Réponse D}
```

| | |
|---|---|
| ☐ Réponse A | ☐ Réponse C |
| ☐ Réponse B | ☐ Réponse D |

```
\ReponsesQCM[Largeur=10cm,NbCols=2,Filets]%
    {$\displaystyle\frac1x$ § $1+\displaystyle\frac1x$ § $-2x^2+5$ § $-\infty$}
    <rows={2cm}>
```

| | |
|---|---|
| **a.** $\dfrac{1}{x}$ | **c.** $-2x^2+5$ |
| **b.** $1+\dfrac{1}{x}$ | **d.** $-\infty$ |

# 4 Environnement Centrage

## 4.1 Principe

L'idée est de proposer un environnement, basé sur `center`, avec une gestion plus fine des espacements avant et après.
Le fait est qu'un environnement `center` génère des espacements (parfois) un peu trop grands autour (on peut également utiliser `\centering` ou `\centered` du package `spacingtricks`), et donc il s'agit ici de garder l'architecture *environnement* et proposant des solutions pour modifier les espacements.

```
\begin{Centrage}[options]
    %corps
\end{Centrage}
```

Les `options` disponibles sont :

— `Avant` pour spécifier l'espacement avant l'environnement, `0.33\baselineskip` par défaut ;

— `Apres` pour spécifier l'espacement après l'environnement, `0.33\baselineskip` par défaut.

À noter que les espacements peuvent être donnés de manière absolue, ou via des dimensions existantes.

## 4.2 Exemples

```
%environnement center, par défaut
\lipsum[1][1-3]

\begin{center}
    \lipsum[1][1]
\end{center}

\lipsum[1][1-2]
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis.

```
%centering
\lipsum[1][1-3]\par

{\centering\lipsum[1][1]\par}

\lipsum[1][1-2]
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis.

```
%environnement Centrage, par défaut
\lipsum[1][1-3]

\begin{Centrage}
    \lipsum[1][1]
\end{Centrage}

\lipsum[1][1-2]
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis.

```
%environnement Centrage, personnalisé
\lipsum[1][1-3]

\begin{Centrage}[Avant=0pt,Apres=0pt]
    \lipsum[1][1]
\end{Centrage}

\lipsum[1][1-2]
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis.

```
%environnement Centrage, avec des listes
\lipsum[2][3]

\begin{itemize}
    \item \lipsum[1][1]
    \item \lipsum[1][2]
\end{itemize}

\begin{Centrage}[Avant=-0.25\baselineskip]
    \lipsum[1][1]
\end{Centrage}

\lipsum[1][1-2]
```

Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus.

- Lorem ipsum dolor sit amet, consectetuer adipiscing elit.
- Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis.

# 5 Environnement multi-colonnes

## 5.1 Principe

L'idée est de proposer un environnement basé sur `multicols` (donc le package `multicol` est à charger), pour lequel les espacements avant et après peuvent être personnalisés.
C'est la longueur `\multicolsep`, qui vaut `12pt plus 4pt minus 3pt` par défaut, qui gère ces espacements.

L'idée est donc de proposer une environnement *simplifié* intégrant une modification de cette longueur.
Il sera également possible de créer automatiquement un environnement multi-colonnes combiné avec une liste d'énumération (avec `enumitem` chargé, par exemple) !
De plus, si le multi-colonnes est destiné à accueillir une liste, les items seront correctement alignés avec une liste sans multi-colonnes.

```
\begin{MultiColsAlt}[options](nbcols)<options enumitem>!options beamer!
    %corps
\end{MutiCols}
```

Les `options` disponibles sont :

— `Type` pour spécifier le type d'environnement qui sera inclus en multi-colonnes, et valant `texte` par défaut ;

à choisir parmi `texte / enum / item`

— `CoeffEspVert` pour spécifier le coefficient à appliquer à la longueur par défaut, et valant `0.5` par défaut ;

à choisir parmi `0 / 0.25 / 0.33 / 0.5 / 0.66 / 0.75 / 1 / 1.25`

— `EpTrait` pour l'épaisseur éventuelle du trait de séparation, et valant `0pt` par défaut.

Le nombre de colonnes, obligatoire, est à donner entre `(...)`.
L'argument optionnel et entre `<...>` est passé à l'environnement `enumitem` ou `itemize` si spécifié.
Le dernier argument, entre `!...!`, permet de spécifier des options pour l'environnement `enumitem` avec `beamer`.

## 5.2 Exemples

```
%par défaut
\lipsum[1][1-2]

\begin{MultiCols}(2)
    \lipsum[2]
\end{MultiCols}

\lipsum[1][3-4]
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna.

```
%espacement réduit + filet
\lipsum[1][1-2]

\begin{MultiCols}[CoeffEspVert=0.25,EpTrait=1pt](3)
  \lipsum[2]
\end{MultiCols}

\lipsum[1][3-4]
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna.

```
%type enumitem
\begin{enumerate}
    \item \lipsum[1][1-2]
    \begin{MultiCols}[Type=enum](4)
        \item bla
        \item bla
        \item bla
        \item bla
    \end{MultiCols}
    \item \lipsum[1][3-4]
    \begin{MultiCols}[Type=item](3)<label=$\bullet$>
        \item bla
        \item bla
        \item bla
    \end{MultiCols}
\end{enumerate}

\lipsum[3][1]
```

1. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis.

   (a) bla (b) bla (c) bla (d) bla

2. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna.

   - bla
   - bla
   - bla

Nulla malesuada porttitor diam.

# 6 Énumération avec choix des items, parmi une liste

## 6.1 Principe et fonctionnement

L'idée est ici de :

— créer une liste d'items qui servira de base pour le(s) choix ;

— afficher la liste avec choix des items, de manière aléatoire ou par items choisis

À noter que l'environnement `MultiCols` du package peut être utilisé comme environnement de listes !

```
\CreerListeItems{liste}{macro}{nomliste}
```

```
\ListeChoixItems[clés]{macro}{nomliste}(numéros)<options enumitem>!options beamer!
```

Les `clés` disponibles sont :

— `Type` pour spécifier le type d'environnement, et valant `enum` par défaut ;
à choisir parmi `enum` `item` ou `MultiCols/Type/NbCols`

— `Alea` pour forcer un affichage aléatoire, `false` par défaut.

Le deuxième argument, obligatoire et entre `{...}` est la macro créée précédemment.
Le troisième argument, obligatoire et entre `{...}` est le nom de la liste créée précédemment.
Le quatrième argument, obligatoire et entre `(...)` permet de spécifier :

— le nombre d'items à afficher en mode `Alea=true` ;

— les items à afficher, sous la forme `num1,num2,...`.

L'avant-dernier argument, optionnel et entre `<...>` correspond à des options spécifiques à passer à l'environnement de liste `enumitem` créé.
Le dernier argument, entre `!...!`, permet de spécifier des options pour l'environnement `enumitem` avec `beamer`.

À noter que des contrôles sont effectués lors de l'appel aux macros pour :

— vérifier que la liste n'existe pas déjà (pour la macro de création) ;

— vérifier que la liste existe déjà (pour la macro d'affichage des items).

## 6.2 Exemples

```
%création de la liste ListeItems, avec la macro \malisteditems
\CreerListeItems%
    {Réponse A,Réponse B,Réponse C,Réponse D,Réponse E,Réponse F,Réponse G,Réponse H}%
    {\malisteditems}{ListeItems}
```

```
%affichage d'items aléatoires
\ListeChoixItems[Alea]{\malisteditems}{ListeItems}(5)
```

1. Réponse H
2. Réponse D
3. Réponse F
4. Réponse C
5. Réponse E

```
%affichage de certains items
\ListeChoixItems{\malisteditems}{ListeItems}(1,4,3,8,2)
```

1. Réponse A
2. Réponse D
3. Réponse C
4. Réponse H
5. Réponse B

```
%création de la liste ListeItemsB, avec la macro \malisteditemsb
\CreerListeItems%
    {{$\int_0^1 x^2 dx$},{$\int_0^1 x^3 dx$},{$\int_0^1 x^4 dx$},...}%
    {\malisteditemsb}{ListeItemsB}
```

```
%affichage d'items aléatoires, via MultiCols
\ListeChoixItems[Alea,Type={MultiCols/enum/2}]{\malisteditemsb}{ListeItemsB}(4)
```

1. $\int_0^1 x^6 dx$
2. $\int_0^1 x^3 dx$
3. $\int_0^1 x^7 dx$
4. $\int_0^1 x^4 dx$

```
%affichage de certains items
\ListeChoixItems[Type=item]{\malisteditemsb}{ListeItemsB}(7,2,1,5,3)<label=$\bullet$>
```

- $\int_0^1 x^8 dx$
- $\int_0^1 x^3 dx$
- $\int_0^1 x^2 dx$
- $\int_0^1 x^6 dx$
- $\int_0^1 x^4 dx$

# 7 Tableau de compétences

## 7.1 Principe et fonctionnement

L'idée est de proposer un environnement pour créer un tableau de compétences, via 2/3/4 niveaux :

— basé sur `tblr`, qui doit donc être chargé (par défaut il l'est) ;

— basé sur `xcolor`, qui doit donc être chargé (par défaut il l'est) ;

— avec personnalisations possibles.

Si `xcolor` est déjà chargé, avec des options particulières, le package peut ne pas le charger, grâce à l'option `nonxcolor`.

```
\TableauCompetences[clés]{listecompétences}
```

```
\TableauCompetences{Compétence A § Compétence B}
```

| DS01 | NA | ECA | A | Note |
|---|---|---|---|---|
| Compétence A | | | | |
| Compétence B | | | | |
| **NA : Non acquis / ECA : En cours d'acquisition / A : Acquis** | | | | |

## 7.2 Éléments prédéfinis

Pour simplifier la saisie de certains paramètres, certaines macros ont été définies en interne, et pourront être utilisées, ou redéfinies si besoin.

```
%patch fa vcenter
\NewDocumentCommand{\vcenterfa}{ O{} m }{$\vcenter{\hbox{\faIcon[#1]{#2}}}$}
%labelnote
\def\LabelNoteComp{Note}
%listeniveaux
\def\LstDeuxNiv{NA § A}
\def\LstTroisNiv{NA § ECA § A}
\def\LstQuatreNiv{NA § PA § ECA § A}
%noticeniveaux
\def\NoticeDeuxNiv{Non acquis § Acquis}
\def\NoticeTroisNiv{Non acquis § En cours d'acquis. § Acquis}
\def\NoticeQuatreNiv{Non acquis § Part. acquis § En cours d'acquis. § Acquis}
```

Niveaux par « batterie » :
`\DeuxNivBatterie` : 🪫 § 🔋
`\TroisNivBatterie` : 🪫 § 🪫 § 🔋
`\QuatreNivBatterie` : ▭ § 🪫 § 🪫 § 🔋

Niveaux par « smiley » :
`\DeuxNivSmiley` : 😖 § 😊
`\TroisNivSmiley` : 😖 § ☹ § 😊
`\QuatreNivSmiley` : 😖 § ☹ § 😊 § 😀

Les `clés` disponibles sont :

- `Largeur` : largeur globale du tableau ; `0.95\linewidth` par défaut
- `LargeurNivs` : largeur des colonnes Niv + Note (séparées par `§`) ; `0.75cm § 1.25cm` par défaut
- `Niveaux` : liste des niveaux (séparés par `§`) ; `NA § ECA § A` par défaut
- `NoticeNiveaux` : notice des niveaux (séparés par `§`) ; `Non acquis § En cours d'acquisition § Acquis` par défaut ;
  *(il faut que le nombre d'éléments de cette clé soit en adéquation avec le nombre de niveaux !)*
- `Titre` : titre du tableau ; `DS01` par défaut
- `PolTitre` : police de la 1ere ligne ; `\small\sffamily\bfseries` par défaut
- `PolNotice` : police de la notice (dernière ligne) ; `\small\sffamily\bfseries` par défaut
- `PolComp` : police des lignes des compétences ; `\small\sffamily` par défaut
- `LigneSep` : séparation entre les lignes ; `2pt` par défaut
- `CouleurNotice` : couleur de la notice ; `black` par défaut
- `CouleurNiveaux` : couleur de la première ligne ; `black` par défaut
- `CouleurFond` : fond de la première ligne ; `lightgray!25` par défaut
- `Note` : booléen pour afficher la colonne note ; `true` par défaut
- `Notice` : booléen pour afficher la ligne notice ; `true` par défaut.

L'argument, obligatoire et entre `{...}` est la liste des compétences, sous la forme `Comp A § Comp B § ...`.
À noter que les clés `Niveaux` et `NoticeNiveaux` doivent avoir le même nombre d'éléments.

## 7.3 Exemples

```
%Note + Notice
\TableauCompetences{Utiliser le compas § Utiliser l'équerre}
```

| DS01 | NA | ECA | A | Note |
|---|---|---|---|---|
| Utiliser le compas | | | | |
| Utiliser l'équerre | | | | |
| **NA : Non acquis / ECA : En cours d'acquisition / A : Acquis** | | | | |

```
%- Note + Notice
\TableauCompetences[Note=false]{Utiliser le compas § Utiliser l'équerre}
```

| DS01 | NA | ECA | A |
|---|---|---|---|
| Utiliser le compas | | | |
| Utiliser l'équerre | | | |
| **NA : Non acquis / ECA : En cours d'acquisition / A : Acquis** | | | |

```
%Note - Notice
\TableauCompetences[Notice=false]{Utiliser le compas § Utiliser l'équerre}
```

| DS01 | NA | ECA | A | Note |
|---|---|---|---|---|
| Utiliser le compas | | | | |
| Utiliser l'équerre | | | | |

```
%- Note - Notice
\TableauCompetences[Note=false,Notice=false]{Utiliser le compas § Utiliser l'équerre}
```

| DS01 | NA | ECA | A |
|---|---|---|---|
| Utiliser le compas | | | |
| Utiliser l'équerre | | | |

```
%Personnalisations
\TableauCompetences[Titre=Eval n°01,Niveaux=\TroisNivBatterie]{Utiliser le compas § Utiliser
    l'équerre}\par
\TableauCompetences[Titre={},Niveaux=\TroisNivSmiley]{Utiliser le compas § Utiliser
    l'équerre}\par
\TableauCompetences[Largeur=10cm,Notice=false,Titre={},Niveaux=\TroisNivSmiley]{Utiliser le
    compas § Utiliser l'équerre}\par
```

| Eval n°01 | 🪫 | 🔋 | 🔋 | Note |
|---|---|---|---|---|
| Utiliser le compas | | | | |
| Utiliser l'équerre | | | | |
| 🪫 : Non acquis / 🔋 : En cours d'acquisition / 🔋 : Acquis | | | | |

| | 😠 | ☹ | ☺ | Note |
|---|---|---|---|---|
| Utiliser le compas | | | | |
| Utiliser l'équerre | | | | |
| 😠 : Non acquis / ☹ : En cours d'acquisition / ☺ : Acquis | | | | |

| | 😠 | ☹ | ☺ | Note |
|---|---|---|---|---|
| Utiliser le compas | | | | |
| Utiliser l'équerre | | | | |

```
%deux niveaux
\TableauCompetences[Niveaux=\LstDeuxNiv,NoticeNiveaux=\NoticeDeuxNiv]{Utiliser le compas §
    Utiliser l'équerre}\par
\TableauCompetences[Largeur=10cm,Titre={},Niveaux=\DeuxNivBatterie,
    NoticeNiveaux=\NoticeDeuxNiv]{Utiliser le compas § Utiliser l'équerre}
```

| DS01 | NA | A | Note |
|---|---|---|---|
| Utiliser le compas | | | |
| Utiliser l'équerre | | | |
| NA : Non acquis / A : Acquis | | | |

| | 🪫 | 🔋 | Note |
|---|---|---|---|
| Utiliser le compas | | | |
| Utiliser l'équerre | | | |
| 🪫 : Non acquis / 🔋 : Acquis | | | |

```
%quatre niveaux, personnalisation 'complète'
\def\LabelNoteComp{Appréc.}
\TableauCompetences[%
  Largeur=14cm,%
  LargeurNivs={1cm§3.5cm},%
  Niveaux={N0§N1§N2§N3},
  NoticeNiveaux={Très bof§Bof§Moyen§Bien},
  CouleurNotice=orange,%
  CouleurNiveaux=blue,%
  PolTitre=\large\ttfamily\itshape\bfseries
  ]%
  {Utiliser la règle § Utiliser le compas § Utiliser l'équerre}
```

| DS01 | N0 | N1 | N2 | N3 | Appréc. |
|---|---|---|---|---|---|
| Utiliser la règle | | | | | |
| Utiliser le compas | | | | | |
| Utiliser l'équerre | | | | | |
| NO : Très bof / N1 : Bof / N2 : Moyen / N3 : Bien | | | | | |

# 8 Crayon de compétences

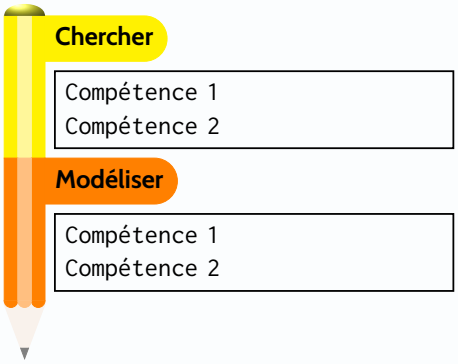
## 8.1 Principe et fonctionnement

L'idée est de proposer un environnement pour créer un *crayon* de compétences, basé sur `TikZ`.
Le code (en licence CC-BY-SA 4.0) est largement inspiré du fil :

https://tex.stackexchange.com/questions/504092/replicating-a-fancy-bordered-text-style-in-latex/504145#504145

```
\CrayonDeCompetences[clés]<options tikz>{listecompétences}
```

```
\CrayonDeCompetences{Chercher/Compétence 1\\Compétence 2,Modéliser/Compétence 1\\Compétence
    2}
```




La forme générale est fixée, et seuls quelques éléments de personnalisation(s) sont modifiables.

## 8.2 La commande

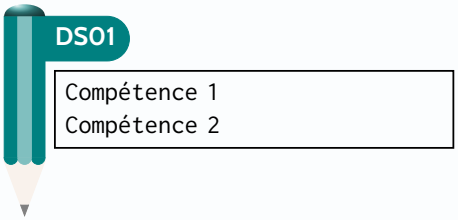
Les `clés` disponibles, à donner entre `[...]`, sont :

- `PoliceCateg` : police des catégories ; `\bfseries\sffamily` par défaut
- `PoliceBloc` : police des compétences ; `\small\ttfamily` par défaut
- `Couleurs` : liste des couleurs de catégories, sous la forme
  `FondCatég1/PoliceCatég1,FondCatég2/PoliceCatég2,...`
  (Si `PolicCatég` est absent, `black` est utilisé par défaut)
- `LargeurBloc` : largeur des blocs *texte* ; `5cm` par défaut ;
- `Echelle` : échelle globale du schéma ; `1` par défaut
- `NoirBlanc` : booléen pour un affichage N&B. `false` par défaut.

L'argument optionnel, et entre `<...>`, permet de spécifier des options à la figure `TikZ` (comme une rotation, un alignement, etc)

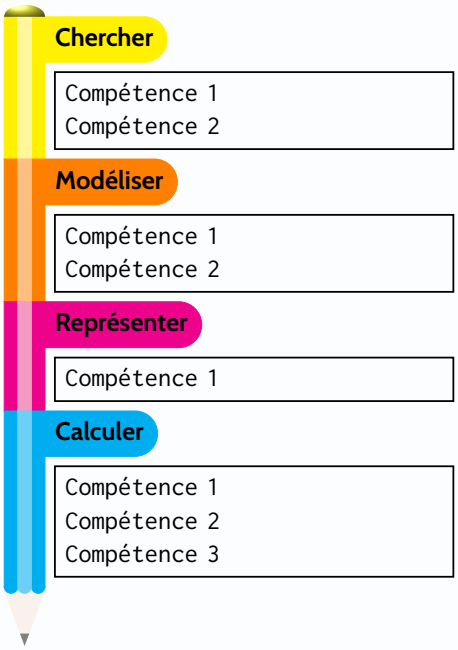
L'argument, obligatoire et entre `{...}` est la liste des catégories/compétences, sous la forme `Catég1/ListeCompétences1,Catég2/ListeCompétences2,...`.

```
\CrayonDeCompetences[Largeur=3cm,Couleurs=teal/white]{%
  DS01/Compétence 1\\Compétence 2
}
```

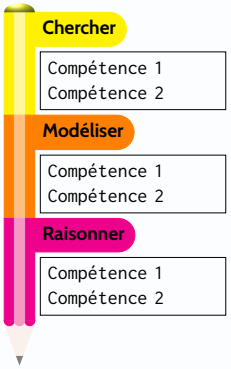


## 8.3 Exemples

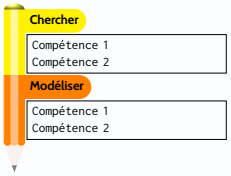
```
%Sortie par défaut
\CrayonDeCompetences{%
  Chercher/Compétence 1\\Compétence 2,
  Modéliser/Compétence 1\\Compétence 2,
  Représenter/Compétence 1,
  Calculer/Compétence 1\\Compétence 2\\Compétence 3
}
```



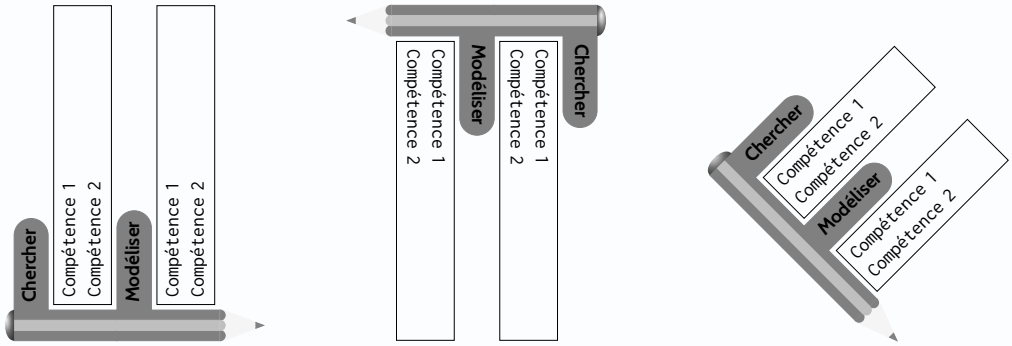

```
\CrayonDeCompetences[Echelle=0.75,LargeurBloc=3cm]%
  {Chercher/Compétence 1\\Compétence 2,
  Modéliser/{Compétence 1\\Compétence 2},Raisonner/{Compétence 1\\Compétence 2}}
```




```
\CrayonDeCompetences[Echelle=0.5]%
  {Chercher/Compétence 1\\Compétence 2,Modéliser/{Compétence 1\\Compétence 2}}
```



```
\CrayonDeCompetences[Echelle=0.75,NoirBlanc]<rotate=90>%
  {Chercher/Compétence 1\\Compétence 2,Modéliser/{Compétence 1\\Compétence 2}}%
\hspace{1cm}
\CrayonDeCompetences[Echelle=0.75,NoirBlanc]<rotate=-90>%
  {Chercher/Compétence 1\\Compétence 2,Modéliser/{Compétence 1\\Compétence 2}}
\hspace{1cm}
\CrayonDeCompetences[Echelle=0.75,NoirBlanc,Largeur,LargeurBloc=3cm]<rotate=45>%
  {Chercher/Compétence 1\\Compétence 2,Modéliser/{Compétence 1\\Compétence 2}}
```



# 9 Roue de compétences

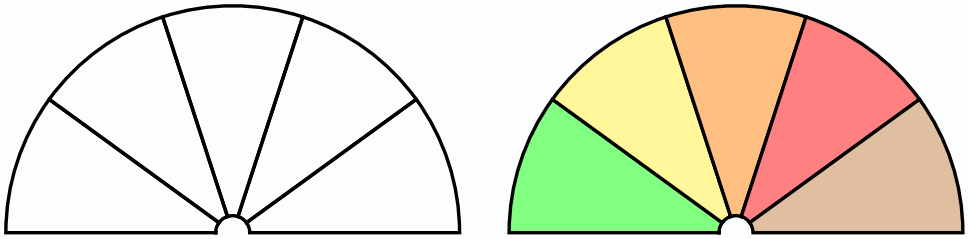
## 9.1 Principe et fonctionnement

L'idée est de proposer une commande et un environnement pour créer une *demie-roue* de compétences, basé sur `TikZ`.

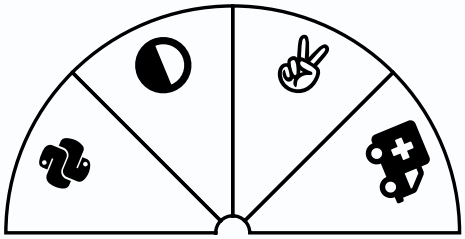
```
\RoueNiveaux[clés]<options tikz>{couleurs ou nb de niveaux}%
```

```
\begin{EnvRoueNiveaux}[clés]<options tikz>{couleurs ou nb de niveaux}%
  \PlacerIconesNiveaux[clés]{liste d'icônes}
\end{EnvRoueNiveaux}
```

```
\RoueNiveaux{5}\hspace{5mm}
\RoueNiveaux{green!50,yellow!50,orange!50,red!50,brown!50}
```



```
\begin{EnvRoueNiveaux}{4}
  \PlacerIconesNiveaux{\faPython,\faAdjust,\faAngellist,\faAmbulance}
\end{EnvRoueNiveaux}
```



## 9.2 La commande et l'environnement

Les `clés` (communes) disponibles, à donner entre `[...]`, sont :

— `Rayon` : rayon de la roue ; `4cm` par défaut

— `Police` : police des éventuels labels ; `\bfseries\sffamily` par défaut

— `ListeNiveaux` : liste éventuelle des niveaux, sous la forme `Label1,Label2,...`

— `Marqueur` : position relative d'un éventuel marqueur.

L'argument optionnel, et entre `<...>`, permet de spécifier des options à la figure `TikZ` (comme une rotation, un alignement, etc)

L'argument, obligatoire et entre `{...}` est la liste des couleurs, ou le nombre de catégories pour une sortie en N&B.

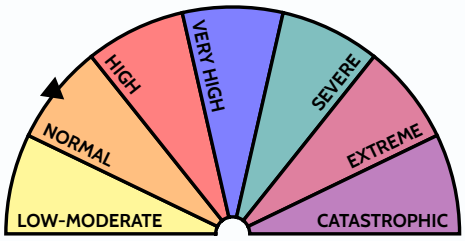
Pour l'environnement, la commande `\PlacerIconesNiveaux` permet de placer des icônes en lieu et place des libellés. les clés disponibles sont :

— `Pos` : position (excentricité) des icônes ; `0.8` par défaut

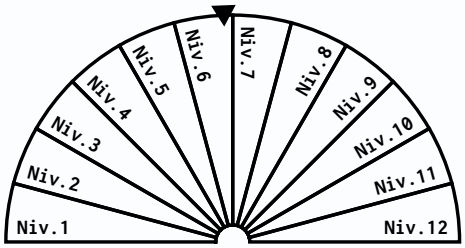
— `Echelle` : échelle des icônes ; `2`.

## 9.3 Exemples

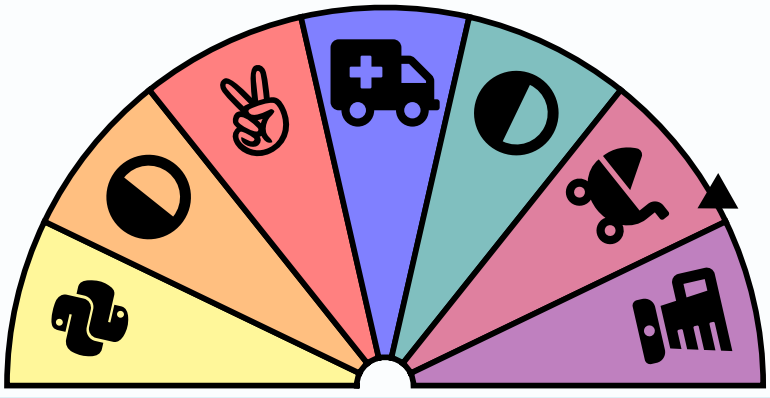
```
\RoueNiveaux[%
  Marqueur=1.5,%
  Police=\scriptsize\bfseries\sffamily,%
  ListeNiveaux={LOW-MODERATE,NORMAL,HIGH,VERY HIGH,SEVERE,EXTREME,CATASTROPHIC}
  ]%
  {yellow!50,orange!50,red!50,blue!50,teal!50,purple!50,violet!50}%
```




```
\RoueNiveaux[%
  Rayon=3cm,%
  Marqueur=5.85,%
  Police=\scriptsize\bfseries\ttfamily,%
  ListeNiveaux={%
    Niv.1,Niv.2,Niv.3,Niv.4,Niv.5,Niv.6,Niv.7,Niv.8,Niv.9,Niv.10,Niv.11,Niv.12}%
  ]%
  {12}
```




```
\begin{EnvRoueNiveaux}[Rayon=5cm,Marqueur=5.85]%
    {yellow!50,orange!50,red!50,blue!50,teal!50,purple!50,violet!50}
  \PlacerIconesNiveaux[Echelle=3]%
    {\faPython,\faAdjust,\faAngellist,\faAmbulance,\faAdjust,\faBabyCarriage,\faBlender}
\end{EnvRoueNiveaux}
```



### 9.4 « Speed-o-meter »

En marge de la roue *détaillée*, il est possible d'utiliser une version simplifiée, soit en mode en ligne, soit en mode autonome.

À noter qu'en mode en ligne, les dimensions sont calculées automatiquement en fonction de la police courante.

```
%à utiliser en mode en ligne
\MiniCompteurNiveaux[clés]<options tikz>{nb niveaux}
```

```
%mode autonome
\tkzspeedometer[clés]<options tikz>{nb niveaux}
```

Les `clés` (communes) disponibles, à donner entre `[...]`, sont :

— `Couleurs` : couleur(s) (sous la forme `couleur` ou `couleurp/couleurs`) ; `black` par défaut

— `Marqueur` : position relative d'un éventuel marqueur ;

— `Taille` : taille du cadran (en mode autonome uniquement !)

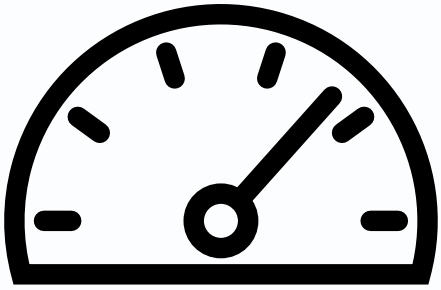
L'argument optionnel, et entre `<...>`, permet de spécifier des options à la figure `TikZ` (comme une rotation, un alignement, etc)

L'argument, obligatoire et entre `{...}` est le nombre de niveaux.

```
\scalebox{3}[3]{Petit compteur \MiniCompteurNiveaux[Couleurs=blue/red,Marqueur=4.33]{6} en
    ligne.}
```

Petit compteur  en ligne.

```
\tkzspeedometer[Marqueur=3.66]{5}
```



# 10 Barres de niveau

## 10.1 Principe et fonctionnement

L'idée est de proposer une commande pour insérer un petit pictogramme présentant des *barres de niveau*, à la manière de certains livres scolaires.

```
\pictoskills[clés]{nb}
```

```
\pictoskills{0}~\pictoskills{1}~\pictoskills{2}~\pictoskills{3}
```



```
%raccourcis pour les pictos 'de base'
\pictoskillsgreen~\pictoskillsorange~\pictoskillsred
```



## 10.2 La commande

La commande permet d'insérer, *en ligne*, le petit pictogramme à trois barres, avec une hauteur adaptée à la police courante (et de forme fixée).
Trois couleurs prédéfinies ont été déclarées, à savoir `skillgreen`, `skillorange` et `skillred`.

Les `clés` disponibles, à donner entre `[...]`, sont :

— `colors` : couleur(s) principale(s) ; `gray/lightgray` par défaut

— `height` : hauteur, donnée en unité ou bien parmi `auto/dauto` du pictogramme ; `dauto` par défaut

— `hoffset` : *vide* (en décimal) entre les barres ; `0.05` par défaut

— `opacity` : opacité de la partie *claire* ; `0.5` par défaut.

L'argument obligatoire permet de préciser le *niveau*, c'est-à-dire le nombre barres pleines dans le pictogramme.
À noter que la version étoilée permet d'insérer le pictogramme dans un environnement TikZ existant.

```
%hauteur personnalisée
\pictoskills[height=1cm]{1}

%hauteur automatique, basée sur la hauteur des majuscules sans jamabage
\begin{LARGE}
En ligne sans profondeur, on a \pictoskills[height=auto,colors=skillorange]{2} et
    \pictoskills[colors=skillred,hoffset=0.1,height=auto]{1}
\end{LARGE}

%hauteur automatique, avec intégration du jambage
\begin{Huge}
En ligne avec profondeur, on a \pictoskills[colors=skillorange]{2} et
    \pictoskills[colors=skillred,hoffset=0.025]{1}
\end{Huge}
```



# 11 PictoCible

## 11.1 Principe et fonctionnement

L'idée est de proposer une commande pour insérer un petit pictogramme du type cible/flèche.

```
\pictobullseye[clés]<options tikz>
```

```
\pictobullseye~\pictobullseye[height=auto]<rotate=90>
```

🎯 🎯

En mode `auto/dauto`, la hauteur du pictogramme est basée sur la hauteur de la lettre H ou bien la hauteur totale des lettres qH dans la police courante ! On peut modifier cette base via les macros suivantes.

```
\def\calcheightofchars{H}
\def\calctotheightofchars{qH}
\def\calcdepthofchars{qH}
```

## 11.2 La commande

La commande permet d'insérer, *en ligne*, le petit pictogramme, avec une hauteur adaptée à la police courante (et de forme fixée).

Les `clés` disponibles, à donner entre `[...]`, sont :

— `color` : booléen pour colorier ; `false` par défaut

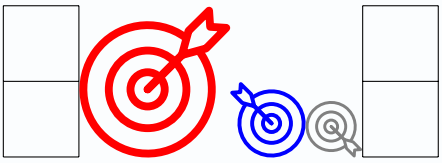
— `height` : hauteur, donnée en unité ou `auto/dauto` du pictogramme ; `dauto` par défaut

— `maincolor` : couleur principale, si différente du texte courant ; `vide` par défaut

— `listcolors` : liste éventuelle des couleurs des zones ; `yellow,red,blue!50!cyan` par défaut.

L'argument obligatoire permet de préciser des options spécifiques à la figure Ti*k*Z.

```
Hauteurs fixées := {\tikz\draw (0,0) grid (1,2);}%
\pictobullseye[height=2cm,maincolor=red]%
\pictobullseye[height=1cm,maincolor=blue]<rotate=90>%
\pictobullseye[height=0.75cm,maincolor=gray]<rotate=-90>%
{\tikz\draw (0,0) grid (1,2);}
```

Hauteurs fixées := 

```
Un essai q\pictobullseye\pictobullseye[height=auto,maincolor=red]H en ligne.
```

Un essai q🎯🎯H en ligne.

```
{\Huge\sffamily\textcolor{orange}{Un essai q\pictobullseye\pictobullseye[height=auto]H en
    ligne.}}
```

Un essai q🎯🎯H en ligne.

```
\scalebox{4}[4]{Tests en
    ligne\,\pictobullseye[height=auto,colors,listcolors={green!50,orange!50,red!50}] et
    \,\pictobullseye[colors,listcolors={red!50,yellow!50,teal!50}]!}
```

Tests en ligne et !

```
\def\calcheightofchars{É}
\def\calctotheightofchars{()}
\def\calcdepthofchars{()}
\scalebox{3}[3]{En
    ligne\,\pictobullseye[height=auto,colors,listcolors={green!50,orange!50,red!50}].
    Évidemment !}

{\huge Et (\pictobullseye[colors,listcolors={red!50,yellow!50,teal!50}]<rotate=90>)!}
```

En ligne. Évidemment !
Et ()!

# 12 Bandeau de score

## 12.1 Principe et fonctionnement

L'idée est de proposer une commande pour insérer un bandeau de score, type *nutriscore*.
La majorité des éléments sont personnalisables.

```
\BandeauScore[clés]{numéro}
```

```
%sortie par défaut
\BandeauScore{}
```




## 12.2 La commande

Les `clés` disponibles, à donner entre `[...]`, sont :

- `Hauteur` : hauteur du bandeau (sans la légende) ; `1` par défaut
- `Ratio` : rapport V/H des cases ; `0.6` par défaut
- `Symboles` : contenu des cases ; `A,B,C,D,E` par défaut
- `Legende` : texte de la légende (qui sera en majuscule) ; `score` par défaut ;
- `Police` : Police globale ; `\bfseries\sffamily` par défaut
- `AffLegende` : booléen pour afficher la légende ; `false` par défaut ;
- `Couleurs` : couleurs des cases ;
  `couleurNS1,couleurNS2,couleurNS3,couleurNS4,couleurNS5` par défaut ;
- `EchelleSymboles` : échelle(s) H/V des symboles ; `1.25,1.65` par défaut ;
- `CouleurFond` : couleur du fond (pour la case choisie) ; `white` par défaut.

L'argument obligatoire correspond quant à lui au numéro de la case à mettre en valeur (`vide` par défaut).
Si la liste des couleurs ne permet pas de remplir toutes les cases, la couleur `lightgray` est utilisée.

```
\BandeauScore[Legende=Géométrie,Hauteur=2]{4}
```




```
%le fond de la boîte a été défini sur yellow!25
\def\lstcouleurs{couleurNS1,couleurNS2,couleurNS3,couleurNS4,couleurNS5,purple}
\BandeauScore%
  [EchelleSymboles={1.33,2},Hauteur=2,AffLegende=false,
  Symboles={1,2,3,4,5,6},Couleurs=\lstcouleurs,
  CouleurFond=yellow!25]{1}
```



# 13 Fenêtre type conversation instantanée

## 13.1 Principe et fonctionnement

L'idée est de proposer un environnement pour créer une fenêtre type *conversation instantanée*, basée sur `tcolorbox`.

```
\begin{EnvSMS}[Clés]{nom}
  \SMSrec(*){heure}{msg}
  \SMSenv(*){heure}{msg}
\end{EnvSMS}
```

```
\begin{EnvSMS}{\LaTeX}
  \SMSrec{19:23}{Salut !}
  \SMSenv{19:23}{Salut!\\ Comment ça va~?}
\end{EnvSMS}
```




La forme générale est fixée, et seuls quelques éléments de personnalisation(s) sont modifiables.

## 13.2 L'environnement

Les `clés` disponibles pour l'environnement, à donner entre `[...]`, sont :

- `Hauteur` : hauteur de la fenêtre (automatique ou spécifique) ; `auto` par défaut
- `Largeur` : largeur de la fenêtre (un minimum de 5 cm est conseillé) ; `7cm` par défaut
- `Marge` : marge (G ou D) pour les bulles `1.5cm` par défaut
- `Couleur` : couleur *principale* (bandeau) ; `teal!75!cyan!75!white` par défaut ;
- `CouleurFond` : couleur du fond ; `lightgray!5` par défaut
- `CouleurR` : couleur des bulles de réception ; `lime!25` par défaut
- `CouleurE` : couleur des bulles d'envoi ; `teal!5` par défaut
- `TxtEcrire` : texte dans la zone d'envoi ; `Écrire` par défaut
- `PoliceTxt` : police des textes ; `\normalfont` par défaut
- `Avatar` : avatar du contact ; `\faAddressCard` par défaut
- `AffAvatar` : booléen pour ajouter l'avatar aux bulles de réception ; `false` par défaut
- `NoirBlanc` : booléen pour forcer un affichage N&B. `false` par défaut

L'argument, obligatoire et entre `{...}`, est le nom du contact à afficher.

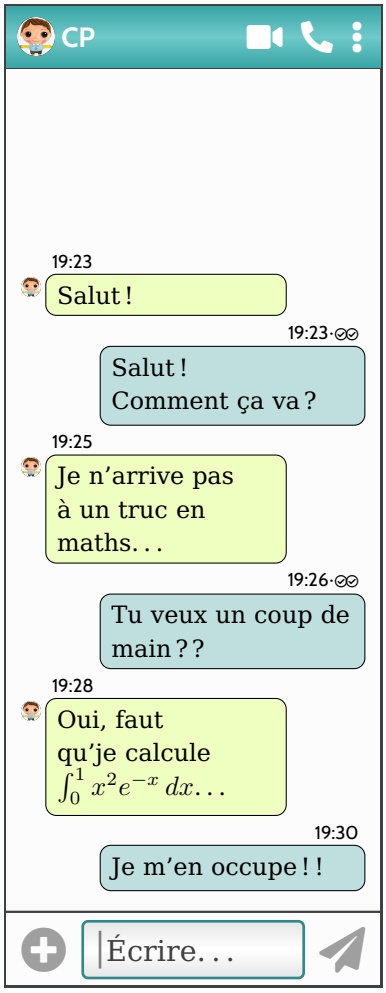
## 13.3 Les commandes de création des bulles

En ce qui concerne les commandes de création des bulles, `\SMSrec` et `\SMSenv` :

- la version *étoilée* n'affiche pas les *coches* de *bonne réception* ;
- le premier argument obligatoire est l'heure à afficher ;
- le deuxième argument obligatoire est le message à afficher (y compris multi-lignes).

## 13.4 Exemples

```
%avec une image personnelle
\begin{EnvSMS}%
  [Largeur=5cm,Hauteur=13cm,AffAvatar,Avatar=Image/avatar]{CP}
  \SMSrec{19:23}{Salut !}
  \SMSenv{19:23}{Salut!\\ Comment ça va~?}
  \SMSrec{19:25}{Je n'arrive pas à un truc en maths\ldots}
  \SMSenv{19:26}{Tu veux un coup de main ??}
  \SMSrec{19:28}{Oui, faut qu'je calcule $\int_{0}^{1} x^2e^{-x}\,dx$\ldots}
  \SMSenv*{19:30}{Je m'en occupe !!}
\end{EnvSMS}
```



```
\begin{EnvSMS}%
  [Largeur=8cm,PoliceTxt=\sffamily,NoirBlanc]{CP}
  \SMSrec{19:23}{Salut !}
  \SMSenv{19:23}{Salut!\\ Comment ça va~?}
  \SMSrec{19:25}{Je n'arrive pas à un truc en maths\ldots}
  \SMSenv{19:26}{Tu veux un coup de main ??}
  \SMSrec{19:28}{Oui, faut que je calcule $\mathsf{\int_{0}^{1} x^2e^{-x}\,dx}$\ldots}
  \SMSenv*{19:30}{Je m'en occupe !!}
\end{EnvSMS}
```

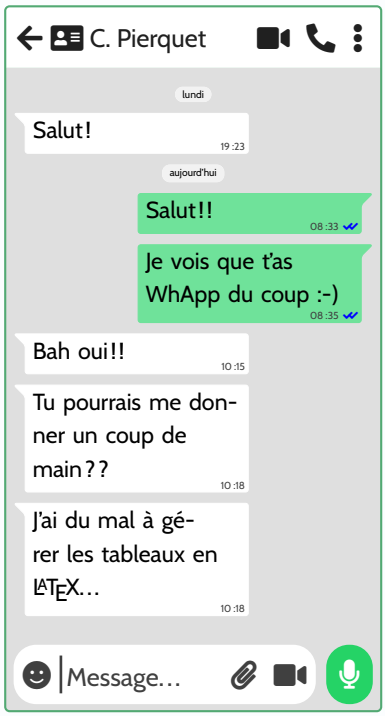



## 13.5 Style WhatsApp

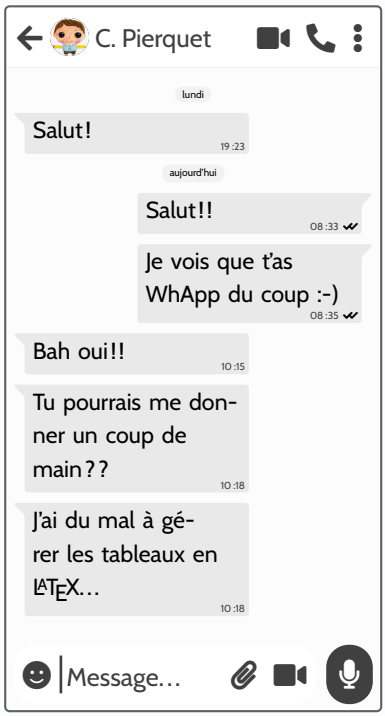
Un style type *WhatsApp* est également disponible, avec un fonctionnement similaire à celui présenté précédemment.
Les clés disponibles sont :

- `height` : hauteur ; `auto` par défaut
- `width` : largeur ; `5cm` par défaut
- `bgcolor` : couleur de fond ; `lightgray!50` par défaut
- `receivecolor` : couleur de réception ; `greenwa!66!white` par défaut
- `sendcolor` : couleur d'envoi ; `white` par défaut
- `txtwrite` : texte zone saisie ; `Message...` par défaut
- `fonttxt` : police ; `sffamily` par défaut
- `avatar` : avatar ; `\faAddressCard` par défaut
- `showavatar` : booléen pour afficher l'avatar ; `false` par défaut
- `bw` : booléen pour passer en noir et blanc ; `false` par défaut
- `txtwidth` : largeur des bulles ; `0.55` par défaut.

```
\begin{EnvChatWA}{C. Pierquet}
  \WaDate{lundi}
  \WaRec{19:23}{Salut !}
  \WaDate{aujourd'hui}
  \WaSend*{08:33}{Salut !!}
  \WaSend*{08:35}{Je vois que t'as WhApp du coup :-)}
  \WaRec{10:15}{Bah oui !!}
  \WaRec{10:18}{Tu pourrais me donner un coup de main ??}
  \WaRec{10:18}{J'ai du mal à gérer les tableaux en \LaTeX\ldots}
\end{EnvChatWA}
```



```
\begin{EnvChatWA}[bw,showavatar,avatar=Image/avatar]{C. Pierquet}
  \WaDate{lundi}
  \WaRec{19:23}{Salut !}
  \WaDate{aujourd'hui}
  \WaSend*{08:33}{Salut !!}
  \WaSend*{08:35}{Je vois que t'as WhApp du coup :-)}
  \WaRec{10:15}{Bah oui !!}
  \WaRec{10:18}{Tu pourrais me donner un coup de main ??}
  \WaRec{10:18}{J'ai du mal à gérer les tableaux en \LaTeX\ldots}
\end{EnvChatWA}
```



# 14 Titre d'exercices

## 14.1 Principe et définition du style global

L'idée est de proposer une commande ou un environnement pour afficher, *facilement*, un *titre* pour les exercices d'une évaluation ou d'une fiche d'exercices.

Les éléments personnalisables sont :

— le libellé via la clé `[Libelle]`, qui vaut `Exercice` par défaut ;

— la couleur via la clé `[Couleur]`, qui vaut `blue!50!black` par défaut ;

— la couleur de la décoration (si textuelle) via la clé `[CouleurDeco]`, qui vaut `blue!50!black` par défaut ;

— la couleur de fond la décoration (si il existe) via la clé `[CouleurFondDeco]`, qui vaut 50 % de `blue!50!black` par défaut ;

— la police via la clé `[Police]`, qui vaut `\bfseries\LARGE\sffamily` par défaut ;

— l'épaisseur du trait (éventuel) via la clé `[EpTrait]`, qui vaut `1.1pt` par défaut.

L'utilisateur peut personnaliser *globalement* les éléments précédents, les styles seront dans ce cas propagés à tous les environnements ou à toutes les commandes créant les titres.

```
%personnalisation(s)
\StyleEnvtExo[clés]

%paramètres par défaut
\StyleEnvtExoDefaut
```

Une fois les styles *choisis*, il suffit d'utiliser la commande d'environnement ou la commande *directe* :

```
%exo1, environnement par défaut
\begin{EnvtExo}
Ceci est un exercice\ldots
\end{EnvtExo}

%exo1 alt, environnement par défaut
\begin{EnvtExoAlt}
Ceci est un exercice\ldots
\end{EnvtExoAlt}
```

**Exercice 1**

Ceci est un exercice…

**Exercice 1**

Ceci est un exercice…

```
%exo2, macro par défaut
\TitreExo

Ceci est un exercice\ldots
```

**Exercice 2**

Ceci est un exercice…

## 14.2 Options de personnalisations

Hormis le style global, il est possible de modifier/ajouter certains éléments :

— le trait peut être défini :

  — plein (par défaut) ;

    - en pointillés, via `dotfill` ;
    - en *dashillés*, via le package `dashrulex` ;
- un complément de titre peut être rajouté ;
- une décoration peut être rajoutée en fin de ligne :
    - des points, sous la forme `(xx points)` ;
    - une image, via `graphicx`, dont la hauteur est adaptée à la hauteur de la ligne en cours ;
    - une icône, via `fontawesome5`, dont la hauteur est adaptée à la hauteur de la ligne en cours ;
    - des étoiles, via `fontawesome5` ou Ti*k*Z, dont la hauteur est adaptée à la hauteur de la ligne en cours ;
    - un petit chronomètre, via `pictochrono`, dont la hauteur est adaptée à la hauteur de la ligne en cours ;
    - un petit *speedometer*, dont la hauteur est adaptée à la hauteur de la ligne en cours ;
    - une ceinture colorée, via `coloredbelts`, dont la hauteur est adaptée à la hauteur de la ligne en cours ;
    - une barre de niveaux colorée dont la hauteur est adaptée à la hauteur de la ligne en cours ;
- le compteur (nommé `numeroexo` ou `numeroexoalt`) peut être désactivé et *adpaté* via les commandes classiques de compteurs.

Les `clés` disponibles, à donner entre `[...]`, sont :

- `Type` : type de libellés, parmi `Classique` (défaut) ou `Perso/titrepersonnalisé`
- `ComplementTitre` : complément pour la titre, après le numéro (attention aux espaces) ;
- `CodeDebut` : code LaTeX qui sera *rajouté* entre le titre et l'énoncé ;
- `EchelleImage` : pour modifier ponctuellement l'image de décoration ;
- `Compteur` : booléen pour activer/désactiver (`true` par défaut) le compteur (non affiché et non incrémenté) ;
- `Decoration` : choix de la décoration parmi :
    - `Icone/...` := afficher l'icône `...` à la fin de la ligne ;
    - `Perso/...` := afficher une commande personnelle ;
    - `Image/...` := afficher l'image `...` à la fin de la ligne ;
    - `Ceinture/...` := afficher la ceinture couleur `...` à la fin de la ligne ;
    - `Chrono/...` := afficher le chrono de durée `...` à la fin de la ligne ;
    - `Points/...` := afficher les points `(... points)` à la fin de la ligne ;
    - `Speedo/...§...` := afficher un *speedometer* ;
    - `faEtoiles/...` := afficher des étoiles de difficultés à la fin de la ligne ;
    - `tkzEtoiles/...` := afficher des étoiles de difficultés à la fin de la ligne ;
    - `Barres(*)/...` := un niveau de difficulté (`V0/V1/V2/V3/O0/...`) ;
- `Trait` : choix du trait parmi :
    - `plein` := un trait plein (par défaut) ;
    - `non` := aucun trait ;
    - `pointilles` := pointilles (`\dotfill`) ;
    - `dashilles` := pointilles *dash* (`\hdashrule`).

La commande d'affichage des étoiles de difficultés est indépendante de l'environnement `EnvtExo`, la clé `Decoration=xxxEtoiles/niv` ou `Decoration=xxxEtoiles/niv§max` permet de l'y intégrer.

```
%commande d'étoiles (max=3 par défaut) avec fontawesome5
\NiveauDiffExos[max]{nb}
%commande d'étoiles (max=3 par défaut) avec tikz
\tkzEtoiles[Couleur=...,CouleurFond=...,Offset=...,NiveauMax=...,AlignV=...]{nb}
```

```
%des demies étoiles sont possibles
\NiveauDiffExos{0}\par
\NiveauDiffExos{2.5}\par
\textcolor{teal}{\LARGE\NiveauDiffExos[5]{4}}\par
\NiveauDiffExos[5]{1.5}\par
```



```
%des portions d'étoiles sont possibles
%Offset=... permet de gérer l'espacement horizontal
%AlignV est un booléen pour décaler verticalement les étoiles (inline)
\tkzEtoiles{2.5}\par
{\LARGE On essaye en ligne \tkzEtoiles{2.5} avec une note de 2.5}\par
{\LARGE On essaye en ligne \tkzEtoiles[AlignV=false]{2.5} avec une note de 2.5}\par
\tkzEtoiles[Couleur=red,CouleurFond=yellow,NiveauMax=5]{3.5}
```




## 14.3 Exemples

Les exemples suivants ont été obtenus via un document externe (fourni avec le package), du fait du chargement de packages spécifiques.
Le code proposé utilise l'environnement, mais la commande simple est complètement compatible également !

⚠ Les petites *images* ne sont pas incluses dans le package, elles sont là pour illustrer l'utilisation d'images personnelles.

```
\documentclass[a5paper,11pt]{article}
\usepackage[margin=1cm]{geometry}
\usepackage{customenvs}
%facultatif, pour les décorations
\usepackage{graphicx}
\usepackage{dashrulex}
\usepackage{coloredbelts}
\usepackage{pictochrono}
%mise en page
\usepackage{ipsum}
\setlength{\parindent}{0pt}
```

```
\begin{EnvtExo}\ipsum<Lang=FR,Type=sent>\end{EnvtExo}%

\begin{EnvtExo}[Trait=pointilles]\ipsum<Lang=FR,Type=sent>\end{EnvtExo}%

\begin{EnvtExo}[Trait=dashilles]\ipsum<Lang=FR,Type=sent>\end{EnvtExo}%

\begin{EnvtExo}[Decoration=Icone/\faPython]\ipsum<Lang=FR,Type=sent>\end{EnvtExo}%

\begin{EnvtExo}[Decoration=Ceinture/rouge]\ipsum<Lang=FR,Type=sent>\end{EnvtExo}%

\begin{EnvtExo}[Decoration=Chrono/20]\ipsum<Lang=FR,Type=sent>\end{EnvtExo}%

\begin{EnvtExo}[Decoration=Image/goku_ssj4]\ipsum<Lang=FR,Type=sent>\end{EnvtExo}%

\begin{EnvtExo}[Decoration=faEtoiles/1.5§4]\ipsum<Lang=FR,Type=sent>\end{EnvtExo}%

\begin{EnvtExo}[Decoration=tkzEtoiles/3.5§5]\ipsum<Lang=FR,Type=sent>\end{EnvtExo}%

\begin{EnvtExo}[Decoration=Barres/V2]\ipsum<Lang=FR,Type=sent>\end{EnvtExo}%

\begin{EnvtExo}[Decoration=Points/7.5]\ipsum<Lang=FR,Type=sent>\end{EnvtExo}%

\begin{EnvtExo}[Decoration=Speedo/4.25§6]\ipsum<Lang=FR,Type=sent>\end{EnvtExo}%

\begin{EnvtExo}[Type=Perso/{Titre perso }]\ipsum<Lang=FR,Type=sent>\end{EnvtExo}%

\StyleEnvtExo[Couleur=red,CouleurDeco=teal,Police=\bfseries\ttfamily,EpTrait=2pt,
    Libelle={Exercice n°}]

\begin{EnvtExo}[Decoration=Icone/\faAddressBook]
\ipsum<Lang=FR,Type=sent>
\end{EnvtExo}%


\begin{EnvtExo}[Type=Perso/{Titre perso },Decoration=Chrono/25,Trait=dashilles]%
\ipsum<Lang=FR,Type=sent>
\end{EnvtExo}

\StyleEnvtExoDefaut
\begin{EnvtExo}%
  [Type=Perso/{Titre perso~},Decoration=Image/goku_ssj4,Trait=non,Compteur=false]%
\ipsum<Lang=FR,Type=sent>
\end{EnvtExo}

\TitreExo[Type=Perso/{Annexe Exercice 3},Compteur=false,Decoration=Image/sseiya_aiolos]%
\ipsum<Lang=FR,Type=sent>
```

```
\includegraphics{envtexo_exemples.pdf}
```

## Exercice 1

Sa réponse fut parfaite, et surtout longue comme un mandement ; elle laissait

## Exercice 2

Sa réponse fut parfaite, et surtout longue comme un mandement ; elle laissait

## Exercice 3

Sa réponse fut parfaite, et surtout longue comme un mandement ; elle laissait

## Exercice 4

Sa réponse fut parfaite, et surtout longue comme un mandement ; elle laissait

## Exercice 5

Sa réponse fut parfaite, et surtout longue comme un mandement ; elle laissait

## Exercice 6

Sa réponse fut parfaite, et surtout longue comme un mandement ; elle laissait

## Exercice 7 (★★☆)

Sa réponse fut parfaite, et surtout longue comme un mandement ; elle laissait

## Exercice 8

Sa réponse fut parfaite, et surtout longue comme un mandement ; elle laissait

## Exercice 9

Sa réponse fut parfaite, et surtout longue comme un mandement ; elle laissait

## Exercice 10

Sa réponse fut parfaite, et surtout longue comme un mandement ; elle laissait

## Exercice 11 (7,5 points)

Sa réponse fut parfaite, et surtout longue comme un mandement ; elle laissait

## Exercice 12

Sa réponse fut parfaite, et surtout longue comme un mandement ; elle laissait

## Titre perso 13 (★★⯪☆☆)

Sa réponse fut parfaite, et surtout longue comme un mandement ; elle laissait

## Exercice n°14

Sa réponse fut parfaite, et surtout longue comme un mandement ; elle laissait

## Exercice n°15

Sa réponse fut parfaite, et surtout longue comme un mandement ; elle laissait

## Titre perso 16 (⯪☆☆)

Sa réponse fut parfaite, et surtout longue comme un mandement ; elle laissait

## Titre perso

Sa réponse fut parfaite, et surtout longue comme un mandement ; elle laissait

## Annexe Exercice 3

Sa réponse fut parfaite, et surtout longue comme un mandement ; elle laissait

# 15 Boîtes divers

## 15.1 Introduction

L'idée est de proposer (modestement) des commandes, basées sur `tcolorbox`, pour, avec un style prédéfini :

— créer des boîtes de présentation ;

— créer des titres de sujets d'examens, par exemple ;

— créer des titres d'exercices, pour des sujets d'examens par exemple ;

— créer des numéros encadrés ;

— créer de vignettes.

## 15.2 'Petites' boîtes

### 15.2.1 Numéros encerclés

```
\CircledNumber[bg=...,txt=...,bthick=...,bcol=...,raise=true/false]%
    {nombre}{noeud tikz}
```

```
En ligne \CircledNumber{1} avec texte après.
```

En ligne ① avec texte après.

```
{\bfseries\sffamily\Huge En ligne
    (\CircledNumber[bthick=0.25mm,bcol=red,bg=cyan!25,txt=darkgray]{7}) avec texte après}
```

**En ligne (⑦) avec texte après**

```
{\Large En ligne \CircledNumber[raise=false,bthick=0.5mm,bcol=cyan,bg=cyan!25,txt=orange]{4}
    avec texte après}
```

En ligne ④ avec texte après

```
\begin{enumerate}[label={\CircledNumber[raise=false]{\arabic*}}]
  \item A
  \item B
  \item C
\end{enumerate}
```

① A

② B

③ C

### 15.2.2 Boîte arrondie, petite boîte type marker

```
\BoiteArrondie[Fond=...,Texte=...,EspH=...,Style=...]{texte}[noeud tikz]
```

```
On lance le logiciel \BoiteArrondie[Fond=cyan!33,Texte=violet,EspH=2mm,Style=rect]{situé sur
    le bureau} en cliquant \BoiteArrondie[Fond=lightgray!25,Texte=darkgray]{droit}.
```

On lance le logiciel situé sur le bureau en cliquant droit.

```
\tbcmarker[Couleur=...,Largeur=...,Police=...]{contenu}
```

```
\tbcmarker{contenu}
```

contenu

```
\tbcmarker[Couleur=olive,Police=\normalfont\normalsize]{contenu}
```

contenu

### 15.2.3 Numéro type 'boîte'

```
\tkzboxnumber[clés]{numéro}

%---Les clés disponibles sont :
%bg color       = darkgray
%shadow color   = gray
%font           = \bfseries
%shadow         = true
%shadow offset  = 0.25pt
%num color      = white
%width auto     = true
```

```
%style global (pour visualiser l'influence de width auto)
\setKVdefault[TikzBoxNumber]{font=\bfseries\sffamily}

\tkzboxnumber{1}~\lipsum[1][1]\par
\tkzboxnumber[width auto=false]{1}~\lipsum[1][1]\par
\tkzboxnumber[bg color=red]{2}~\lipsum[1][1]\par
\tkzboxnumber[num color=yellow,shadow=false]{4}~\lipsum[1][1]\par
$\ldots$\par
\tkzboxnumber{11}~\lipsum[1][1]\par
\tkzboxnumber{77}~\lipsum[1][1]\par
$\ldots$\par
\tkzboxnumber{697}~\lipsum[1][1]\par
\tkzboxnumber[width auto=false]{697}~\lipsum[1][1]\par
\tkzboxnumber{800}~\lipsum[1][1]\par
$\ldots$\par
\tkzboxnumber{1564}~\lipsum[1][1]\par
\tkzboxnumber[width auto=false]{1564}~\lipsum[1][1]\par
\tkzboxnumber{7124}~\lipsum[1][1]\par
```

1 Lorem ipsum dolor sit amet, consectetuer adipiscing elit.
1 Lorem ipsum dolor sit amet, consectetuer adipiscing elit.
2 Lorem ipsum dolor sit amet, consectetuer adipiscing elit.
4 Lorem ipsum dolor sit amet, consectetuer adipiscing elit.
...
11 Lorem ipsum dolor sit amet, consectetuer adipiscing elit.
77 Lorem ipsum dolor sit amet, consectetuer adipiscing elit.
...
697 Lorem ipsum dolor sit amet, consectetuer adipiscing elit.
697 Lorem ipsum dolor sit amet, consectetuer adipiscing elit.
800 Lorem ipsum dolor sit amet, consectetuer adipiscing elit.
...
1564 Lorem ipsum dolor sit amet, consectetuer adipiscing elit.
1564 Lorem ipsum dolor sit amet, consectetuer adipiscing elit.
7124 Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

## 15.3 Boîtes de 'titres'

### 15.3.1 Titres de sujets d'examens, titres d'exercices

```
\begin{TitreSujet}[Couleur=...,AlignH=...]<options tcbox>{titre onglet}
...
\end{TitreSujet}
```

```
\begin{TitreSujet}[Couleur=red!50!black]{SUJET}
Métropole, SIO, 16 Mai 2024
\end{TitreSujet}
```

SUJET

**Métropole, SIO, 16 Mai 2024**

```
\begin{TitreSujet}[Couleur=teal,AlignH=center]{CORRIGÉ}
Baccalauréat Centres étrangers Groupe 1\\
14 mars 2023
\end{TitreSujet}
```

CORRIGÉ

**Baccalauréat Centres étrangers Groupe 1**
**14 mars 2023**

```
\SujetTitreExo[couleur]{titre}
```

```
\SujetTitreExo{Exercice 4 (5 points)}

\SujetTitreExo[olive]{Exercice 1 [Matrices]\dotfill(5 points)}
```

**Exercice 4 (5 points)**

**Exercice 1 [Matrices]. . . . . (5 points)**

### 15.3.2 Bannière de titre

L'idée est de proposer une bannière, réalisée en Ti*k*Z, pour présenter par exemple un titre.
Il sera ensuite possible de redéfinir un sectionnement du document (`part/section/subsection/...`).
Le style global est fixé, mais des éléments de personnalisations sont possibles (des calculs en interne sont effectués pour adapter la taille des textes).

```
\tkzBannerTri[clés]{numéro}{titre}
```

```
\tkzBannerTri{01}{Titre du document}
```

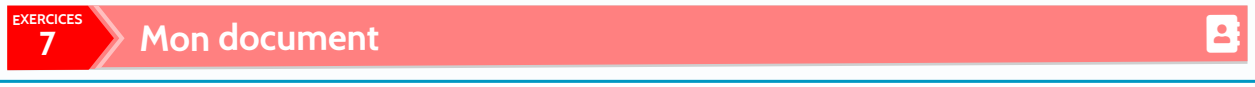
01 Titre du document

Les principales `clés` (anglicisées) sont :

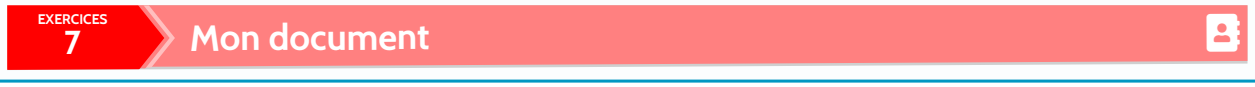
— `height` : hauteur de la bannière (`2.5em` par défaut)
— `width` : largeur de la bannière (`\linewidth` par défaut)
— `blockwidth` : largeur du bloc gauche (`2.75em` par défaut, mais si `auto`, la largeur s'adaptera à son contenu)
— `coltxt` : couleur des textes (`white` par défaut)

— `fonttxt` : police globale des textes (`white` par défaut)
— `swap` : booléen pour changer le style de la partie droite (`false` par défaut)
— `maincolor` : couleur principale, les dégradés étant calculés automatiquement (`darkgray` par défaut)
— `collight` : couleur la plus claire (`darkgray!25` par défaut)
— `colmedium` : couleur du milieu (`darkgray!50` par défaut)
— `coldark` : couleur la plus foncée (`darkgray` par défaut)
— `logo` : logo éventuel, placé tout à droite
— `type` : type éventuel du document, qui est dans le bloc gauche
— `dispblock` : booléen pour afficher le bloc de gauche (actif par défaut)
— `num` : booléen pour afficher le numéro (actif par défaut)
— `customtype` : texte personnalisé éventuel du bloc gauche
— `custommulti` : booléen pour un texte personnalisé éventuel du bloc gauche sur deux lignes

```
\tkzBannerTri
  [maincolor=red,type=EXERCICES,blockwidth=auto,logo=\faAddressBook]
  {7}{Mon document}
```




```
\tkzBannerTri
  [maincolor=red,type=EXERCICES,blockwidth=5em,logo=\faAddressBook]
  {7}{Mon document}
```




```
\tkzBannerTri[%
  maincolor=red,type=EXERCICES,blockwidth=auto,logo=\faAddressBook,swap]
  {07}{Mon document}
```




```
\tkzBannerTri
  [dispblock=false,maincolor=teal,logo=\faSchool]
  {}{Mon document}
```




```
\tkzBannerTri
  [maincolor=olive,customtype=TP,blockwidth=4em,logo=\faAddressBook,height=4em]
  {7}{Mon document}
```




Il est de ce fait *normalement* possible (à vérifier quand même avec une insertion dans une toc...) de modifier le comportement de la commande `\part`, dans l'optique de l'utilisation avec une table des matières notamment.

### 15.3.3 Boîte avec ornements

```
\begin{tcboxornaments}[clés]{options tcbox}
%contenu
\end{tcboxornaments}

%---Les clés disponibles sont :
%v size deco        = hauteur de la déco
%h size deco        = largeur de la déco
%linewidth deco     = épaisseur de la déco
%color deco         = couleur de la décoration
%alt deco           = booléen décoration 'intérieure'
%h stretch deco     = coeff v -> h
%inner size deco    = taille du coin
```

```
\begin{tcboxornaments}{}
\lipsum[1][1-5]
\end{tcboxornaments}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque.

```
\begin{tcboxornaments}
  [color deco=red!50!black,h size deco=3cm]
  {center,width=12cm,fontupper=\LARGE\sffamily,colupper=red!50!black}
CHAPITRE 01 : Éléments de logique
\end{tcboxornaments}
```

CHAPITRE 01 : Éléments de logique

### 15.3.4 Titre 'bicolore'

```
%inspiré d'un document de Joan Riguet
\tkzbicolortitle[clés]<options tikz>{titre ou ligne1§ligne2}

%---Les clés disponibles sont :
%main color     = blue!50!black
%alt color      = yellow!75!orange
%font           = \bfseries\sffamily
%left upper     = {}
%left lower     = {}
%height         = 1.15cm
%left part      = 3cm
%inner sep      = 0.5cm
```

```
\tkzbicolortitle%
  [left upper=CH-01,left lower=Fiche 1]%
  {TEST}
```

CH-01
Fiche 1

TEST

```
\tkzbicolortitle%
  [left upper=CH-01,left lower=Fiche 1]%
  {une semaine pour travailler§mes automatismes}
```

CH-01
Fiche 1

une semaine pour travailler
mes automatismes

```
\tkzbicolortitle%
  [main color=red,alt color=yellow,left upper=CH-01,
  left lower=Fiche 1,inner sep=1.5cm,left part=2cm,height=0.75cm]%
  {une semaine pour travailler§mes automatismes}
```

CH-01
Fiche 1

une semaine pour travailler
mes automatismes

## 15.4 Boîtes de présentation

```
\begin{BoiteSimple}[couleur]<options tcbox>{titre}
  ...
\end{BoiteSimple}
```

```
\begin{BoiteSimple}[red]{Propriété}
Si M est la matrice d'adjacence d'un graphe simple orienté de sommets $A_1$, $A_2$, \dots,
    $A_n$, le nombre de chemins de longueur $p$ d'un sommet $A_i$  à un sommet $A_j$ est le
    nombre situé ligne $i$ et colonne $j$ dans la matrice $M^p$.
\end{BoiteSimple}
```

**Propriété**

Si M est la matrice d'adjacence d'un graphe simple orienté de sommets $A_1$, $A_2$, ..., $A_n$, le nombre de chemins de longueur $p$ d'un sommet $A_i$ à un sommet $A_j$ est le nombre situé ligne $i$ et colonne $j$ dans la matrice $M^p$.

```
\begin{BoiteSimple}[blue]<width=0.75\linewidth,flush right>{Propriété}
Si M est la matrice d'adjacence d'un graphe simple orienté de sommets $A_1$, $A_2$, \dots,
    $A_n$, le nombre de chemins de longueur $p$ d'un sommet $A_i$  à un sommet $A_j$ est le
    nombre situé ligne $i$ et colonne $j$ dans la matrice $M^p$.
\end{BoiteSimple}
```

**Propriété**

Si M est la matrice d'adjacence d'un graphe simple orienté de sommets $A_1$, $A_2$, ..., $A_n$, le nombre de chemins de longueur $p$ d'un sommet $A_i$ à un sommet $A_j$ est le nombre situé ligne $i$ et colonne $j$ dans la matrice $M^p$.

## 15.5 Vignettes

```
\AffVignette(*)[Type=...,Couleur=...,Police=...]{texte}
% la version étoilée active le \NoAutoSpacing
```

```
%vignette de base
\AffVignette{test} ou \AffVignette[Couleur=magenta]{test}
```

test ou test

```
%vignette type algo
\AffVignette[Type=algo]{test} ou \AffVignette[Type=algo,Couleur=teal]{Renvoyer}
```

test ou Renvoyer

```
%vignette type python, classique
\AffVignette[Type=py]{test} ou \AffVignette[Type=py,Couleur=lime]{return}

%vignette type python, avec piton (et lualatex)
%\AffVignette[Type=pypit]{from math import sqrt}

%vignette type python, avec piton et pyluatex (et lualatex + shell-escape)
%\AffVignette[Type=pyl,Couleur=blue]{1+4/5}
```

test ou return

```
%vignette type graphique
\AffVignette[Type=grph]{fonction} ou \AffVignette[Couleur=olive,Type=grph]{intersection}
```

fonction ou intersection

```
%vignette type MPM
\AffVignette[Type=mpm]{marge} ou \AffVignette[Type=mpm,Couleur=orange]{chemin}
```

marge ou chemin

```
%vignette type xcas
\AffVignette[Type=xcas]{calcul formel} ou \AffVignette[Type=xcas,Couleur=brown]{calcul
    formel}
```

calcul formel ou calcul formel

```
%vignette type shell
\AffVignette[Type=shell,Couleur=red!50!orange]{fenêtre cmd}
```

fenêtre cmd

```
%vignette type LaTeX
\AffVignette[Type=tex]{code LaTeX}
```

code LaTeX

```
%vignette type tableur
\AffVignette[Type=sheet,Couleur=green!50!black,Police=\footnotesize\sffamily]
    {cellule A3}
```

cellule A3

```
%vignette type tableur
\AffVignette[Type=perso/CRYPT,Couleur=blue!50!teal,Echelle=0.5]
    {vignette personnalisée}
```

vignette personnalisée

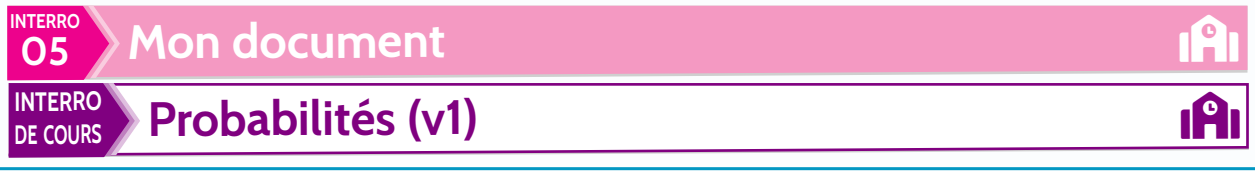
```
%création d'une macro personnelle
\NewDocumentCommand\VignetteTableur{ m }{%
  \AffVignette*[Type=sheet,Couleur=green!50!black,Police=\footnotesize\sffamily]
  {#1}
}
On se place dans la plage \VignetteTableur{A3:B5} pour...
```

On se place dans la plage A3:B5 pour...

À noter qu'il existe une commande *alternative*, avec un fonctionnement légèrement différent, puisque la hauteur de la bannière et la largeur du bloc sont fixés, de sorte que les labels s'adaptent automatiquement.
Les `clés` sont les mêmes, le fonctionnement global également, à l'exception de la clé `[customtype]` en mode `custommulti` qui doit être donné sous la forme `[customtype=L1+L2]`.

```
\tkzBannerTriAlt
  [maincolor=magenta,logo=\faSchool,type=INTERRO]
  {05}{Mon document}

\tkzBannerTriAlt
  [maincolor=violet,swap,logo=\faSchool,customtype={INTERRO+DE
    COURS},custommulti,blockwidth=1.25cm]
  {}{Probabilités (v1)}
```

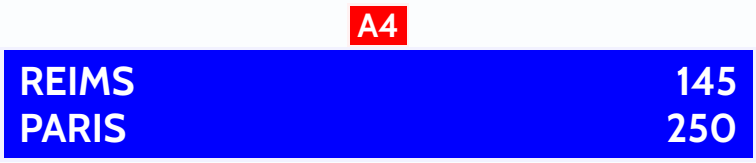


## 15.6 Panneaux autoroutiers

L'idée est de proposer des panneaux type *autoroutes*, réalisés via `tcolorbox`.
Le style global est fixé, mais des éléments de personnalisations sont possibles..

```
\begin{PanneauAutoroute}[clés]{titre}
%contenu
\end{PanneauAutoroute}
```

```
\begin{PanneauAutoroute}{A4}
REIMS\hfill~145\\
PARIS\hfill~250
\end{PanneauAutoroute}
```

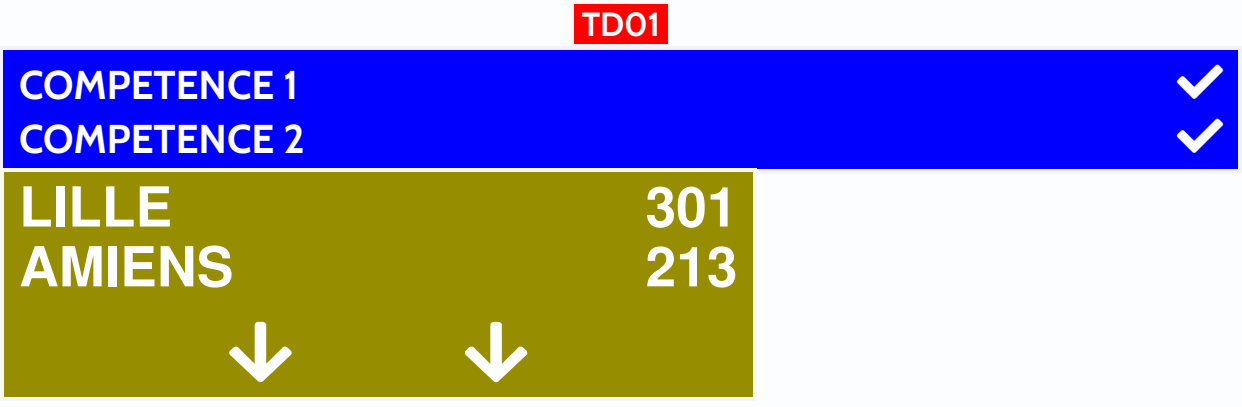



Les clés, optionnelles et entre `[...]` sont :

— `Fleches` : nombre de flèches à afficher (`0` par défaut)
— `CouleurFond` : couleur de fond (`blue` par défaut)
— `CouleurTitre` : couleur de la boîte *titre* (`red` par défaut)
— `Largeur` : largeur du panneau (`10cm` par défaut)
— `Police` : police du panneau
— `PoliceCartouche` : police du titre
— `TypeFleche` : flèche(s) à afficher
— `EspacementV` : interligne (`2.5mm` par défaut)
— `Dernier` : booléen pour préciser que la boîte sera la dernière de la série (pour l'espace après).

```
\begin{PanneauAutoroute}[Largeur=\linewidth,Dernier=false]{TD01}
  \VilleDist{COMPETENCE 1}{\faCheck}  %macro interne pour afficher une ligne
  \VilleDist*{COMPETENCE 2}{\faCheck} %macro pour afficher la dernière ligne
\end{PanneauAutoroute}

\begin{PanneauAutoroute}%
  [CouleurFond=olive,Fleches=2,Police=\bfseries\huge\fontfamily{phv}\selectfont]{}
  LILLE\hfill~301\\
  AMIENS\hfill~213
\end{PanneauAutoroute}
```



## 15.7 Affichettes de soldes

L'idée est de proposer des petites affichettes pour des réductions, réalisées via `tcolorbox` (sur une idée de *Sébastien Lozano*).
Le style global est fixé, mais des éléments de personnalisations sont possibles..

```
\AfficheSoldes[clés]{prix avant}{prix après}{réduction}
```

```
\AfficheSoldes{99\texteuro}{79\texteuro}{-20\%}
```




Les clés, optionnelles et entre `[...]` sont :

- `Couleur` : couleur principale (`gray` par défaut)
- `Largeur` : largeur (`auto` ou par défaut) à définir
- `PoliceEntete` : police du titre
- `PolicePrix` : police des prix
- `PoliceReduc` : police de la réduction
- `Titre` : titre de l'affiche (`GRANDE DÉMARQUE` par défaut)
- `OffsetReduc` : *pente* pour le bandeau de réduction (`1.25cm` par défaut)
- `AgrandirReduc` : augmenter la hauteur de la partie réduction

Donc la hauteur est fixée par *calculs*, et également la largeur si `Largeur=auto`.
La largeur `auto` permet de faire *marcher* des prix jusque `99 999 €`.

```
\AfficheSoldes{99\texteuro}{79\texteuro}{-20\,\%}~%
\AfficheSoldes[Couleur=red]{99\texteuro}{79\texteuro}{-20\,\%}~%
\AfficheSoldes[Couleur=blue]{\ldots\texteuro}{79\texteuro}{-20\,\%}~%
\AfficheSoldes[Couleur=teal]{\ldots}{\ldots}{\ldots}
```



```
\AfficheSoldes[%
    Couleur=magenta,%
    Largeur=6cm,%
    PoliceEntete=\huge\sffamily\bfseries,%
    PolicePrix=\ttfamily,%
    PoliceReduc=\bfseries\Huge\sffamily,%
    AgrandirReduc=8mm
    ]%
  {1\,500\texteuro}{1\,299\texteuro}{-\ldots\,\%}
```




```
\AfficheSoldes[%
    Couleur=cyan,%
    PoliceEntete=\bfseries\huge\fontfamily{qhv}\selectfont,%
    PolicePrix=\fontfamily{qhv}\selectfont,%
    PoliceReduc=\bfseries\Huge\fontfamily{qhv}\selectfont,%
    AgrandirReduc=1cm
    ]%
  {\makebox[1cm]{\dotfill}\texteuro}{1\,299\texteuro}{-30\,\%}
```



## 15.8 Boîtes diverses

### 15.8.1 Boîte type message de forum

```
\begin{tcforumpost}[clés]{options tcbox}
%contenu
\end{tcforumpost}

%---Les clés disponibles sont :
%height-pseudo  = hauteur pseudo/alt pseudo
%deco-length    = largeur de la déco (triangle)
%left-margin    = marge gauche (pour la boîte avatar/pseudo...)
%pseudo         = pseudo affiché
%alt-pseudo     = pseudo alternatif affiché
%fill-color     = couleur de fond
%rule-color     = couleur du bord
%left           = booléen pour passer en mode avatar à droite
%swap-title    = booléen pour échanger les lignes du bloc avatar
%avatar         = code avatar (emoji, graphics, texte...)
```

```
\begin{tcforumpost}{}
\lipsum[1][1-5]
\end{tcforumpost}
```




Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque.

```
%style des labels (en l3e)
\ExplSyntaxOn
\tikzset{
  forum~style~post~font~pseudo/.style={%
    inner~sep=0pt,font=\ttfamily\bfseries,text=white},
  forum~style~post~font~altpseudo/.style={%
    inner~sep=0pt,font=\ttfamily,text=\g_tcbox_forum_post_rule_color},
}
\ExplSyntaxOff

\begin{tcforumpost}%
  [%
    fill-color=pink!10,rule-color=magenta,%
    right,swap-title
  ]%
  {width=12cm}
\lipsum[1][1-5]
\end{tcforumpost}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque.



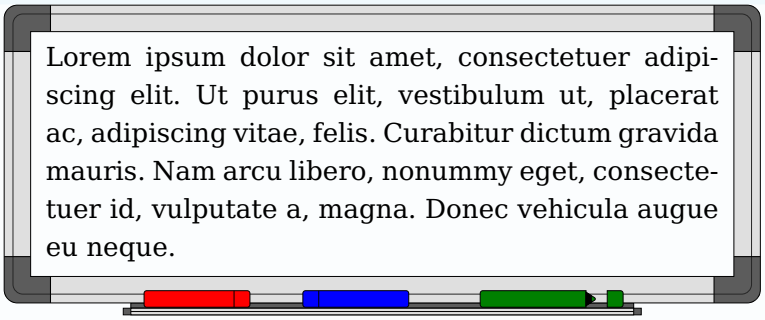
### 15.8.2 Boîte type 'tableau blanc'

```
%inspiré d'un document de Joan Riguet
\begin{tcboxwhiteboard}[clés]{options tcbox}
%contenu
\end{tcboxwhiteboard}

%---Les clés disponibles sont :
%width         = 10cm
%border        = 3.5mm
%border color  = lightgray!50
%alt color     = darkgray
%pens          = true
%bottom        = true
%left align    = true
```

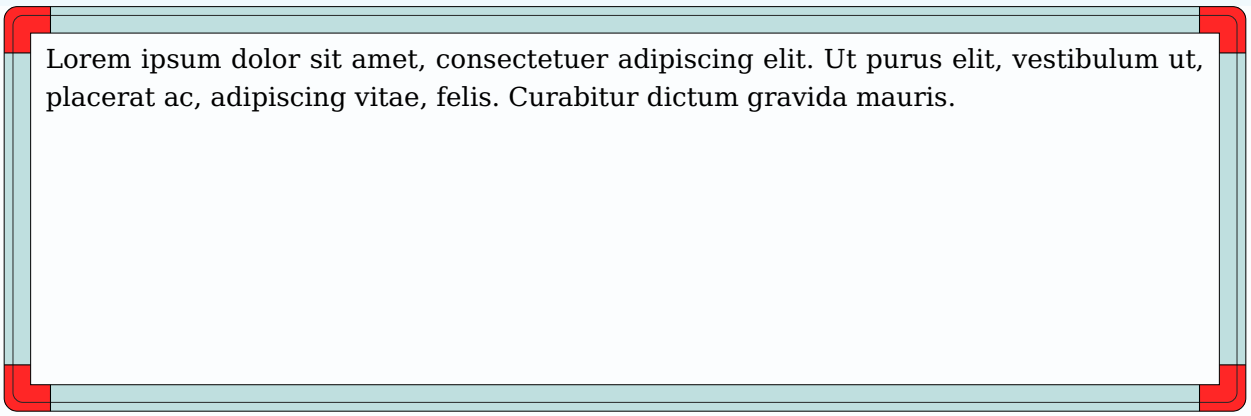
```
\begin{tcboxwhiteboard}{}
\lipsum[1][1-5]
\end{tcboxwhiteboard}

\medskip
```




```
\begin{tcboxwhiteboard}%
  [border color=teal!25,alt color=red,width=\linewidth,bottom=false]%
  {height=5cm}
\lipsum[1][1-3]
\end{tcboxwhiteboard}

\medskip
```



### 15.8.3 Boîte type 'feuille perforée'

```
%boîte
\begin{tcboxnotebook}[clés]{options tcbox}
%contenu
\end{tcboxnotebook}

%---Les clés disponibles sont :
% orientation = orientation de la boîte (V)
% bar         = affichage de la barre, en mode V (false)
% bg          = couleur de fond (white)
```

```
%formatage du titre
\NewDocumentCommand\tcboxnotebooktitle{ O{blue!50!black} m }{%
  \hspace{2.5mm}%
  \textcolor{#1}{%
    \bfseries\large\sffamily{\vphantom{(qÉ)}#2}%
  }\par\smallskip
}
```

```
\begin{tcboxnotebook}[orientation=H]{width=10cm}
\tcboxnotebooktitle{Rappels utiles}
\lipsum[1]
\end{tcboxnotebook}
```

**Rappels utiles**

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

```
\begin{tcboxnotebook}[bg=violet!25]{}
\tcboxnotebooktitle{Rappels utiles}
\lipsum[1]
\end{tcboxnotebook}
```

**Rappels utiles**

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

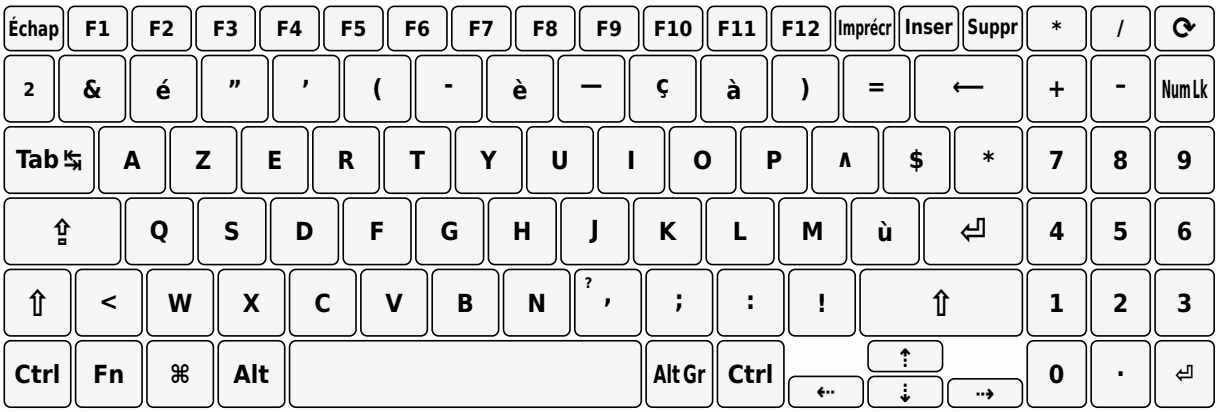
# 16 Clavier français

## 16.1 Introduction

L'idée est de proposer une commande pour afficher un clavier (type `fr 15'`), avec éventuellement des touches mises en valeur.

```
\ClavierFrancais[clés]<options tikzpicture{touches}
```

```
\ClavierFrancais{}
```




## 16.2 Liste des touches

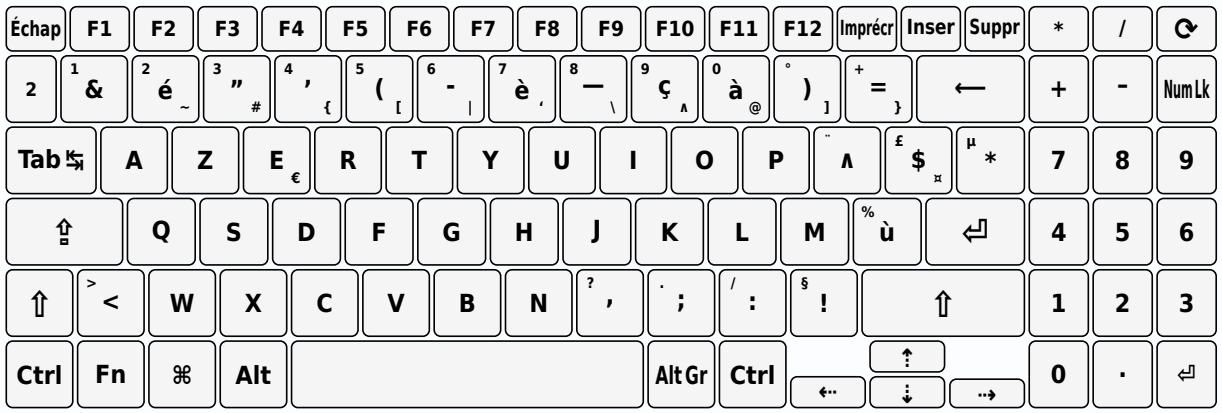
Les touches disponibles sont :
`echap,F1,F2,F3,F4,F5,F6,F7,F8,F9,F10,F11,F12,printscreen,inser,suppr,NUMstar,NUMslash,off`
`square,N1,N2,N3,N4,N5,N6,N7,N8,N9,N0,rparen,equal,backspace,NUMplus,NUMminus,numlock`
`tab,A,Z,E,R,T,Y,U,I,O,P,circum,dollar,star,NUM7,NUM8,NUM9`
`capslock,Q,S,D,F,G,H,J,K,L,M,ugrave,enter,NUM4,NUM5,NUM6`
`leftmaj,geqleq,W,X,C,V,B,N,comma,semicolon,colon,excla,rightmaj,NUM1,NUM2,NUM3`
`leftctrl,fn,win,alt,space,altgr,rightctrl,NUM0,NUMdot,NUMenter,left,right,down,up`

## 16.3 Clés de personnalisations

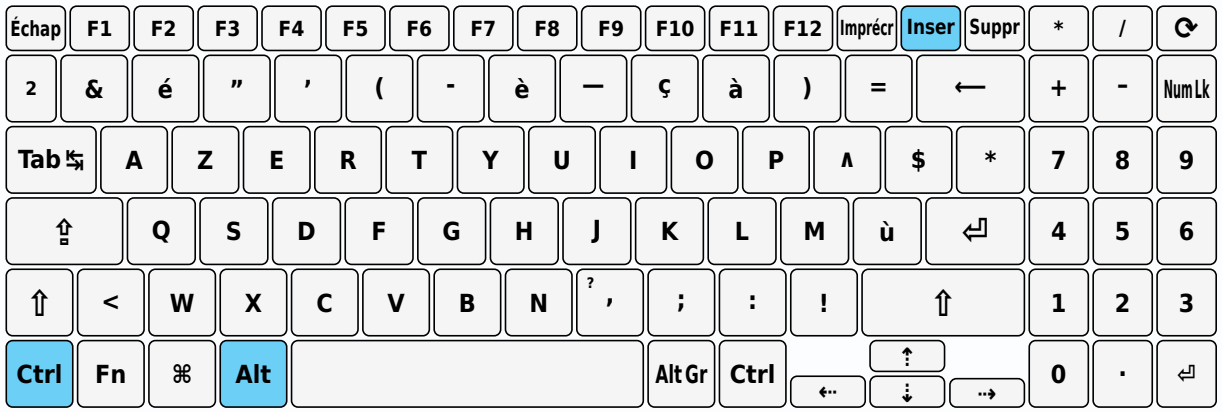
Les `clés` disponibles, à donner entre `[...]`, sont :

- `Largeur=...` : largeur du clavier (`16cm` par défaut)
- `CouleurBordure=...` : bordure des touches
- `CouleurFond=...` : couleur de fond des touches
- `CouleurSelect=...` : couleur des touches sélectionnées
- `AfficheTouches=TF` : booléen pour afficher les étiquettes des touches
- `Complet=TF` : booléen pour afficher les étiquettes complètes.

```
\ClavierFrancais[Complet]{}
```




```
\ClavierFrancais{leftctrl,alt,inser}
```



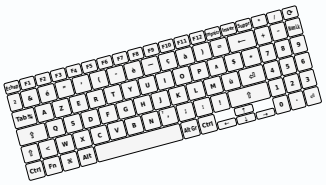

```
\ClavierFrancais[%
  Largeur=10cm,CouleurBordure=violet,%
  CouleurFond=orange!10,CouleurSelec=blue!50%
  ]{altgr,N4}
```




```
\ClavierFrancais[Largeur=6cm]{}
```



```
\ClavierFrancais[Largeur=4cm]<rotate=15>{}
```



# 17 Patchs divers

## 17.1 Gérer l'espacement vertical du mode hors-ligne

L'idée est de proposer une commande, à insérer dans le préambule, pour *patcher* l'espacement vertical proposé par les environnements mathématiques hors-ligne.

```
...
\ChangerDisplaySkip[clés]
...
\begin{document}
```

Les `clés` disponibles, à donner entre `[...]`, sont :

— `Avant=...` : modification de l'espacement avant ;

— `AvantS=...` : modification de l'espacement avant (version *short*) ;

— `Apres=...` : modification de l'espacement après ;

— `ApresS=...` : modification de l'espacement après (version *short*) ;

— `Global=...` : modification globale de **tous** les espacements ;

— `Dense` : booléen pour modification globale en `0.33\baselineskip`.

## 17.2 Quelques ajouts à pas-tableur (customenvs-pastableur, 0.1.2)

L'idée est de proposer quelques commandes complémentaires pour le package pas-tableur, de Stéphane Pasquet (https://ctan.org/pkg/pas-tableur), en respectant autant que faire se peut la syntaxe du package (voir la documentation pour de plus amples informations).

Certains patchs/ajouts ont été intégrés à la version `3.0.1` du package `pas-tableur`, du 5 septembre 2026, donc elles ne sont plus dans le sous-package `customenvs-pastableur`.

À noter que le sous-package `customenvs-pastableur` peut être chargé de manière autonome.

Le package pas-tableur n'est pas chargé par défaut, mais il peut l'être grâce à l'option `[pastableur]`.

Les *ajouts* disponibles sont :

— fusionner des celules, via `\celfusion` ;

— *casser la numérotation* avec des pointillés, via `\celnumbreak` ;

```
%chargement du package
\usepackage[pastableur]{customenvs} %via customenvs
\usepackage{customenvs-pastableur}  %autonome

%fusionner des cellules
\celfusion[options pas-tableur]{col-lig}{col-lig}{label}
%coin
\coinTableur
%barre de formule
\barreFormule[hauteur]{référence cellule}{dernière colonne}{contenu formule}
%pointillés dans la numérotation
\celnumbreak{num ligne pointillés}{début numéro suite}{fin numéro suite}
```

L'exemple suivant illustre les différentes commandes proposées dans ce package.

```
\begin{tikzpicture}{}
  \tableur*[10]{A/2.5cm,B/2.5cm,C/2.5cm,D/2.5cm,E/2.5cm,F/2.5cm}
  %ligne n°1
  \celtxt*[align=center]{A}{1}{$n$}
  \celfusion[align=center]{B-1}{C-1}{$M_n$}
  \celfusion[align=center]{D-1}{E-1}{$S_n$}
  \celtxt*[align=center]{F}{1}{$d_n$}
  %Ligne2/3/4
  \celcouleur{A-2}
  \lignetxt*[align=center]{2}<2>{$x_n$,$y_n$,5,$n$,$n$} %à partir de la colonne n°2
  \lignetxt*[align=center]{3}{0,0,0,5,0,5}
  \lignetxt*[align=center]{4}{1,1,0,5,1,4.12310563}%
  \celnumbreak{8}{27}{28}
  %colonneA
  \colonnetxt*[align=center]{A}<5>{2,3,4,\dots,24,25}   %à partir de la ligne n°5
\end{tikzpicture}
```

| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | $n$ | $M_n$ | | $S_n$ | | $d_n$ |
| 2 | | $x_n$ | $y_n$ | 5 | $n$ | $n$ |
| 3 | 0 | 0 | 0 | 5 | 0 | 5 |
| 4 | 1 | 1 | 0 | 5 | 1 | 4.12310563 |
| 5 | 2 | | | | | |
| 6 | 3 | | | | | |
| 7 | 4 | | | | | |
| ... | ... | | | | | |
| 27 | 24 | | | | | |
| 28 | 25 | | | | | |

Un autre exemple :

```
\storewidthtolength[4mm]{20\,000,00\,}{\pastableurcolA}%
\storewidthtolength[4mm]{\bfseries Capital restant dû}{\pastableurcolB}%
\storewidthtolength[4mm]{\bfseries Mensualité}{\pastableurcolC}%
\storewidthtolength[4mm]{\bfseries Mensualités}{\pastableurcolD}%
\storewidthtolength[4mm]{\bfseries Amortissement}{\pastableurcolE}%
\begin{tikzpicture}
  \tabnumlinewidth{1.57em}
  \gentableur[6]%avec des hauteurs de lignes variables
    {A/\pastableurcolA,%
     B/\pastableurcolB,%
     C/\pastableurcolC,%
     D/\pastableurcolD,%
     E/\pastableurcolE}%
    [2em,2.15em,1.85em,3em,2em,1.75em]
  %personnalisations
  \coinTableur
  \barreFormule{C5}{C}{=\$B\$5*\$B\$2}
  %fond
  \plagecouleur[lightgray]{A-1}{C-1}
  \plagecouleur[lightgray]{A-4}{E-4}
  %bordures
  \plagebordures{A-1}{C-2}
  \plagebordures{A-4}{E-6}
  %cellules tableau 1
  \celtxt[align=center,font={\bfseries},width=\pastableurcolA]{A}{1}{Emprunt}
  \celtxt[align=center,font={\bfseries},width=\pastableurcolB]{B}{1}{Taux mensuel}
  \celtxt[align=center,font={\bfseries},width=\pastableurcolC]{C}{1}{Mensualité}
  \celtxt[align=center,width=\pastableurcolA]{A}{2}{20\,000,00\,}
  \celtxt[align=center,width=\pastableurcolB]{B}{2}{0,40\,\%}
  \celtxt[align=center,width=\pastableurcolC]{C}{2}{325,59\,}
  %cellules tableau 2
  \celtxt[align=center,font={\bfseries},width=\pastableurcolA]{A}{4}{Mois}
  \celtxt[align=center,font={\bfseries},width=\pastableurcolB]{B}{4}{Capital restant dû en
    début de mois}
  \celtxt[align=center,font={\bfseries},width=\pastableurcolC]{C}{4}{Intérêts mensuels}
  \celtxt[align=center,font={\bfseries},width=\pastableurcolD]{D}{4}{Mensualités constantes}
  \celtxt[align=center,font={\bfseries},width=\pastableurcolE]{E}{4}{Amortissement}
  \lignetxt[align=center]{5}{1,{20\,000,00\,},{80,00\,},{375,59\,},{295,59\,}}
  \lignetxt[align=center]{6}{2,{19\,704,41\,},{78,82\,},{375,59\,},{296,78\,}}
  %cellules sélectionnées
  \multiSelec{C-5}{D-6}
  \renumLine{3}{...}
\end{tikzpicture}
```

C5 | =$B$5*$B$2

| | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | **Emprunt** | **Taux mensuel** | **Mensualité** | | |
| 2 | 20 000,00 € | 0,40 % | 325,59 € | | |
| ... | | | | | |
| 4 | **Mois** | **Capital restant dû en début de mois** | **Intérêts mensuels** | **Mensualités constantes** | **Amortissement** |
| 5 | 1 | 20 000,00 € | 80,00 € | 375,59 € | 295,59 € |
| 6 | 2 | 19 704,41 € | 78,82 € | 375,59 € | 296,78 € |

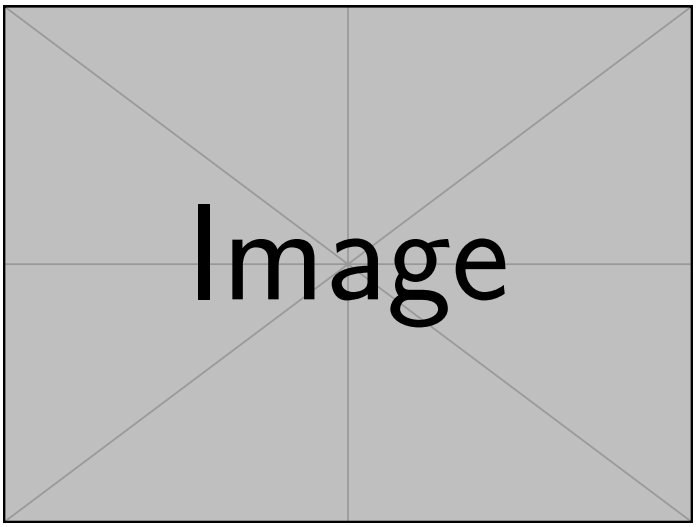
## 17.3 Insertion d'images avec remplissage de l'espace vertical (customenvs-macros)

L'idée est de proposer une commande pour adapter la hauteur d'une image à la hauteur restante en base de page.
Cela repose en fait simplement sur l'utilisation de `[height=\dimexpr\pagegoal-\pagetotal\relax]`.

```
\InsererImage<vRemplir=true/false,vOffset=...,vCentrer=...>[clés includegraphics]<image>
```

La clé `vOffset` permet de spécifier une longueur à retirer au calcul automatique (par défaut elle est fixée à `0pt`)..
La clé `vCentrer` (parmi `hg/h/hd/g/m/d/bg/b/db/n` permet de spécifier un type de centrage H-V, si besoin, parmi `milieu/haut/bas/non` (`n` par défaut).

```
\InsererImage<vRemplir,vOffset=30mm,vCentrer=m>{example-image}
```



## 17.4 Grille automatique pour les environnements tikzpicture

L'idée est de proposer de quoi afficher, en *surimpression,* une grille automatique sur une figure `tikzpicture`.
Cela peut-être utile dans la construction de graphes ou de figures géométriques pour placer plus facilement des objets.

Il est possible de :

— demander ponctuellement l'affichage avec une macro à placer en fin d'environnement ;

— automatiser l'affichage pour une figure avec une `clé` et une macro locale ;

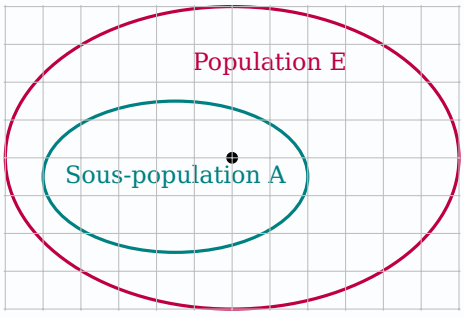
— automatiser l'affichage pour toutes les figures avec une macro globale et une `clé`.

### Version manuelle

```
%macro locale, utilisation ponctuelle
\begin{tikzpicture}
  %codes
  \tkzGrilleAuto(*){pasX§pasY}
\end{tikzpicture}
```

La version étoilée marque également l'origine $(0;0)$.
L'argument obligatoire permet de spécifier les pas (H§V) pour la grille.

```
\begin{tikzpicture}
  %population
  \draw[very thick,purple] (0,0) circle[x radius=3cm,y radius=2cm] ;
  \draw[purple] (0.5,1.25) node[font=\small] {Population E} ;
  %ss population
  \draw[very thick,teal] (-0.75,-0.25) circle[x radius=1.75cm,y radius=1cm] ;
  \draw[teal] (-0.75,-0.25) node[font=\small] {Sous-population A} ;
  %grille auto manuelle, avec origine
  \tkzGrilleAuto*{0.5§0.5}
\end{tikzpicture}
```

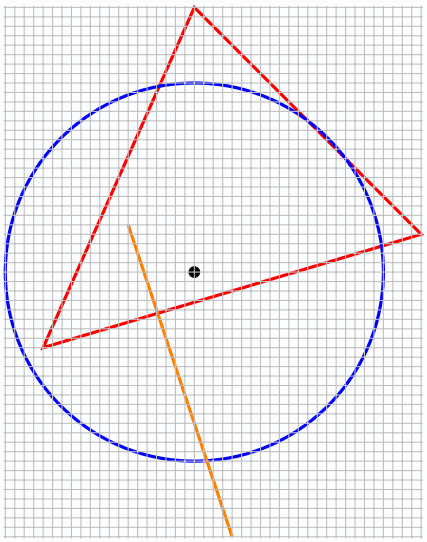



### Version automatique locale, avec clé

```
%clé locale, utilisation ponctuelle
\begin{tikzpicture}[grilleauto=pasX§pasY]
  \tkzAutoGridLocal(*)
  %codes
\end{tikzpicture}
```

La version étoilée marque également l'origine $(0;0)$.

```
\begin{tikzpicture}[x=0.5cm,y=0.5cm,grilleauto=0.25§0.25]
  %grille (qui sera affichée en dernier)
  \tkzAutoGridLocal*
  %triangle
  \draw[very thick,red] (-4,-2)--(0,7)--(6,1)--cycle ;
  %cercle
  \draw[very thick,blue] (0,0) circle[radius=5] ;
  %segment
  \draw[very thick,orange] (-1.75,1.25)--(1,-7);
\end{tikzpicture}
```



### Version automatique globale, avec clé

```
%préambule, la commande sera active pour tous les environnements tikz
\tkzAutoGridActivate(*)

%document
\begin{tikzpicture}[grilleauto=pasX§pasY]
  %codes
\end{tikzpicture}
```

La version étoilée marquera également l'origine $(0;0)$.

## 17.5 Flèche évasée, en TikZ

L'idée est de proposer une commande, en TikZ, pour représenter une flèche évasée, avec des paramètres optionnels.

```
\tkzFlecheEvasee[clés]{départ}{arrivée ou déplacement}
```

```
\begin{tikzpicture}
  \tkzFlecheEvasee{0,0}{5,1.5}
\end{tikzpicture}
```



Les `clés` disponibles, à donner entre `[...]`, sont :

- `Couleur=...` : couleur de la flèche ;
- `Epaisseur=...` : taille initiale du trait (`0.5mm` par défaut) ;
- `Direction=...` : direction de la flèche, avec angle éventuel ;
- `Coeff=...` : coefficient pour la taille finale du trait (`4` par défaut) ;
- `TypeFleche=...` : type de la flèche, issue de `arrows.meta` ;
- `TailleFleche=...` : booléen pour la taille de la partie flèche (`auto` par défaut), en langage `arrows.meta` ;
- `Deplacement` : booléen pour spécifier qu'on travaille avec un déplacement pour l'arrivée.

Le premier argument obligatoire, permet de spécifier le nœud de départ ou les coordonnées du point de départ.
Le second argument obligatoire, permet de spécifier le nœud d'arrivée ou les coordonnées du point d'arrivée, mais également le déplacement dans le cas ou `Deplacement=true`.

Par défaut, et en mode `auto` :

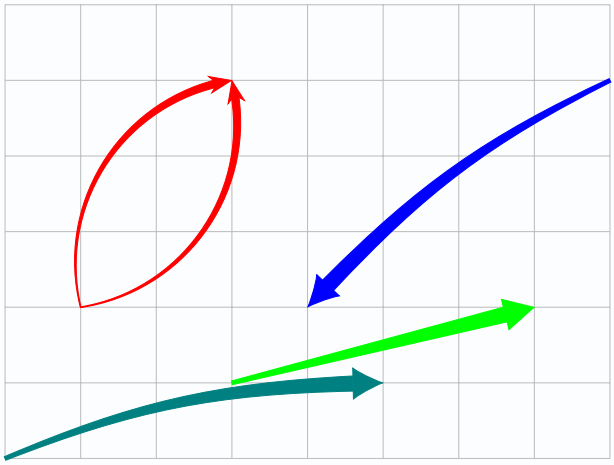
- le début du trait mesure `Epaisseur=0.5mm` ;
- la fin du trait mesure `4*Epaisseur=2mm` ;
- les dimensions de la flèche sont `H=2*4*Epaisseur=4mm` et `L=2.1*4*Epaisseur=4.2mm`.

À noter que pour les flèches type `Stealth`, un ajustement est sans doute à prévoir pour compenser le *creux*.

```
%flèche 0.5mm -> 1.25mm
\begin{tikzpicture}
  \tkzFlecheEvasee%
    [Epaisseur=0.5mm,Coeff=2.5,Direction=left/30,Couleur=red,TypeFleche=Triangle]%
    {0,0}{5,1.5}
\end{tikzpicture}
```



```
\begin{tikzpicture}
  \draw[thin,lightgray] (-3,-1) grid (5,5) ;
  \coordinate (A) at (0,0) ; \coordinate (B) at (4,1) ;
  \coordinate (C) at (1,1) ; \coordinate (D) at (5,4) ;
  \coordinate (E) at (0,1) ; \coordinate (F) at (0,5) ;
  \coordinate (G) at (-2,0) ;
  \tkzFlecheEvasee[Couleur=green,TypeFleche=Triangle]{A}{B}
  \tkzFlecheEvasee[Couleur=blue,Direction=right/10]{D}{C}
  \tkzFlecheEvasee%
    [Couleur=red,Direction=left/45,TypeFleche=Stealth,Epaisseur=0.1mm,Coeff=10]%
    {-2,1}{0,4}
  \tkzFlecheEvasee%
    [Couleur=red,Direction=right/45,Epaisseur=0.1mm,Coeff=10,TypeFleche=Stealth]%
    {-2,1}{0,4}
  \tkzFlecheEvasee[Couleur=teal,Deplacement,Direction=left/10]{-3,-1}{5,1}
\end{tikzpicture}
```

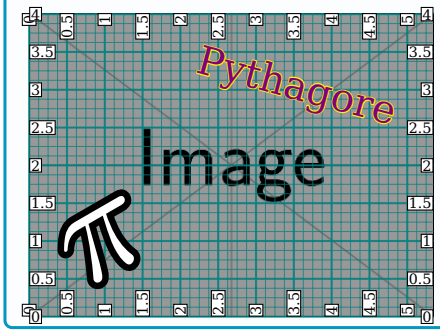


## 17.6 Annoter une image

L'idée est de proposer de quoi annoter une image, en utilisant un environnement et une commande qui sont liés à Ti*k*Z.

```
\begin{AnnoterImage}[clés][options includegraphics]{image}
  \PlacerTxtSurImg[options node tikz]{coordonnées}{texte}
  \PlacerTxtSurImg*[options node tikz]{coordonnées en pourcentage}{texte}
  %code tikz
\end{AnnoterImage}
```

```
%\usepackage[auto,outline]{contour}
\begin{AnnoterImage}[Grille=0.5][height=4cm]{example-image.png}
  \PlacerTxtSurImg[scale=5,rotate=30]
    {1,1}{\contourlength{0.05em}\color{white}\contour{black}{$\pi$}}
  \PlacerTxtSurImg*[scale=1.5,rotate=-15]
    {0.66,0.75}{\contourlength{0.025em}\color{violet}\contour{yellow}{Pythagore}}
\end{AnnoterImage}
```




Concernant l'environnement, les `clés` disponibles (premier argument), à donner entre `[...]`, sont :

— `Noeud=...` : nom du nœud créé (si réutilisation avec un `remember picture`) ;

— `Grille=...` : Grille éventuelle d'aide (en unité) (`vide` par défaut) ;

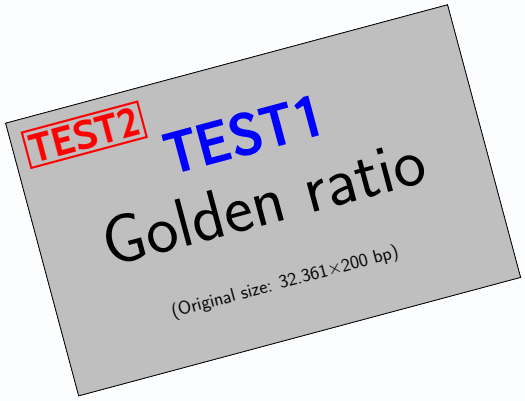
— `SousGrille=...` : sous-grille éventuelle (`4` par défaut) ;

— `CouleurGrille=...` : couleur de la grille d'aide (`teal` par défaut) ;

— `Clip=...` : booléen pour fixer les annotations à l'intérieur de l'image (`true`).

Le deuxième argument, optionnel et entre `[...]`, correspond aux options spécifiques liées à `include graphics`.
Le fichier est à préciser avec sont extension, car le code se charge de vérifier que le fichier existe.

```
\begin{AnnoterImage}[Noeud=TESTIMG][height=3.75cm]<rotate=15>{example-image-golden.pdf}
  %commandes manuelles
  \draw (TESTIMG.center) node[above=5mm,font=\Huge\ttfamily\bfseries,text=blue] {TEST1} ;
  \draw (TESTIMG.north west) node[draw,thick,red,inner sep=0.5mm,below
    right=2.5mm,font=\LARGE\sffamily\bfseries,text=red] {TEST2} ;
\end{AnnoterImage}
```



## 17.7 Travailler sur des longueurs (customenvs-macros)

L'idée est de proposer de quoi déterminer la plus grande largeur d'une liste de textes. Il est ainsi possible (même si le but des polices à chasses non fixes n'est pas celui-là...) d'*aligner* verticalement des textes de présentation par exemple.

```
\getwideststring[macro de stockage]{elt1,elt2,...eltn}

\halignmakebox[alig option]{elt}{list of elements}
```

```
%récupération de la plus grande largeur (\tmpwideststring par défaut)
\getwideststring{Exercice 1,Évaluation 2,Test n°3}\the\tmpwideststring
```

62.80528pt

```
\sffamily\Large
Exercice 1 (10 points)\\
Évaluation 2 (8 points) \\
Test n°3 (4 points)
```

Exercice 1 (10 points)
Évaluation 2 (8 points)
Test n°3 (4 points)

```
%avec alignement
\sffamily\Large
\halignmakebox[l]{Exercice 1}{Exercice 1,Évaluation 2,Test n°3}
(\halignmakebox[r]{10}{10,8,4} points)

\halignmakebox[l]{Évaluation 2}{Exercice 1,Évaluation 2,Test n°3}
(\halignmakebox[r]{8}{10,8,4} points)

\halignmakebox[l]{Test n°3}{Exercice 1,Évaluation 2,Test n°3}
(\halignmakebox[r]{4}{10,8,4} points)
```

Exercice 1 (10 points)
Évaluation 2 ( 8 points)
Test n°3 ( 4 points)

À noter que des commandes sont disponibles pour stocker des dimensions d'une *boîte* dans une macro (existante ou non) et éventuellement lui rajouter une valeur.

```
%largeur
\storewidthtolength[delta]{box}{\macro}
%hauteur
\storeheighttolength[delta]{box}{\macro}
%hauteur totale
\storetotalheighttolength[delta]{box}{\macro}
%profondeur
\storedepthtolength[delta]{box}{\macro}
```

```
\def\tmpbox{\large $1+\frac{1}{x}$}
%
\storewidthtolength{\tmpbox}{\mytmpboxwidth}
\verb|largeur -> \the\mytmpboxwidth        | : \the\mytmpboxwidth

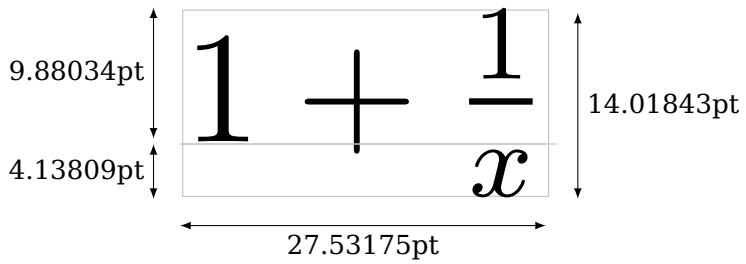
\storewidthtolength[10pt]{\tmpbox}{\mytmpboxwidthdelta}
\verb|largeur+ -> \the\mytmpboxwidthdelta  | : \the\mytmpboxwidthdelta

\storeheighttolength{\tmpbox}{\mytmpboxheight}
\verb|hauteur -> \the\mytmpboxheight       | : \the\mytmpboxheight

\storetotalheighttolength{\tmpbox}{\mytmpboxtotheight}
\verb|hauteur tot -> \the\mytmpboxtotheight| : \the\mytmpboxtotheight

\storedepthtolength{\tmpbox}{\mytmpboxdepth}
\verb|profondeur -> \the\mytmpboxdepth     | : \the\mytmpboxdepth
```

```
largeur -> \the\mytmpboxwidth         : 27.53175pt
largeur+ -> \the\mytmpboxwidthdelta   : 37.53175pt
hauteur -> \the\mytmpboxheight        : 9.88034pt
hauteur tot -> \the\mytmpboxtotheight : 14.01843pt
profondeur -> \the\mytmpboxdepth      : 4.13809pt
```




Il est également possible de déterminer la hauteur (totale ou non) maximale de boîtes différentes.

```
%plus grande hauteur (sans profondeur pour la version étoilée)
\getheighestring(*)[\macro]{liste d'éléments}
```

```
%boîtes individuelles
\storetotalheighttolength{\large\sffamily qB}{\malongueurA}\mytightbox{\large\sffamily qB}
    := \the\malongueurA

\storetotalheighttolength{\huge Éducation}{\malongueurB}\mytightbox{\huge Éducation} :=
    \the\malongueurB

\storetotalheighttolength{$\tfrac{1}{x}$}{\malongueurC}\mytightbox{$\tfrac{1}{x}$} :=
    \the\malongueurC

%macro globale
\getheighestring[\maplusgrandelongueur]{\large\sffamily qB,\huge Éducation,$\tfrac{1}{x}$}%
\mytightbox{\large\sffamily qB}\mytightbox{\huge Éducation}\mytightbox{$\tfrac{1}{x}$} :=
    \the\maplusgrandelongueur
```

qB := 9.91724pt
Éducation := 17.69522pt
$\frac{1}{x}$ := 13.24292pt
qBÉducation$\frac{1}{x}$ := 17.69522pt

Il est, pour finir, possible d'*adapter* du texte à une *boîte* de dimensions fixées, en modifiant la largeur ou la hauteur de manière à optimiser la taille (sans modifier l'aspect).

```
%version étoilée avec affichage (en tikz) de la boîte support
\fittexttobox(*){texte}{largeur}{hauteur}
```

```
%avec les boîtes
\fittexttobox*{TÉLÉPHONE}{2cm}{1cm}\\
\fittexttobox*{\bfseries\sffamily TÉLÉPHONE}{7cm}{1cm}\\
\fittexttobox*{TÉLÉPHONE}{3cm}{1cm}\\
\fittexttobox*{\ttfamily TÉLÉPHONE}{3cm}{1cm}\\
\fittexttobox*{TÉLÉPHONE}{2cm}{2cm}\\
\fittexttobox*{COMPLIMENTS}{10cm}{3.5cm}\\
\fittexttobox*{COMPLIMENTS}{14cm}{1.25cm}
```



```
%sans les boîtes
\fittexttobox{TÉLÉPHONE}{2cm}{1cm}\\
\fittexttobox{\bfseries\sffamily TÉLÉPHONE}{7cm}{1cm}\\
\fittexttobox{TÉLÉPHONE}{3cm}{1cm}\\
\fittexttobox{\ttfamily TÉLÉPHONE}{3cm}{1cm}\\
\fittexttobox{TÉLÉPHONE}{2cm}{2cm}\\
\fittexttobox{COMPLIMENTS}{10cm}{3.5cm}\\
\fittexttobox{COMPLIMENTS}{14cm}{1.25cm}
```

