TikZ cheat sheet

Document and picture

```
\documentclass{article}
\usepackage{tikz}
\usetikzlibrary{positioning, arrows.meta}
\begin{document}
\begin{tikzpicture}[thick, scale=0.8]
  \draw (0,0) -- (2,1);
  \node at (1,1) {text};
\end{tikzpicture}
A dot \tikz \fill (0,0) circle (2pt); inline
\end{document}
```

Every path command ends with a semicolon.

Plain numbers are centimetres: `(2,1)` is 2 cm right and 1 cm up. 1 cm = 28.45 pt, 1 in = 72.27 pt.

Options `[key=value, ...]` go on the picture, a scope, a path or a node; the innermost ones win.

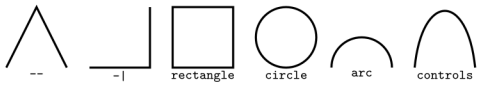
Libraries: `\usetikzlibrary{...}` after `\usepackage{tikz}`. Packages built on TikZ (pgfplots, tikz-cd, circuitikz) use `\usepackage`.

The standalone class crops the page to the picture: `\documentclass[border=4pt]{standalone}`

Paths

```
\draw (0,0) -- (1,1);          % stroke
\fill (0,0) rectangle (1,1);   % fill
\filldraw[fill=red!20] ...;    % both
\path (0,0) node {x};          % invisible
\clip (0,0) circle (1);        % clip
\shade[left color=red, right color=blue]
```

```
(a) -- (b)              straight line
(a) -| (b)  (a) |- (b) right angle
(a) rectangle (b)       (a) circle (1)
(a) ellipse[x radius=2, y radius=1]
(a) arc[start angle=0, end angle=90,
        radius=1]
(a) .. controls (c) and (d) .. (b)
(a) to[bend left] (b)   (a) grid (b)
-- cycle                close the path
```

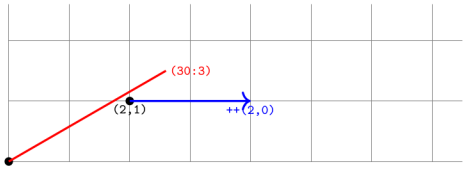


Coordinates

```
(2,1)          2 cm right, 1 cm up
(2cm,10pt)     with units
(30:2)         2 cm at 30 degrees (polar)
++(1,0)        relative, moves on
+(1,0)         relative, stays put
\coordinate (A) at (1,2);
(A)  (n.north)  (n.south west)  (n.30)
(A |- B)       x of A, y of B
(A -| B)       x of B, y of A
```

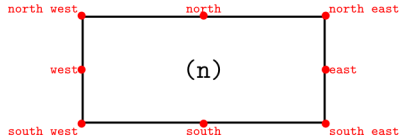
```
\usetikzlibrary{calc}
($(A)!0.5!(B)$)     halfway from A to B
($(A)!1cm!(B)$)     1 cm from A to B
($(A)!(C)!(B)$)     C projected on AB
($(A)+(1,2)$)       ($2*(A)$)
```

```
\usetikzlibrary{intersections}
\draw[name path=a] ...
name intersections={of=a and b}
  gives (intersection-1), ...
```



Nodes and anchors

```
\node[draw] (n) at (1,2) {text};
\draw (0,0) node[above] {label};
\draw (0,0) -- node[above] {x} (2,0);
pos=0.3 near start midway near end
above  below left  right=2pt
anchor=west  inner sep=2pt  outer sep
minimum size=1cm  minimum width=2cm
circle  rounded corners  fill=yellow
label=above:text   pin=right:text
\node[coordinate] (p) at (1,1) {};
```

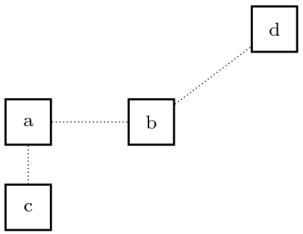


A node name used as a coordinate means its centre, but a line drawn to it stops at its border. Angles work too: `(n.30)`.

Positioning

```
\usetikzlibrary{positioning}
\node (a) {a};
\node[right=of a] (b) {b};
\node[below=5mm of a] (c) {c};
\node[above right=of b] (d) {d};
node distance=6mm and 10mm
      (vertical and horizontal gap)
on grid    measure centre to centre
```

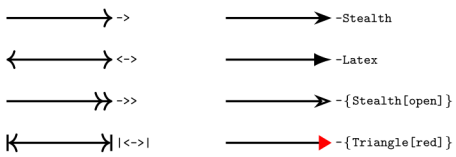
```
% without the library:
\node[anchor=west] at (a.east) {b};
\node[right=1cm] at (a) {b};
```



Arrows

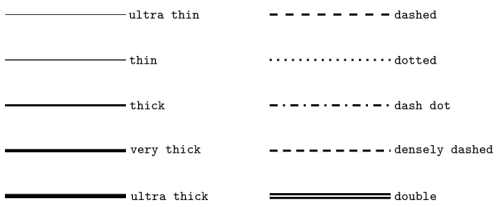
```
\draw[->] (0,0) -- (1,0);
->   <-   <->   ->>   |<->|   -|
```

```
\usetikzlibrary{arrows.meta}
-Stealth  -Latex  -Triangle  -Circle
-Bar  -Bracket  -Hooks  -To
-{Stealth[length=3mm]}   (braces!)
-{Latex[open]}  -{Stealth[round]}
-{Triangle[red, scale=1.5]}
>=Stealth            % every -> in scope
shorten >=2pt   shorten <=2pt
\usetikzlibrary{bending} % curved tips
```



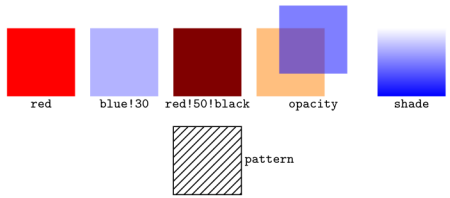
Line styles

```
ultra thin 0.1pt     very thin 0.2pt
thin 0.4pt (default) semithick 0.6pt
thick 0.8pt          very thick 1.2pt
ultra thick 1.6pt    line width=2pt
solid  dashed  dotted  dash dot
densely dashed  loosely dotted
dash pattern=on 4pt off 2pt
line cap=round   line join=round
rounded corners=3pt   double
draw opacity=0.5   help lines
```



Colours and fills

```
red green blue cyan magenta yellow
black white gray darkgray lightgray
brown lime olive orange pink purple
teal violet                 (xcolor)
blue!30       30% blue, 70% white
red!50!black  half red, half black
draw=blue  fill=yellow  text=red
\definecolor{mine}{RGB}{30,120,200}
opacity=0.5  fill opacity  draw opacity
\shade[top color=white,
       bottom color=blue] ...
\usetikzlibrary{patterns}
pattern=north east lines
```

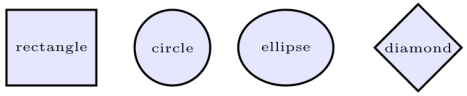


TikZ cheat sheet

Node shapes

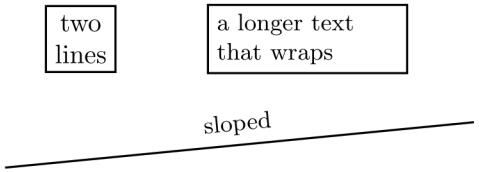
```
% built in: rectangle, circle, coordinate
\usetikzlibrary{shapes.geometric}
ellipse  diamond  star  trapezium
cylinder  isosceles triangle  kite
regular polygon, regular polygon sides=6
semicircle  circular sector  dart
shapes.symbols    cloud  starburst  signal
shapes.arrows     single arrow
shapes.multipart  rectangle split
   {top \nodepart{second} bottom}
shapes.callouts   ellipse callout
shapes.misc       rounded rectangle

\node[draw, diamond, aspect=2] {?};
\node[draw, circle, minimum size=1cm] {};
```



Text in nodes

```
\node[align=center] {two\\lines};
\node[text width=3cm] {wraps ...};
align=left  align=right  align=center
font=\small\bfseries   (switches only,
                        not \textbf)
font=\sffamily  font=\Large  text=red
{$x_1^2$}       maths needs $...$
inner sep=0pt  text height  text depth
rotate=90       sloped (along a line)
label={[red]above:text}
\node[fill=white] {opaque};
```



Loops: \foreach

```
\foreach \x in {1,2,5} {...}
\foreach \x in {1,...,10}     step 1
\foreach \x in {0,0.5,...,3}  step 0.5
\foreach \x in {a,...,e}      letters
\foreach \x/\c in {1/red,2/blue}
\foreach \x [count=\i] in {a,b,c}
\foreach \x [evaluate=\x as \y using 2*\x]
\foreach \x [remember=\x as \last]
\foreach \x in {1,...,4}
  \foreach \y in {1,2} {...}   % nested
\draw (0,0) foreach \x in {1,2} {--(\x,1)};
(\x, {sin(\x r)})   braces around (...)
{0,0.1,...,0.5} stops at 0.4: use 0.501
```



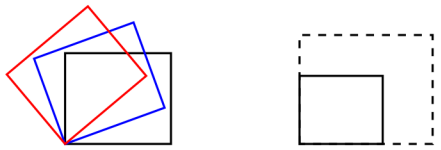
Styles: \tikzset

```
\tikzset{box/.style={draw, fill=blue!10}}
\node[box] {A};
\tikzset{box/.style={draw, fill=#1!10},
         box/.default=blue}
\node[box=red] {B};
tag/.style 2 args={fill=#1, text=#2}
\node[tag={black}{white}] {C};
box/.append style={thick}
every node/.style={font=\small}
every path/.style={thick}
\begin{tikzpicture}[box/.style={...}]
% local: only in this picture
```



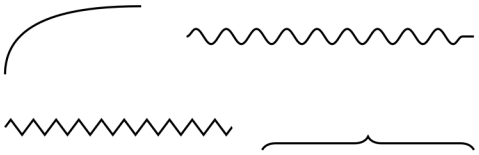
Scopes and transformations

```
\begin{scope}[red, thick]
  \draw ...;   % options apply inside
\end{scope}
shift={(2,1)}   xshift=2cm   yshift=-1cm
rotate=30       rotate around={30:(1,1)}
scale=2         xscale=2     yscale=-1
transform shape    also scale the nodes
x=2cm, y=1cm       new unit vectors
\usetikzlibrary{backgrounds}
\begin{scope}[on background layer]
```



Curves and decorations

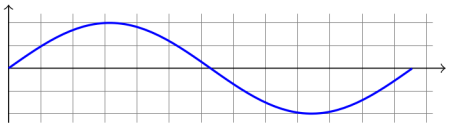
```
(a) to[out=90, in=180] (b)
(a) to[bend left=30] (b)  looseness=1.5
(a) .. controls (c) and (d) .. (b)
(a) .. controls +(90:1) and +(90:1) .. (b)
plot[smooth] coordinates {(0,0) (1,1)}
\usetikzlibrary{decorations.pathmorphing}
decorate, decoration={snake, amplitude=1mm}
zigzag  coil  bumps  saw  random steps
\usetikzlibrary{decorations.pathreplacing}
decoration={brace, mirror, raise=2pt}
\usetikzlibrary{decorations.markings}
postaction={decorate}, decoration={markings,
  mark=at position 0.5 with {\arrow{>}}}
```



Plots and grids

```
\draw[help lines, step=0.5] (0,0) grid (4,3);
\draw[->] (0,0) -- (4.2,0) node[right] {$x$};
\draw[domain=0:6.28, samples=60, smooth]
  plot (\x, {sin(\x r)});
\draw plot coordinates {(0,0) (1,2) (2,1)};
% sin, cos take degrees: \x r = radians
\foreach \x in {1,...,4}
  \draw (\x,0) -- (\x,-3pt) node[below] {\x};

% data plots: the pgfplots package
\usepackage{pgfplots}
\pgfplotsset{compat=1.18}
\begin{axis} \addplot {x^2}; \end{axis}
```



Common errors

Did you forget a semicolon? Add the ; at the end of the path, or braces round a formula: `(1,{sin(30)})`.

I do not know the key '/tikz/...' A typo, or a missing `\usetikzlibrary` (`diamond` needs `shapes.geometric`).

Unknown arrow tip kind 'Stealth' Load `arrows.meta`; with options write `-{Stealth[length=3mm]}`.

Missing $ inserted Maths in a node needs `$...$`: `{$x_1$}`.

Dimension too large A number above about 16384 pt: scale the data down.

No shape named 'a' is known The node name is misspelt or defined later.

`\\` **ignored in a node** Add `align=center` or `text width`.

Undefined control sequence A command from a package that is not loaded.

More: www.labcanvas.com/tikz/errors