

Comment faire de beaux graphiques avec Ti^kZ et PGFPLOTS

et si vous êtes sages on fera peut-être des camemberts.

— Et bien non on ne fera pas de camemberts.

— Ben pourquoi ?

— Parce que c'est moche et ça sert à rien.

Bertrand Masson

Les fiches de Bébert

30 septembre 2010

Le but de ce tutoriel est d'apprendre à réaliser des graphiques à l'aide de **Ti**kZ et **PGFPLOTS**. Pourquoi ne pas utiliser un logiciel spécialisé comme un tableur ? Il y a plusieurs raisons.

- l'exportation d'un graphique fait à l'aide d'un tableur en un document compris par \LaTeX n'est pas toujours aisé et tu te retrouves souvent à exporter ton graphique sous un format image, qui perd beaucoup en qualité ;
- l'insertion dans \LaTeX notamment au niveau de l'échelle, n'est pas toujours heureuse. Tu es amené à souvent réduire sa taille, rendant souvent des parties du graphique illisibles (taille de caractère trop petite, pixelisation, ...) ;
- tu perds la continuité typographique de ton document (polices différentes).

Donc pour des graphiques parfaitement intégrés à ton document, utilise le couple **Ti**kZ et **PGFPLOTS**, en plus ce n'est guère plus compliquer à utiliser qu'un tableur.

Dans cette fiche on va se limiter à des opérations de base, on se contentera de graphique simple avec un minimum de réglages. Ultérieurement d'autres fiches aborderont en détail des points particuliers.

Pour illustrer notre propos nous allons utiliser les précipitations et les températures relevées dans la région lilloise pour le mois de décembre 2004. Les deux premières choses à faire c'est de charger les packages `tikz` et `pgfplots` :

```
\usepackage{tikz}  
\usepackage{pgfplots}
```

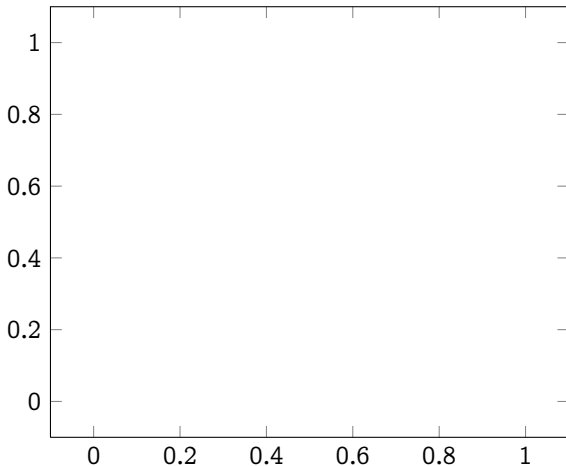
Maintenant nous allons commencer notre graphique. Tu ouvres un environnement `TikZ` :

```
\begin{tikzpicture}  
  
\end{tikzpicture}
```

Pour être lisible un graphique se doit d'avoir des axes, ce que tu vas réaliser grâce à l'environnement `axis`. Comme tu ne lui as encore précisé aucune donnée **PGFLOTS** affiche un truc par défaut avec des coordonnées comprise entre 0 et 1 :

```
\begin{tikzpicture}
\begin{axis}

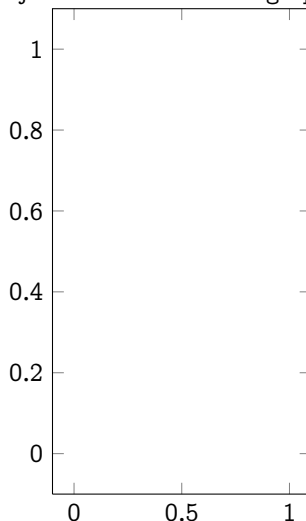
\end{axis}
\end{tikzpicture}
```



Taille du graphique

Évidemment l'environnement `axis` accepte des options (**Attention** les options sont séparées par des virgules). Tu peux jouer sur la taille du graphique :

```
\begin{tikzpicture}  
  \begin{axis}[height=8cm,width=5cm]  
  
  \end{axis}  
\end{tikzpicture}
```



Taille du graphique

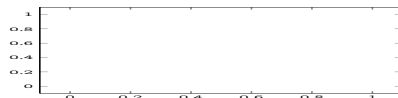
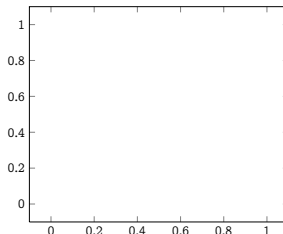
d'autre exemples :

```
\begin{tikzpicture}  
\begin{axis}[scale=0.5]
```

```
\end{axis}  
\end{tikzpicture}
```

```
\begin{tikzpicture}  
\begin{axis}[xscale=0.7, yscale=0.2]
```

```
\end{axis}  
\end{tikzpicture}
```



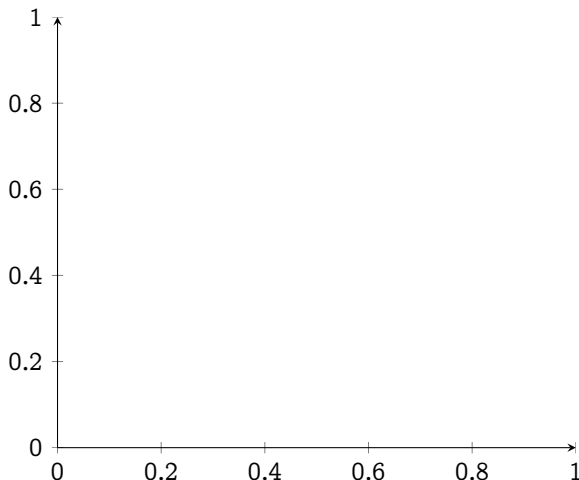
Je te rappelle que tu peux jouer aussi avec les options liées à [tikzpicture](#) pour adapter ton graphique à ton document :

```
\begin{tikzpicture}[scale=0.5]
```

Tu peux décider de n'afficher que 2 côtés du graphique

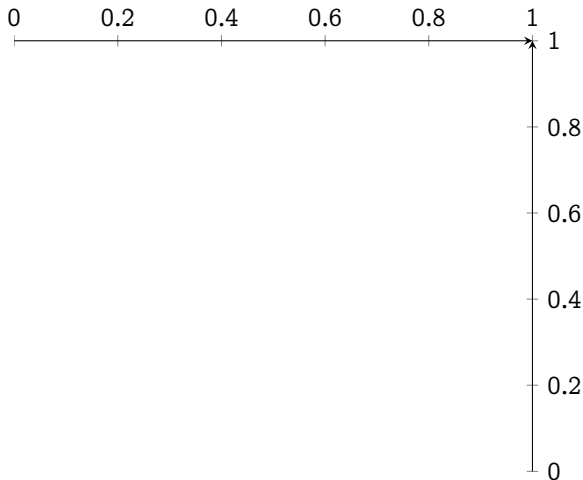
```
\begin{tikzpicture}
\begin{axis}[
axis x line=bottom,
axis y line = left]

\end{axis}
\end{tikzpicture}
```



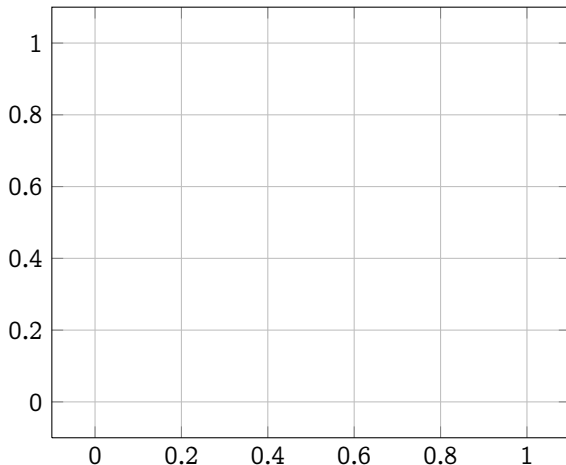
Dans l'autre sens

```
\begin{tikzpicture}  
  \begin{axis}[  
    axis x line=top,  
    axis y line=right]  
  
  \end{axis}  
\end{tikzpicture}
```



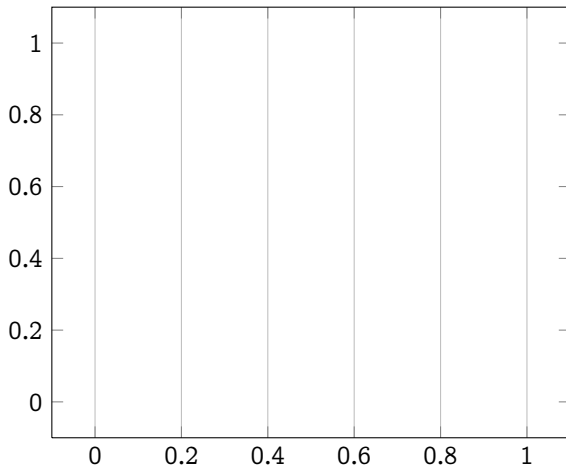
Ajouter une grille

```
\begin{tikzpicture}  
  \begin{axis}[  
    grid=major  
  ]  
  
  \end{axis}  
\end{tikzpicture}
```



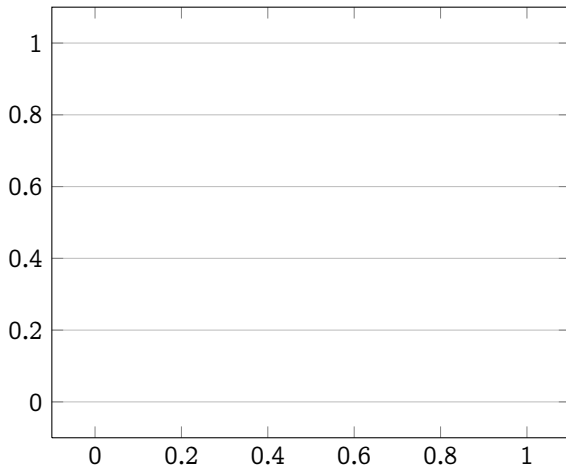
Ajouter une grille uniquement pour les x

```
\begin{tikzpicture}  
  \begin{axis}[  
    xmajorgrids]  
  
  \end{axis}  
\end{tikzpicture}
```



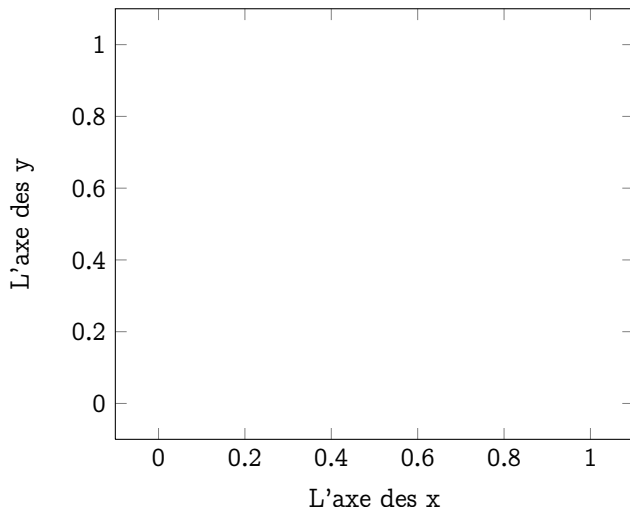
Ajouter une grille uniquement pour les y

```
\begin{tikzpicture}  
  \begin{axis}[  
    ymajorgrids  
  ]  
  
  \end{axis}  
\end{tikzpicture}
```



Ajouter un label pour les x ou les y

```
\begin{tikzpicture}  
  \begin{axis}[  
    xlabel={L'axe des x},  
    ylabel={L'axe des y}]  
  
  \end{axis}  
\end{tikzpicture}
```

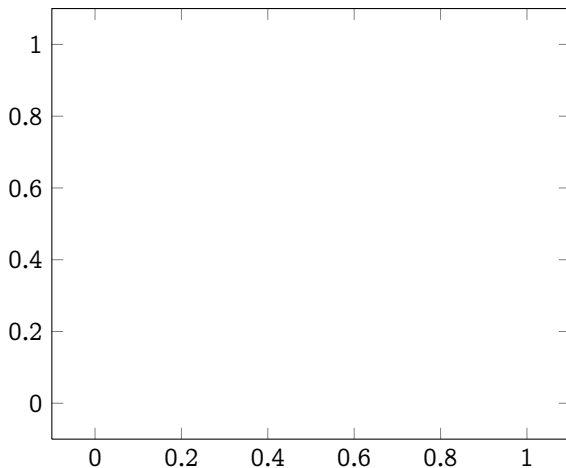


Ajouter un titre pour ton graphique

Le titre du graphique

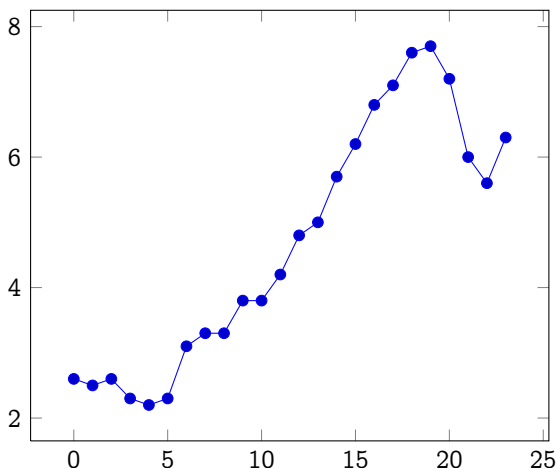
```
\begin{tikzpicture}
\begin{axis}[
title={Le titre du
graphique}]

\end{axis}
\end{tikzpicture}
```



Il est temps d'ajouter des données

```
\begin{tikzpicture}
\begin{axis}
\addplot coordinates {
(0,2.6) (1,2.5) (2,2.6)
(3,2.3) (4,2.2) (5,2.3)
(6,3.1) (7,3.3) (8,3.3)
(9,3.8) (10,3.8) (11,4.2)
(12,4.8) (13,5) (14,5.7)
(15,6.2) (16,6.8) (17,7.1)
(18,7.6) (19,7.7) (20,7.2)
(21,6) (22,5.6) (23,6.3)};
\end{axis}
\end{tikzpicture}
```



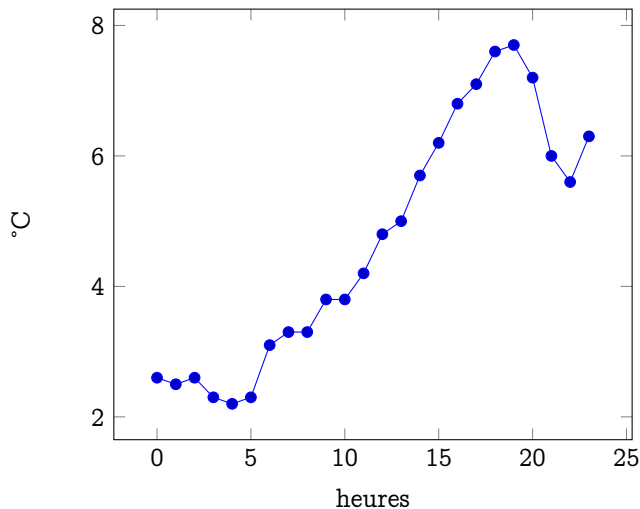
Le système de coordonnées est le même que celui de *TikZ*, voir la fiche de Bébert « *TikZ & PGF, introduction* ». Je te rappelle que les décimales sont notées à l'anglo-saxonne avec un point. Comme toutes commande *TikZ*, n'oublie pas le ; à la fin `\addplot coordinates {...};`

Avec un titre et des labels : le code

```
\begin{tikzpicture}
\begin{axis}[
xlabel={heures},
ylabel={\degres C},
title={Températures relevées à Lille le 16 décembre 2004},
]
\addplot coordinates {
(0, 2.6) (1, 2.5) (2, 2.6)
(3, 2.3) (4, 2.2) (5, 2.3)
(6, 3.1) (7, 3.3) (8, 3.3)
(9, 3.8) (10, 3.8) (11, 4.2)
(12, 4.8) (13, 5) (14, 5.7)
(15, 6.2) (16, 6.8) (17, 7.1)
(18, 7.6) (19, 7.7) (20, 7.2)
(21, 6) (22, 5.6) (23, 6.3)};
\end{axis}
\end{tikzpicture}
```

Avec un titre et des labels : le résultat

Températures relevées à Lille le 16 décembre 2004



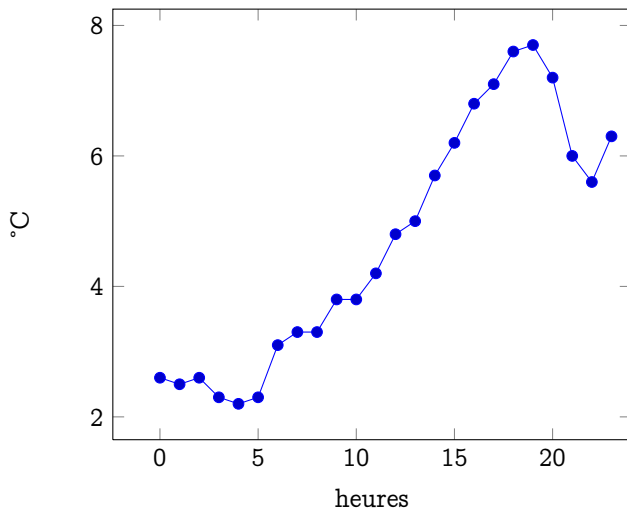
Premier problème

Alors il n'est pas joli notre graphique et pas si compliqué à faire. Bon il y a quand même un petit problème. Si la 25^e heure est très cinématographique, elle n'a rien de scientifique. Il faut donc donner à **PGFPLOTS** des limites. Tu les fixes par `xmin`, `xmax`, `ymin`, `ymax`.

```
\begin{tikzpicture}
\begin{axis}[xmax=24,
xlabel={heures},
ylabel={\degres C},
title={Températures relevées à Lille le 16 décembre 2004}]
\addplot coordinates {
(0,2.6) (1,2.5) (2,2.6) (3,2.3) (4,2.2) (5,2.3)
(6,3.1) (7,3.3) (8,3.3) (9,3.8) (10,3.8) (11,4.2)
(12,4.8) (13,5) (14,5.7) (15,6.2) (16,6.8) (17,7.1)
(18,7.6) (19,7.7) (20,7.2) (21,6) (22,5.6) (23,6.3)};
\end{axis}
\end{tikzpicture}
```

Première solution

Températures relevées à Lille le 16 décembre 2004

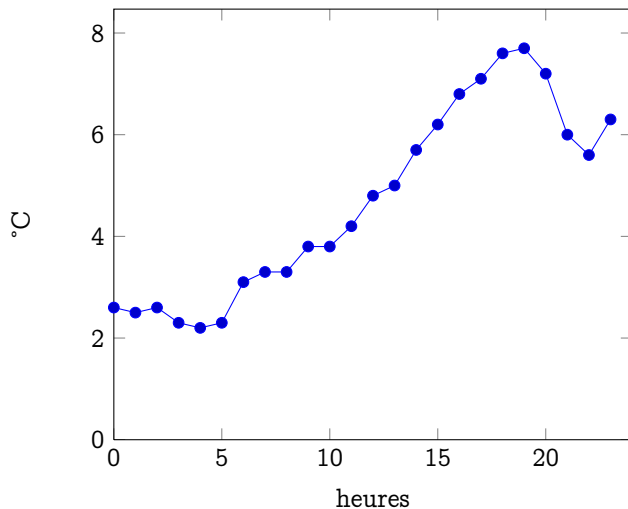


Fixer des limites à son graphique : le code

```
\begin{tikzpicture}
\begin{axis}[xmax=24,xmin=0,ymin=0,
xlabel={heures},
ylabel={\degres C},
title={Températures relevées à Lille le 16 décembre 2004}]
\addplot coordinates {
(0,2.6) (1,2.5) (2,2.6) (3,2.3) (4,2.2) (5,2.3)
(6,3.1) (7,3.3) (8,3.3) (9,3.8) (10,3.8) (11,4.2)
(12,4.8) (13,5) (14,5.7) (15,6.2) (16,6.8) (17,7.1)
(18,7.6) (19,7.7) (20,7.2) (21,6) (22,5.6) (23,6.3)};
\end{axis}
\end{tikzpicture}
```

Fixer des limites à son graphique : le résultat

Températures relevées à Lille le 16 décembre 2004



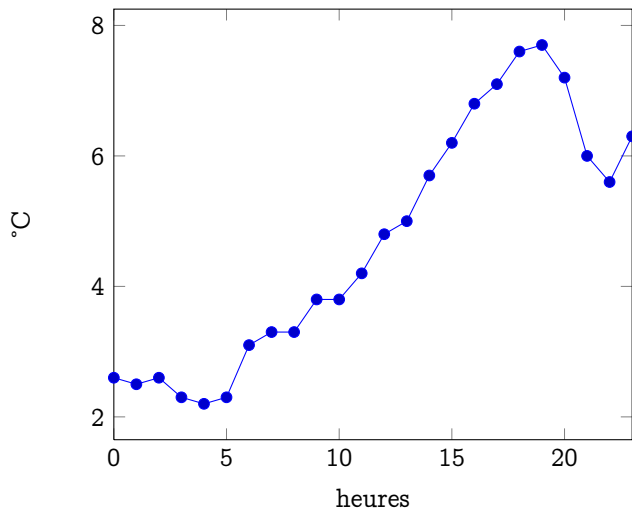
Enlever les marges : le code

Tu as remarqué que **PGFPLOTS** ajoute des marges entre les axes et le graphique. Si tu veux les supprimer tu utilises l'option `enlarge x limits=false`. Note que dans ce cas tu n'as plus besoin de mettre `xmax=24`, puisque les axes s'arrêtent aux limites du graphique. Il existe également un `enlarge y limits=false`, pas très utile dans notre exemple.

```
\begin{tikzpicture}
\begin{axis}[enlarge x limits=false,
xlabel={heures},
ylabel={\degres C},
title={Températures relevées à Lille le 16 décembre 2004}]
\addplot coordinates {
(0,2.6) (1,2.5) (2,2.6) (3,2.3) (4,2.2) (5,2.3)
(6,3.1) (7,3.3) (8,3.3) (9,3.8) (10,3.8) (11,4.2)
(12,4.8) (13,5) (14,5.7) (15,6.2) (16,6.8) (17,7.1)
(18,7.6) (19,7.7) (20,7.2) (21,6) (22,5.6) (23,6.3)};
\end{axis}
\end{tikzpicture}
```

Enlever les marges : le résultat

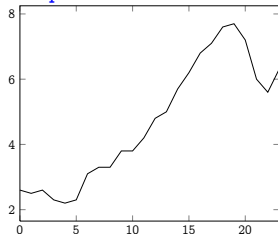
Températures relevées à Lille le 16 décembre 2004



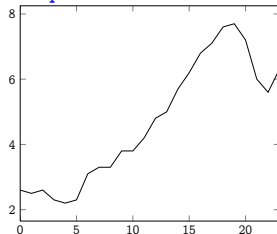
Changer l'aspect du graphique

Sympa cette couleur bleue avec les petits ronds comme marqueur, mais comment on fait pour changer ? Deux possibilités, soit tu veux tout changer, soit tu veux ajouter une modification, comme par exemple supprimer les marqueurs tout en gardant la couleur bleue. Dans le premier cas tu utilises `\addplot[options]` et dans le deuxième `\addplot+[options]`. Voici 3 exemples du même graphique :

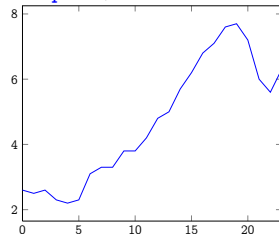
`\addplot[]`



`\addplot[mark=none]`



`\addplot+[mark=none]`




Les marqueurs

Puisque que l'on a commencé avec les marqueurs continus. Donc `mark=none` supprime les marqueurs. Il y a 3 marqueurs de base (ce sont des options à ajouter à `\addplot`, pas à `\begin{axis}`) :

`mark=*` 

`mark=x` 


`mark=+` 

En utilisant la librairie `\usetikzlibrary{plotmarks}` (à placer avant le `\begin{document}`) tu as :

`mark=-` 

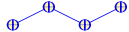
`mark=|` 

`mark=o` 


`mark=asterisk` 

`mark=star` 


`mark=oplus` 


`mark=oplus*` 


`mark=otimes` 


`mark=otimes*` 


`mark=square` 


`mark=square*` 


`mark=triangle` 


`mark=triangle*` 

`mark=diamond` 

`mark=diamond*` 

`mark=pentagon` 

`mark=pentagon*` 

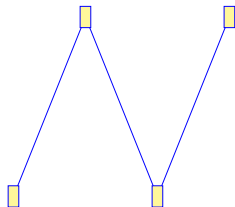
`mark=text` 

Pour `mark=texte` tu changes le texte en ajoutant l'option `text mark={texte}`

Changer la forme et la couleur des marqueurs

Tu utilises l'option `mark options={option1,option2,...}`. Voici un exemple :

```
\addplot+[mark=square*,  
mark options={fill=yellow!50,xscale=2,yscale=4}]  
coordinates {...};
```

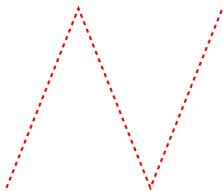


Je te renvoie aux fiches de Bébert consacrées à **TikZ & PGF** pour voir toutes les options possibles pour modifier la forme et les couleurs du marqueur.

Changer la forme et la couleur des lignes

Je te renvoie aux fiches de Bébert consacrées à **Ti**kZ & **P**GF pour voir toutes les options possibles pour modifier la forme et les couleurs des lignes. Voici un exemple :

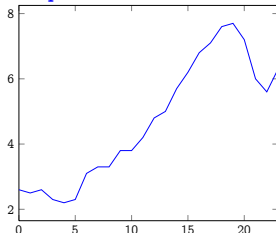
```
\addplot[mark=none, draw=red, dashed, ultra thick]  
coordinates {...};
```



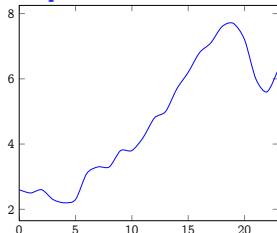
Encore des modifications

Tu peux arrondir les angles par l'option `smooth` et supprimer les lignes avec `only marks`

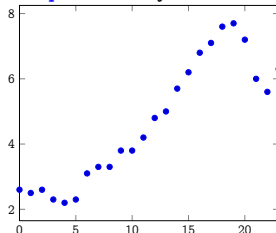
`\addplot+[mark=none]`



`\addplot+[mark=none,smooth]`



`\addplot+[only marks]`

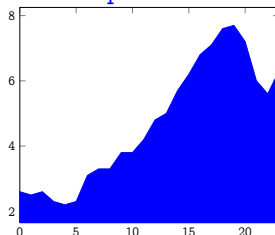


Remplir la surface sous le graphique

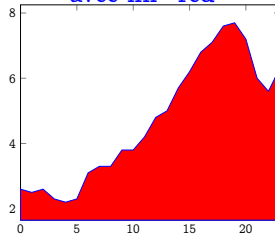
Il faut utiliser l'option `fill` et refermer le graphique avec `\closedcycle`

```
\begin{tikzpicture}
\begin{axis}[enlarge x limits=false]
\addplot+[mark=none,fill] coordinates {
(0,2.6) (1,2.5) (2,2.6) (3,2.3) (4,2.2) (5,2.3)
(6,3.1) (7,3.3) (8,3.3) (9,3.8) (10,3.8) (11,4.2)
(12,4.8) (13,5) (14,5.7) (15,6.2) (16,6.8) (17,7.1)
(18,7.6) (19,7.7) (20,7.2) (21,6) (22,5.6) (23,6.3)}\closedcycle;
\end{axis}
\end{tikzpicture}
```

l'exemple ci-dessus



avec `fill=red`



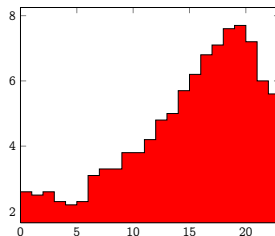
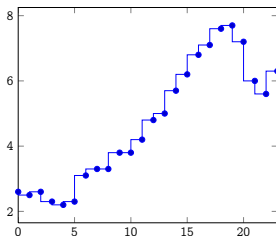
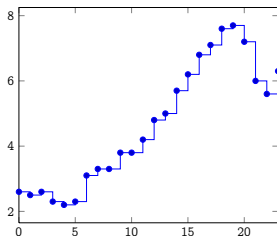
Type de graphique : les graphiques en pallier

`\addplot+[const plot]coordinate {...}` (exemple 1^{er} colonne)

Chaque point est relié par des lignes horizontales et verticales.

Par défaut les marqueurs sont placés à gauche. Tu peux le mettre à droite avec `\addplot+[const plot mark right]coordinate {...}` (2^e colonne).

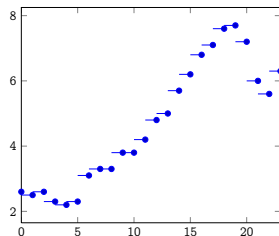
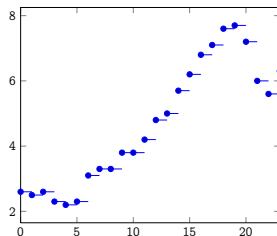
Tu peux remplir la zone `\addplot+[const plot,fill=red,draw=black]coordinate {...}` et `\closedcycle`.



Type de graphique : les graphiques en pallier, variante

Il existe une variante où tu ne traces pas les lignes verticales.

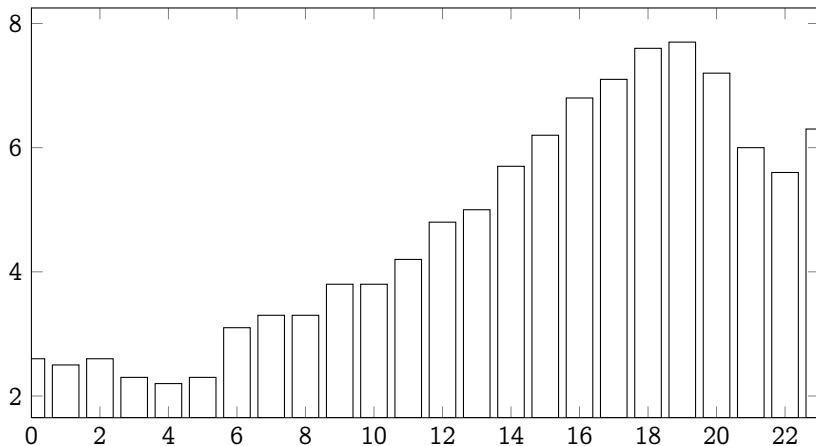
`\addplot+[jump mark left]coordinate {...}` avec les marqueurs à gauche (1^{er} colonne) et `\addplot+[jump mark right]coordinate {...}`, avec les marqueurs à droite (2^e colonne)



Type de graphique : les graphiques en barres verticales

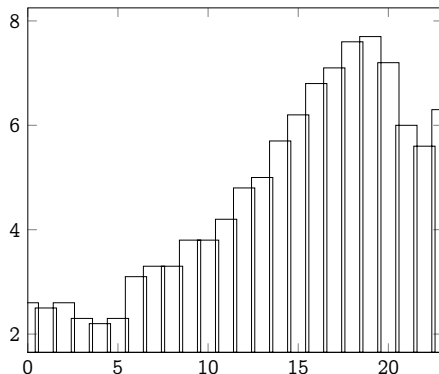
`\addplot[ybar] coordinate {...}.`

Les barres sont centrées sur la coordonnée.



Graphiques en barres : réglages

Les réglages de ces graphiques sont un peu plus délicats que les précédents, car il faut jouer sur plus de paramètres. Si je reprends notre graphique des températures et que je le transforme en graphique en barres sans options, voilà ce que l'on obtient :

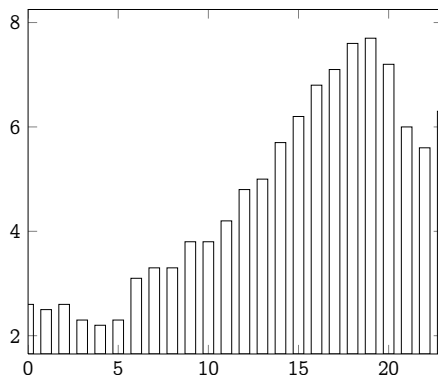


Tu as donc deux possibilités pour améliorer ton graphique, soit jouer sur la largeur des barres soit augmenter la taille de ton graphique, c'est cette dernière option que j'ai prise page précédente où j'ai mis

```
\begin{axis}[enlarge x limits=false,  
width=12cm,height=7cm]
```


Graphiques en barres : réglages

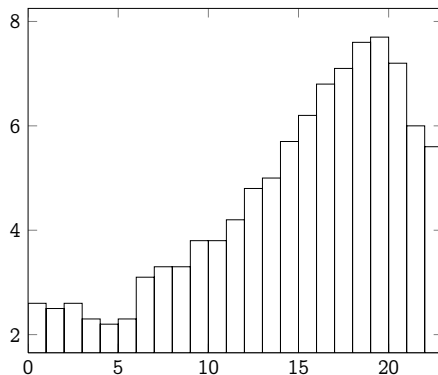
Pour modifier la largeur des barres tu utilises `bar width=`. La valeur par défaut est de 10 pt. Comme pour `TikZ` si tu ne précises pas les unités ce sont les centimètres qui sont employés. Voici notre graphique avec `\addplot[ybar,bar width=5pt] coordinate {...}` :



Graphiques en barres : réglages

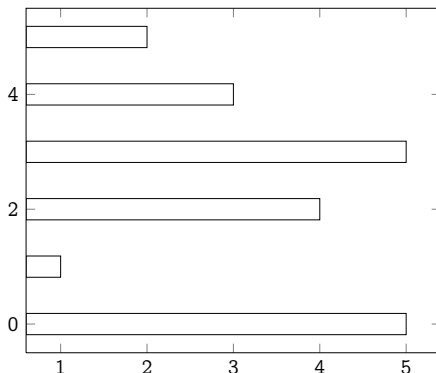
Tu peux supprimer l'espace entre les barres par `ybar interval=0`. Voici l'exemple précédent avec

```
\addplot[ybar, bar width=5pt, ybar interval=0] coordinate {...}
```



Type de graphique : les barres horizontales

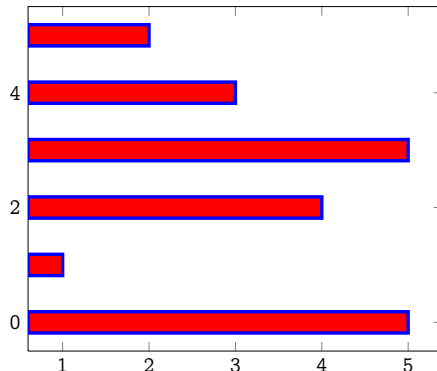
`xbar` permet de réaliser des barres horizontales. Il y a les mêmes réglages qu'avec `ybar`



Changer la couleur des barres

C'est le même principe que pour changer l'aspect d'un objet avec `TikZ`. Pour avoir des barres rouges avec un contour épais bleu, tu ajoutes les options suivantes :

```
\addplot [xbar,draw=blue,fill=red,ultra thick] coordinates  
{(1,1) (2,5) (3,4) (5,3) (4,2) (5,0)};
```

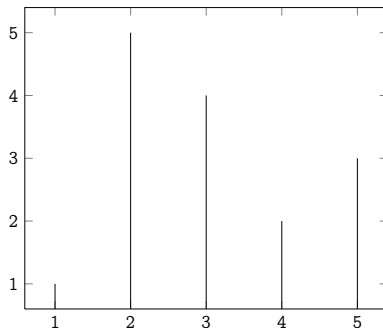
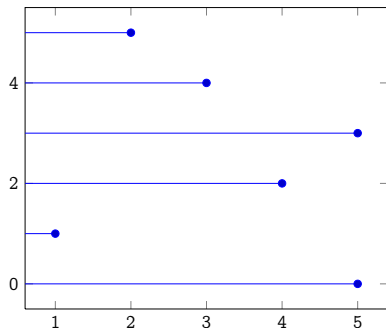


Type de graphique : les graphiques en peignes

`xcomb` et `ycomb` permettent de réaliser des graphiques en peigne, respectivement horizontalement et verticalement (sans les marqueurs pour `ycomb`, la différence est l'utilisation du `addplot+` pour `xcomb`).

```
\addplot+[xcomb] coordinate {...}
```

```
\addplot[ycomb] coordinate {...}
```

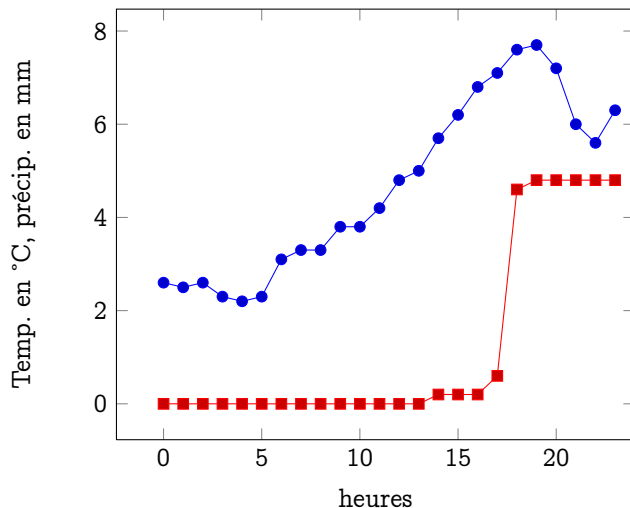


Une deuxième série de mesures, les précipitations : le code

```
\begin{tikzpicture}
\begin{axis}[xmax=24,
xlabel={heures},
ylabel={Temp. en \degres C, précip. en mm},
title={Températures relevées à Lille le 16 décembre 2004}]
\addplot coordinates {
(0,2.6) (1,2.5) (2,2.6) (3,2.3) (4,2.2) (5,2.3) (6,3.1) (7,3.3)
(8,3.3) (9,3.8) (10,3.8) (11,4.2) (12,4.8) (13,5)
(14,5.7) (15,6.2) (16,6.8) (17,7.1) (18,7.6) (19,7.7)
(20,7.2) (21,6) (22,5.6) (23,6.3)};
\addplot coordinates {
(0,0) (1,0) (2,0) (3,0) (4,0) (5,0) (6,0) (7,0) (8,0) (9,0)
(10,0) (11,0) (12,0) (13,0) (14,0.2) (15,0.2) (16,0.2) (17,0.6)
(18,4.6) (19,4.8) (20,4.8) (21,4.8) (22,4.8) (23,4.8)};
\end{axis}
\end{tikzpicture}
```

Une deuxième série de mesures, les précipitations : le résultat

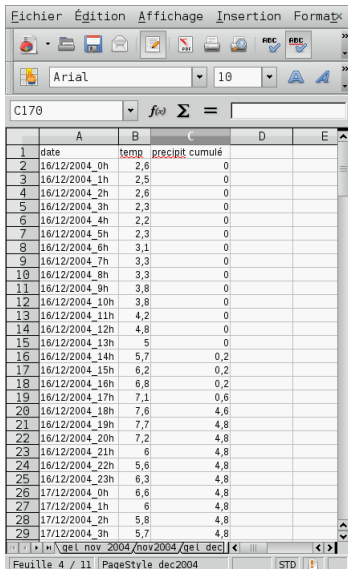
Températures et précipitations relevées à Lille le 16 décembre 2004



Charger les coordonnées à partir d'un fichier

Ré-écrire tes données dans ton source \LaTeX peut être fastidieux. Heureusement **PGFPLOTS** sait lire les données dans un fichier. Ce fichier doit être un fichier texte. **Attention** pas un document « word » (extension .doc) ou « open office » (.opt). En informatique, un fichier texte ou fichier texte brut ou fichier texte simple ou fichier ASCII, est un fichier dont le contenu représente uniquement une suite de caractères imprimables, d'espaces et de retours à la ligne (wikipédia). Ils ont en général une extension en .txt. Pour tes données tu peux leur donner une autre extension comme .dat pour les distinguer, du moment que ce sont de véritable fichier ASCII. Tu as plusieurs façons pour les obtenir soit tu les tapes à l'aide d'un éditeur de texte de type « bloc note » sous windows, soit avec « gedit » ou « Kedit », sous Linux, soit tu les exportes à partir d'un tableur.

Transformer une feuille de calcul en .txt

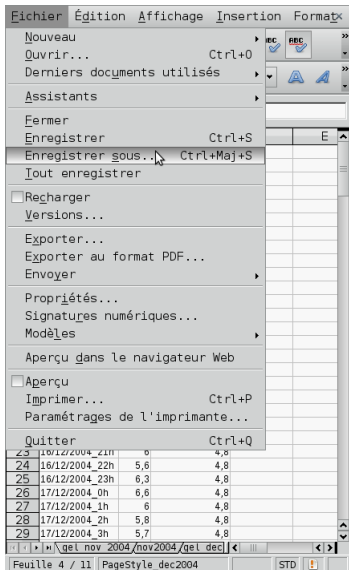


The screenshot shows a spreadsheet application window with a menu bar (Fichier, Édition, Affichage, Insertion, Format), a toolbar with icons for file operations and formatting, and a data table. The table has columns A, B, and C, with rows numbered 1 to 29. The data in the table is as follows:

	A	B	C	D	E
1	date	temp	precipit cumulé		
2	16/12/2004_0h	2,6	0		
3	16/12/2004_1h	2,5	0		
4	16/12/2004_2h	2,6	0		
5	16/12/2004_3h	2,3	0		
6	16/12/2004_4h	2,2	0		
7	16/12/2004_5h	2,3	0		
8	16/12/2004_6h	3,1	0		
9	16/12/2004_7h	3,3	0		
10	16/12/2004_8h	3,3	0		
11	16/12/2004_9h	3,8	0		
12	16/12/2004_10h	3,8	0		
13	16/12/2004_11h	4,2	0		
14	16/12/2004_12h	4,8	0		
15	16/12/2004_13h	5	0		
16	16/12/2004_14h	5,7	0,2		
17	16/12/2004_15h	6,2	0,2		
18	16/12/2004_16h	6,8	0,2		
19	16/12/2004_17h	7,1	0,6		
20	16/12/2004_18h	7,6	4,6		
21	16/12/2004_19h	7,7	4,8		
22	16/12/2004_20h	7,2	4,8		
23	16/12/2004_21h	6	4,8		
24	16/12/2004_22h	5,6	4,8		
25	16/12/2004_23h	6,3	4,8		
26	17/12/2004_0h	6,6	4,8		
27	17/12/2004_1h	6	4,8		
28	17/12/2004_2h	5,8	4,8		
29	17/12/2004_3h	5,7	4,8		

Je vais te montrer comment transformer une feuille de calcul en un document texte. Je vais le faire à partir d'un document Open Office, car il est multiplateforme, connu par beaucoup de monde et la manipulation est identique sous Linux, Windows et Mac Os X. Mais si tu es sous Linux je te conseille d'utiliser Gnumeric beaucoup plus léger (il existe une version Windows). Voici notre feuille de calculs, pour la transformer il suffit de choisir dans le menu [Fichier](#)

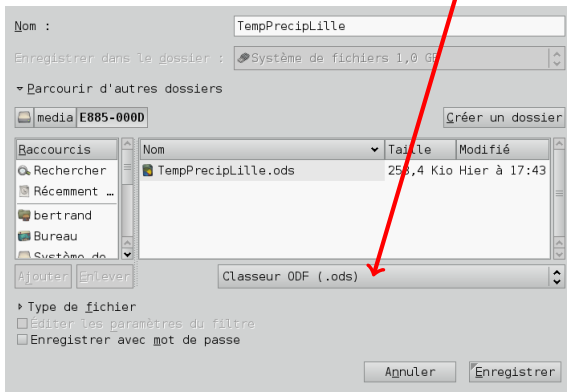
Transformer une feuille de calcul en .txt



Je vais te montrer comment transformer une feuille de calcul en un document texte. Je vais le faire à partir d'un document Open Office, car il est multiplateforme, connu par beaucoup de monde et la manipulation est identique sous Linux, Windows et Mac Os X. Mais si tu es sous Linux je te conseille d'utiliser Gnumeric beaucoup plus léger (il existe une version Windows). Voici notre feuille de calculs, pour la transformer il suffit de choisir dans le menu [Fichier](#) -> [Enregistrer sous](#).

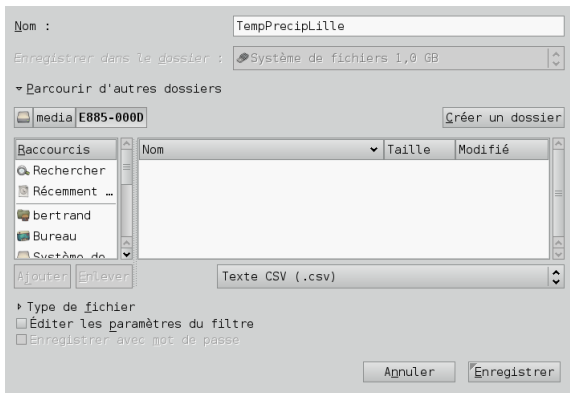
Transformer une feuille de calcul en .txt

Tu arrives sur la fenêtre suivante et tu choisis dans ce menu déroulant



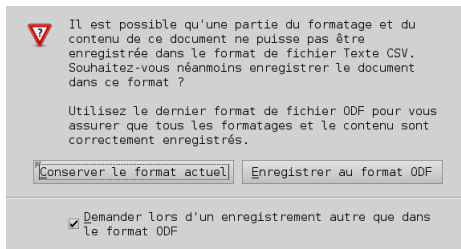
Transformer une feuille de calcul en .txt

Tu arrives sur la fenêtre suivante et tu choisis dans ce menu déroulant Texte CVS (.cvs).



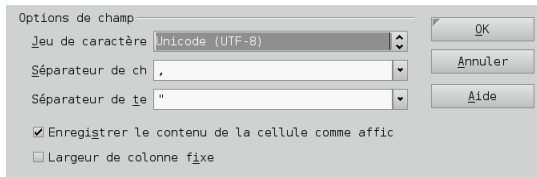
Transformer une feuille de calcul en .txt

Tu conserves le format actuel.



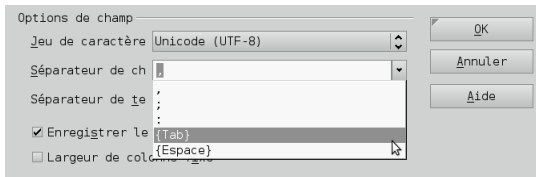
Transformer une feuille de calcul en .txt

Si le jeu de caractères n'est pas en Unicode (UTF-8), je te conseille de le mettre. Dans la mesure du possible essaye, quelque soit le logiciel de travailler en Unicode, tu seras moins embêté par les problèmes de compatibilité entre les divers systèmes d'exploitation. Tu ouvres le menu déroulant [Séparateur de ch\(amp\)](#)



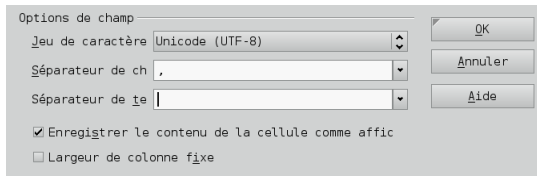
Transformer une feuille de calcul en .txt

Si le jeu de caractères n'est pas en Unicode (UTF-8), je te conseille de le mettre. Dans la mesure du possible essaye, quelque soit le logiciel de travailler en Unicode, tu seras moins embêté par les problèmes de compatibilité entre les divers systèmes d'exploitation. Tu ouvres le menu déroulant **Séparateur de ch(amp)** et tu choisis **{Tab}** ou **{Espace}**.



Transformer une feuille de calcul en .txt

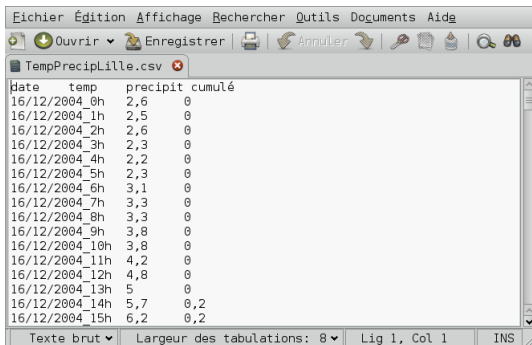
On a pas besoin de séparateur de texte, donc dans **Séparateur de Te(xte)** tu sélectionnes les guillemets et tu les effaces (touche Suppr du clavier).



Il ne te reste plus qu'à confirmer la sauvegarde en ayant éventuellement changé le nom du fichier. Tu obtiens un fichier avec une extension .cvs. Tu peux changer le nom de cette extension en .txt, .dat...

Transformer une feuille de calcul en .txt

Si ton tableur est correctement configuré pour le français en théorie le séparateur des décimales est la virgule. **PGFPLOTS** ne supportant que le point, il faut modifier le fichier. Tu l'ouvres avec un éditeur de texte (gedit sous Linux). Un Ctrl H (ou [Édition](#)→[remplacer](#)).

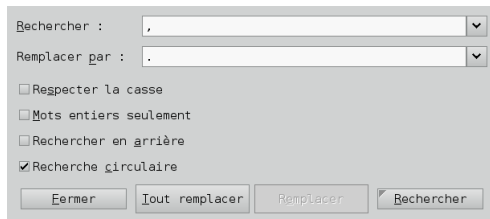


The screenshot shows a text editor window titled 'TempPrecipLille.csv'. The menu bar includes 'Fichier', 'Édition', 'Affichage', 'Rechercher', 'Outils', 'Documents', and 'Aide'. The toolbar contains icons for opening, saving, printing, undo, redo, find, and zoom. The text area displays a table with three columns: 'date', 'temp', and 'precipit cumulé'. The data represents hourly weather for December 16, 2004. The decimal separator is a comma. The status bar at the bottom shows 'Texte brut', 'Largeur des tabulations: 8', 'Lig 1, Col 1', and 'INS'.

date	temp	precipit cumulé
16/12/2004_0h	2,6	0
16/12/2004_1h	2,5	0
16/12/2004_2h	2,6	0
16/12/2004_3h	2,3	0
16/12/2004_4h	2,2	0
16/12/2004_5h	2,3	0
16/12/2004_6h	3,1	0
16/12/2004_7h	3,3	0
16/12/2004_8h	3,3	0
16/12/2004_9h	3,8	0
16/12/2004_10h	3,8	0
16/12/2004_11h	4,2	0
16/12/2004_12h	4,8	0
16/12/2004_13h	5	0
16/12/2004_14h	5,7	0,2
16/12/2004_15h	6,2	0,2

Transformer une feuille de calcul en .txt

Tu remplaces toutes les virgules par des points



N'oublie pas éventuellement de modifier les noms de champ qui contiennent des espaces. Tu peux aussi ajouter des lignes de commentaires en les précédant de # ou %. Tu sauvegardes ton fichier et tu peux maintenant l'utiliser pour tes graphiques dans **PGFPLOTS**.

Charger les coordonnées à partir d'un fichier

Quelque soit la méthode employée pour obtenir ton fichier de donnée, il doit avoir les caractéristiques suivantes :

- les colonnes doivent être séparées par un ou des espaces ou des tabulations (tu n'es pas obligé d'aligner les colonnes bien que cela soit plus facile à lire) ;
- les lignes commençant par # ou % sont considérées comme des commentaires et ne sont pas lues ;
- le format des nombres est anglo-saxon donc le séparateur de décimale est le point ;
- si il y a une colonne dans la première ligne qui n'est pas un nombre, la ligne complète est considérée comme un en-tête qui contient les noms de colonnes. **Attention** à ne pas mettre d'espace dans les noms de colonnes (« precipit_cumul » et pas « precipit cumul ») car tu créerais une nouvelle colonne (cumul) sans donnée et une erreur de compilation.

Exemples de fichiers de données

#préci pitations à Lille le 16 déc. 2004

heure	temp	precipit_cumul
-------	------	----------------

0	2.6	0
1	2.5	0
2	2.6	0
3	2.3	0
4	2.2	0
5	2.3	0
6	3.1	0
14	5.7	0.2
15	6.2	0.2
16	6.8	0.2
17	7.1	0.6
18	7.6	4.6
19	7.7	4.8
20	7.2	4.8

#préci pitations à Lille le 16 d

heure	temp	precipit_cumul
-------	------	----------------

0	2.6	0
1	2.5	0
2	2.6	0
3	2.3	0
4	2.2	0
5	2.3	0
6	3.1	0
14	5.7	0.2
15	6.2	0.2
16	6.8	0.2
17	7.1	0.6
18	7.6	4.6
19	7.7	4.8
20	7.2	4.8

Ces deux fichiers sont valables, le premier réalisé avec des tabulations est quand même plus lisible que le second où les colonnes ne sont séparées que par des espaces.

Ouvrir le fichier de données : 1^{er} méthode

```
#préci p i t a t i o n s à
#L i l l e l e 16 d é c . 2004
h      temp
0      2.6
1      2.5
2      2.6
3      2.3
4      2.2
5      2.3
6      3.1
...
14     5.7
15     6.2
16     6.8
17     7.1
18     7.6
19     7.7
20     7.2
...
```

Il y a deux façons d'ouvrir un fichier avec **PGFPLOTS**. La première est utilisée pour les fichiers ne contenant que 2 colonnes comme le fichier ci-contre. Tu utilises :

```
\addplot file {fichier.txt};
```

Attention dans l'exemple ci-contre les 2 premières ligne commençant par des **#** sont ignorées, car considérées comme des commentaires, pas de problème. Par contre la troisième ligne pose problème car il s'agit de l'entête des colonnes et pas le début des données, dans ce cas pour éviter une erreur de compilation tu dois ajouter :

```
\addplot file[skip first]{fichier.txt};
```

Tu peux également commenter la ligne des noms de colonnes dans ton fichier. Tu peux bien évidemment continuer de mettre des options :

```
\addplot[mark=none] file[skip first]{fichier.txt};
```

Ouvrir le fichier de données : 2^e méthode

La deuxième méthode permet d'ouvrir des fichiers contenant plusieurs colonnes de données comme le fichier ci-dessous (tu peux l'utiliser aussi si tu n'as que deux colonnes, c'est d'ailleurs la seule méthode que j'emploie, je la trouve plus facile à utiliser).

```
#préci pitation Lille 24-12-2004
heure    temp    precipit_cumulé
0        2.6     0
1        2.5     0
...
22       5.6     4.8
23       6.3     4.8
```

La commande est la suivante :

```
\addplot table[x=nomColonne,y=nomColonne]{fichier.txt};
```

donc pour notre exemple

```
\addplot table[x=heure,y=temp]{fichier.txt};
```

Si ton fichier n'a pas de nom de colonne tu utilises

```
\addplot table[x index=1,y index=2]{fichier.txt};
```

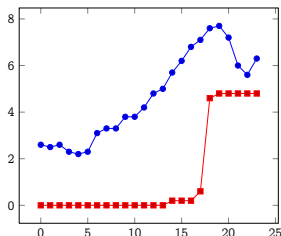
Attention la numérotation des colonnes commence à 0, ici la colonne « precipit_cumulé » a le numéro 2. Tu peux mélanger les deux

```
\addplot table[x=heure,y index=2]{fichier.txt};
```

Ouvrir le fichier de données : 2^e méthode

Notre précédent exemple de graphique, avec la superposition de la courbe des températures et celle des précipitations, peut se coder, si l'on utilise un fichier texte semblable à celui de la page précédente :

```
\begin{tikzpicture}
\begin{axis}
\addplot table[x index=0, y=temp]{fichier.txt};
\addplot table[x=heure, y=precip_t_cumulé]{fichier.txt};
\end{axis}
\end{tikzpicture}
```



La légende

Il est temps de s'occuper de la légende, en effet quand il y a plusieurs courbes sur un graphe, il est souhaitable de pouvoir les identifier.

`\begin{axis}[legend entry={légende 1,légende 2}]` permet d'afficher une légende où « légende 1 » correspond au premier `\addplot` saisi, « légende 2 » au deuxième et ainsi de suite.

```
\begin{tikzpicture}
```

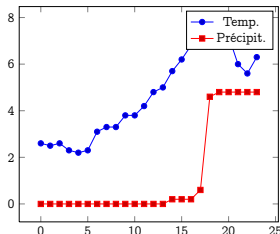
```
\begin{axis}[legend entries={Temp., Précipit.}]
```

```
\addplot table[x index=0, y=temp]{TempC.txt};
```

```
\addplot table[x=heure, y=precipit_cumulé]{TempC.txt};
```

```
\end{axis}
```

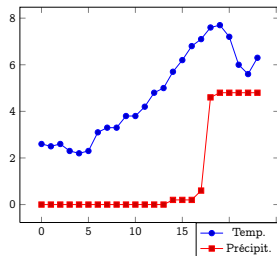
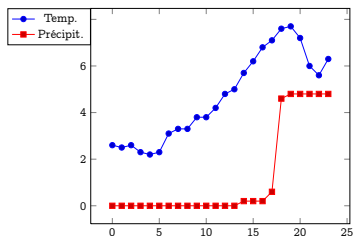
```
\end{tikzpicture}
```



Bon, l'emplacement par défaut de la légende, n'est pas du plus bel effet dans notre graphe. On va changer cela. Les dispositions du texte dans **PGFLOTS** obéissent aux mêmes règles que la position des nodes dans **TikZ**. Donc relis la fiche de Bébert « **TikZ le positionnement des nodes** » et notamment les pages 21 à 23 sur l'ancrage, tu vas en avoir besoin.

Positionner la légende

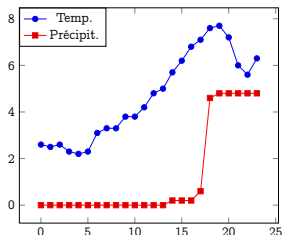
Quelque soit la taille de ton graphe, pour le positionnement de la légende **PGFPLOTS** considère que $(0,0)$ est l'angle en bas à gauche de ton graphe, $(1,1)$ l'angle en haut à droite, $(1,0)$ en haut à gauche et $(0,1)$ en bas à droite. Par défaut l'option de placement est `anchor=north east` (quand je te dis qu'il faut relire la la fiche de Bébert « **TikZ le positionnement des nodes** »). L'option pour gérer le style des légendes est : `legend style=liste options`. Pour le positionnement il faut utiliser `at={(x,y)}`. **Attention** ce n'est pas la même commande que sous **TikZ**, `at(0,1)` par exemple. Ici on a un signe égal et des accolades. Donc `\begin{axis}[legend entries={Temp.,Précipit.},legend style={at={(0,1)}}]` et `\begin{axis}[legend entries={Temp.,Précipit.},legend style={at={(1,0)}}]` donne respectivement :



Positionner la légende

Bon évidemment c'est pas le bon ancrage (t'ai-je déjà dis de lire la la fiche de Bébert « [TikZ le positionnement des nodes](#) » ?) ceci doit aller mieux :

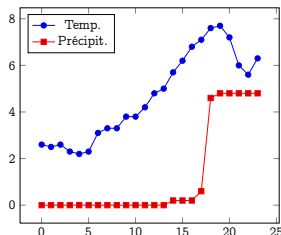
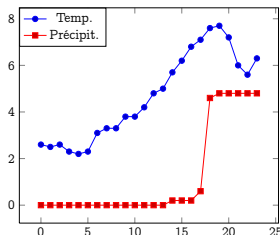
```
\begin{axis}[legend entries={Temp.,Précipit.},  
legend style={at={(0,1)},anchor=north west}]
```



Positionner la légende

Bon évidemment c'est pas le bon ancrage (t'ai-je déjà dis de lire la la fiche de Bébert « [TikZ le positionnement des nodes](#) » ?) ceci doit aller mieux :

```
\begin{axis}[legend entries={Temp.,Précipit.},  
legend style={at={(0,1)},anchor=north west}]
```

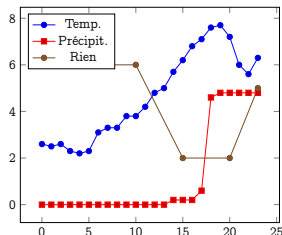


Si tu trouves que le cadre de ta légende est trop collé au cadre du graphe, fait comme ceci :

```
\begin{axis}[legend entries={Temp.,Précipit.},  
legend style={at={(0.03,0.97)},anchor=north west}]
```

Positionner la légende

Pour illustrer la commande suivante j'ai ajouté une troisième courbe qui ne signifie rien et donc appelée « Rien ».



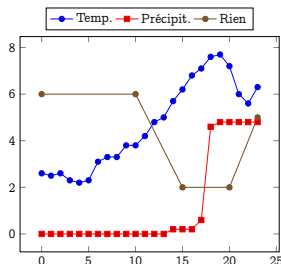
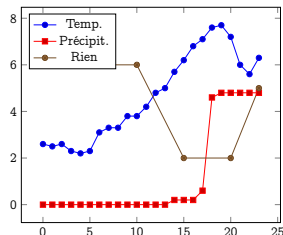
Il peut être intéressant dans ce cas d'avoir la légende disposée en ligne et non pas en colonne. La commande `legend columns={nombre}` configure le nombre d'entrée par ligne. J'ai placé la légende au-dessus du graphe :

```
\begin{axis}[legend entries={Temp.,Précipit.,Rien},  
legend style={at={(0.5,1.03)},anchor=south},legend columns=3]
```

Je te laisse essayer avec `nombre=2`.

Positionner la légende

Pour illustrer la commande suivante j'ai ajouté une troisième courbe qui ne signifie rien et donc appelée « Rien ».



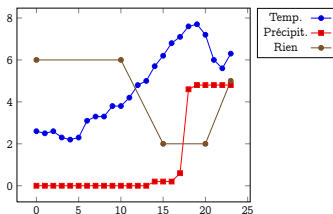
Il peut être intéressant dans ce cas d'avoir la légende disposée en ligne et non pas en colonne. La commande `legend columns={nombre}` configure le nombre d'entrée par ligne. J'ai placé la légende au-dessus du graphe :

```
\begin{axis}[legend entries={Temp.,Précipit.,Rien},  
legend style={at={(0.5,1.03)},anchor=south},legend columns=3]
```

Je te laisse essayer avec `nombre=2`.

Positionner la légende

Une dernière petite chose et on en aura fini avec les légendes. `legend plot pos=left|right|none` avec `left` par défaut permet de changer la position du symbole et de la description, ou de supprimer le symbole avec `none`.



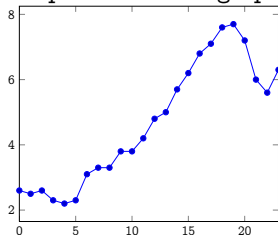
```
\begin{tikzpicture}[scale=0.5]
\begin{axis}[legend entries={Temp.,Précipit.,Rien},
legend style={at={(1.03,1)},anchor=north west},
legend plot pos=right]
\addplot table[x index=0,y index=1]{TempC.txt};
\addplot table[x=heure,y index=2]{TempC.txt};
\addplot coordinates {(0,6) (10,6) (15,2) (20,2)
(23,5)}; \end{axis}
\end{tikzpicture}
```

Je te laisse jouer avec d'autre type d'ancrage ou de coordonnées. Avec un peu de pratique tu dois être en mesure de placer la légende où tu veux et comme tu veux.

Mettre des éléments du graphiques en évidence

Pour pouvoir intervenir à l'intérieur d'un graphique il faut utiliser un système particulier de coordonnées « *axis coordinate system* », `axis cs`.

Reprenons notre graphique des températures :



Si je veux placer un point dans mon graphique, je dois utiliser les coordonnées du graphique. Pour l'exemple les coordonnées d'un point à (18,3.5) s'écriront (`axis cs :18,3.5`). Voici le code complet :

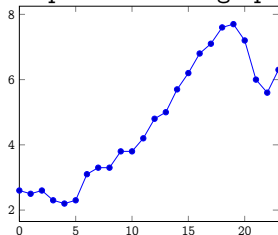
```
\begin{tikzpicture}[scale=0.5]
\begin{axis}[enlarge x limits=false]
\addplot table[x index=0,y index=1]{TempC.txt};
\draw [green,fill=green] (axis cs:18,3.5) circle (4pt);
\end{axis}
\end{tikzpicture}
```

Tu peux remplacer le point par tout objet `TikZ`, le principe reste le même et, hormis les coordonnées, reste identique au commande `TikZ`.

Mettre des éléments du graphiques en évidence

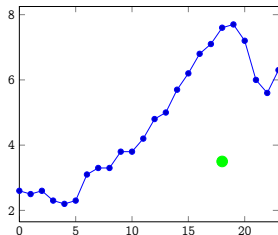
Pour pouvoir intervenir à l'intérieur d'un graphique il faut utiliser un système particulier de coordonnées « *axis coordinate system* », `axis cs`.

Reprenons notre graphique des températures :



Si je veux placer un point dans mon graphique, je dois utiliser les coordonnées du graphique. Pour l'exemple les coordonnées d'un point à (18,3.5) s'écriront (`axis cs :18,3.5`). Voici le code complet :

```
\begin{tikzpicture}[scale=0.5]
\begin{axis}[enlarge x limits=false]
\addplot table[x index=0,y index=1]{TempC.txt};
\draw [green,fill=green] (axis cs:18,3.5) circle (4pt);
\end{axis}
\end{tikzpicture}
```

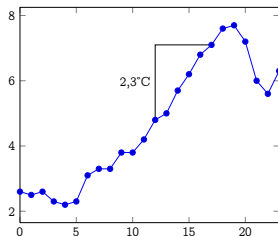


Tu peux remplacer le point par tout objet `TikZ`, le principe reste le même et, hormis les coordonnées, reste identique au commande `TikZ`.

Mettre des éléments du graphiques en évidence

Voici un exemple un peu plus complexe qui te donne une idée de ce que tu peux faire. Supposons que tu veuilles faire apparaître la différence de température entre midi et 17 h. Voici le code :

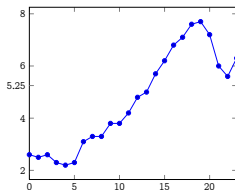
```
\begin{tikzpicture}[scale=0.5]
\begin{axis}[enlarge x limits=false]
\addplot table[x index=0,y index=1]{TempC.txt};
\draw (axis cs:12,4.8) |- (axis cs:17,7.1)
node[near start,left]{2,3°C};
\end{axis}
\end{tikzpicture}
```



Ajouter une ordonnée particulière

Dans l'exemple qui te préoccupe, tu considères que la température de 5,25°C est optimum et tu veux la faire apparaître :

```
\begin{axis}[extra y ticks={5.25}]
```



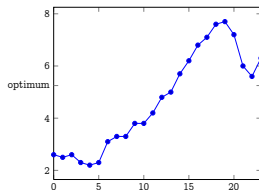
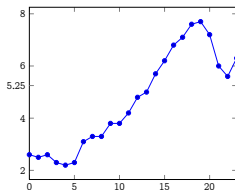
Ajouter une ordonnée particulière

Dans l'exemple qui te préoccupe, tu considères que la température de 5,25°C est optimum et tu veux la faire apparaître :

```
\begin{axis}[extra y ticks={5.25}]
```

À la place de 5.25 tu veux faire apparaître « optimum », tu ajoutes

```
\begin{axis}[extra y ticks={5.25}, extra y tick  
labels={optimum}]
```



Ajouter une ordonnée particulière

Dans l'exemple qui te préoccupe, tu considères que la température de 5,25°C est optimum et tu veux la faire apparaître :

```
\begin{axis}[extra y ticks={5.25}]
```

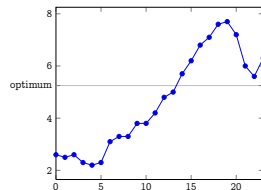
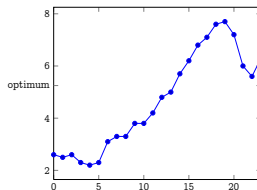
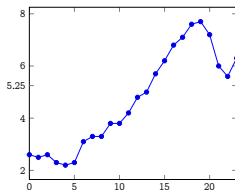
À la place de 5.25 tu veux faire apparaître « optimum », tu ajoutes

```
\begin{axis}[extra y ticks={5.25}, extra y tick
```

```
labels={optimum}]
```

Et si en plus tu veux faire figurer une ligne :

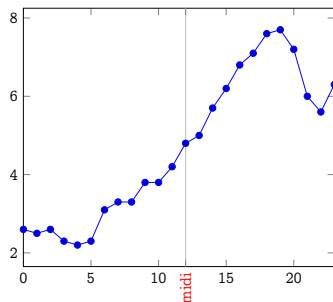
```
\begin{axis}[extra y ticks={5.25}, extra y tick  
labels={optimum}, extra y tick style={grid=major}]
```



Ajouter une ordonnée particulière

Tu peux jouer de la même manière avec les x.

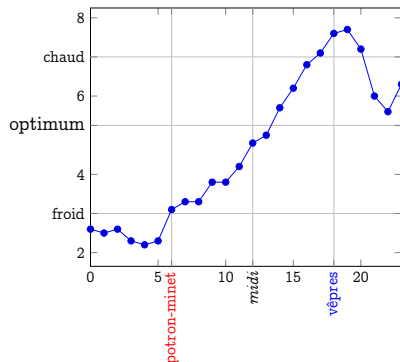
```
\begin{axis}[  
  enlarge x limits=false, extra x ticks={12},  
  extra x tick labels={\color{red}midi},  
  extra x tick style={grid=major},  
  tick label style={rotate=90,anchor=east}}  
]
```



Ajouter une ordonnée particulière

Avec encore plus d'ordonnées :

```
\begin{axis}[enlarge x limits=false,  
extra x ticks={6,12,18},  
extra x tick  
labels={\color{red}potron-minet,  
\itshape midi,\color{blue}vêpres},  
extra x tick style={grid=major},  
tick label style={rotate=90,anchor=east}},  
extra y ticks={3,5.25,7},  
extra y tick labels={froid,\large  
optimum,chaud},  
extra y tick style={grid=major}  
]
```



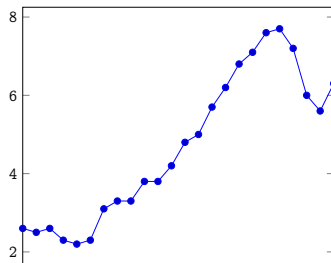
Ce graphique, c'est quand même du grand n'importe quoi, mais il te donne un aperçu des possibilités.

Les graduations

Les graduations de ton graphique sont contrôlées par `xtick` et `ytick` qui peuvent prendre les valeurs suivantes : `\empty`, `data`, `{liste de coordonnées}`.

Le rôle d' `\empty` est trivial, il supprime les divisions (**Attention**, n'oublie pas le `\`).

```
\begin{tikzpicture}[scale=0.6]
\begin{axis}[enlarge x limits=false,
xtick=\empty]
\addplot table[x index=0,
y index=1]{TempC.txt};
\end{axis}
\end{tikzpicture}
```

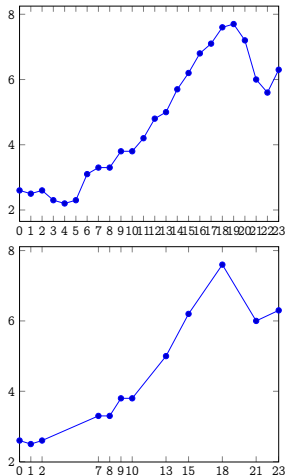


Les graduations

L'option `data` écrira des coordonnées pour tous les points et uniquement eux, précisé dans la liste de données.

```
\begin{tikzpicture}[scale=0.5]
\begin{axis}[enlarge x limits=false, xtick=data]
\addplot table[x index=0, y index=1]{TempC.txt};
\end{axis}
\end{tikzpicture}

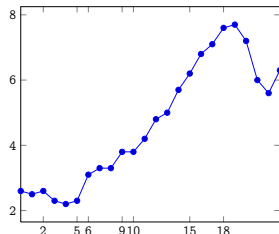
\begin{tikzpicture}[scale=0.5]
\begin{axis}[enlarge x limits=false, xtick=data]
\addplot table[x index=0, y index=1]{TempC.txt};
\addplot coordinates {
(0,2.6) (1,2.5) (2,2.6) (7,3.3) (8,3.3) (9,3.8)
(10,3.8) (13,5) (15,6.2) (18,7.6) (21,6) (23,6.3)};
\end{axis}
\end{tikzpicture}
```



Les graduations

L'option `liste coordonnées` écrira uniquement les coordonnées précisées dans la liste.

```
\begin{tikzpicture}[scale=0.5]
\begin{axis}[enlarge x limits=false,
xtick={2,5,6,9,10,15,18}]
\addplot table[x index=0, y index=1]{TempC.txt};
\end{axis}
\end{tikzpicture}
```



Comme toutes les listes `TikZ`, tu peux les abrégé par ... :

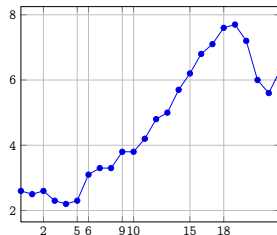
- $\{0, \dots, 9\}$ est équivalent à $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$
- $\{0, 2, \dots, 10\}$ est équivalent à $\{0, 2, 4, 6, 8, 10\}$
- $\{9, 12, \dots, 27\}$ est équivalent à $\{9, 12, 15, 18, 21, 24, 27\}$
- $\{10, \dots, 0\}$ est équivalent à $\{10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0\}$

Attention les trois points doivent être encadrés de virgules ,...,

Les graduations

Le choix des coordonnées affichées influe sur la grille :

```
\begin{tikzpicture}[scale=0.5]
\begin{axis}[enlarge x limits=false,
xtick={2,5,6,9,10,15,18},grid=major]
\addplot table[x index=0, y index=1]{TempC.txt};
\end{axis}
\end{tikzpicture}
```



Si tu ne précises pas les coordonnées à afficher, le nombre et la positions des graduations peut-être configuré par :

- `max space between ticks=nombre` où *nombre* correspond a l'espace maximum entre chaque graduation. Sa valeur par défaut est de 35 pt soit 1,2 cm. **Attention** tu ne peux pas changer les unités, c'est obligatoirement des pt, tu ne dois donc pas préciser l'unité.
- `try min ticks=nombre` où *nombre* correspond au nombre de graduations. La valeur par défaut est 4. Ce nombre, dépendant d'autres paramètres, est une indication et ne sera pas appliqué exactement, il faut le prendre comme un ordre de grandeur.

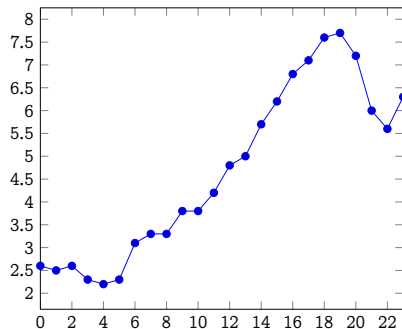
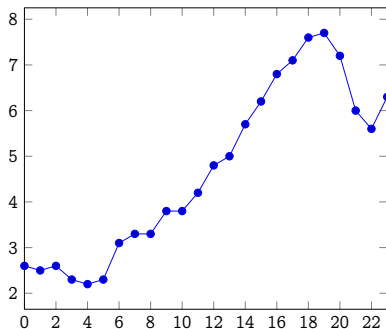
Les graduations

Un exemple à gauche avec :

```
\begin{axis}[enlarge x limits=false,max space between  
ticks={25}]
```

et à droite avec

```
\begin{axis}[enlarge x limits=false,try min ticks={10}]
```



Les graduations intermédiaires

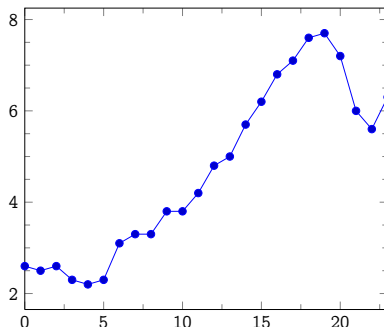
Les commandes suivantes permettent l'ajout de graduation intermédiaires.

```
minor tick num={nombre}
```

```
minor x tick num={nombre}
```

```
minor y tick num={nombre}
```

Par défaut nombre = 0. Voici un exemple avec `minor x tick num={1}`
et `minor y tick num={3}`



Les ordonnées non numériques

Le problème majeur avec **PGFPLOTS** est qu'il n'accepte que des données numériques. Imaginons que tu veuilles faire un graphique des consommations de bières des participants du Grand Concours de Buveur de Bière de Knokke-Le-Zoute. Les données sont les suivantes : Sophie 21 litres, Ginette 2l, Robert 12l, Roland 5l, et Marcel 7,5l. La commande suivante va produire une erreur :

```
\addplot coordinates{(Sophie,21)(Ginette,2)(Robert,12),(Roland,5)(Marcel,7.5)};
```

! Package PGF Math Error: Could not parse input 'Sophie' as a floating point number, sorry. The unreadable part was near 'Sophie'...

Tu n'auras pas plus de chance avec les données dans un fichier .txt
Tu vas devoir transformer tes données de cette façon :

```
\addplot coordinates{(1,21)(2,2)(3,12)(4,5)(5,7.5)};
```

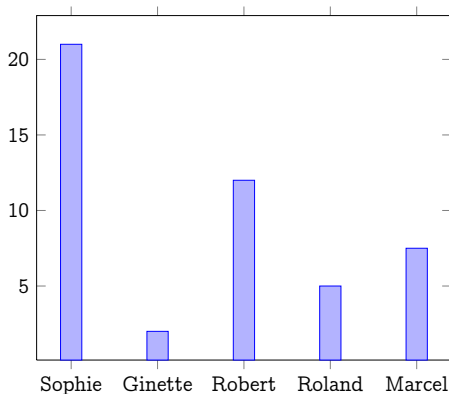
Puis remplacer les abscisses par

```
xticklabels={Sophie,Ginette,Robert,Roland,Marcel}
```

La commande `yticklabels={...}` modifie les ordonnées.

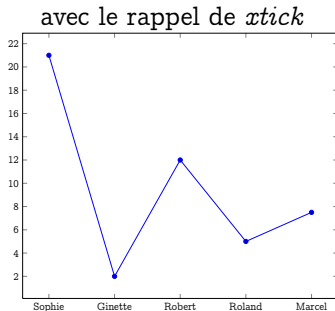
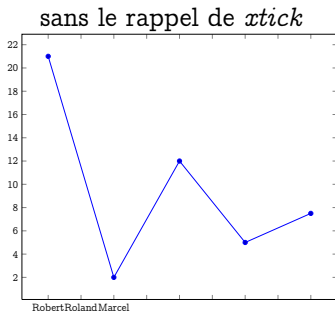
Les ordonnées non numériques

```
\begin{tikzpicture}[scale=0.7]  
\begin{axis}[ybar, xticklabel s={Sophie, Ginette, Robert, Roland, Marcel }]  
\addplot coordinates{(1, 21)(2, 2)(3, 12)(4, 5)(5, 7.5)};  
\end{axis}  
\end{tikzpicture}
```



Attention : Il est préférable de rappeler les valeurs de *xtick* avec l'option `xtick={1,2,3,4,5}` sinon tu peux te retrouver dans certains cas avec les premiers éléments perdus ou les ordonnées mal placées comme ci-dessous à gauche :

```
\begin{axis}[ybar,xtick={1,2,3,4,5},xticklabels={Sophie,Ginette,Robert,Roland,Marcel}]
```

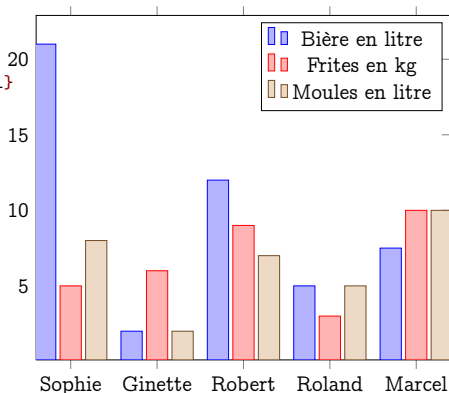


Ce problème est apparu avec des versions récentes de **PGFPLOTS**.

Les graphiques à barres multiples

Poursuivons avec notre Grand Concours de Buveur de Bière de Knokke-Le-Zoute. Les concurrents pour faire passer la bière avaient le droit d'ingurgiter des moules et des frites.

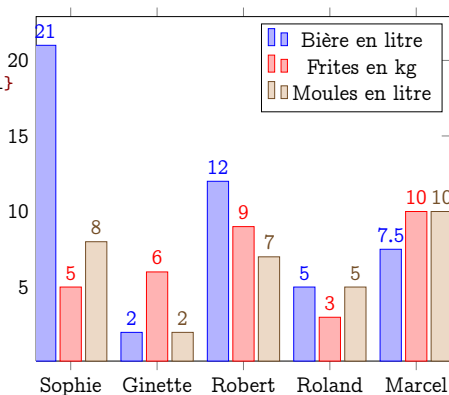
```
\begin{tikzpicture}[scale=0.7]
\begin{axis} [ybar,xtick={1,2,3,4,5},
xticklabels={Sophie,Ginette,Robert,Roland,Marcel}
legend entries=Bière en litre, Frites en kg,
Moules en litre]
\addplot
coordinates{(1,21)(2,2)(3,12)(4,5)(5,7.5)};
\addplot coordinates{(1,5)(2,6)(3,9)
(4,3)(5,10)};
\addplot coordinates{(1,8)(2,2)(3,7)
(4,5)(5,10)};
\end{axis}
\end{tikzpicture}
```



Les graphiques à barres multiples

Tu peux vouloir marquer la valeur au-dessus de chaque barre. C'est réalisable avec la commande : `nodes near coords`

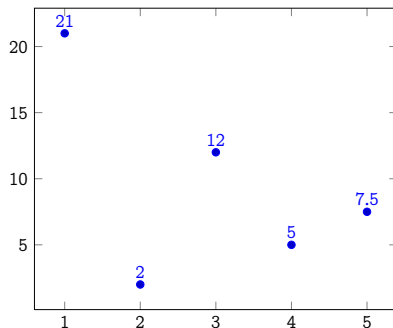
```
\begin{tikzpicture}[scale=0.7]
\begin{axis} [ybar,xtick={1,2,3,4,5},
xticklabels={Sophie,Ginette,Robert,Roland,Marcel}
legend entries=Bière en litre, Frites en kg,
Moules en litre,nodes near coords]
\addplot
coordinates{(1,21)(2,2)(3,12)(4,5)(5,7.5)};
\addplot coordinates{(1,5)(2,6)(3,9)
(4,3)(5,10)};
\addplot coordinates{(1,8)(2,2)(3,7)
(4,5)(5,10)};
\end{axis}
\end{tikzpicture}
```



Valeurs au-dessus des points

Ça marche aussi pour les autres types de graphes :

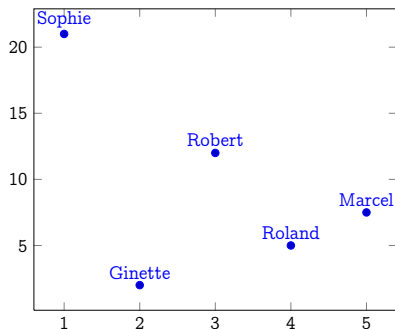
```
\begin{tikzpicture}[scale=0.7]
\begin{axis} [nodes near coords]
\addplot+[only marks]
coordinates{(1,21) (2,2) (3,12) (4,5) (5,7.5)};
\end{axis}
\end{tikzpicture}
```



Valeurs au-dessus des points

Répéter les valeurs n'est peut-être pas très utile, tu préférerais indiquer quelque chose de plus explicite. Il faut utiliser `point meta=explicit symbolic` et indiquer les valeurs à insérer entre `[]`.

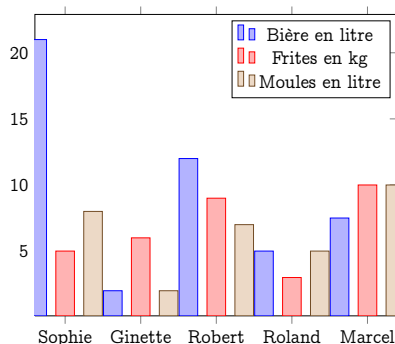
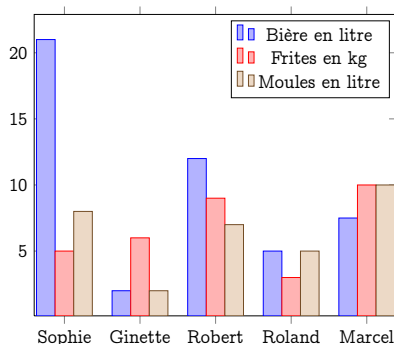
```
\begin{tikzpicture}[scale=0.7]
\begin{axis} [nodes near coords,
point meta=explicit symbolic]
\addplot+[only marks]
coordinates{(1,21) [Sophie]
(2,2) [Ginette]
(3,12) [Robert]
(4,5) [Roland]
(5,7.5) [Marcel]};
\end{axis}
\end{tikzpicture}
```



Les graphiques à barres multiples

L'espace entre les barres d'une même ordonnée se règle avec `ybar=valeurUnité`, la valeur par défaut est de 2 pt. Voici par exemple le graphique précédent avec 0 pt à gauche et 5 pt à droite. :

```
\axis[ybar=0pt,xticklabels={Sophie,...}]
```

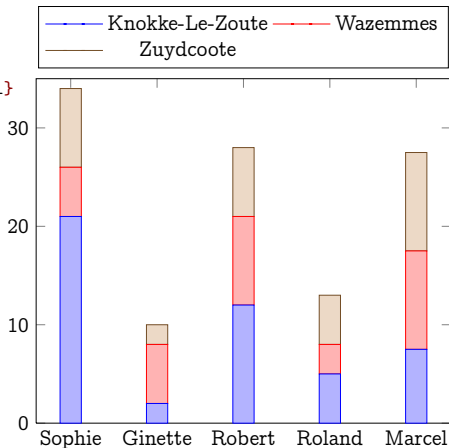


Attention si tu augmentes la valeur de `ybar=`, tu seras certainement obligé d'augmenter la largeur de ton graphique avec `width=`. Mêmes options avec `xbar=`

Les graphiques cumulés

ybar stacked ou xbar stacked : Sophie, Ginette, Robert, Roland et Marcel sont des professionnels et participent à de nombreux concours comptant pour la coupe du monde des buveurs de bières.

```
\begin{tikzpicture}[scale=0.7]
\begin{axis} [ybar stacked,xtick={1,2,3,4,5},
xticklabels={Sophie,Ginette,Robert,Roland,Marcel}
legend style={at={(0.5,1.03)},anchor=south},
legend entries={Knokke-Le-Zoute,Wazemmes,
Zuydcoote},ymin=0,ymax=35,
legend columns={2}]
\addplot coordinates{(1,21)(2,2)(3,12)
(4,5)(5,7.5)};
\addplot coordinates{(1,5)(2,6)(3,9)
(4,3)(5,10)};
\addplot coordinates{(1,8)(2,2)(3,7)
(4,5)(5,10)};
\end{axis}
\end{tikzpicture}
```



Te voilà aptes à réaliser de beaux graphiques parfaitement intégrés à ton document. Si tu es un scientifique le manuel de **PGFPLOTS** (tape dans une console texdoc pgfplots) contient beaucoup d'autre chose pour toi. Il permet également de réaliser des graphes en 3D, mais c'est une autre fiche, peut-être.