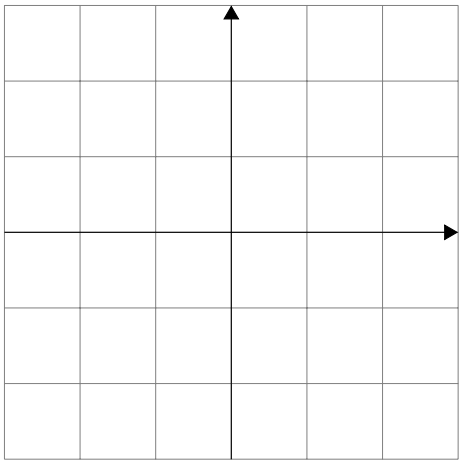
LaTeX: 2D Graphs

Dr. Yihsiang Liow (April 4, 2023)

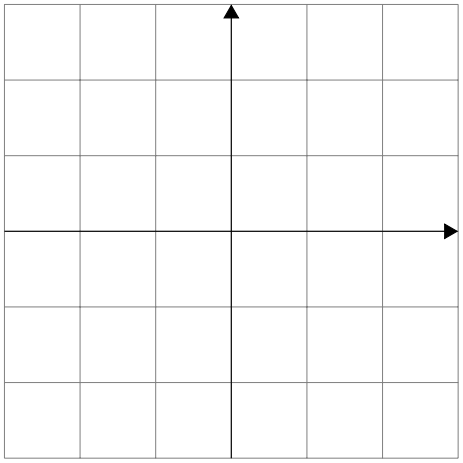
Contents

1 The xy-plane (xy-plane.tex)

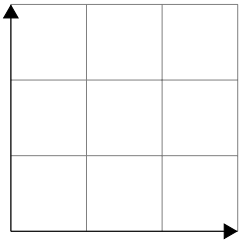
Here's the xy-plane:



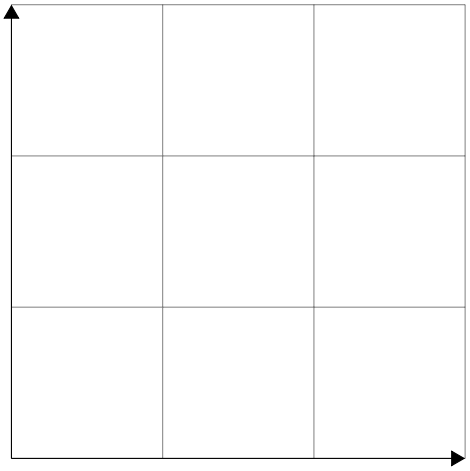
You can change the arrow tip:



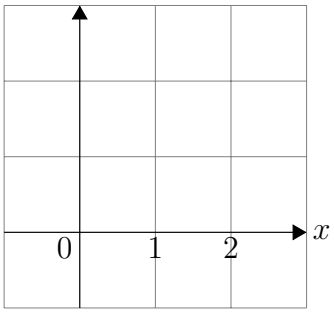
This one only has the first quadrant:



After drawing your diagram if you find it too cramped up, you don't have to rescale by hand. You just do this:



Let me add values on the x–axis:



LaTeX code

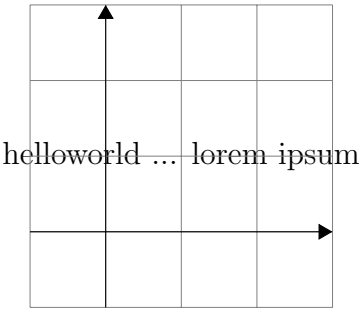
```
Here's the $xy$-plane:

\begin{center}
\begin{tikzpicture}
\draw[step=1cm,gray,very thin] (-3,-3) grid (3,3);
\draw[->] (-3,0) -- (3,0);
\draw[->] (0,-3) -- (0,3);
\end{tikzpicture}
\end{center}

You can change the arrow tip:
\begin{center}
\begin{tikzpicture}
\draw[step=1cm,gray,very thin] (-3,-3) grid (3,3);
\draw[->, >=triangle 60] (-3,0) -- (3,0);
\draw[->, >=triangle 60] (0,-3) -- (0,3);
```

```
\end{tikzpicture}
\end{center}

This one only has the first quadrant:
\begin{center}
\begin{tikzpicture}
\draw[step=1cm,gray,very thin] (0,0) grid (3,3);
\draw[->] (0,0) -- (3,0);
\draw[->] (0,0) -- (0,3);
\end{tikzpicture}
\end{center}

After drawing your diagram if you find it too cramped up,
you don't have to rescale by hand.
You just do this:
\begin{center}
\begin{tikzpicture}[scale=2]
\draw[step=1cm,gray,very thin] (0,0) grid (3,3);
\draw[->] (0,0) -- (3,0);
\draw[->] (0,0) -- (0,3);
\end{tikzpicture}
\end{center}

Let me add values on the $x$--axis:

\begin{center}
\begin{tikzpicture}
\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) -- (3,0);
\draw[->] (0,-1) -- (0,3);

\draw (-0.2,-0.2) node {0};
\draw (1,-0.2) node {1};
\draw (2,-0.2) node {2};
\draw (3.2,0) node {$x$};
\end{tikzpicture}
\end{center}
```

2 The xy-plane: text

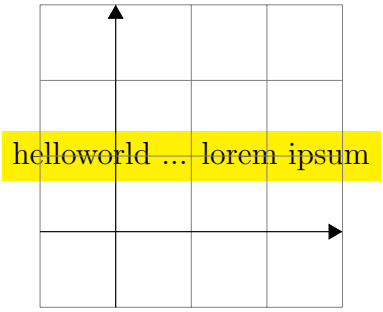
You can put a text box at $(1, 1)$:

```
\draw (1,1) node[] {helloworld ... lorem ipsum};

\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) --(3,0);
\draw[->] (0,-1) -- (0,3);
```



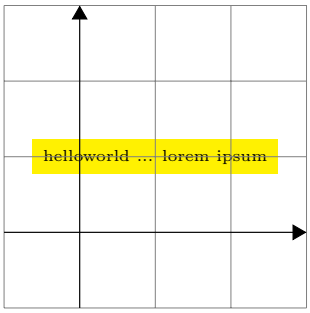
You can give text a background color:

```
\draw (1,1) node[fill=yellow] {helloworld ... lorem ipsum};

\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) --(3,0); \draw[->] (0,-1) -- (0,3);
```



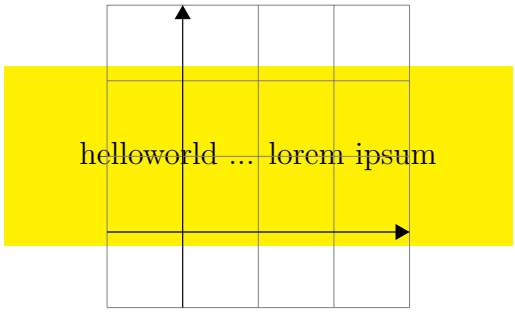
You can change the font size:

```
\draw (1,1) node[fill=yellow, font=\tiny] {helloworld ... lorem ipsum};

\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) --(3,0); \draw[->] (0,-1) -- (0,3);
```



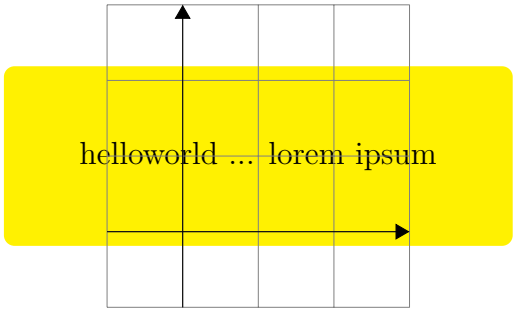
You can vary the inner sep (the border spacing):

```
\draw (1,1) node[fill=yellow, inner sep=1cm] {helloworld ... lorem ipsum};

\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) --(3,0); \draw[->] (0,-1) -- (0,3);
```



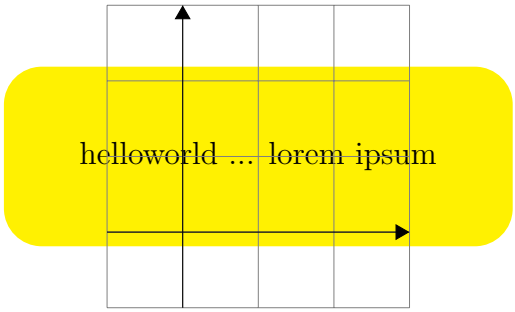
You can have rounded corners:

```
\draw (1,1) node[fill=yellow, inner sep=1cm, rounded corners]
            {helloworld ... lorem ipsum};

\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) --(3,0); \draw[->] (0,-1) -- (0,3);
```



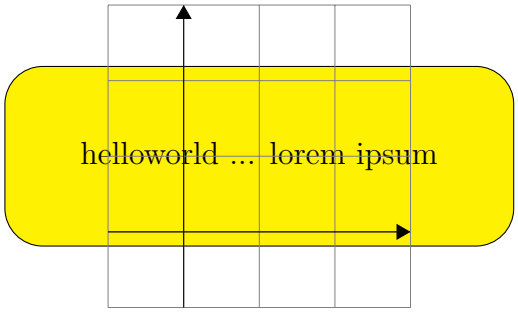
You can specify the radius of the rounded corners:

```
\draw (1,1) node[fill=yellow, inner sep=1cm, rounded corners=0.5cm]
            {helloworld ... lorem ipsum};

\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) --(3,0); \draw[->] (0,-1) -- (0,3);
```



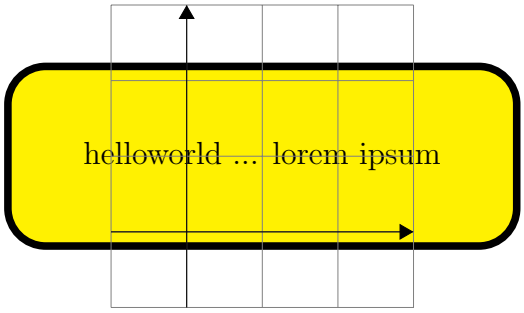
You can draw the boundary:

```
\draw (1,1) node[draw, fill=yellow, inner sep=1cm, rounded corners=0.5cm]
            {helloworld ... lorem ipsum};

\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) --(3,0); \draw[->] (0,-1) -- (0,3);
```



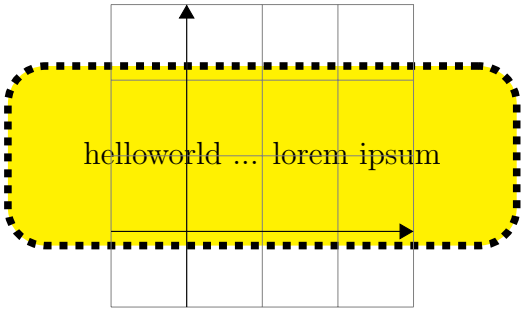
You can specify the line width:

```
\draw (1,1) node[draw, fill=yellow, inner sep=1cm, rounded corners=0.5cm,
                 line width=0.1cm]
            {helloworld ... lorem ipsum};

\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) --(3,0); \draw[->] (0,-1) -- (0,3);
```



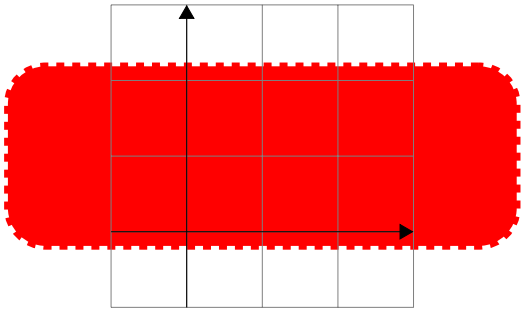
You can specify the line style:

```
\draw (1,1) node[draw, fill=yellow, inner sep=1cm, rounded corners=0.5cm,
                 line width=0.1cm, dashed]
            {helloworld ... lorem ipsum};

\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) --(3,0); \draw[->] (0,-1) -- (0,3);
```



If you specify the foreground color something goes wrong (see the section on minipage on how to fix this problem):

```
\draw (1,1) node[draw, fill=yellow, inner sep=1cm, rounded corners=0.5cm,
                 line width=0.1cm, dashed, color=red]
            {helloworld ... lorem ipsum};

\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) --(3,0); \draw[->] (0,-1) -- (0,3);
```

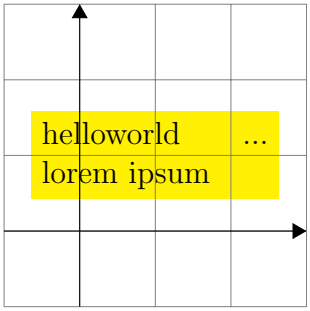


3 minipage

You can put a minipage at a node. You can specify the width and height of the minipage.

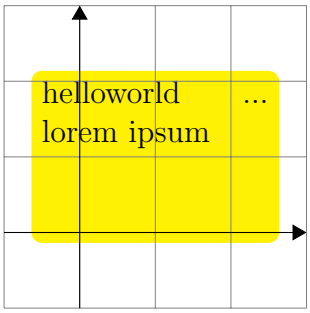
If you leave out the height:

```
\draw (1,1) node[fill=yellow] {
  \begin{minipage}[t][]{3cm}
    helloworld ... lorem ipsum
  \end{minipage}
};

\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) --(3,0); \draw[->] (0,-1) -- (0,3);
```



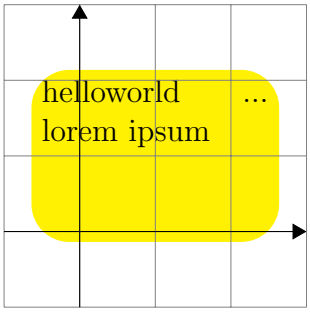
And use rounded corners and a minipage:

```
\draw (1,1) node[fill=yellow, rounded corners] {
  \begin{minipage}[t][2cm]{3cm}
    helloworld ... lorem ipsum
  \end{minipage}
};

\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) --(3,0); \draw[->] (0,-1) -- (0,3);
```



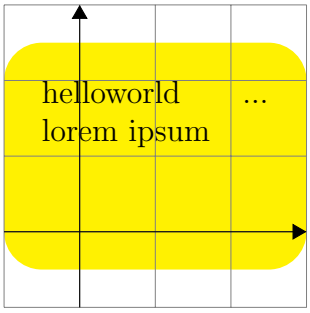
Rounded corners with specified radius:

```
\draw (1,1) node[fill=yellow, rounded corners=0.5cm] {
  \begin{minipage}[t][2cm]{3cm}
    helloworld ... lorem ipsum
  \end{minipage}
};

\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) --(3,0); \draw[->] (0,-1) -- (0,3);
```



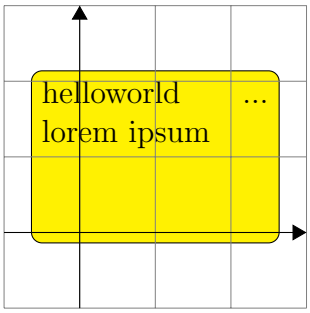
It looks better in this case to increase the inner sep (say inner sep = radius):

```
\draw (1,1) node[fill=yellow, rounded corners=0.5cm, inner sep=0.5cm] {
  \begin{minipage}[t][2cm]{3cm}
    helloworld ... lorem ipsum
  \end{minipage}
};

\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) --(3,0); \draw[->] (0,-1) -- (0,3);
```



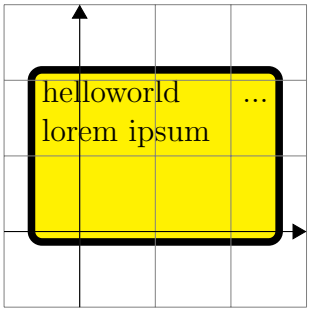
You can draw the boundary:

```
\draw (1,1) node[draw, fill=yellow, rounded corners] {
  \begin{minipage}[t][2cm]{3cm}
    helloworld ... lorem ipsum
  \end{minipage}
};

\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) --(3,0); \draw[->] (0,-1) -- (0,3);
```



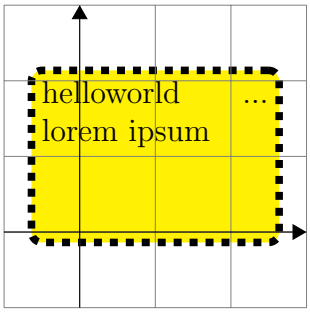
You can choose the line width for the boundary:

```
\draw (1,1) node[draw, fill=yellow, rounded corners, line width=0.1cm] {
  \begin{minipage}[t][2cm]{3cm}
    helloworld ... lorem ipsum
  \end{minipage}
};

\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) --(3,0); \draw[->] (0,-1) -- (0,3);
```



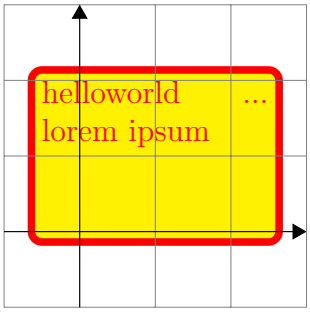
You can choose the the line style for the boundary:

```
\draw (1,1) node[draw, fill=yellow, rounded corners, line width=0.1cm,
                 dashed] {
  \begin{minipage}[t][2cm]{3cm}
    helloworld ... lorem ipsum
  \end{minipage}
};

\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) --(3,0); \draw[->] (0,-1) -- (0,3);
```



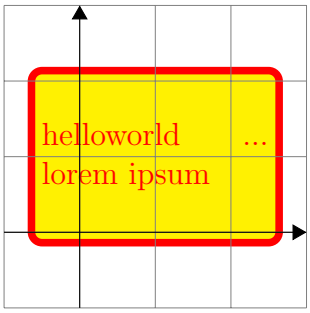
You can choose foregound color:

```
\draw (1,1) node[draw, color=red, fill=yellow, line width=0.1cm,
                 rounded corners] {
 \begin{minipage}[t][2cm]{3cm}
  helloworld ... lorem ipsum
 \end{minipage}
};

\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) --(3,0); \draw[->] (0,-1) -- (0,3);
```



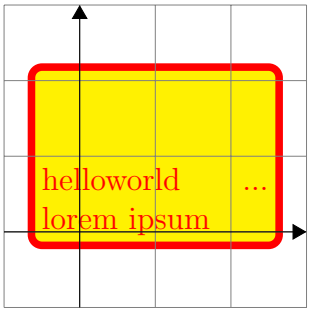
Center-justify the contents of the node:

```
\draw (1,1) node[draw, color=red, fill=yellow, line width=0.1cm,
                 rounded corners] {
 \begin{minipage}[c][2cm]{3cm}
  helloworld ... lorem ipsum
 \end{minipage}
};

\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) --(3,0); \draw[->] (0,-1) -- (0,3);
```



Bottom-justify the contents of the node:

```
\draw (1,1) node[draw, color=red, fill=yellow, line width=0.1cm,
                 rounded corners] {
 \begin{minipage}[b][2cm]{3cm}
  helloworld ... lorem ipsum
 \end{minipage}
};

\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) --(3,0); \draw[->] (0,-1) -- (0,3);
```

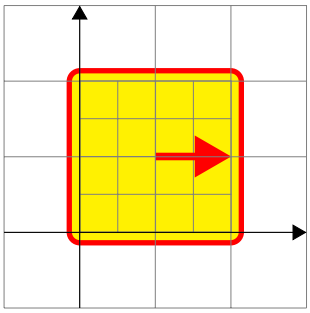


Note that the extra border (due to the inner sep) is beyond the specified width and height.

4 Nested tikzpicture

Nested tikzpicture works:

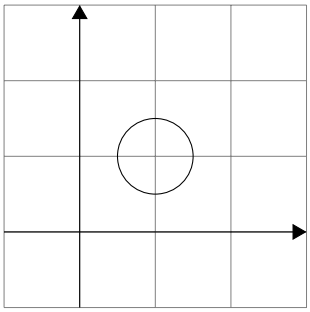
```
\begin{tikzpicture}
\draw (1,1) node[draw, color=red, fill=yellow, line width=2,
                 style=rounded corners] {%
 \begin{minipage}[b][2cm]{2cm}
  \begin{tikzpicture}
   \draw[step=0.5cm,gray,very thin] (-1,-1) grid (1,1);
   \draw[->, line width=0.1cm] (0,0) -- (1,0);
  \end{tikzpicture}
 \end{minipage}
};

\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) --(3,0);
\draw[->] (0,-1) -- (0,3);
\end{tikzpicture}
```



5 Circles and Dots

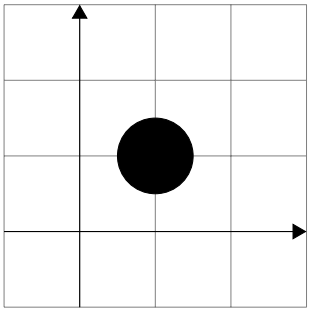
Here's a circle at (1,1) of radius 0.5:

```
\begin{center}
\begin{tikzpicture}
\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) -- (3,0);
\draw[->] (0,-1) -- (0,3);

\draw (1,1) circle (0.5cm);
\end{tikzpicture}
\end{center}
```



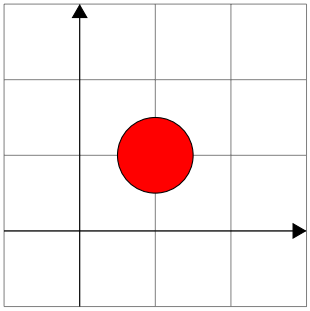
I can fill it:

```
\begin{center}
\begin{tikzpicture}
\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) -- (3,0);
\draw[->] (0,-1) -- (0,3);

\filldraw (1,1) circle (0.5cm);
\end{tikzpicture}
\end{center}
```



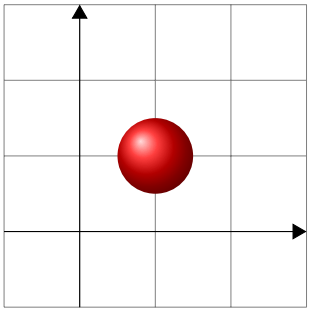
I can fill with different colors:

```
\begin{center}
\begin{tikzpicture}
\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) -- (3,0);
\draw[->] (0,-1) -- (0,3);

\filldraw[fill=red] (1,1) circle (0.5cm);
\end{tikzpicture}
\end{center}
```



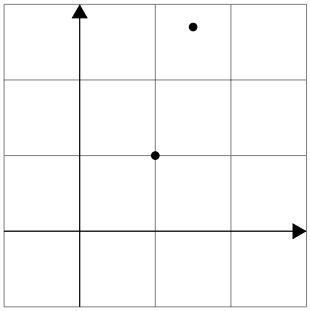
I can also shade it:

```
\begin{center}
\begin{tikzpicture}
\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) -- (3,0);
\draw[->] (0,-1) -- (0,3);

\shade[ball color=red] (1,1) circle (0.5cm);
\end{tikzpicture}
\end{center}
```



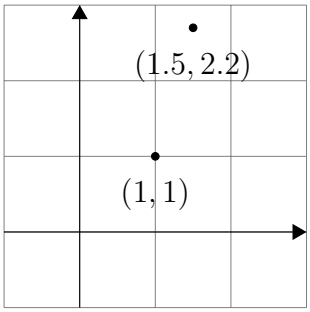
OK ... I don't think you need shaded balls on a *flat* 2-d plane. However you probably want to draw point which are just small circles:

```
\begin{center}
\begin{tikzpicture}
\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) -- (3,0);
\draw[->] (0,-1) -- (0,3);

\filldraw (1,1)     circle (0.05cm);
\filldraw (1.5,2.7) circle (0.05cm);
\end{tikzpicture}
\end{center}
```



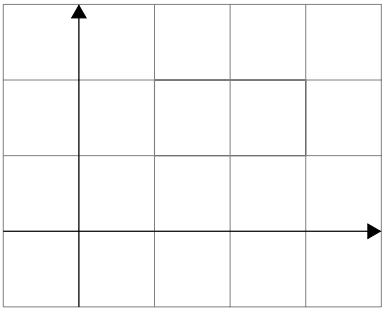
And you might want to label their coordinates:

```
\begin{center}
\begin{tikzpicture}
\draw[step=1cm,gray,very thin] (-1,-1) grid (3,3);
\draw[->] (-1,0) -- (3,0);
\draw[->] (0,-1) -- (0,3);

\filldraw (1,1)     circle (0.05cm);
\draw (1,0.5) node {$(1,1)$};

\filldraw (1.5,2.7) circle (0.05cm);
\draw (1.5,2.2) node {$(1.5,2.2)$};

\end{tikzpicture}
\end{center}
```



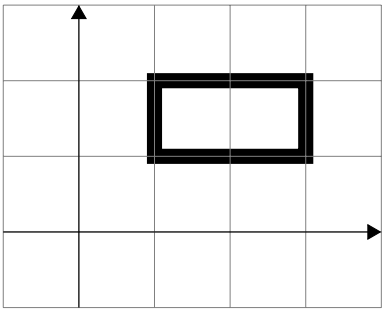
6 Rectangle

```
\draw (1,1) rectangle (3, 2);

\draw[step=1cm,gray,very thin] (-1,-1) grid (4,3);
\draw[->] (-1,0) --(4,0); \draw[->] (0,-1) -- (0,3);
```



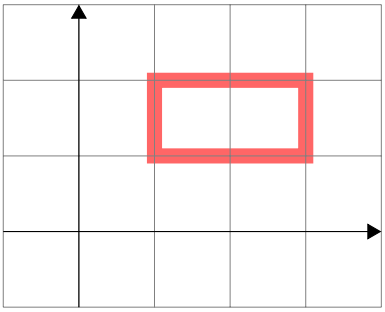
You can choose a line width:

```
\draw[line width=0.2cm] (1,1) rectangle (3, 2);

\draw[step=1cm,gray,very thin] (-1,-1) grid (4,3);
\draw[->] (-1,0) --(4,0); \draw[->] (0,-1) -- (0,3);
```



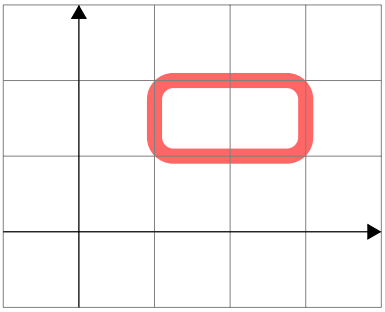
You can choose a color:

```
\draw[line width=0.2cm, color=red!60!white] (1,1) rectangle (3, 2);

\draw[step=1cm,gray,very thin] (-1,-1) grid (4,3);
\draw[->] (-1,0) --(4,0); \draw[->] (0,-1) -- (0,3);
```



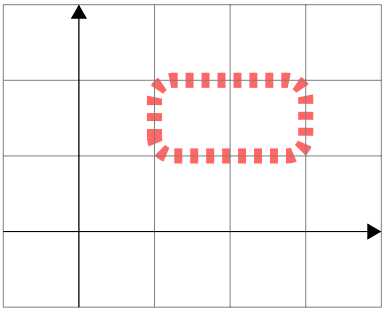
Rounded corners:

```
\draw[line width=0.2cm, color=red!60!white, rounded corners=0.25cm]
     (1,1) rectangle (3, 2);

\draw[step=1cm,gray,very thin,radius=0.5cm] (-1,-1) grid (4,3);
\draw[->] (-1,0) --(4,0); \draw[->] (0,-1) -- (0,3);
```



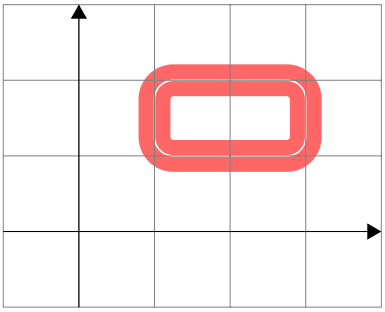
Dashed line style:

```
\draw[line width=0.2cm, color=red!60!white, rounded corners=0.25cm,
      dashed]
     (1,1) rectangle (3, 2);

\draw[step=1cm,gray,very thin,radius=0.5cm] (-1,-1) grid (4,3);
\draw[->] (-1,0) --(4,0); \draw[->] (0,-1) -- (0,3);
```



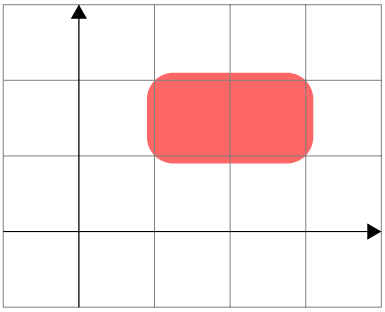
Double line style:

```
\draw[line width=0.2cm, color=red!60!white, rounded corners=0.25cm,
      double]
     (1,1) rectangle (3, 2);

\draw[step=1cm,gray,very thin,radius=0.5cm] (-1,-1) grid (4,3);
\draw[->] (-1,0) --(4,0); \draw[->] (0,-1) -- (0,3);
```



You can color the interior:

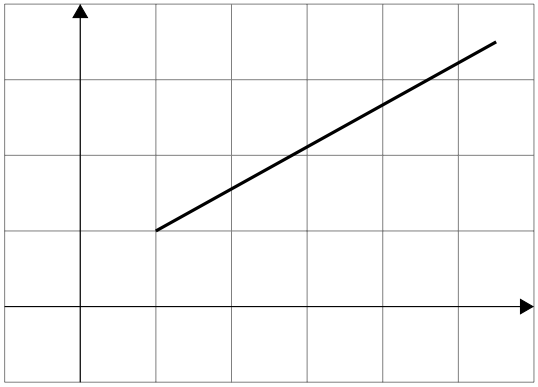
```
\filldraw[line width=0.2cm, color=red!60!white, rounded corners=0.25cm]
     (1,1) rectangle (3, 2);

\draw[fill=yellow,step=1cm,gray,very thin,radius=0.5cm] (-1,-1) grid (4,3);
\draw[->] (-1,0) --(4,0); \draw[->] (0,-1) -- (0,3);
```



7 Line

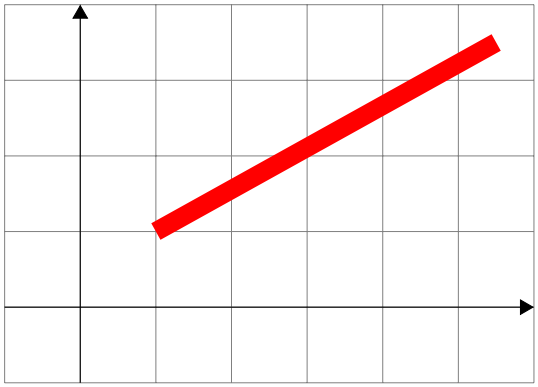
Here's a line:

```
\begin{center}
\begin{tikzpicture}
\draw[step=1cm,gray,very thin] (-1,-1) grid (6,4);
\draw[->] (-1,0) -- (6,0);
\draw[->] (0,-1) -- (0,4);

\draw[very thick] (1,1) -- (5.5,3.5);
\end{tikzpicture}
\end{center}
```



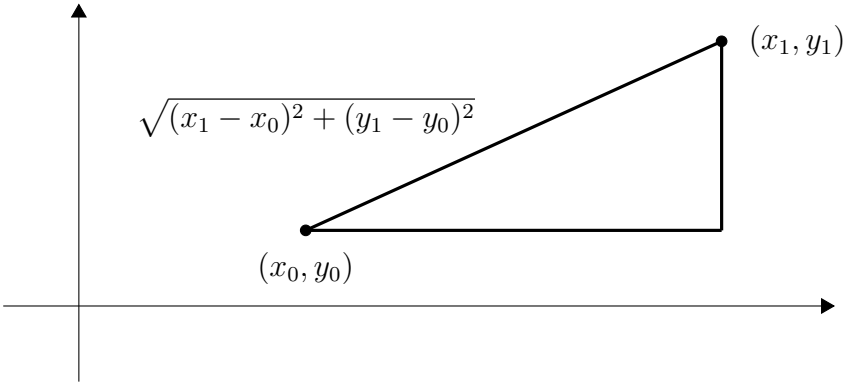
I can change the thickness and the color:

```
\begin{center}
\begin{tikzpicture}
\draw[step=1cm,gray,very thin] (-1,-1) grid (6,4);
\draw[->] (-1,0) -- (6,0);
\draw[->] (0,-1) -- (0,4);

\draw[red, line width=0.25cm] (1,1) -- (5.5,3.5);
\end{tikzpicture}
\end{center}
```



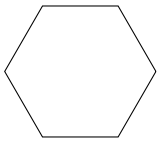
Here's Pythagorus' theorem in action (I removed the grid):

```
\begin{center}
\begin{tikzpicture}
\draw[->] (-1,0) -- (10,0);
\draw[->] (0,-1) -- (0,4);

\draw[very thick] (3,1) -- (8.5,3.5);
\draw[very thick] (8.5,1) -- (8.5,3.5);
\draw[very thick] (3,1) -- (8.5,1);

\filldraw (3,1) circle (0.07);
\filldraw (8.5,3.5) circle (0.07);

\draw (3,0.5) node {$(x_0, y_0)$};
\draw (9.5,3.5) node {$(x_1, y_1)$};
\draw (3,2.5) node {$\sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2}$};

\end{tikzpicture}
\end{center}
```



8 The xy-plane: Polar Coordinates (xy-plane-polar-coord.tex)

You can specify points in polar coordinates in the form (angle:radius). WARNING: The angle is in degrees, not radians.



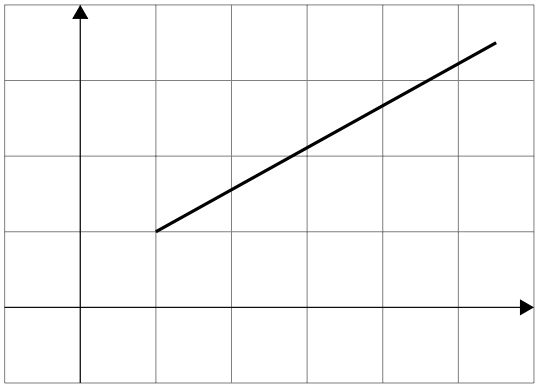
LaTeX code

```
You can specify points in polar coordinates in the form
(angle:radius). WARNING: The angle is in degrees, not radians.

\begin{center}
\begin{tikzpicture}
\draw (0:1) -- (60:1) -- (120:1)
   -- (180:1) -- (240:1) -- (300:1) -- (360:1)
;
\end{tikzpicture}
\end{center}
```

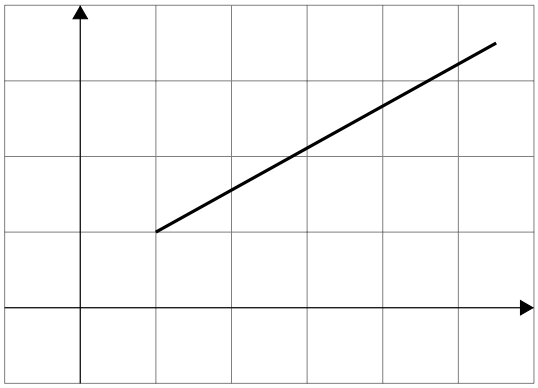
9 Variable Points and Operations (variable.tex)

You can create variables for points and do simple operations on them.

The following hardcode points:



The following is the same but uses variables and operations:



LaTeX code

```
You can create variables for points and do simple operations on them.

The following hardcode points:

\begin{center}
\begin{tikzpicture}
\draw[step=1cm,gray,very thin] (-1,-1) grid (6,4);
\draw[->] (-1,0) -- (6,0);
\draw[->] (0,-1) -- (0,4);
```

```
\draw[very thick] (1,1) -- (5.5,3.5);
\end{tikzpicture}
\end{center}

The following is the same but uses variables and operations:

\begin{center}
\begin{tikzpicture}
\draw[step=1cm,gray,very thin] (-1,-1) grid (6,4);
\draw[->] (-1,0) -- (6,0);
\draw[->] (0,-1) -- (0,4);

\path (1,1) coordinate (A);
\path (A) ++(4.5, 2.5) coordinate (B);

\draw[very thick] (A) -- (B);
\end{tikzpicture}
\end{center}
```

10 PgfPlots (pgfplots.tex)

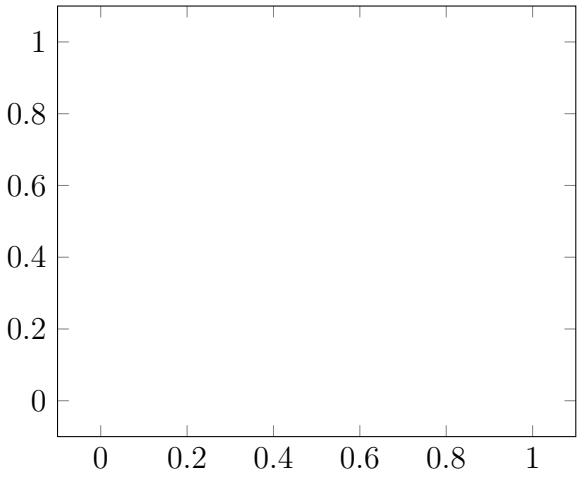
The following sections use the pgfplots library for 2D diagrams. The pgfplots library is built on top of the pgf/tikz library. It makes drawing 2D plots easier.

Pgfplots can also draw 3D plots.

LaTeX code

```
The following sections use the pgfplots library for 2D diagrams.
The pgfplots library is built on top of the pgf/tikz library.
It makes drawing 2D plots easier.

Pgfplots can also draw 3D plots.
```

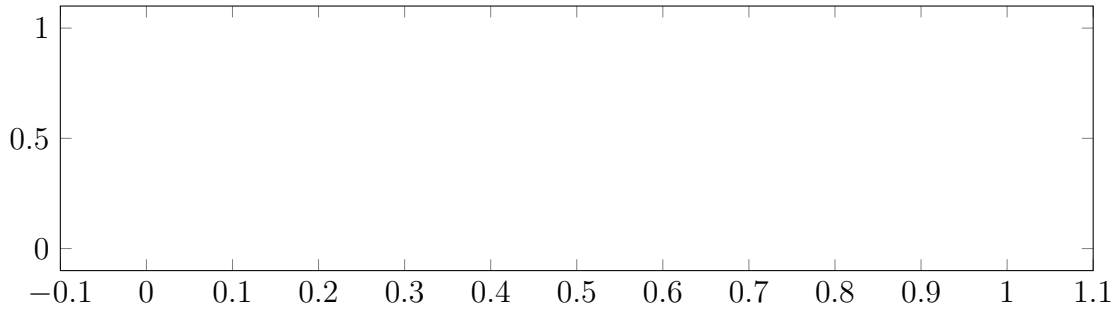
11 pgfplots: Default Window (xy-plane-default-window.tex)



LaTeX code

```
\begin{center}
\begin{tikzpicture}
\begin{axis}
\end{axis}
\end{tikzpicture}
\end{center}
```

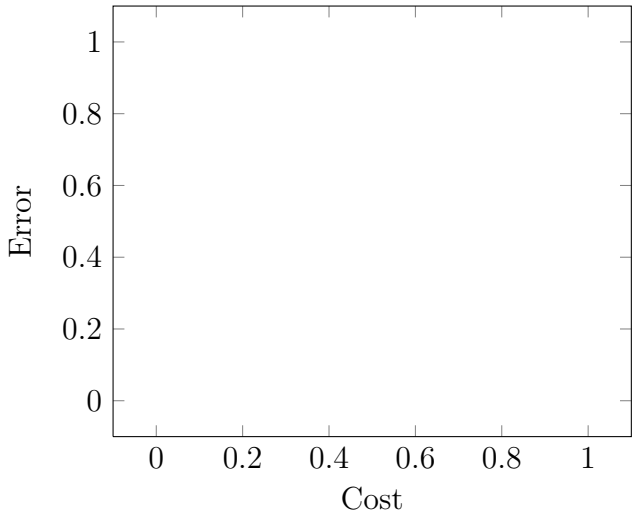
12 pgfplots: Width and Height (pgfplots-width-height.tex)



LaTeX code

```
\begin{center}
\begin{tikzpicture}
\begin{axis}[width=6in, height=2in]
\end{axis}
\end{tikzpicture}
\end{center}
```

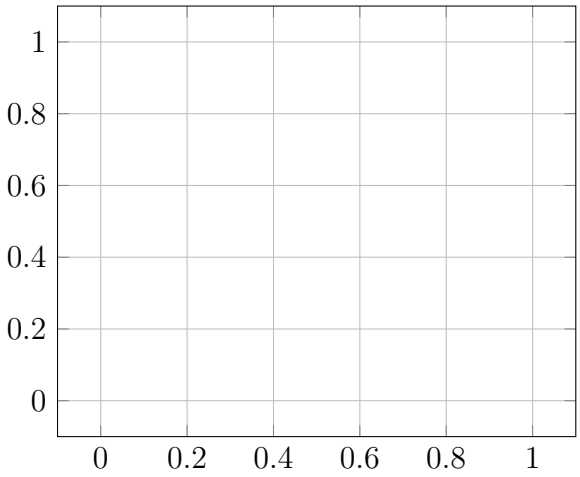
13 pgfplots: Label Axes (pgfplots-label-axis.tex)



LaTeX code

```
\begin{center}
\begin{tikzpicture}
\begin{axis}[
xlabel=Cost,
ylabel=Error]
\end{axis}
\end{tikzpicture}
\end{center}
```

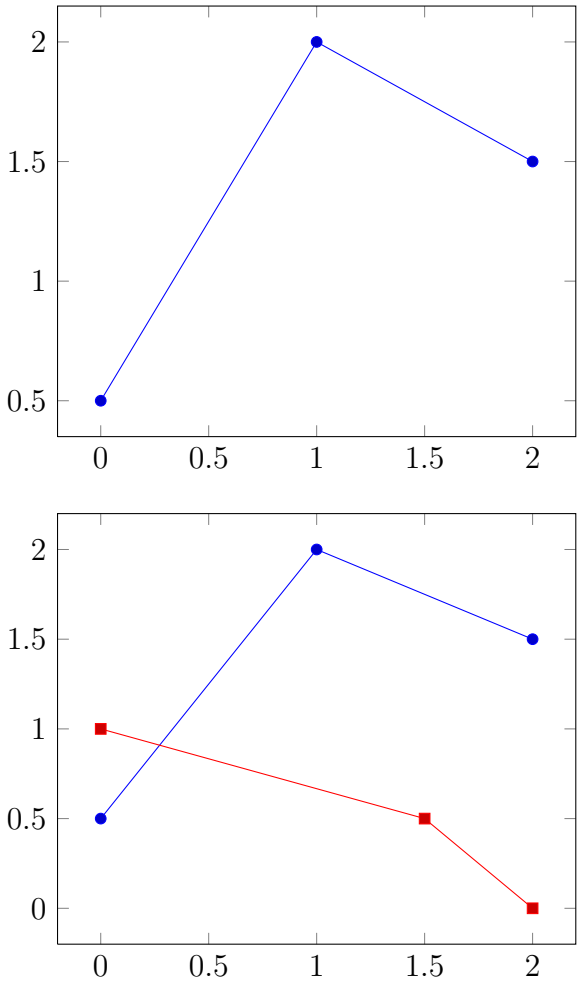
14 pgfplots: Grid (pgfplots-2d-grid.tex)



LaTeX code

```
\begin{center}
\begin{tikzpicture}[line width=2]
\begin{axis}[grid=major]
\end{axis}
\end{tikzpicture}
\end{center}
```

15 pgfplots: Line Plots (pgfplots-line.tex)

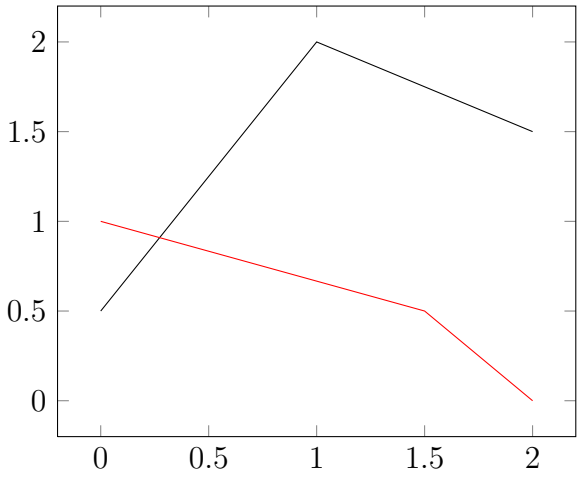


LaTeX code

```
\begin{center}
\begin{tikzpicture}
\begin{axis}
\addplot coordinates {(0,0.5) (1,2) (2,1.5)};
\end{axis}
\end{tikzpicture}
\end{center}

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

```
\addplot coordinates {(0,0.5) (1,2) (2,1.5)};
\addplot coordinates {(0,1) (1.5,0.5) (2,0)};
\end{axis}
\end{tikzpicture}
\end{center}
```

16 pgfplots: Color (pgfplots-2d-color.tex)



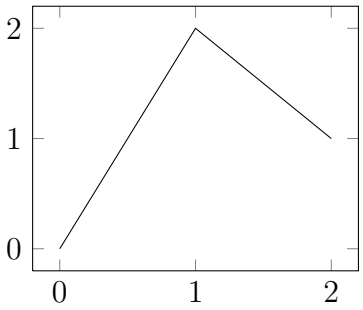
Note: If you specify a style `[...]`, then the default style is removed.

LaTeX code

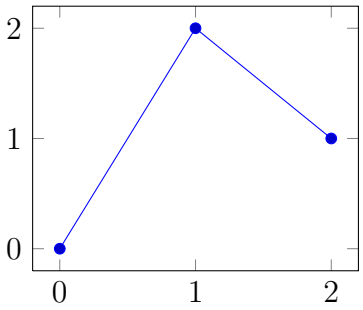
```
\begin{center}
\begin{tikzpicture}
\begin{axis}
\addplot [black] coordinates {(0,0.5) (1,2) (2,1.5)};
\addplot [red] coordinates {(0,1) (1.5,0.5) (2,0)};
\end{axis}
\end{tikzpicture}
\end{center}

Note: If you specify a style \verb![...]!, then the default style
is removed.
```

17 pgfplots: Marker styles (pgfplots-marker.tex)

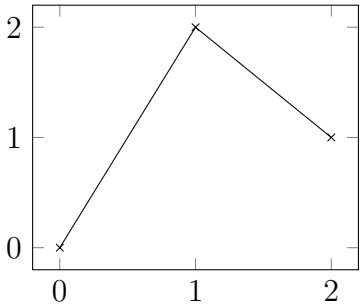
No marker:



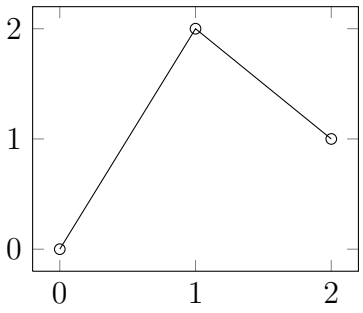
Filled dot marker (default, color default):



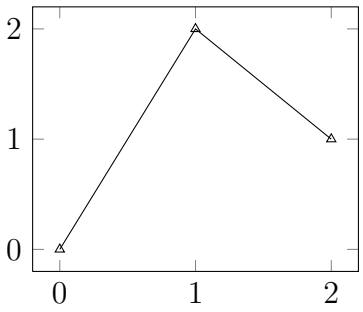
x Marker:



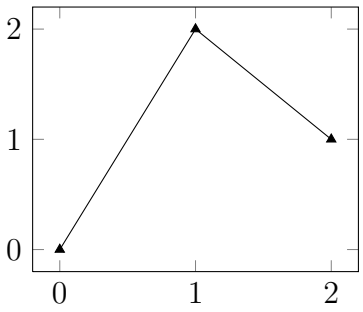
o Marker:



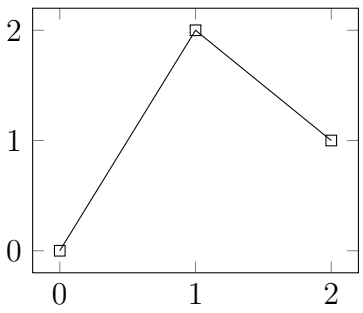
Triangle Marker:



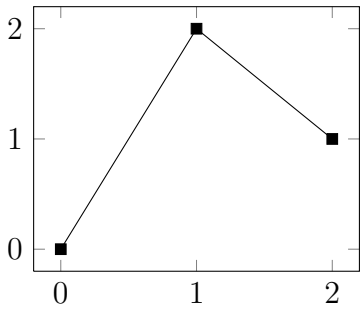
Filled Triangle Marker:



Square Marker:



Filled Square Marker:



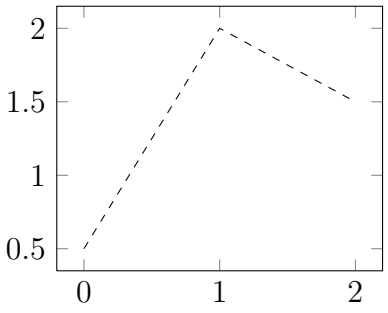
LaTeX code

```
No marker:
\begin{center}
\begin{tikzpicture}
\begin{axis}[height=2in]
\addplot[no markers] coordinates {(0,0) (1,2) (2,1)};
\end{axis}
\end{tikzpicture}
\end{center}

Filled dot marker (default, color default):
\begin{center}
\begin{tikzpicture}
\begin{axis}[height=2in]
\addplot+[sharp plot] coordinates {(0,0) (1,2) (2,1)};
\end{axis}
\end{tikzpicture}
\end{center}

x Marker:
\begin{center}
\begin{tikzpicture}
\begin{axis}[height=2in]
\addplot[mark=x] coordinates {(0,0) (1,2) (2,1)};
\end{axis}
\end{tikzpicture}
\end{center}

o Marker:
\begin{center}
\begin{tikzpicture}
```

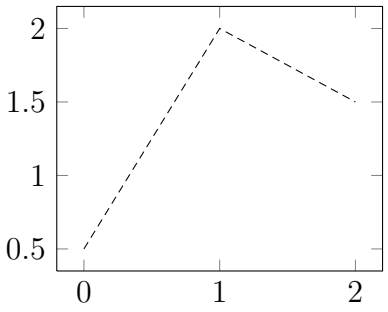
```
\begin{axis}[height=2in]
\addplot[mark=o] coordinates {(0,0) (1,2) (2,1)};
\end{axis}
\end{tikzpicture}
\end{center}

Triangle Marker:
\begin{center}
\begin{tikzpicture}
\begin{axis}[height=2in]
\addplot[mark=triangle] coordinates {(0,0) (1,2) (2,1)};
\end{axis}
\end{tikzpicture}
\end{center}

Filled Triangle Marker:
\begin{center}
\begin{tikzpicture}
\begin{axis}[height=2in]
\addplot[mark=triangle*] coordinates {(0,0) (1,2) (2,1)};
\end{axis}
\end{tikzpicture}
\end{center}

Square Marker:
\begin{center}
\begin{tikzpicture}
\begin{axis}[height=2in]
\addplot[mark=square] coordinates {(0,0) (1,2) (2,1)};
\end{axis}
\end{tikzpicture}
\end{center}

Filled Square Marker:
\begin{center}
\begin{tikzpicture}
\begin{axis}[height=2in]
\addplot[mark=square*] coordinates {(0,0) (1,2) (2,1)};
\end{axis}
\end{tikzpicture}
\end{center}
```

18 pgfplots: Dashed Styles (pgfplots-dashed.tex)

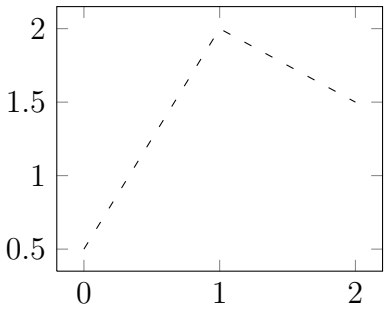
Dashed:



Densely dashed:



Loosely dashed:



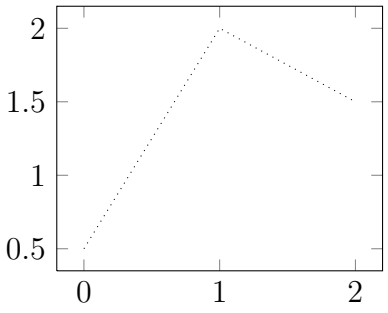
LaTeX code

```
Dashed:

\begin{center}
\begin{tikzpicture}
\begin{axis}[height=2in]
```

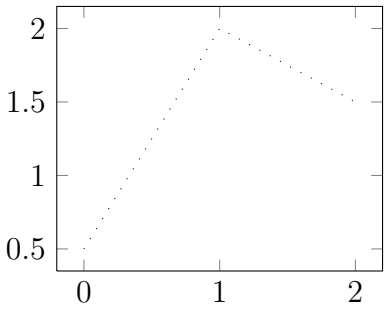
```
\addplot[sharp plot, dashed] coordinates
{(0,0.5) (1,2) (2,1.5)};
\end{axis}
\end{tikzpicture}
\end{center}

Densely dashed:

\begin{center}
\begin{tikzpicture}
\begin{axis}[height=2in]
\addplot[sharp plot, densely dashed] coordinates
{(0,0.5) (1,2) (2,1.5)};
\end{axis}
\end{tikzpicture}
\end{center}

Loosely dashed:

\begin{center}
\begin{tikzpicture}
\begin{axis}[height=2in]
\addplot[sharp plot, loosely dashed] coordinates
{(0,0.5) (1,2) (2,1.5)};
\end{axis}
\end{tikzpicture}
\end{center}
```

19 pgfplots: Dotted Styles (pgfplots-dotted.tex)

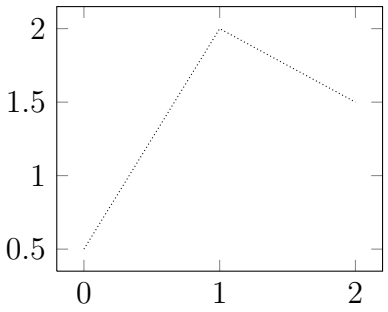
Dotted:



Loosely dotted:



Densely dotted:



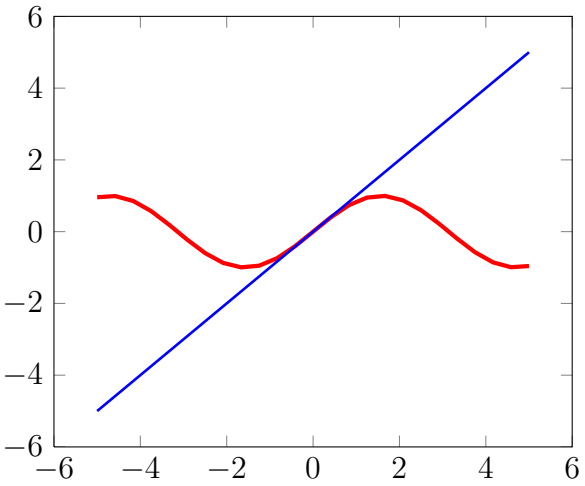
LaTeX code

```
Dotted:

\begin{center}
\begin{tikzpicture}
\begin{axis}[height=2in]
```

```
\addplot[sharp plot, dotted] coordinates
{(0,0.5) (1,2) (2,1.5)};
\end{axis}
\end{tikzpicture}
\end{center}

Loosely dotted:

\begin{center}
\begin{tikzpicture}
\begin{axis}[height=2in]
\addplot[sharp plot, loosely dotted] coordinates
{(0,0.5) (1,2) (2,1.5)};
\end{axis}
\end{tikzpicture}
\end{center}

Densely dotted:

\begin{center}
\begin{tikzpicture}
\begin{axis}[height=2in]
\addplot[sharp plot, densely dotted] coordinates
{(0,0.5) (1,2) (2,1.5)};
\end{axis}
\end{tikzpicture}
\end{center}
```

20 pgfplots: Functions (pgfplots-2d-function.tex)

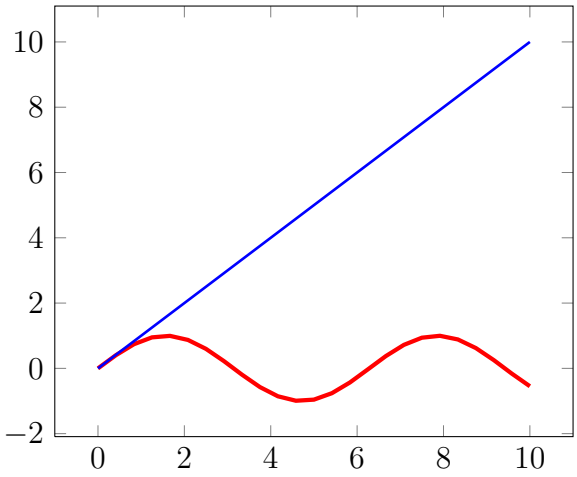
[WARNING: angles are in degrees, not radians.]



LaTeX code

```
[WARNING: angles are in degrees, not radians.]

\begin{center}
\begin{tikzpicture}[line width=2]
\begin{axis}
\addplot[draw=red, ultra thick] {sin(180*x/3.14)};
\addplot[draw=blue, line width=1] {x};
\end{axis}
\end{tikzpicture}
\end{center}
```

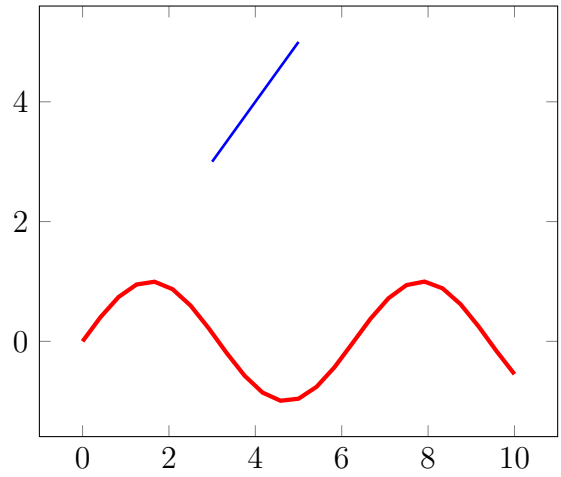
21 pgfplots: Domain for all Functions (xy-plane-domain-all.tex)



LaTeX code

```
\begin{center}
\begin{tikzpicture}[line width=2]
\begin{axis}[domain=0:10]
\addplot[draw=red, ultra thick] {sin(180*x/3.14)};
\addplot[draw=blue, line width=1] {x};
\end{axis}
\end{tikzpicture}
\end{center}
```

22 pgfplots: Domain for each Function (xy-plane-domain-each-function.tex)

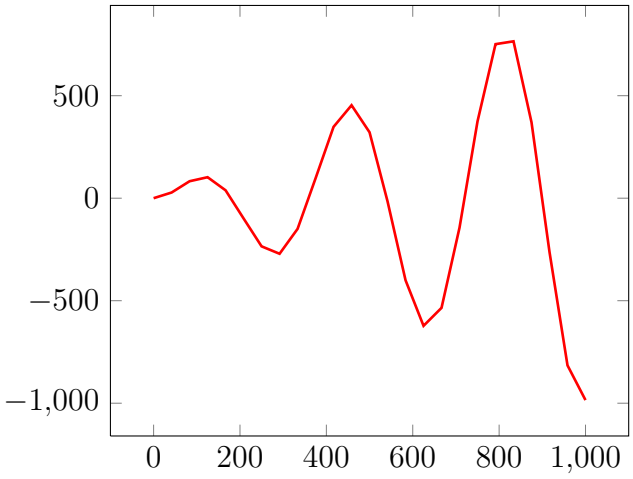


LaTeX code

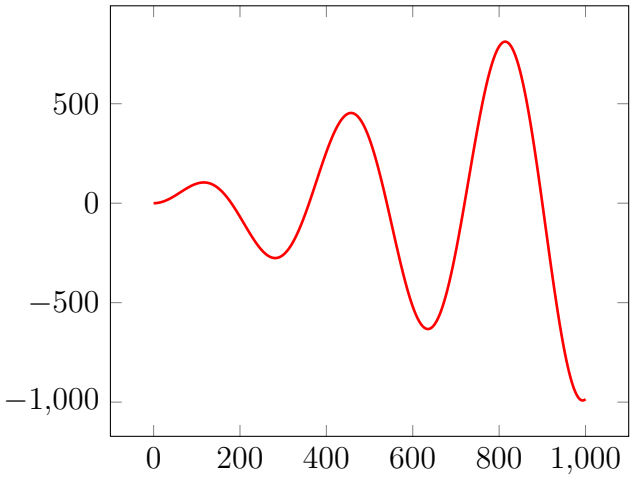
```
\begin{center}
\begin{tikzpicture}[line width=2]
\begin{axis}
\addplot[draw=red, domain=0:10, ultra thick] {sin(180*x/3.14)};
\addplot[draw=blue, domain=3:5, line width=1] {x};
\end{axis}
\end{tikzpicture}
\end{center}
```

23 pgfplots: Sample Size (pgfplots-sample-size.tex)

Default sample size:



Sample size = 1000:

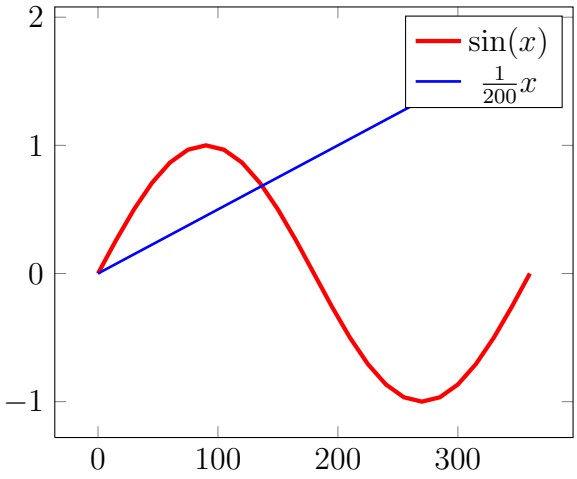


LaTeX code

```
Default sample size:

\begin{center}
\begin{tikzpicture}[line width=2]
\begin{axis}[domain=0:1000]
\addplot[draw=red, line width=1] {x * sin(x)};
\end{axis}
```

```
\end{tikzpicture}
\end{center}

Sample size = 1000:

\begin{center}
\begin{tikzpicture}[line width=2]
\begin{axis}[domain=0:1000]
\addplot[draw=red, line width=1, samples=1000] {x * sin(x)};
\end{axis}
\end{tikzpicture}
\end{center}
```

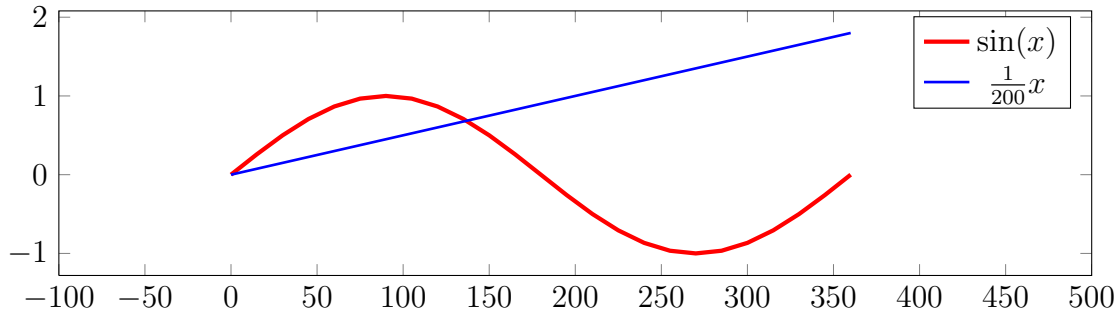
24 pgfplots: Legend (xy-plane-legend.tex)



LaTeX code

```
\begin{center}
\begin{tikzpicture}[line width=2]
\begin{axis}
\addplot[draw=red, domain=0:360, ultra thick] {sin(x)};
\addplot[draw=blue, domain=0:360, line width=1] {x/200};
\legend{$\sin(x)$,$\frac{1}{200}x$}
\end{axis}
\end{tikzpicture}
\end{center}
```

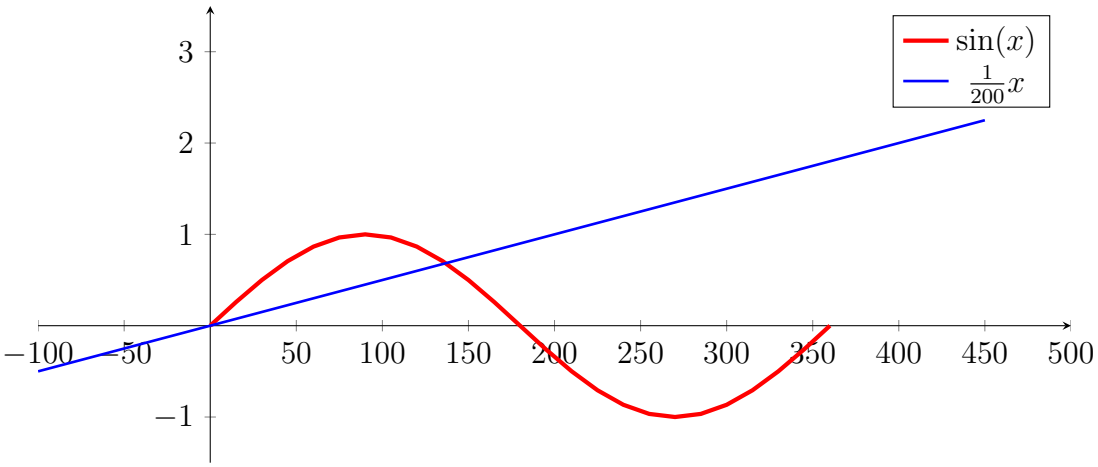
25 pgfplots: Max and min for axes (xy-plane-axes-max-min.tex)



LaTeX code

```
\begin{center}
\begin{tikzpicture}[line width=2]
\begin{axis}[width=6in, height=2in, xmin=-100, xmax=500]
\addplot[draw=red, domain=0:360, ultra thick] {sin(x)};
\addplot[draw=blue, domain=0:360, line width=1] {x/200};
\legend{$\sin(x)$,$\frac{1}{200}x$}
\end{axis}
\end{tikzpicture}
\end{center}
```

26 pgfplots: Place Axes in Middle (xy-plane-place-axes.tex)

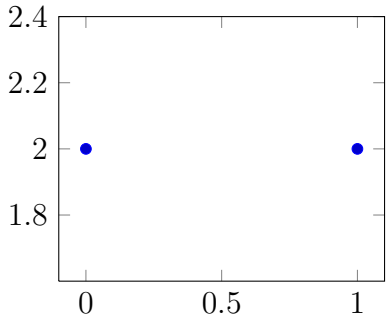


LaTeX code

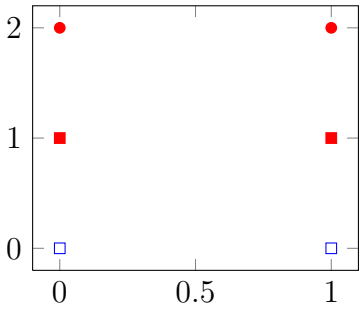
```
\begin{center}
\begin{tikzpicture}[line width=2]
\begin{axis}[
  width=6in, height=3in,
  xmin=-100, xmax=500,
  ymin=-1.5, ymax=3.5,
  axis x line=middle,
  axis y line=middle,
]
\addplot[draw=red, domain=0:360, ultra thick] {sin(x)};
\addplot[draw=blue, domain=-100:450, line width=1] {x/200};
\legend{$\sin(x)$,$\frac{1}{200}x$}
\end{axis}
\end{tikzpicture}
\end{center}
```

27 pgfplots: Points (pgfplots-points.tex)

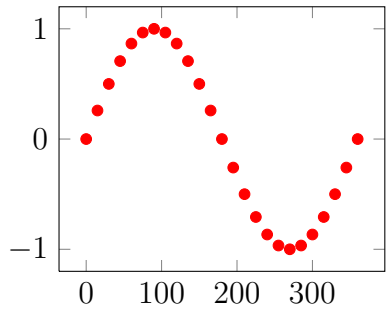
Default style:



User-specified styles:



Points from functions:



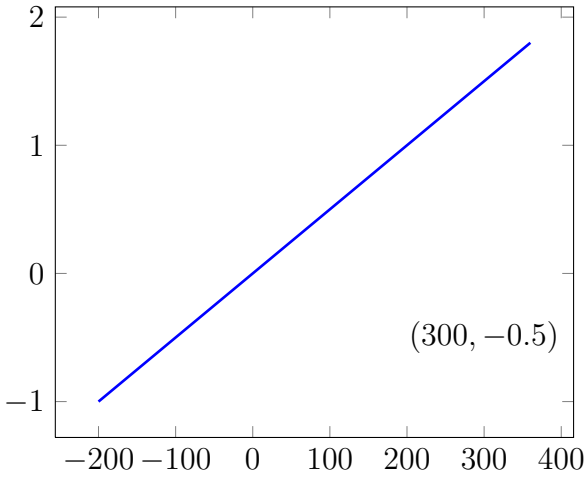
LaTeX code

```
Default style:

\begin{center}
\begin{tikzpicture}
```

```
\begin{axis}[height=2in]
\addplot+[only marks] coordinates {(0, 2) (1, 2)};
\end{axis}
\end{tikzpicture}
\end{center}

User-specified styles:

\begin{center}
\begin{tikzpicture}
\begin{axis}[height=2in]

\addplot[mark=*,draw=red,fill=red,only marks] coordinates
{(0, 2) (1, 2)};

\addplot[mark=square*, draw=red, fill=red, only marks] coordinates
{(0, 1) (1, 1)};

\addplot[mark=square, draw=blue, fill=blue, only marks] coordinates
{(0, 0) (1, 0)};

\end{axis}
\end{tikzpicture}
\end{center}

Points from functions:
\begin{center}
\begin{tikzpicture}
\begin{axis}[height=2in]

\addplot[mark=*,draw=red,fill=red,only marks, domain=0:360] {sin(x)};

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

28 pgfplots: Insert PGF node (pgfplots-insert-node.tex)

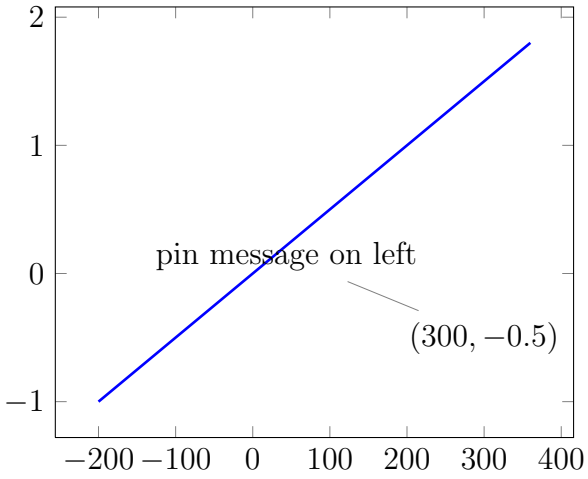
The coordinate space of PGF is not the same as the coordinate space of pgfplots. To insert a PGF node into the right coordinates do the following.

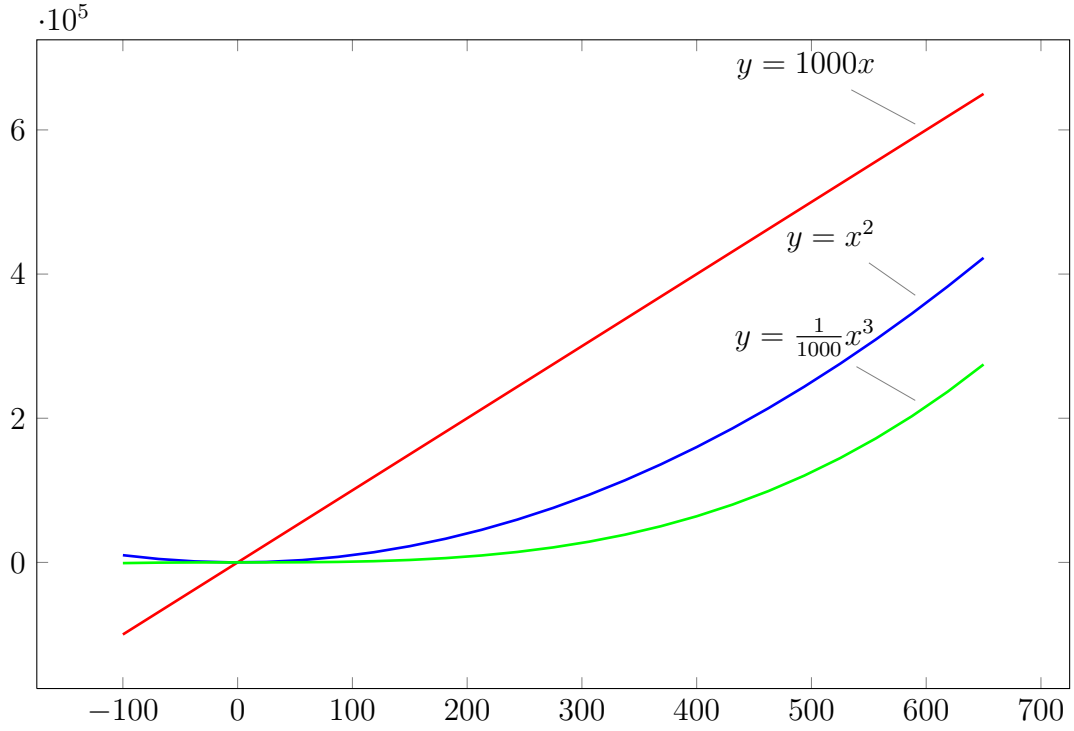


LaTeX code

```
The coordinate space of PGF is not the same as the coordinate space of
pgfplots.
To insert a PGF node into the right coordinates do the following.

\begin{center}
\begin{tikzpicture}[line width=2]
\begin{axis}
\addplot[draw=blue, domain=-200:360, line width=1] {x/200};

\node[]      at (axis cs:300,-0.5)    {$(300,-0.5)$};

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

29 pgfplots: Node with pin (pgfplots-pin.tex)



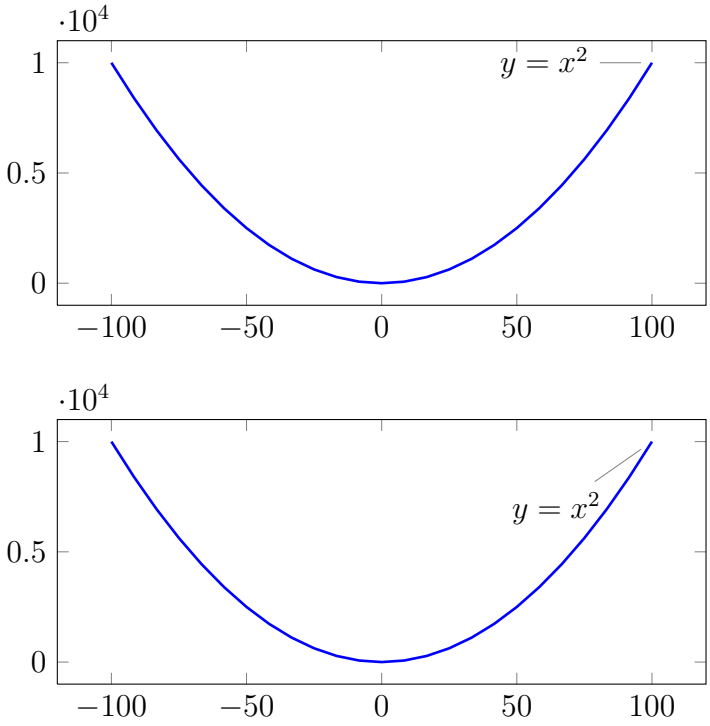


LaTeX code

```
\begin{center}
\begin{tikzpicture}[line width=2]
\begin{axis}
\addplot[draw=blue, domain=-200:360, line width=1] {x/200};
```

```
\node[pin=above left:{pin message on left}] at (axis cs:300,-0.5) {$(300,-0.5)$};

\end{axis}
\end{tikzpicture}
\end{center}


\begin{center}
\begin{tikzpicture}[line width=2]
\begin{axis}
  [width=6in, height=4in]

\addplot[draw=red, domain=-100:650, line width=1] {1000*x};
\addplot[draw=blue, domain=-100:650, line width=1] {x^2};
\addplot[draw=green, domain=-100:650, line width=1] {x^3/1000};

\node[pin=above left:{$y=1000x$}] at (axis cs:600,600000) {};
\node[pin=above left:{$y=x^2$}] at (axis cs:600,360000) {};
\node[pin=above left:{$y=\frac{1}{1000}x^3$}] at (axis cs:600,216000) {};

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

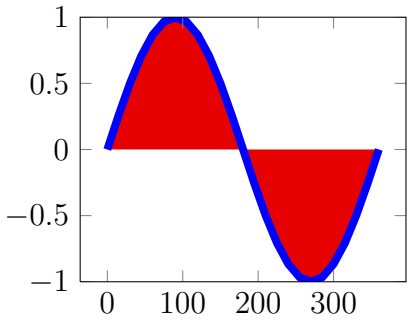
30 pgfplots: Pin Direction (pgfplots-pin-direction.tex)



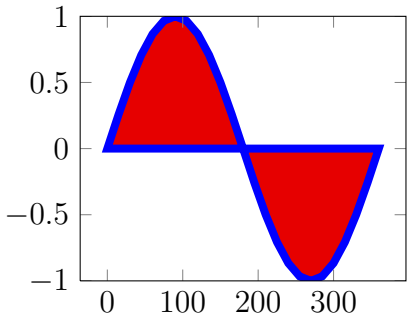
LaTeX code

```
\begin{center}
\begin{tikzpicture}
\begin{axis}[height=2in, width=4in]
\addplot[draw=blue, domain=-100:100, line width=1] {x^2};
\node[pin=180:{$y=x^2$}] at (axis cs:100,10000) {};
\end{axis}
\end{tikzpicture}
\end{center}

\begin{center}
\begin{tikzpicture}
\begin{axis}[height=2in, width=4in]
\addplot[draw=blue, domain=-100:100, line width=1] {x^2};
\node[pin=225:{$y=x^2$}] at (axis cs:100,10000) {};
\end{axis}
\end{tikzpicture}
\end{center}
```

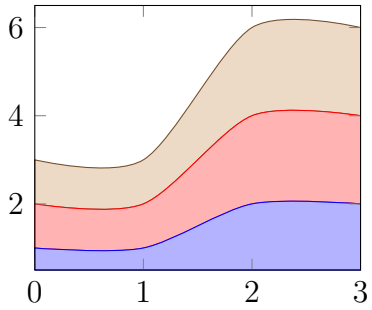
31 pgfplots: Shade Area (xy-plane-area.tex)



Same as above but with y–axis redrawn (i.e. complete cycle)



Area under curve specified with points:



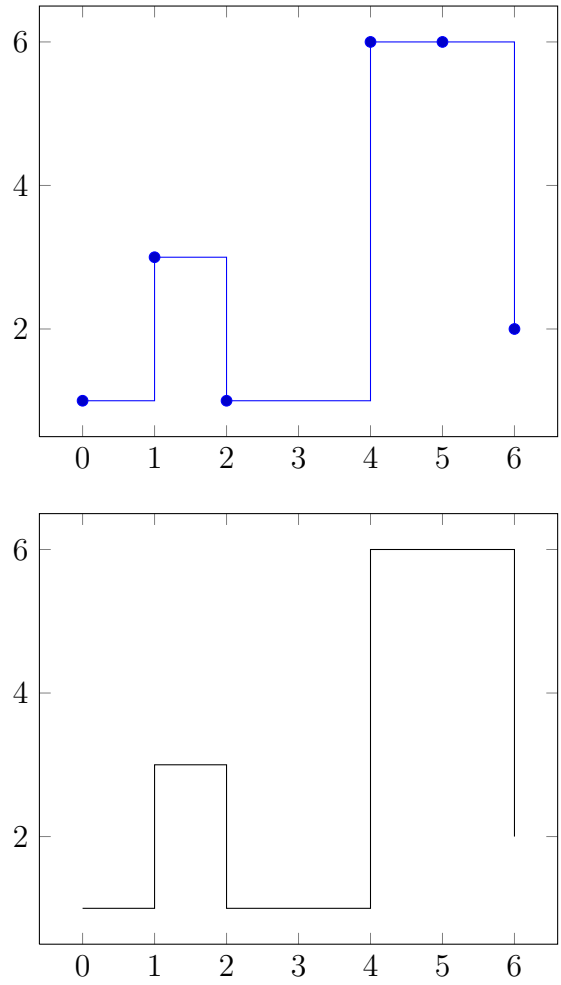
LaTeX code

```
\begin{center}
\begin{tikzpicture}
\begin{axis}[height=2in,
stack plots=y,
area style,
```

```
enlarge y limits=false
]

\addplot [domain=0:360,
          blue,fill=red!90!black,
          line width=3pt
]
{sin(x)};

\end{axis}
\end{tikzpicture}
\end{center}

Same as above but with $y$--axis redrawn (i.e. complete cycle)

\begin{center}
\begin{tikzpicture}
\begin{axis}[height=2in,
stack plots=y,
area style,
enlarge y limits=false
]

\addplot [domain=0:360,
          blue,fill=red!90!black,
          line width=3pt
]
{sin(x)}
|- (axis cs:0,0) -- cycle;

\end{axis}
\end{tikzpicture}
\end{center}

Area under curve specified with points:

\begin{center}
\begin{tikzpicture}
\begin{axis}[height=2in,
smooth,
stack plots=y,
area style,
```

```
enlarge x limits=false
]
\addplot coordinates {(0,1) (1,1) (2,2) (3,2)} \closedcycle;
\addplot coordinates {(0,1) (1,1) (2,2) (3,2)} \closedcycle;
\addplot coordinates {(0,1) (1,1) (2,2) (3,2)} \closedcycle;
\end{axis}
\end{tikzpicture}
\end{center}
```

32 pgfplots: Steps (xy-plane-step.tex)



LaTeX code

```
\begin{center}
\begin{tikzpicture}
\begin{axis}
\addplot+[const plot]
coordinates
{
  (0,1) (1,3) (2,1) (4,6) (5,6) (6,2)
};
\end{axis}
\end{tikzpicture}
\end{center}
```

```
\begin{center}
\begin{tikzpicture}
\begin{axis}
\addplot[const plot]
coordinates
{
  (0,1) (1,3) (2,1) (4,6) (5,6) (6,2)
};
\end{axis}
\end{tikzpicture}
\end{center}
```