customenvs [en]

Some custom environments,
or small patches.

Version 0.2.6 -- 04/11/2024

Cédric Pierquet
c pierquet -- at -- outlook . fr
https://github.com/cpierquet/customenvs

Contents

1 History

v0.2.6 : Whell of skills, speedometer
v0.2.5 : Bugfix with exercices ([fr] macro)
v0.2.4 : Small box *marker*
v0.2.3 : Highway signs + sold banners (see [fr] doc)
v0.2.2 : Flared arrow, with Ti*k*Z
v0.2.1 : Enhancements for *stars skills* + AutoGrid for Ti*k*Z (see [fr] doc)
v0.2.0 : Skills with stars (fontawesome5 or Ti*k*Z)
v0.1.9 : Title banner
v0.1.8 : Score banner
v0.1.7 : Small patch for Vignette macro (see [fr] documentation)
v0.1.6 : Small patchs for displayskip + pas-tableur (see [fr] documentation)
v0.1.5 : New macros for boxes with tcolorbox (see [fr] documentation)
v0.1.4 : Create a SMS conversation
v0.1.3 : Environment for exercise(s) (in french doc)
v0.1.2 : Pencil of skills
v0.1.1 : Skills table (only french for the moment...)
v0.1.0 : Initial version

2 The package customenvs

2.1 Idea

The idea is to propose some classics environments with customizations (some are, for the moment, only in french) :

- write in *multicols*, with spacings enhancements ;

- present answers for a *MCQ* ;

- create a list with *choosen items* (randomly or by numbers) ;

- present a skill table.

The globa idea is ti propose *user-friendly* environments, with explicit customizations, without using verbose syntax ; but there's other solutions, using for example `\vspace` ou `\setlength` or `spacingtricks` package.

2.2 Loading

The package loads within the preamble with `\usepackage{customenvs}`.
Loaded packages are

- `xstring`, `simplekv`, `listofitems`, `randomlist` and `xintexpr` ;

- `enumitem` ;

- `multicol` ;

- `tabularray` ;

- `fontawesome5` ;

Due to limitations, `enumitem/multicol/tabularrayfontawesome5` can be *un*loaded by `customenvs` (user must load them manually) via options :

- ⟨`noenum`⟩ ;

- ⟨`nomulticol`⟩ ;

- ⟨`notblr`⟩ ;

- ⟨`nofa`⟩ ;

```
%with all packages
\usepackage{customenvs}

%with option to no load some packages
\usepackage[option(s)]{customenvs}
```

3 Answers for a MCQ

3.1 Idea

The idea is to propose an environment to present answers for a MQC with `tabularray` (and not `multicols`). It's possible to use 2, 3 or 4 answers (and with 4 answers it's possible to use 2 columns.)

```
\AnswersMCQ[options]{list of answers}<tblr options>
```

The avalailable `options` are :

- `Width` : `0.99\linewidth` by default ;
- `Lines` : `false` by default ;
- `SpaceCR` for Columns/Rows spacing, within `col/row` or `global` : `6pt/2pt` by default ;
- `NumCols`, 2 or 4 : `4` by default ;
- `Labels` for the labels : `a.` by default ;
  - with `a` to *enumerate* `a b c d` ;
  - with `A` to *enumerate* `A B C D` ;
  - with `1` to *enumerate* `1 2 3 4` ;
- `FontLabels` : `\bfseries` by default ;
- `SpaceLabels` : `\kern5pt` by default ;
- `Swap`, for ACBD instead of ABCD : `false` by default.

The list of answers must be given within `answA § answB § ...`.
Specific options for `tblr` are given between last optionnal argument, between `<...>`.

3.2 Examples

```
%default output
\AnswersMCQ{Answer A § Answer B § Answer C § Answer D}
```

| **a.** Answer A | **b.** Answer B | **c.** Answer C | **d.** Answer D |

```
\AnswersMCQ[Lines]{Answer A § Answer B § Answer C § Answer D}
```

| **a.** Answer A | **b.** Answer B | **c.** Answer C | **d.** Answer D |

```
\AnswersMCQ[Lines,Labels=(1.),SpaceLabels={~~~}]{Answer A § Answer B § Answer C}
```

| **(1.)** Answer A | **(2.)** Answer B | **(3.)** Answer C |

```
\AnswersMCQ[Labels={A.},FontLabels={\color{red}\bfseries}]%
    {Answer A § Answer B § Answer C § Answer D}
```

| **A.** Answer A | **B.** Answer B | **C.** Answer C | **D.** Answer D |

```
\AnswersMCQ[Labels={1.},FontLabels={\color{red}\bfseries}]%
    {Answer A § Answer B § Answer C § Answer D}
```

| **1.** Answer A | **2.** Answer B | **3.** Answer C | **4.** Answer D |

```
\AnswersMCQ[NumCols=2,Labels={A.},FontLabels={\color{red}\bfseries}]%
    {Answer A § Answer B § Answer C § Answer D}
```

| **A.** Answer A | **C.** Answer C |
| **B.** Answer B | **D.** Answer D |

```
\AnswersMCQ[NumCols=2,Swap,Labels={A.},FontLabels={\color{red}\bfseries}]%
    {Answer A § Answer B § Answer C § Answer D}
```

| **A.** Answer A | **B.** Answer B |
| **C.** Answer C | **D.** Answer D |

```
\AnswersMCQ[Lines,NumCols=2,SpaceCR=6pt/10pt]%
    {Answer A § Answer B § Answer C § Answer D}
```

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

```
\AnswersMCQ[Width=10cm,NumCols=2,Lines]%
    {$\displaystyle\frac1x$ § $1+\displaystyle\frac1x$ § $-2x^2+5$ § $-\infty$}
    <rows={1.5cm}>
```

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

4 List avec with picked elements (random or not)

4.1 Global use

The idea is to :

- create a list of items, the base for choices ;

- print the list with picked items.

```
\CreateItemsList{list}{macro}{listname}
```

```
\ListItemsChoice[keys]{macro}{listname}(numbers)<enumitem options>
```

The available `keys` are :

- `Type` : `enum` or `item` ;

- `Random` : `false` by default.

The second argument, mandatory and between `{...}` is the macro for the list.
The third argument, mandatory and between `{...}` is the name of the list.
The fourth argument, mandatory and between `(...)` give :

- the number of random items to display, with `Random=true` ;

- the numbers of picked itemps, within `num1,num2,...`.

The last argument, optional and between `<...>` gives specific options to `enumitem` environment.

Controls are done :

- to verify that the liste doesn't exist (for the creation) ;

- to verify that that the list still exist (for the display).

4.2 Examples

```
%creation of list ListItems, with macro \mylistofitems
\CreateItemsList%
    {Answer A,Answer B,Answer C,Answer D,Answer E,Answer F,Answer G,Answer H}%
    {\mylistofitems}{ListItems}
```

```
%items random
\ListItemsChoice[Random]{\mylistofitems}{ListItems}(5)
```

1. Answer H
2. Answer A
3. Answer B
4. Answer C
5. Answer E

```
%items picked
\ListItemsChoice{\mylistofitems}{ListItems}(1,4,3,8,2)
```

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

```
%creation of list ListItemsB, with macro \mylistofitemsb
\CreateItemsList%
    {{$\int_0^1 x^2 dx$},{$\int_0^1 x^3 dx$},{$\int_0^1 x^4 dx$},...}%
    {\mylistofitemsb}{ListItemsB}
```

```
%items picked
\ListItemsChoice[Type=item]{\mylistofitemsb}{ListItemsB}(7,2,1,5,3)<label=$--$>
```

—— $\int_0^1 x^8 dx$

—— $\int_0^1 x^3 dx$

—— $\int_0^1 x^2 dx$

—— $\int_0^1 x^6 dx$

—— $\int_0^1 x^4 dx$

5 Pencil of skills

5.1 Global use

The idea is to :

- present of list of categories and skills ;

- presented like a pencil.

The code (within CC-BY-SA 4.0 license) is adapted from :

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

```
\PencilSkills[keys]<tikz options>{listofskills}
```

The style is globally fixed, but there's some customization available.

5.2 The macro

Available `keys` are :

- `FontCateg` : font for the categories ;

- `FontBlock` : font for the skills ;

- `Colors` : list of category's colors
  `BgCateg1/FgCateg1,BgCateg1/FgCateg1,...`
  (if `FgCateg1` est missing, `black` is used)

- `BlockWidth` : width of skill's block ;

- `Scale` : global scale

- `BlackWhite` : boolean for B&W.

The second argument, optional and between `<...>` gives specific options to `enumitem` environment.

The last argument, mandatory and between `(...)` give the list of categories/skills, within `Categ1/ListSkills1,Categ2/ListSkills2,...`.
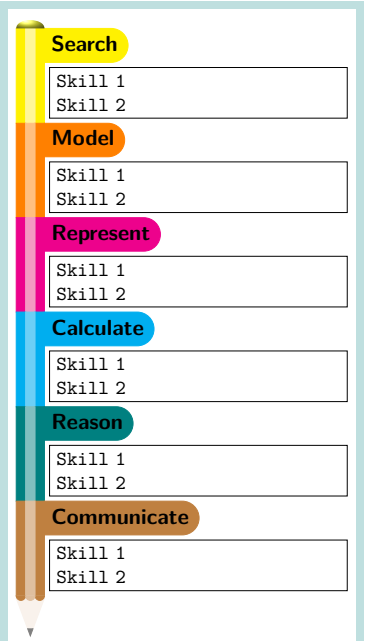
5.3 Examples

```
%default output
\PencilSkills{Search/Skill 1\\ Skill 2,Model/{Skill 1\\ Skill 2}}
```
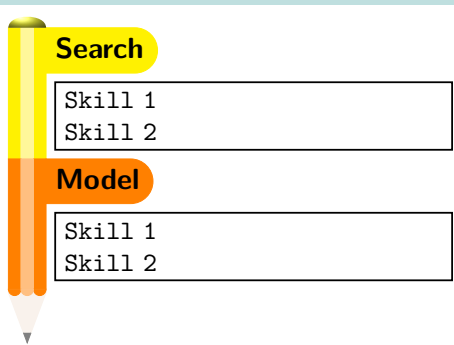
```
\PencilSkills[Scale=0.75]%
    {Search/Skill 1\\Skill 2,Model/{Skill 1\\Skill 2},%
    Represent/{Skill 1\\Skill 2},Calculate/{Skill 1\\Skill 2},%
    Reason/{Skill 1\\Skill 2},Communicate/{Skill 1\\Skill 2}}
```
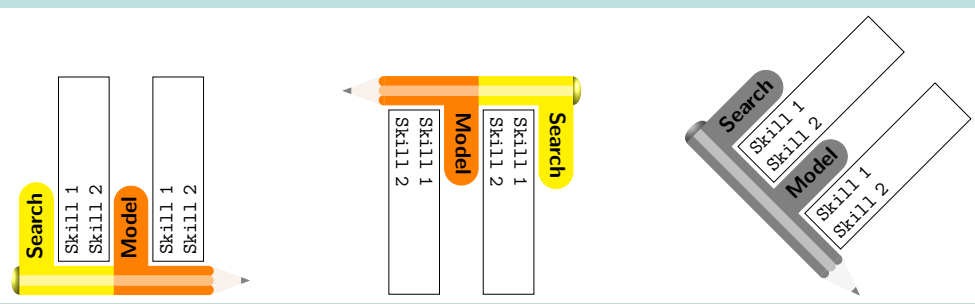
Search

| Skill 1 |
| Skill 2 |

Model

| Skill 1 |
| Skill 2 |

Represent

| Skill 1 |
| Skill 2 |

Calculate

| Skill 1 |
| Skill 2 |

Reason

| Skill 1 |
| Skill 2 |

Communicate

| Skill 1 |
| Skill 2 |

```
\PencilSkills[Scale=0.75,BlockWidth=3cm]<rotate=90>{
    Search/Skill 1\\Skill 2,Model/{Skill 1\\Skill 2}}
\hspace{1cm}
\PencilSkills[Scale=0.75,BlockWidth=3cm]<rotate=-90>{
    Search/Skill 1\\Skill 2,Model/{Skill 1\\Skill 2}}
\hspace{1cm}
\PencilSkills[Scale=0.75,BlockWidth=3cm,BlackWhite]<rotate=45>{
    Search/Skill 1\\Skill 2,Model/{Skill 1\\Skill 2}}
```

6 Score banner

6.1 Global use

The idea is to insert a score banner, with customizations.

```
ScoreBanner[keys]{number}
```

```
%default output
\ScoreBanner{}
```


6.2 The macro

Available `keys` are :

- `Height` : height of the banner (without the legend) ; `1` by default

- `Ratio` : ratio of boxes ; `0.6` by default

- `Symbols` : labels ; `A,B,C,D,E` by default

- `Legend` : legend (uppercase) ; `score` by default ;

- `Font` : global font ; `\bfseries\sffamily` by default

- `ShowLegend` : boolean for the legend ; `false` by default ;

- `Colors` : colors for boxes ;

$$\texttt{colorNS1,colorNS2,colorNS3,colorNS4,colorNS5} \text{ by default ;}$$

- `ScaleSymbols` : scale H/V of labels ; `1.25,1.65` by default ;

- `Colbg` : background color for select box ; `white` by default.

If the list of colors doesn't fill all the boxes, `lightgray` color is used.

```
\ScoreBanner[Legend=Geometry,Height=2]{4}
```



```
%bg of lower part is yellow!25
\def\lstcouleurs{colorNS1,colorNS2,colorNS3,colorNS4,colorNS5,purple}
\ScoreBanner%
  [ScaleSymbols={1.33,2},Height=3.25,ShowLegend=false,Ratio=0.75,
  Symbols={1,2,3,4,5,6},Colors=\lstcouleurs,
  Colbg=yellow!25]{1}
```

7 SMS conversation

7.1 Global use

The idea is to present a conversation of SMS.

```
\begin{ChatSMS}[keys]{name}
  \InSMS(*){time}{msg}
  \OutSMS*(*){time}{msg}
\end{ChatSMS}
```

The style is globally fixed, but there's some customization available.

7.2 The environment

Available `keys` are :

- `height` : height of the window (auto or specific) ; `auto` by default

- `width` : width of the window ; `7cm` by default

- `margin` : margin (L or R) for the bubble `1.5cm` by default

- `color` : *main* color (banner) ; `teal!75!cyan!75!white` by default ;

- `colback` : color for background ; `lightgray!5` by default

- `colorin` : color for incoming SMS ; `lime!25` by default

- `colorout` : color for outcoming SMS ; `teal!25` by default

- `writetxt` : text of sending zone ; `Write` by default

- `fonttxt` : bubble's font ; `\normalfont` by default

- `avatar` : avatar of contact ; `\faAddressCard` by default

- `dispavatar` : boolean for displaying avatar near the bubbles ; `false` by default

- `blackwhite` : boolean pour black&white. `false` by default

The argument, mandatory and between `(...)` give the name of the contact.

7.3 Macros for the bubbles

Regarding the bubble creation commands, `\InSMS` and `\OutSMS`:

- the *starred* version does not display the *checkmarks* of *good reception*;

- the first mandatory argument is the time to display ;

- the second mandatory argument is the message to display (including multi-lines).

7.4 Examples

```
%with a personal image
\begin{ChatSMS}%
  [width=6cm,fonttxt=\sffamily,height=10cm,avatar=img/android,dispavatar]{CP}
  \InSMS{19:23}{Hi !}
  \OutSMS{19:23}{Hi !\\ How are you ?}
  \InSMS{19:25}{Just a problem with a math question\ldots}
  \OutSMS{19:26}{Wanna help ??}
  \InSMS{19:28}{Yes, I need to compute $\mathsf{\int_{0}^{1} x^2e^{-x}\,dx}$\ldots}
  \OutSMS*{19:30}{Take care !!}
\end{ChatSMS}
```
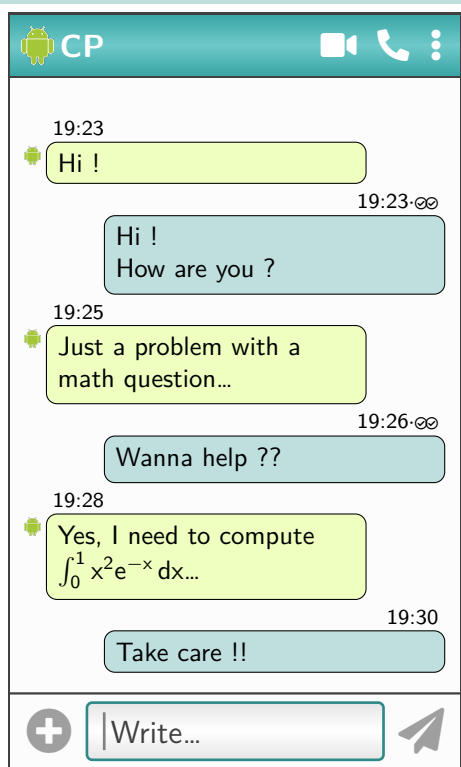
```
\begin{ChatSMS}%
  [width=8cm,fonttxt=\sffamily,avatar=\faCanadianMapleLeaf,blackwhite]{CP}
  \InSMS{19:23}{Hi !}
  \OutSMS{19:23}{Hi !\\ How are you ?}
  \InSMS{19:25}{Just a problem with a math question\ldots}
  \OutSMS{19:26}{Wanna help ??}
  \InSMS{19:28}{Yes, I need to compute $\mathsf{\int_{0}^{1} x^2e^{-x}\,dx}$\ldots}
  \OutSMS*{19:30}{Take care !!}
\end{ChatSMS}
```
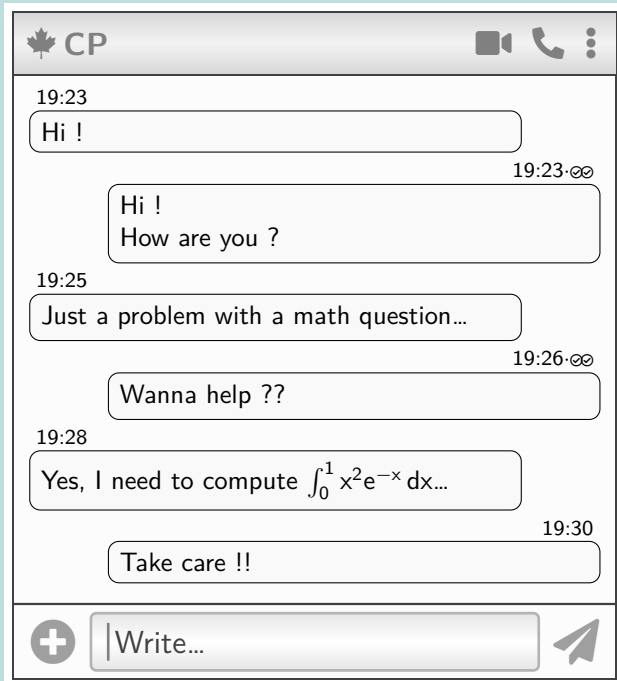
8 Title banner

8.1 Global usage

The idea is to propose a banner, made with TikZ, to present for example a title.
The global style is fixed, but few customizations are possible.

```
\tkzBannerTri[keys]{number}{title}
```

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

```
01 >  Titre du document
```

Available `keys` are :

- `height` (`2.5em` by default)

- `width` (`\linewidth` by default)

- `blockwidth` (`2.75em` by default, but can be set to `auto`)

- `coltxt` (`white` by default)

- `fonttxt`

- `swap` (`false` by default, for an other style)

- `maincolor` (`darkgray` by default)

- `collight` (`darkgray!25` by default)

- `colmedium` (`darkgray!50` by default)

- `coldark` (`darkgray` by default)

- `logo`

- `type`

- `dispblock` (`true` by default)

- `num` (`true` by default)

- `customtype`

8.2 Examples

```
\tkzBannerTri
  [maincolor=red,type=EXERCISES,blockwidth=auto,logo=\faAddressBook]
  {7}{My doc}
```

```
EXERCISES
7  >  My doc
```

```
\tkzBannerTri
  [maincolor=red,type=EXERCISES,blockwidth=5em,logo=\faAddressBook]
  {7}{My doc}
```

```
EXERCISES
7  >  My doc
```

```
\tkzBannerTri
  [maincolor=red,type=EXERCISES,blockwidth=auto,logo=\faAddressBook,swap]
  {07}{My doc}
```

EXERCISES
07 **My doc**

```
\tkzBannerTri
  [dispblock=false,maincolor=teal,logo=\faSchool]
  {}{My doc}
```

My doc

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

TP **My doc**

It's possible to redefine `\part` (for example).

9 Various commands

9.1 Difficulty levels with stars (fontawesome5)

```
\DiffLevelStars[max level (3)]{level}
```

```
\DiffLevelStars{0}\par
\DiffLevelStars{2.5}\par
\textcolor{teal}{\LARGE\DiffLevelStars[5]{4}}\par
\DiffLevelStars[5]{1.5}\par
```

9.2 Difficulty levels with stars (tikz)

```
\tkzLevelStars[colframe=...,colback=...,offset=...,maxlevel=...,valign=...]{level}
```

```
\tkzLevelStars{2.5}\par
{\LARGE We ty inline \tkzLevelStars{2.25} with score 2.25}\par
{\LARGE We ty inline \tkzLevelStars[valign=false]{1.75} with score 1.75}\par
\tkzLevelStars[colframe=red,colback=yellow,maxlevel=5]{3}
```

We ty inline ★★☆ with score 2.25
We ty inline ★★☆ with score 1.75

9.3 Flared arrow

```
\tkzFlaredArrow[%
  color=...,           %color of arrow
  arrowsize=...,       %size (auto or H/W )
  bend=...,            %empty for straigth or left/... or right/...
  thickness=...,       %size for the beginning
  factor=...,          %factor for calculating size for ending
  arrowstyle=...,      %style (arrows.meta)
  move=...             %boolean for moving instead coordinates
  ]%
  {begin}{end or move}
```

```
%arrow 0.5mm -> 1.25mm
\begin{tikzpicture}
\tkzFlaredArrow%
  [thickness=0.5mm,factor=2.5,bend=left/30,color=red,arrowstyle=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) ;
  \tkzFlaredArrow[color=green,arrowstyle=Triangle]{A}{B}
  \tkzFlaredArrow[color=blue,bend=right/10]{D}{C}
  \tkzFlaredArrow%
    [color=red,bend=left/45,arrowstyle=Stealth,thickness=0.1mm,factor=10]%
    {-2,1}{0,4}
  \tkzFlaredArrow%
    [color=red,bend=right/45,thickness=0.1mm,factor=10,arrowstyle=Stealth]%
    {-2,1}{0,4}
  \tkzFlaredArrow[color=teal,move,bend=left/10]{-3,-1}{5,1}
  \end{tikzpicture}
```
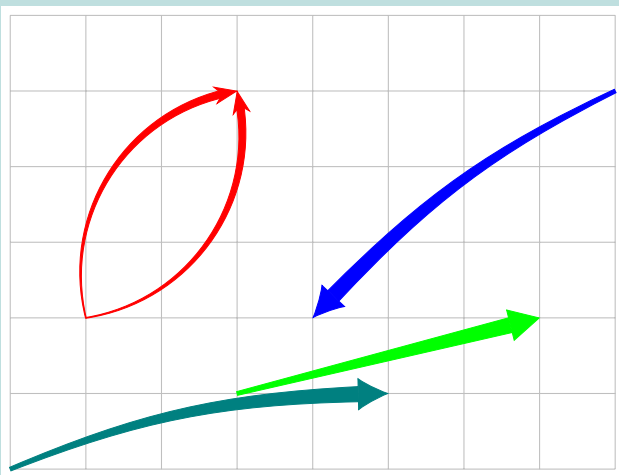

9.4 Small markerbox

```
\tbcmarker[color=...,width=...,font=...]{text}
```

```
\tbcmarker{my text}
```


```
\tbcmarker[color=olive,font=\normalfont\normalsize]{my text}
```


9.5 Wheel of skills / speedometer
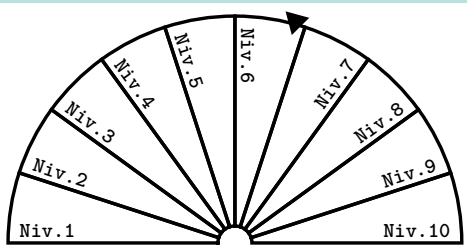
```
\WheelOfSkills[%
    Radius=...,        %radius of wheel
    Mark=...,          %absolute position of optional marker
    Font=...,          %font of optional labels
    SkillsList=...,    %list of optional skill labels
    ]%
    {number of skills or list of colors}
```

```
\WheelOfSkills[%
    Radius=3cm,%
    Mark=5.85,%
    Font=\scriptsize\bfseries\ttfamily,%
    SkillsList={Niv.1,Niv.2,Niv.3,Niv.4,Niv.5,Niv.6,Niv.7,Niv.8,Niv.9,Niv.10}]%
    {10}%
```
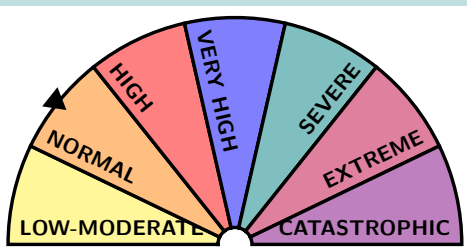


```
\WheelOfSkills[%
    Mark=1.5,%
    Font=\scriptsize\bfseries\sffamily,%
    SkillsList={LOW-MODERATE,NORMAL,HIGH,VERY HIGH,SEVERE,EXTREME,CATASTROPHIC}
    ]%
    {yellow!50,orange!50,red!50,blue!50,teal!50,purple!50,violet!50}%
```
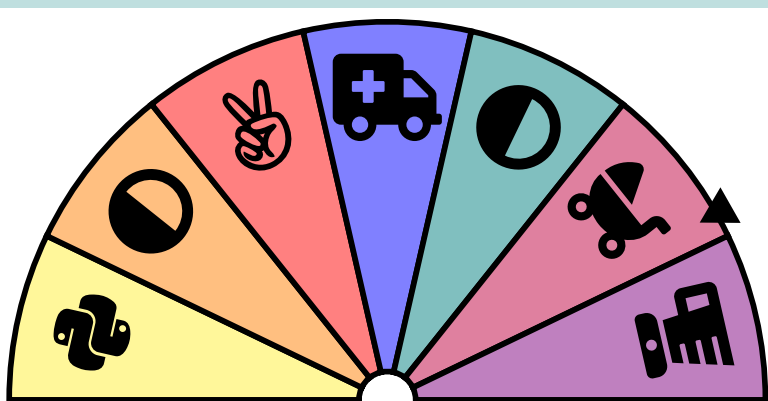


```
\begin{SkillsWheel}[%
    Radius=...,        %radius of wheel
    Mark=...,          %absolute position of optional marker
    Font=...,          %font of optional labels
    SkillsList=...,    %list of optional skill labels
    ]{number of skills or list of colors}
    \PutIconsSkills[Pos=...,Scale=...]{list of icons}
\end{SkillsWheel}
```

```
\begin{SkillsWheel}[Radius=5cm,Mark=5.85]%
        {yellow!50,orange!50,red!50,blue!50,teal!50,purple!50,violet!50}
    \PutIconsSkills[Echelle=3]%
    {\faPython,\faAdjust,\faAngellist,\faAmbulance,\faAdjust,\faBabyCarriage,\faBlender}
\end{SkillsWheel}
```

```
%inline version, with automatic dimensions
\miniskillwheel[Colors=...,Mark=...]{nb of levels}

%normal version
\tkzspeedometer[Size=...,Mark=...,Colors=...]{nb levels}
```

```
%inline version, with automatic dimensions
\scalebox{2.25}[2.25]{\sffamily Small inline \textit{skillwheel}
\miniskillwheel[Colors=red/blue,Mark=4.33]{7} for testing.}

%normal version
\tkzspeedometer[Size=5cm,Mark=2.25,Colors=teal/magenta]{6}
```

Small inline *skillwheel* ⏱ for testing.