

# customenvs [fr]

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

Version 0.3.6 - 10 mai 2025

Cédric Pierquet

c pierquet - at - outlook . fr

<https://forge.apps.education.fr/pierquetcedric/packages-latex>

---

## Table des matières

<b>1</b>	<b>Historique</b>	<b>3</b>
<b>2</b>	<b>Le package customenvs</b>	<b>4</b>
2.1	Idée . . . . .	4
2.2	Chargement . . . . .	4
<b>3</b>	<b>Présentation de réponses à un QCM</b>	<b>5</b>
3.1	Principe . . . . .	5
3.2	Exemples . . . . .	5
<b>4</b>	<b>Environnement Centrage</b>	<b>7</b>
4.1	Principe . . . . .	7
4.2	Exemples . . . . .	7
<b>5</b>	<b>Environnement multi-colonnes</b>	<b>9</b>
5.1	Principe . . . . .	9
5.2	Exemples . . . . .	9
<b>6</b>	<b>Énumération avec choix des items, parmi une liste</b>	<b>11</b>
6.1	Principe et fonctionnement . . . . .	11
6.2	Exemples . . . . .	11
<b>7</b>	<b>Tableau de compétences</b>	<b>13</b>
7.1	Principe et fonctionnement . . . . .	13
7.2	Éléments prédéfinis . . . . .	13
7.3	Exemples . . . . .	14
<b>8</b>	<b>Crayon de compétences</b>	<b>17</b>
8.1	Principe et fonctionnement . . . . .	17
8.2	La commande . . . . .	17
8.3	Exemples . . . . .	18
<b>9</b>	<b>Roue de compétences</b>	<b>20</b>
9.1	Principe et fonctionnement . . . . .	20
9.2	La commande et l'environnement . . . . .	20
9.3	Exemples . . . . .	21
9.4	« Speed-o-meter » . . . . .	22

<b>10 Barres de niveau</b>	<b>23</b>
10.1 Principe et fonctionnement . . . . .	23
10.2 La commande . . . . .	23
<b>11 PictoCible</b>	<b>24</b>
11.1 Principe et fonctionnement . . . . .	24
11.2 La commande . . . . .	24
<b>12 Bandeau de score</b>	<b>26</b>
12.1 Principe et fonctionnement . . . . .	26
12.2 La commande . . . . .	26
<b>13 Fenêtre type conversation instantanée</b>	<b>27</b>
13.1 Principe et fonctionnement . . . . .	27
13.2 L'environnement . . . . .	27
13.3 Les commandes de création des bulles . . . . .	27
13.4 Exemples . . . . .	28
<b>14 Titre d'exercices</b>	<b>30</b>
14.1 Principe et définition du style global . . . . .	30
14.2 Options de personnalisations . . . . .	30
14.3 Exemples . . . . .	32
<b>15 Boîtes divers</b>	<b>35</b>
15.1 Introduction . . . . .	35
15.2 Boîtes de présentation . . . . .	35
15.3 Titres de sujets d'examens, titres d'exercices . . . . .	35
15.4 Numéros encerclés . . . . .	36
15.5 Vignettes . . . . .	36
15.6 Boîte arrondie, petite boîte type marker . . . . .	38
15.7 Bannière de titre . . . . .	38
15.8 Panneaux autoroutiers . . . . .	41
15.9 Affichettes de soldes . . . . .	42
<b>16 Patches divers</b>	<b>44</b>
16.1 Gérer l'espacement vertical du mode hors-ligne . . . . .	44
16.2 Quelques ajouts au package pas-tableur . . . . .	44
16.3 Insertion d'images avec remplissage de l'espace vertical . . . . .	45
16.4 Grille automatique pour les environnements tikzpicture . . . . .	46
16.5 Flèche évasée, en TikZ . . . . .	48
16.6 Annoter une image . . . . .	50

---

# 1 Historique

v0.3.6 : Ajout d'un pictogramme cible/flèche  
v0.3.5 : Bugfix + pre-compatibilité fa5/6  
v0.3.4 : Pictoskill + option pour le package adjustbox  
v0.3.3 : Annoter une image  
v0.3.2 : Version alternative de la bannière Tri  
v0.3.1 : Ajout des cases à cocher pour les QCM  
v0.3.0 : Compatibilité accrue avec beamer  
v0.2.7 : Clé [Melange] pour les QCMs  
v0.2.6 : Roue des compétences / speedometer  
v0.2.5 : Correction du fonctionnement de EnvTeX  
v0.2.4 : Petite boîte type *marker*  
v0.2.3 : Panneaux autoroutiers + Affichettes de soldes  
v0.2.2 : Ajout d'une commande pour des flèches évasées, en TikZ  
v0.2.1 : Amélioration de la gestion des étoiles pour des niveaux de difficultés + Grille auto pour TikZ  
v0.2.0 : Étoiles pour des niveaux de difficultés (compatible avec EnvTeX)  
v0.1.9 : Bannière de titre + Insertion d'images en remplissage vertical  
v0.1.8 : Nutriscore  
v0.1.7 : Possibilité de créer des vignettes perso  
v0.1.6 : Patch displayskip + Patches pas-tableur  
v0.1.5 : La librairie babel de TikZ n'est plus chargée  
v0.1.5 : Vignettes + Numéros encadrés + Création de boîtes 'simples'  
v0.1.4 : Commande pour du texte dans une boîte arrondie, de hauteur 'figée' + 'Chat' SMS  
v0.1.3 : Environnement/commande pour des exercices, avec personnalisation(s)  
v0.1.2 : Crayon de compétences  
v0.1.1 : Tableaux de compétences  
v0.1.0 : Version initiale

## 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`;
- `fontawesome` (en attendant une compatibilité avec la version 6, via les options `[fa6]` / `[fa5fa6]` !);
- `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` 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>`;
- `<nonfa>`.

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

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

## 3 Présentation de réponses à un QCM

### 3.1 Principe

L'idée est de proposer un 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 défaut*

```
\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}
```

<b>a.</b> Réponse A	<b>b.</b> Réponse B	<b>c.</b> Réponse C	<b>d.</b> Réponse D
<b>a.</b> Réponse C1	<b>b.</b> Réponse A1	<b>c.</b> Réponse D1	<b>d.</b> Réponse B1
<b>a.</b> Réponse C2	<b>b.</b> Réponse B2	<b>c.</b> Réponse A2	<b>d.</b> Réponse D2

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

<b>(1.)</b> Réponse A	<b>(2.)</b> Réponse B	<b>(3.)</b> 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

**C.** Réponse C

**B.** Réponse B

**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 cochoer 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]%
{${\displaystyle\frac{1}{x}}$ § $1+{\displaystyle\frac{1}{x}}$ § $-2x^2+5$ § $-\infty$}
<rows={2cm}>
```

**a.**  $\frac{1}{x}$

**c.**  $-2x^2+5$

**b.**  $1+\frac{1}{x}$

**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;
- `Après` 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, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris.

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

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

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

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

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

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

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

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

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Lorem ipsum dolor sit amet, consectetur 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 moins 3pt par défaut, qui gère ces espacements.

L'idée est donc de proposer un 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{MultiCols}[options](nbcols)<options enumitem>!options beamer!  
  %corps  
\end{MultiCols}
```

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, consectetur 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, consectetur 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, consectetur 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 accu-

msan 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 na-

toque penatibus et magnis disparturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur 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, consectetur 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, consectetur 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 E
2. Réponse F
3. Réponse G
4. Réponse B
5. Réponse D

```
%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.  $\int_0^1 x^6 dx$

3.  $\int_0^1 x^4 dx$

2.  $\int_0^1 x^3 dx$

4.  $\int_0^1 x^7 dx$

```
%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}{ 0{} 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;
- **Titre** : titre du tableau; DS01 par défaut
- **PolTitre** : police de la 1<sup>ère</sup> 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					
N0 : Très bof / N1 : Bof / N2 : Moyen / N3 : Bien					



## 8 Crayon de compétences

### 8.1 Principe et fonctionnement

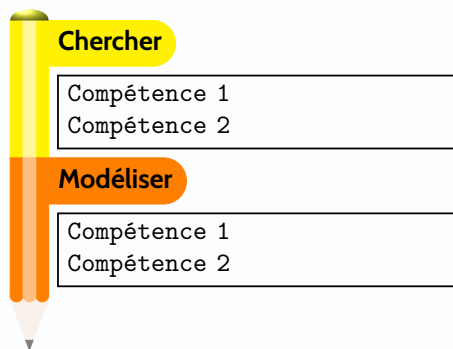
L'idée est de proposer un environnement pour créer un *crayon* de compétences, basé sur `TikZ`.

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>{listecompetences}
```

```
\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

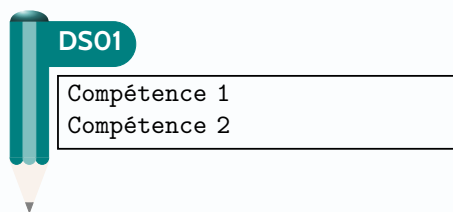
Les clés disponibles, à donner entre [...], sont :

- `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 `PoliceCatég` 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  
}
```

### 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}}
```

### 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}}
```

### Chercher

Compétence 1  
Compétence 2

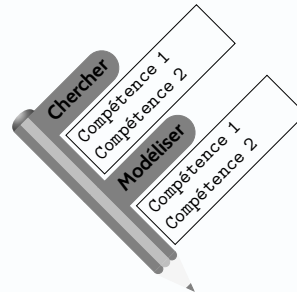
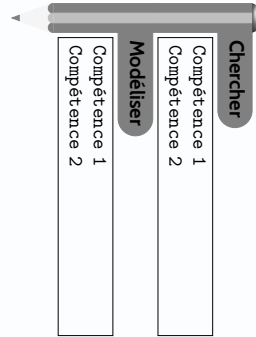
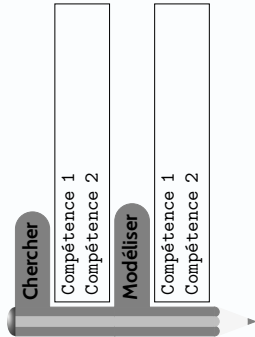
### Modéliser

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

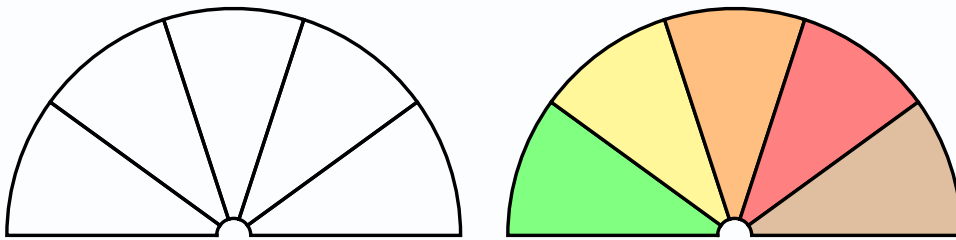
### 9.1 Principe et fonctionnement

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

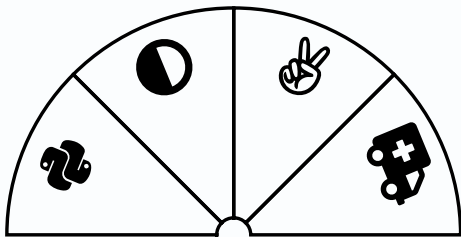
```
\RoueNiveaux[clés]<options tikz>{couleurs ou nb de niveaux}%
```

```
\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)

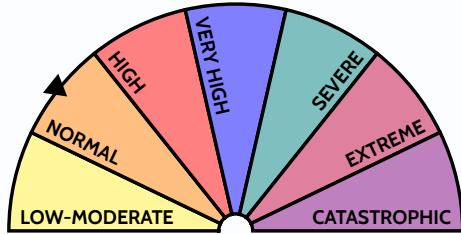
L'argument, obligatoire et entre {...} est la liste des couleurs, ou le nombre de catégories pour une sortie en N&B.

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

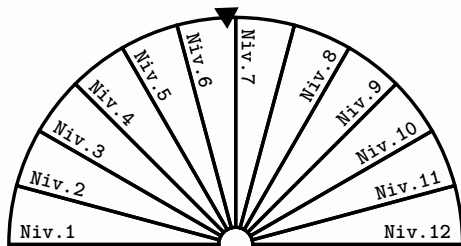
- Pos : position (excentricité) des icônes; 0.8 par défaut
- Echelle : échelle des icônes; 2.

### 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}%
```

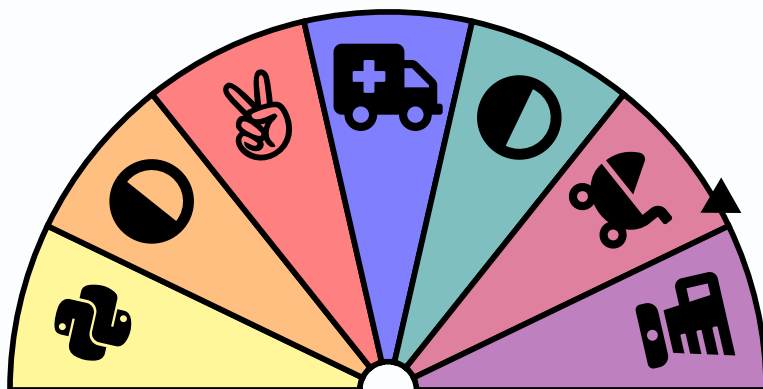


```
\RoueNiveaux[%
  Rayon=3cm,%
  Marqueur=5.85,%
  Police=\scriptsize\bfseries\ttfamily,%
  ListeNiveaux={%
    Niv.1,Niv.2,Niv.3,Niv.4,Niv.5,Niv.6,Niv.7,Niv.8,Niv.9,Niv.10,Niv.11,Niv.12}%
  ]%
{12}
```



```
\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!)

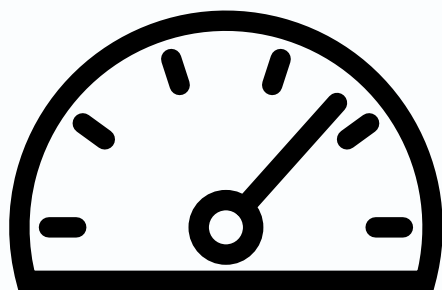
L'argument optionnel, et entre <...>, permet de spécifier des options à la figure TikZ (comme une rotation, un alignement, etc)

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.

```
\pictoskill[clés]{nb}
```

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



*%raccourcis pour les pictos 'de base'*

```
\pictoskillgreen~\pictoskillorange~\pictoskillred
```



### 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 :

- `col` : couleur principale; `skillgreen` par défaut
- `height` : hauteur, donnée en unité ou bien parmi `auto`/`dauto` du pictogramme; `auto` 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  
\pictoskill[height=1cm]{1}
```

*%hauteur automatique, basée sur la hauteur des majuscules sans jambage*

```
\begin{LARGE}
```

En ligne sans profondeur, on a `\pictoskill[col=skillorange]{2}` et

```
\pictoskill[col=skillred,hoffset=0.1]{1}
```

```
\end{LARGE}
```

*%hauteur automatique, avec intégration du jambage*

```
\begin{Huge}
```

En ligne avec profondeur, on a `\pictoskill[col=skillorange,height=dauto]{2}` et

```
\pictoskill[col=skillred,hoffset=0.025,height=dauto]{1}
```

```
\end{Huge}
```



En ligne sans profondeur, on a  et 

En ligne avec profondeur, on a  et 

## 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*{rotate=90}
```



En mode `auto`, 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\calctoheightofchars{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).

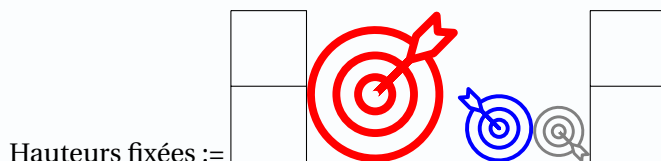
La version étoilée en mode `auto` ne s'occupe pas de la profondeur des caractères à jambage.

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

- `colors` : booléen pour colorier; `false` par défaut
- `height` : hauteur, donnée en unité ou `auto` du pictogramme; `auto` 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 TikZ.

```
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*[maincolor=red]{}H` en ligne.

Un essai `q@H` en ligne.

```
{\Huge\sffamily\textcolor{orange}{Un essai q\pictobullseye{}\pictobullseye*{}H en  
ligne.}}
```

Un essai `q@H` en ligne.



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

# Tests en ligne et !

```
\def\calheightofchars{É}
\def\calctoheightofchars{()}
\def\calcdepthofchars{()}
\scalebox{3}[3]{En
  ligne\,\pictobullseye*[colors,listcolors={green!50,orange!50,red!50}]{}. Évidemment
!}

{\huge Et (\pictobullseye[colors,listcolors={red!50,yellow!50,teal!50}]{rotate=180})!}}
```

# 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{}
```

SCORE



### 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
- **Légende** : texte de la légende (qui sera en majuscule) ; score par défaut ;
- **Police** : Police globale ; \bfseries\sffamily par défaut
- **AffLégende** : 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[Légende=Géométrie,Hauteur=2]{4}
```

GÉOMÉTRIE



*%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,AffLégende=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 `tcolor box`.

```
\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

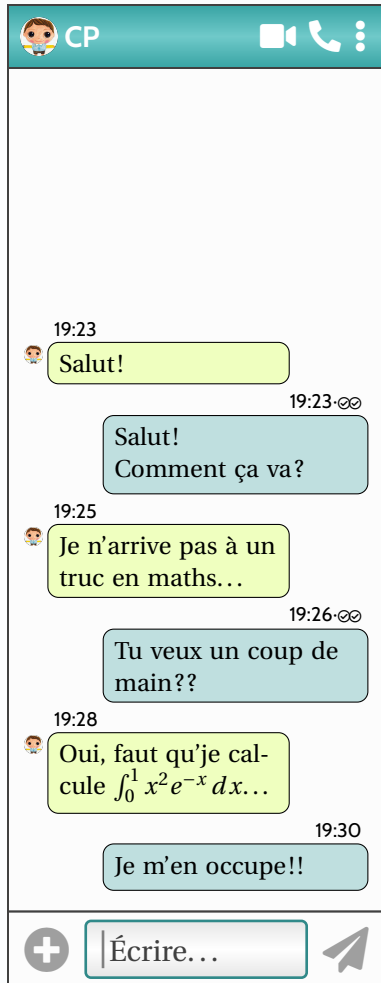
En ce qui concerne les commandes de création des bulles, `\SMSrec` et `\SMSenv` :

- 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  $\int_0^1 x^2 e^{-x} dx$ \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}

```



## 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
\StyleEnvTExoDefault
```

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}
```

**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 `TikZ`, 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`) peut être désactivé et *adapté* 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 :
  - `Icône/...` := afficher l'icône ... à la fin de la ligne;
  - `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/V3/00/...);
- `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}
```

☆☆☆

On essaye en ligne ☆☆☆ avec une note de 2.5

On essaye en ligne ☆☆☆ avec une note de 2.5

☆☆☆☆☆

### 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=Icône/\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_ss4]\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=Icône/\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}

\StyleEnvtExoDefault
\begin{EnvtExo}%
  [Type=Perso/{Titre perso~},Decoration=Image/goku_ss4,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>

```

## 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

## Titre perso 15 (☆☆☆) .....

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 Boîtes de présentation

```
\begin{BoiteSimple}[couleur]<options tcbbox>{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  $A_1, A_2, \dots, A_n$ , 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  $M^p$ .

```
\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  $A_1, A_2, \dots, A_n$ , 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  $M^p$ .

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

```
\begin{TitreSujet}[Couleur=...,AlignH=...]<options tcbbox>{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)**

### 15.4 Numéros encadrés

```
\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

### 15.5 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 pyluater (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 tableau
\AffVignette[Type=sheet,Couleur=green!50!black,Police=\footnotesize\sffamily]
{cellule A3}
```

**cellule A3**

```
%vignette type tableau
\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  pour...
```

## 15.6 Boîte arrondie, petite boîte type marker

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

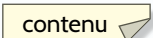
On lance le logiciel

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

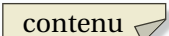
On lance le logiciel  en cliquant .

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

```
\tbcmarker{contenu}
```



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



## 15.7 Bannière de titre

L'idée est de proposer une bannière, réalisée en TikZ, 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}
```



Les principales clés (anglicisées) sont :

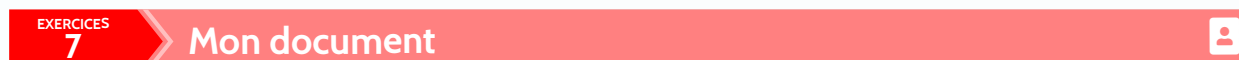
- `height` : hauteur de la bannière (2.5em par défaut)
- `width` : largeur de la bannière (\linewidth par défaut)
- `blockwidth` : largeur du bloc gauche (2.75em par défaut, mais si `auto`, la largeur s'adapte à son contenu)
- `coltxt` : couleur des textes (white par défaut)
- `fonttxt` : police globale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

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)}
```





## 15.8 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}
```

**A4**

<b>REIMS</b>	<b>145</b>
<b>PARIS</b>	<b>250</b>

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}
```

**TD01**

<b>COMPETENCE 1</b>	✓
<b>COMPETENCE 2</b>	✓

<b>LILLE</b>	<b>301</b>
<b>AMIENS</b>	<b>213</b>



## 15.9 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\,\,%}
```



## 16 Patches divers

### 16.1 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.

### 16.2 Quelques ajouts au package pas-tableur

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).

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

Les *ajouts* disponibles sont :

- colorer/griser une case, via \celcouleur;
- fusionner des cellules, via \celfusion;
- casser la numérotation avec des pointillés, via \celnumbreak;
- remplir (partiellement ou non) une ligne, via \lignetxt;
- remplir (partiellement ou non) une colonne, via \colonnetxt.

```
%chargement du package  
\usepackage[pastableur]{customenvs}  
  
%griser une cellule  
\celcouleur[couleur]{col-lig}  
  
%fusionner des cellules  
\celfusion[options pas-tableur]{col-lig}{col-lig}{label}  
  
%pointillés dans la numérotation  
\celnumbreak{num ligne pointillés}{début numéro suite}{fin numéro suite}  
  
%remplir ligne  
\lignetxt(*)[options pas-tableur]{num ligne}<col début>{liste des cellules}  
  
%remplir colonne  
\colonnetxt(*)[options pas-tableur]{num colonne}<lgn début>{liste des cellules}
```

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					

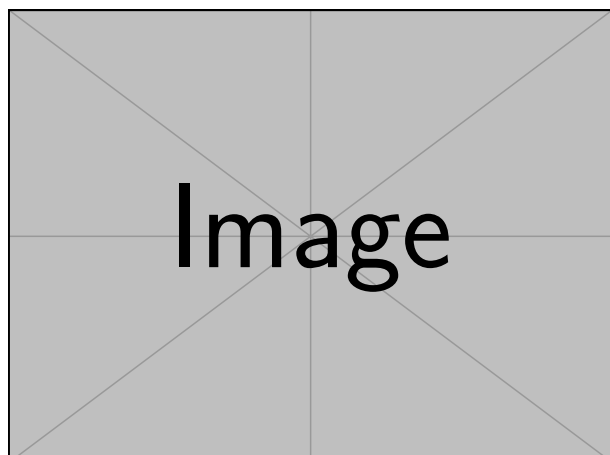
### 16.3 Insertion d'images avec remplissage de l'espace vertical

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 `m/h/b/n` permet de spécifier un type de centrage vertical, si besoin, parmi `milieu/haut/bas/non` (`n` par défaut).

```
\InsererImage<vRemplir,vOffset=5mm,vCentrer=m>{example-image}
```



## 16.4 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`.

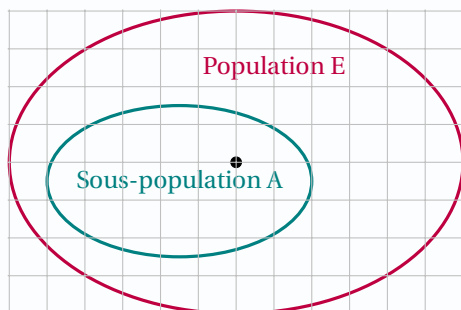
### Version manuelle

```
%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

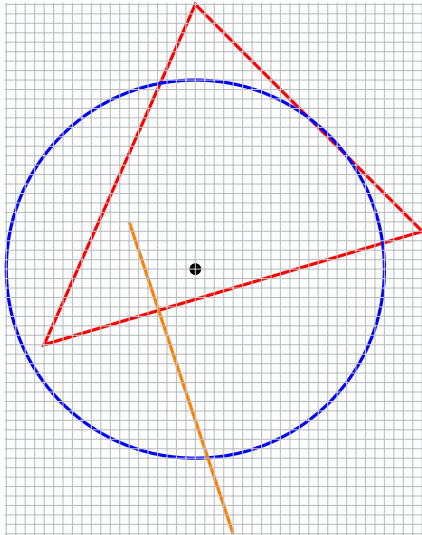
```
%clé locale, utilisation ponctuelle
\begin{tikzpicture}[grilleauto=pasX$pasY]
  \tkzAutoGridLocal(*)
  %codes
\end{tikzpicture}
```

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).

## 16.5 Flèche évasée, en TikZ

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

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

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

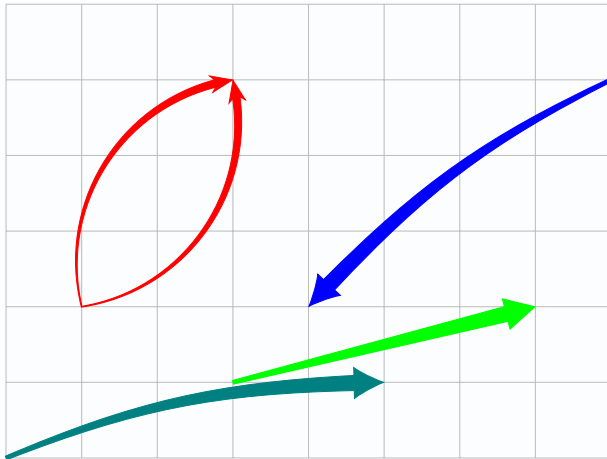




```

\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}

```

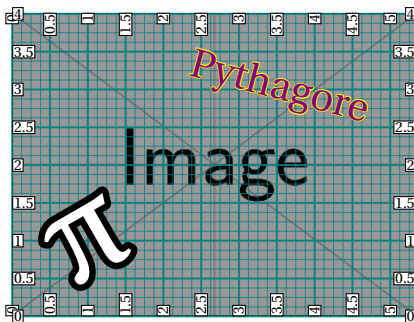


## 16.6 Annoter une image

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

```
\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 `includegraphics`. Le fichier est à préciser avec son 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}
```

