The experimental tikzmusic package

Vũ Văn Dũng

Manual for version 0.0
Last compiled February 10, 2020

Contents

1 Introduction

1.1 What it can do

This package provides commands and syntaxes to help you draw musical notations in a LaTeX document, based on the Ti*k*Z package. You can see next page of this manual for a demo of it.

Note that this package is still *very* experimental.

1.2 Current limitations

Although the syntaxes are quite human-readable, they are very *long*. In the future an easy-to-use front-end is necessary and should be implemented.

The speed is also an issue, as it requires time to draw many complex notations. For instance, the treble clef, `tm-g-clef`, takes 0.03 seconds (in my machine) alone.[1] The first page of *Für Elise*, displayed on the next page, takes about 10 seconds for each PDFLaTeX run. This has no easy fix though, as efficiency and beauty, in these cases, can't live under the same roof. If you need high speed, you can externalize your sheet music and include them as PDFs in your document.

There are many notations which are so advanced that I can't be confident enough to add them to the package. If you need them, feel free to submit a feature request to the package repository.

Also, this documentation is written somewhat hastily, and may contain grammar mistakes from a non-native speaker. If you find typos or errors in the manual, feel free to send me a bug report.

1.3 Thanks

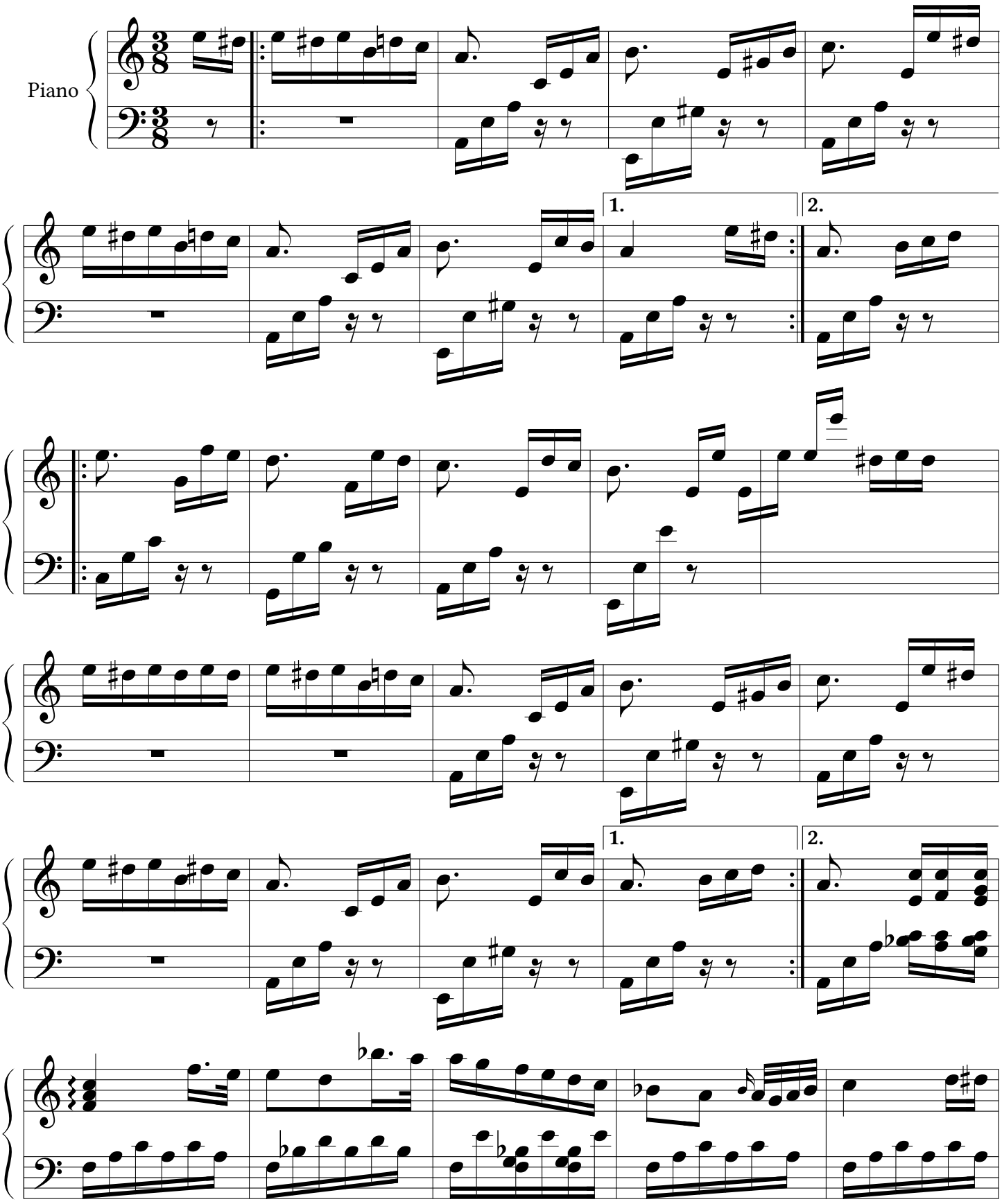
First of all, I would like to thank the following people who have greatly helped me in writing this package:

- Till Tantau and every developer of the Ti*k*Z-PGF package. Obviously, without the powerful PGF and its easy-to-use front-end Ti*k*Z, this package, along with many others, would not be possible.
- The development team of all packages used by this package.
- Every member of the Duck Overflow chat room and the TopTeX community, most notably (but not limited to) marmot, samcarter and Skillmon, for their great help in writing this package. My knowledge in TeX and friends would not be the same without these people.
- Christian Feuersänger for the `pgfmanual` documentation style, most notably the automatic hyperlink in the documentation. The style was adopted in the documentation of this package.
- The Inkscape project and the svg2tikz project. I have used these two tools heavily to generate code for the musical notations from existing SVG files. Although the files in tex/latex/tikzmusic/src/tm-paths are not automatically generated, most parts of them are generated by these tools.
- The MuseScore project. Although I had known some about music notations, that was too little for the package. I have used MuseScore heavily for some ideas about music writing and music notations.

[1] Thanks to the l3benchmark package.

Für Elise

Ludwig van Beethoven



2 Initialization

2.1 Loading the package

This package currently only supports LaTeX 2_ε.

`\usepackage{tikzmusic}`

Loading the tikzmusic package. There are no package options.

This package will automatically load the packages spath3 and Ti*k*Z, as well as Ti*k*Z standard libraries `calc`, `intersection`, `decorations.pathreplacing`. You don't need to load these packages and libraries again in your document.

2.2 Processing options

The tikzmusic package uses the pgfkeys package to handle options. Every option defined in the package is in the same family, `/tm`, e.g. `color`.

`\tmset{`⟨*options*⟩`}`

Process ⟨*options*⟩. where the default path is set to `/tm`.

If you know about Ti*k*Z and its key system, you can think `\tmset` works just like `\tikzset`, only the default path is different. You can now skip to the next section.

If you are not familiar with pgfkeys or `\pgfkeys` or `\tikzset`, ⟨*options*⟩ is a list of ⟨*key*⟩=⟨*value*⟩ pairs, separated by commas. The command will then separate each pair and process them.

- If the key is with ⟨*value*⟩, option ⟨*key*⟩ is processed, with its value being ⟨*value*⟩.
- Otherwise, the command will check whether ⟨*key*⟩ is a defined key. If it is defined, option ⟨*key*⟩ is processed. Otherwise, it will be processed as the value for both `line color` and `color` (see section 8.1).

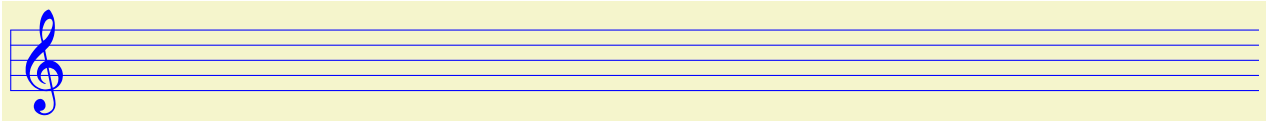
If you want to learn deeper about this, you can read section 88 of the PGF manual (you can read it via `texdoc pgf`).

2.3 Environments for music lines

Each music line will be drawn separately, by using the following environment:

`\begin{tmline}[`⟨*options*⟩`]`
⟨*environment contents*⟩
`\end{tmline}`

Add a music line (consisting of one or many staves).

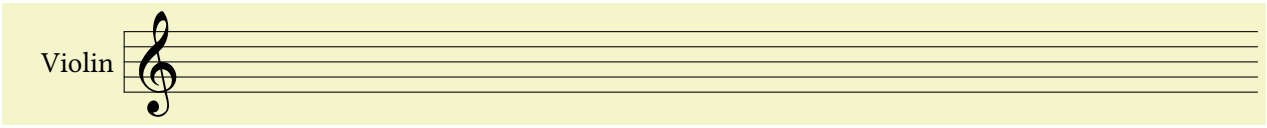


```
\begin{tmline}[blue]
\begin{tmstaff}{g}{}\end{tmstaff}
\end{tmline}
```

If a line consists of more than one staff, you may need to indent the staves a little bit to make room for instrument names and braces/brackets. You can do so by using the following key:

/tm/staff offset=⟨*length*⟩ (no default, initially `0pt`)

Indent all staves in a line by ⟨*length*⟩.




```
\begin{tmline}[staff offset=1.5cm]
\begin{tmstaff}{g}{a}
  \path[overlay] (a-start) node[left] {Violin};%
\end{tmstaff}
\end{tmline}%
```

2.4 Creating a staff

A staff can be created using one of the following environments:

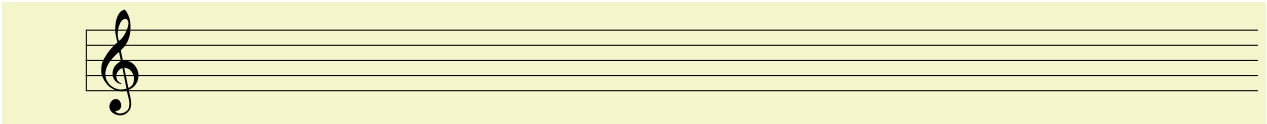
\begin{tmstaff}[⟨*options*⟩]{⟨*clef name*⟩}{⟨*staff name*⟩}
⟨*environment contents*⟩
\end{tmstaff}

Create a staff, with the starting clef is ⟨*clef name*⟩.

⟨*clef name*⟩ can have three values: `g`, `f` and `c`, which stands for the treble (G) clef, the bass (F) clef and the alto (C) clef, respectively.

⟨*staff name*⟩ will be used to make cross-staff barlines or braces, so you should only left it empty if you are sure you will not refer to it later.

⟨*options*⟩ will be executed at the beginning of the environment.

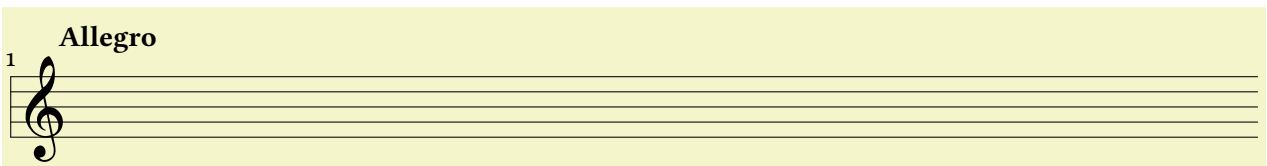


```
\begin{tmline}[staff offset=1cm]
\begin{tmstaff}{g}{}\end{tmstaff}
\end{tmline}
```

\begin{tmstaff*}[⟨*options*⟩]{⟨*staff name*⟩}
⟨*environment contents*⟩
\end{tmstaff*}

Work like `{tmstaff}`, but no clefs will be drawn.

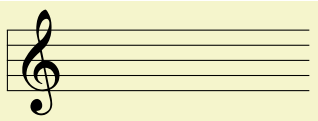
Essentially, `{tmstaff}` and `{tmstaff*}` are extensions of the `{tikzpicture}` environment, where the origin of the canva is the leftmost point of the middle line. That origin is marked as Ti*k*Z remembered coordinate (⟨*staff name*⟩`-start`). There are also two other remembered coordinates: the leftmost points of the top line and the bottom line are marked as Ti*k*Z coordinates (⟨*staff name*⟩`-nw`) and (⟨*staff name*⟩`-sw`) respectively.



```
\begin{tmline}
\begin{tmstaff}{g}{my-staff}
  \path (my-staff-nw) node[above,overlay] {1} ++ (.5,.5) node[right] {\bfseries Allegro};
\end{tmstaff}
\end{tmline}
```

By default, the staff lines will be separated by `2mm`. You can scale this to a different value by using `scale`, see more in section 8.4.1.

The lines will always span over the full line width, so to get a staff having some particular length, you can put them inside a `{minipage}`:



```
\begin{minipage}{4cm}
\begin{tmline}
\begin{tmstaff}{g}{}\end{tmstaff}
\end{tmline}
\end{minipage}
```

3 Multiple-staff operations

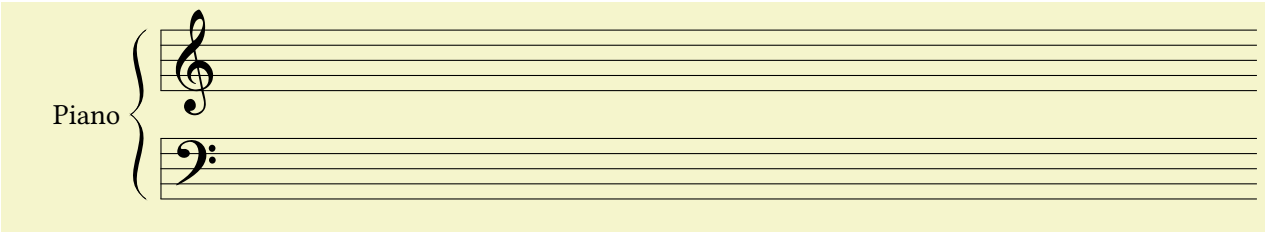
Because the following commands are multiple-staff commands, they should be used outside `{tmstaff}` and `{tmstaff*}` (*except* `\tmbarlineinline`, `\tmdoublebarlineinline`, ...).

3.1 Ensembling staves

Braces that groups some staves inside a `{tmline}` can be drawn using the following command:

`\tmbrace``[`⟨*options*⟩`]{`⟨*staff 1*⟩`}{`⟨*staff 2*⟩`}{`⟨*text*⟩`}`

Draw a brace spanning from ⟨*staff 1*⟩ to ⟨*staff 2*⟩. ⟨*text*⟩ is displayed at the middle of the brace. If you don't want any text to be displayed, you can leave this option empty.

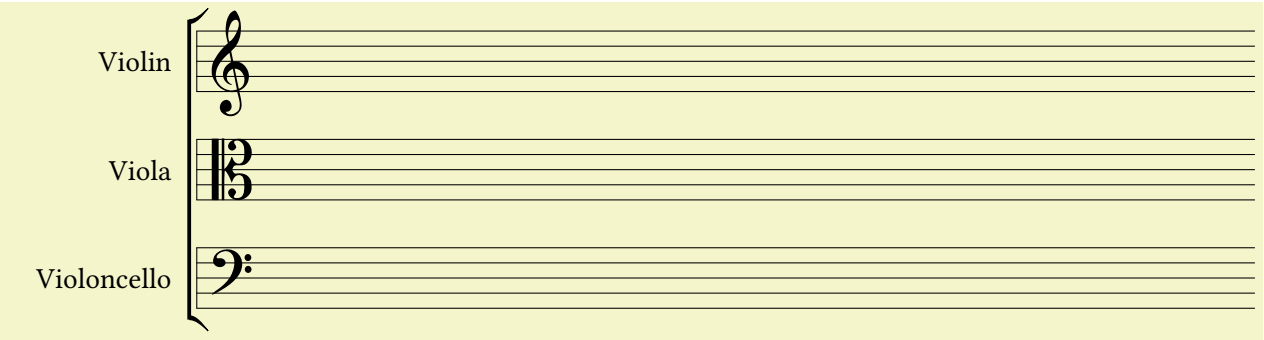



```
\begin{tmline}[staff offset=2cm]%
\begin{tmstaff}{g}{piano-1}\end{tmstaff}%
\begin{tmstaff}{f}{piano-2}\end{tmstaff}%
\tmbrace{piano-1}{piano-2}{Piano}%
\end{tmline}
```

Similarly, brackets can also be drawn:

`\tmbracket``[`⟨*options*⟩`]{`⟨*staff 1*⟩`}{`⟨*staff 2*⟩`}`

Draw a bracket spanning from ⟨*staff 1*⟩ to ⟨*staff 2*⟩. Unlike `\tmbrace`, no text will be displayed.




```
\begin{tmline}[staff offset=2.5cm]%
\begin{tmstaff}{g}{Violin}\end{tmstaff}%
\begin{tmstaff}{c}{Viola}\end{tmstaff}%
\begin{tmstaff}{f}{Violoncello}\end{tmstaff}%
\begin{tikzpicture}[remember picture,overlay]
  \foreach \i in {Violin,Viola,Violoncello}\path (\i-start) node[left=2mm] {\i};
\end{tikzpicture}%
\tmbracket{Violin}{Violoncello}%
\end{tmline}
```

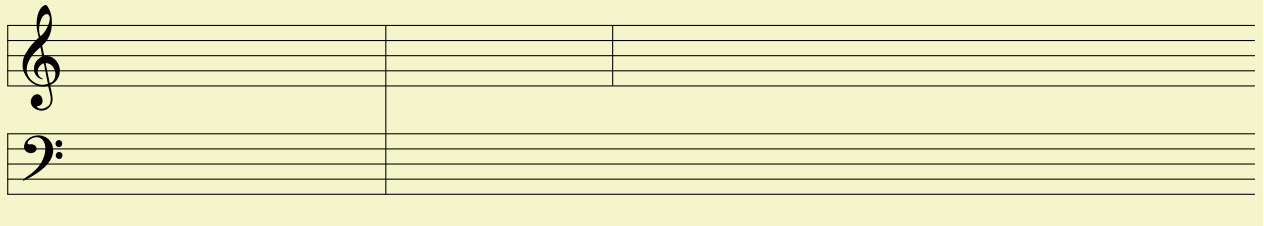
3.2 Barlines

The tikzmusic package supports many different types of barlines.

3.2.1 Normal barlines

\tmbarline[⟨*options*⟩]{⟨*x-pos*⟩}{⟨*staff 1*⟩}{⟨*staff 2*⟩}

Draw a normal barline spanning from ⟨*staff 1*⟩ to ⟨*staff 2*⟩, at *x*-position ⟨*x-pos*⟩ in relative to the origin (⟨*staff name*⟩`-start`) of either staff.

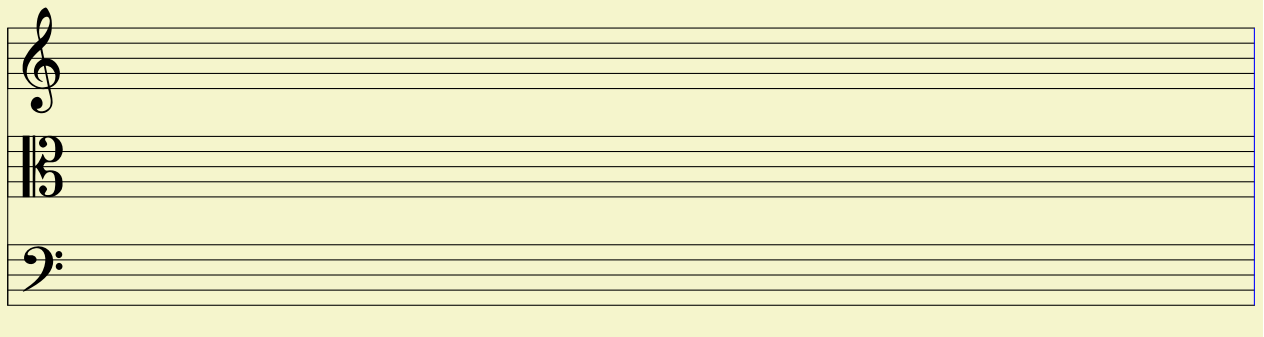


```
\begin{tmline}%
\begin{tmstaff}{g}{staff-1}\end{tmstaff}%
\begin{tmstaff}{f}{staff-2}\end{tmstaff}%
\tmbarline{5}{staff-1}{staff-2}\tmbarline{8}{staff-1}{staff-1}%
\end{tmline}
```

A special case of `\tmbarline` is implemented in the following command:

\tmbarlineendline[⟨*options*⟩]{⟨*staff 1*⟩}{⟨*staff 2*⟩}

Draw a normal barline spanning from ⟨*staff 1*⟩ to ⟨*staff 2*⟩ at the end of the line.

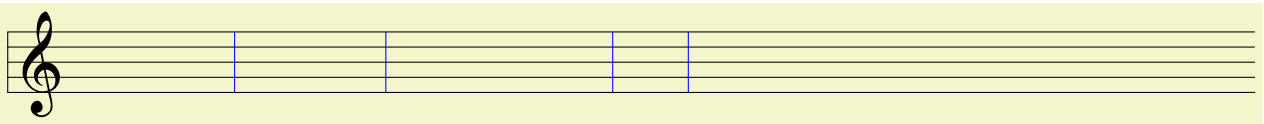


```
\begin{tmline}
\begin{tmstaff}{g}{staff-1}\end{tmstaff}%
\begin{tmstaff}{c}{staff-2}\end{tmstaff}%
\begin{tmstaff}{f}{staff-3}\end{tmstaff}%
\tmbarline{0}{staff-1}{staff-3}%
\tmbarlineendline[blue]{staff-1}{staff-3}%
\end{tmline}
```

If you want to draw the barline inside `{tmstaff}` or `{tmstaff*}`, you can use

\tmbarlineinline[⟨*options*⟩]{⟨*list of x-pos*⟩}

Draw a normal barline at each *x*-position specified in {⟨*list of x-pos*⟩}.



```
\begin{tmline}%
\begin{tmstaff}{g}{}
  \tmbarlineinline[blue]{3,5,8,9}
\end{tmstaff}%
\end{tmline}
```

3.2.2 Double barlines

Like when drawing normal barlines as described in section 3.2.1, we also have four commands for double barlines.

\tmdoublebarline[⟨*options*⟩]{⟨*x-pos*⟩}{⟨*staff 1*⟩}{⟨*staff 2*⟩}

Draw a double barline spanning from ⟨*staff 1*⟩ to ⟨*staff 2*⟩, at *x*-position ⟨*x-pos*⟩ in relative to the origin (⟨*staff name*⟩`-start`) of either staff.

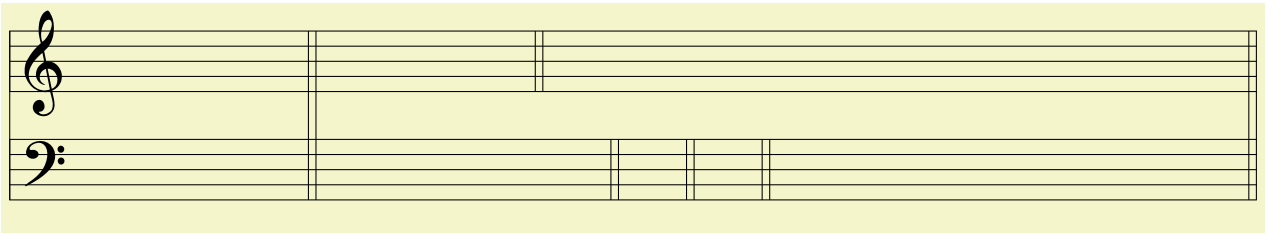
\tmdoublebarlineendline[⟨*options*⟩]{⟨*staff 1*⟩}{⟨*staff 2*⟩}

Draw a double barline spanning from ⟨*staff 1*⟩ to ⟨*staff 2*⟩ at the end of the line.

\tmdoublebarlineinline[⟨*options*⟩]{⟨*list of x-pos*⟩}

Draw a double barline at each *x*-position specified in {⟨*list of x-pos*⟩}.

Example use of all four commands described in this section:



```
\begin{tmline}%
\begin{tmstaff}{g}{staff-1}\end{tmstaff}%
\begin{tmstaff}{f}{staff-2}
  \tmdoublebarlineinline{8,9,10}
\end{tmstaff}%
\tmbarline{0}{staff-1}{staff-2}%
\tmdoublebarline{4}{staff-1}{staff-2}%
\tmdoublebarline{7}{staff-1}{staff-1}%
\tmdoublebarlineendline{staff-1}{staff-2}%
\end{tmline}
```

3.2.3 Dotted barlines

Now you can see the patterns :).

\tmdottedbarline[⟨*options*⟩]{⟨*x-pos*⟩}{⟨*staff 1*⟩}{⟨*staff 2*⟩}

Draw a dotted barline spanning from ⟨*staff 1*⟩ to ⟨*staff 2*⟩, at *x*-position ⟨*x-pos*⟩ in relative to the origin (⟨*staff name*⟩`-start`) of either staff.

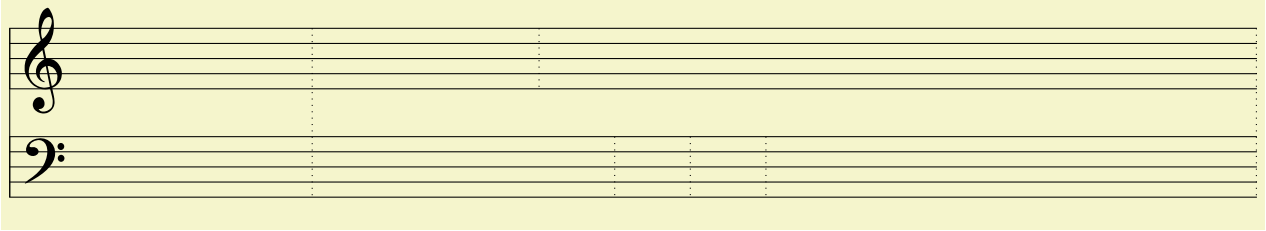
\tmdottedbarlineendline[⟨*options*⟩]{⟨*staff 1*⟩}{⟨*staff 2*⟩}

Draw a dotted barline spanning from ⟨*staff 1*⟩ to ⟨*staff 2*⟩ at the end of the line.

\tmdottedbarlineinline[⟨*options*⟩]{⟨*list of x-pos*⟩}

Draw a double barline at each *x*-position specified in {⟨*list of x-pos*⟩}.

The commands in use:



```
\begin{tmline}%
\begin{tmstaff}{g}{staff-1}\end{tmstaff}%
\begin{tmstaff}{f}{staff-2}
  \tmdottedbarlineinline{8,9,10}
\end{tmstaff}%
\tmbarline{0}{staff-1}{staff-2}%
\tmdottedbarline{4}{staff-1}{staff-2}%
\tmdottedbarline{7}{staff-1}{staff-1}%
\tmdottedbarlineendline{staff-1}{staff-2}%
\end{tmline}
```

3.2.4 Final barlines

\tmfinalbarline[⟨*options*⟩]{⟨*x-pos*⟩}{⟨*staff 1*⟩}{⟨*staff 2*⟩}

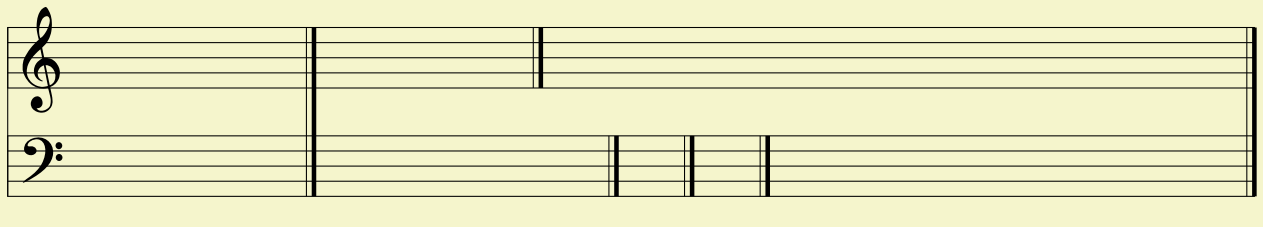
Draw a final barline spanning from ⟨*staff 1*⟩ to ⟨*staff 2*⟩, at *x*-position ⟨*x-pos*⟩ in relative to the origin (⟨*staff name*⟩`-start`) of either staff.

`\tmfinalbarlineendline`[⟨*options*⟩]{⟨*staff 1*⟩}{⟨*staff 2*⟩}

Draw a final barline spanning from ⟨*staff 1*⟩ to ⟨*staff 2*⟩ at the end of the line.

`\tmfinalbarlineinline`[⟨*options*⟩]{⟨*list of x-pos*⟩}

Draw a final barline at each *x*-position specified in {⟨*list of x-pos*⟩}.



```
\begin{tmline}%
\begin{tmstaff}{g}{staff-1}\end{tmstaff}%
\begin{tmstaff}{f}{staff-2}
  \tmfinalbarlineinline{8,9,10}
\end{tmstaff}%
\tmbarline{0}{staff-1}{staff-2}%
\tmfinalbarline{4}{staff-1}{staff-2}%
\tmfinalbarline{7}{staff-1}{staff-1}%
\tmfinalbarlineendline{staff-1}{staff-2}%
\end{tmline}
```

3.2.5 Start repeat barlines

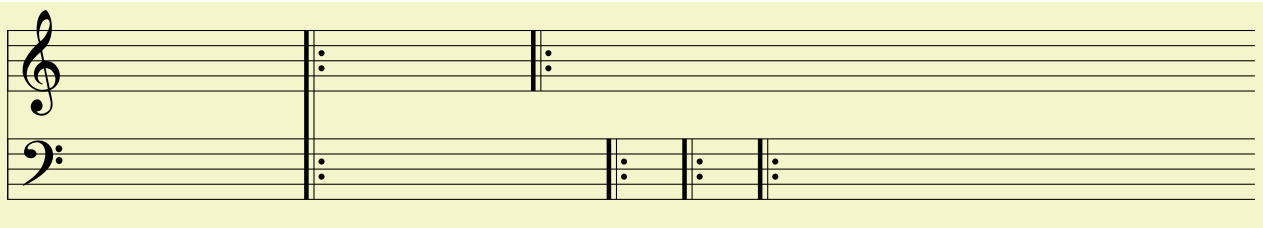
`\tmstartrepeatbarline`[⟨*options*⟩]{⟨*x-pos*⟩}{⟨*staff 1*⟩}{⟨*staff 2*⟩}{⟨*list of staff names*⟩}

Draw a start repeat barline spanning from ⟨*staff 1*⟩ to ⟨*staff 2*⟩, at *x*-position ⟨*x-pos*⟩ in relative to the origin (⟨*staff name*⟩`-start`) of either staff.

Because of some internal problems, you need to specify a full list of the names of the staff that the barline spans over in ⟨*list of staff names*⟩ with a comma-separated list.

`\tmstartrepeatbarlineinline`[⟨*options*⟩]{⟨*list of x-pos*⟩}

Draw a start repeat barline at each *x*-position specified in {⟨*list of x-pos*⟩}.



```
\begin{tmline}%
\begin{tmstaff}{g}{staff-1}\end{tmstaff}%
\begin{tmstaff}{f}{staff-2}
  \tmstartrepeatbarlineinline{8,9,10}
\end{tmstaff}%
\tmbarline{0}{staff-1}{staff-2}%
\tmstartrepeatbarline{4}{staff-1}{staff-2}{staff-1,staff-2}%
\tmstartrepeatbarline{7}{staff-1}{staff-1}{staff-1}%
\end{tmline}
```

Note that there is no `\tmstartrepeatbarlineendline`, because I am sure you will never put a start repeat barline to the end of a line.

3.2.6 End repeat barlines

\tmendrepeatbarline[⟨*options*⟩]{⟨*x-pos*⟩}{⟨*staff 1*⟩}{⟨*staff 2*⟩}{⟨*list of staff names*⟩}

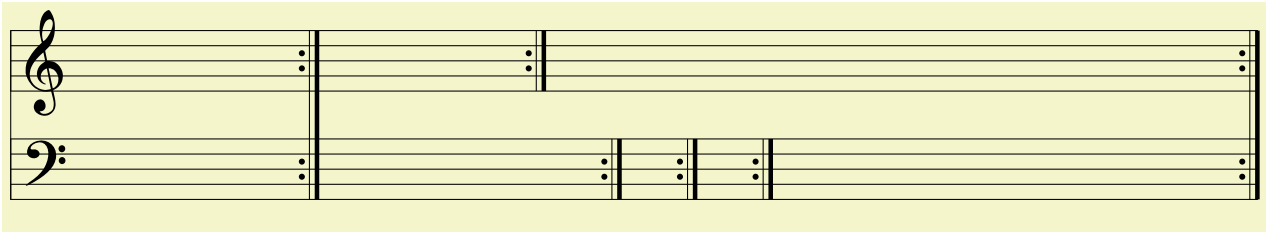
Draw an end repeat barline spanning from ⟨*staff 1*⟩ to ⟨*staff 2*⟩, at *x*-position ⟨*x-pos*⟩ in relative to the origin (⟨*staff name*⟩`-start`) of either staff.

\tmendrepeatbarlineendline[⟨*options*⟩]{⟨*staff 1*⟩}{⟨*staff 2*⟩}{⟨*list of staff names*⟩}

Draw an end repeat barline spanning from ⟨*staff 1*⟩ to ⟨*staff 2*⟩ at the end of the line.

\tmendrepeatbarlineinline[⟨*options*⟩]{⟨*list of x-pos*⟩}

Draw a end repeat barline at each *x*-position specified in {⟨*list of x-pos*⟩}.



```
\begin{tmline}%
\begin{tmstaff}{g}{staff-1}\end{tmstaff}%
\begin{tmstaff}{f}{staff-2}
  \tmendrepeatbarlineinline{8,9,10}
\end{tmstaff}%
\tmbarline{0}{staff-1}{staff-2}%
\tmendrepeatbarline{4}{staff-1}{staff-2}{staff-1,staff-2}%
\tmendrepeatbarline{7}{staff-1}{staff-1}{staff-1}%
\tmendrepeatbarlineendline{staff-1}{staff-2}{staff-1,staff-2}%
\end{tmline}
```

3.2.7 End-start repeat barlines

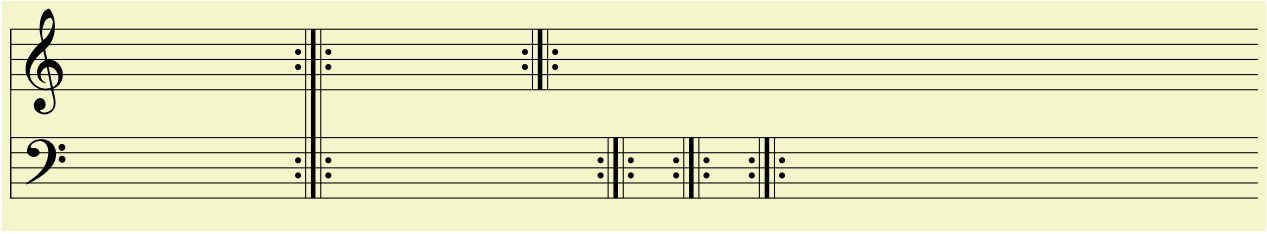
Sometimes, you want a barline to be a start repeat barline and an end repeat barline at the same time. You should not use `\tmstartrepeatbarline` (and similar commands) and `\tmendrepeatbarline` (and similar commands) at the same place, because it will look very bad. In those cases, use the following commands:

\tmendstartrepeatbarline[⟨*options*⟩]{⟨*x-pos*⟩}{⟨*staff 1*⟩}{⟨*staff 2*⟩}{⟨*list of staff names*⟩}

Draw an 'end-start' repeat barline spanning from ⟨*staff 1*⟩ to ⟨*staff 2*⟩, at *x*-position ⟨*x-pos*⟩ in relative to the origin (⟨*staff name*⟩`-start`) of either staff.

\tmendstartrepeatbarlineinline[⟨*options*⟩]{⟨*list of x-pos*⟩}

Draw a end repeat barline at each *x*-position specified in {⟨*list of x-pos*⟩}.



```
\begin{tmline}%
\begin{tmstaff}{g}{staff-1}\end{tmstaff}%
\begin{tmstaff}{f}{staff-2}
  \tmendstartrepeatbarlineinline{8,9,10}
\end{tmstaff}%
\tmbarline{0}{staff-1}{staff-2}%
\tmendstartrepeatbarline{4}{staff-1}{staff-2}{staff-1,staff-2}%
\tmendstartrepeatbarline{7}{staff-1}{staff-1}{staff-1}%
\end{tmline}
```

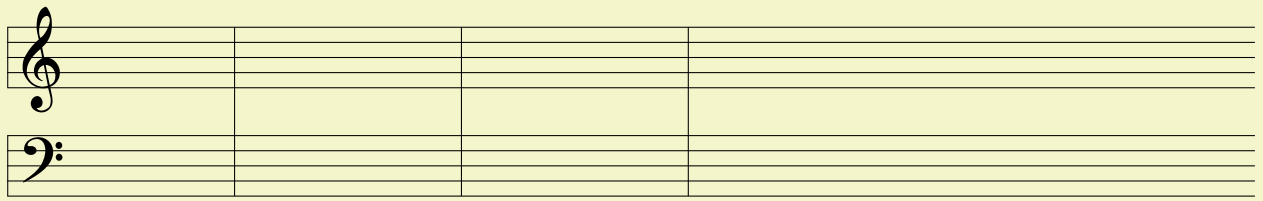
Note that there is no `\tmendstartrepeatbarlineendline`.

3.2.8 Normal barlines loops

Normally there are many barlines in your line, so using `\tmbarline` for each of them is obviously not convenient. You can use the following commands to make drawing barlines easier and more concise.

`\tmbarlineloop`[⟨*options*⟩]{⟨*list of x-pos*⟩}{⟨*staff 1*⟩}{⟨*staff 2*⟩}

Draw a normal barline at each *x*-position in ⟨*list of x-pos*⟩, spanning from ⟨*staff 1*⟩ to ⟨*staff 2*⟩.



```
\begin{tmline}%
\begin{tmstaff}{g}{staff-1}\end{tmstaff}%
\begin{tmstaff}{f}{staff-2}\end{tmstaff}%
\tmbarlineloop{3,6,9}{staff-1}{staff-2}%
\end{tmline}
```

4 Key signatures and time signatures

4.1 Key signatures

Key signatures are added by the following command:

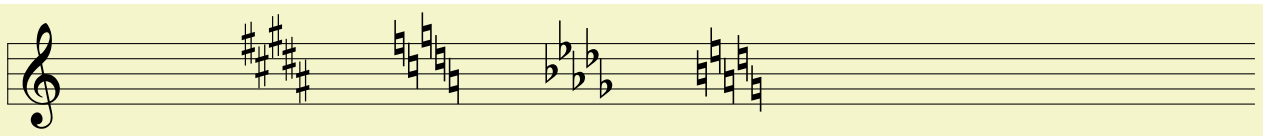
`\tmkeysignature`[⟨*options*⟩]{⟨*x-pos*⟩}{⟨*type*⟩}{⟨*number*⟩}

Add a key signature at *x*-position ⟨*x-pos*⟩. The key signature has type ⟨*type*⟩ and the number of sharps/flats ⟨*number*⟩.

⟨*type*⟩ can be either `sharp`, `flat`, `nsharp` or `nflat`. `sharp` and `flat` will produce a sharp or flat key signature as usual. `nsharp` and `nflat` will produce a 'natural' key signature that has the format of sharp and flat, respectively.

⟨*number*⟩ can be any number from `1` to `7`.

The key signature will be added as in a treble clef. You can use shifting options, e.g. `line shift` (see more in section 8.4) to shift the key signature so that it fits other clefs.



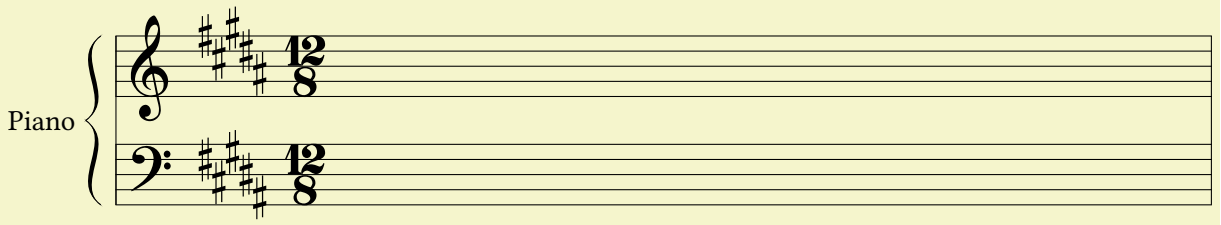
```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmkeysignature{3}{sharp}{5}
  \tmkeysignature{5}{nsharp}{5}
  \tmkeysignature{7}{flat}{5}
  \tmkeysignature{9}{nflat}{5}
\end{tmstaff}
\end{tmline}
```

4.2 Time signatures

Normal time signatures can be added using the following command

\tmtimesignature[⟨*options*⟩]{⟨*x-pos*⟩}{⟨*upper*⟩}{⟨*lower*⟩}

Add a time signature to *x*-position ⟨*x-pos*⟩. The upper part and the lower part of the time signature are ⟨*upper*⟩ and ⟨*lower*⟩ respectively.




```
\begin{tmline}[staff offset=2cm]%
\begin{tmstaff}{g}{piano-1}
  \tmkeysignature{1}{sharp}{5}\tmtimesignature{2.5}{12}{8}
\end{tmstaff}%
\begin{tmstaff}{f}{piano-2}
  \tmkeysignature[line shift=-2]{1}{sharp}{5}\tmtimesignature{2.5}{12}{8}
\end{tmstaff}%
\tmbrace{piano-1}{piano-2}{Piano}%
\tmbarline{0}{piano-1}{piano-2}\tmbarlineendline{piano-1}{piano-2}%
\end{tmline}
```

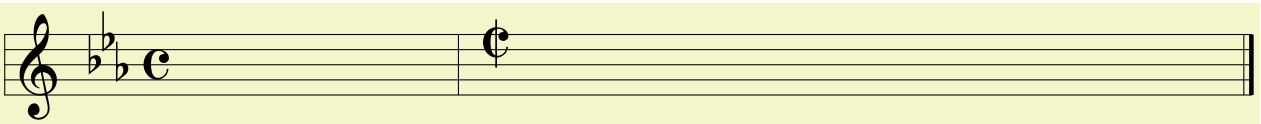
Special time signatures have their own commands:

\tmtimesignaturecommon[⟨*options*⟩]{⟨*x-pos*⟩}

Add the common time signature (𝐂) to *x*-position ⟨*x-pos*⟩.

\tmtimesignatureallabreve[⟨*options*⟩]{⟨*x-pos*⟩}

Add the alla breve time signature (𝐂) to *x*-position ⟨*x-pos*⟩.



```
\begin{tmline}%
\begin{tmstaff}{g}{helloworld}
  \tmkeysignature{1}{flat}{3}
  \tmtimesignaturecommon{2}
  \tmtimesignatureallabreve[yshift=3mm]{6.5}
\end{tmstaff}%
\tmbarline{6}{helloworld}{helloworld}\tmfinalbarlineendline{helloworld}{helloworld}%
\end{tmline}
```

Figure 1: Note value – the letter part

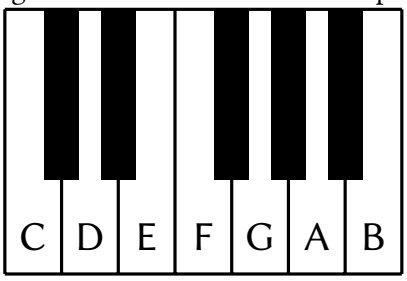
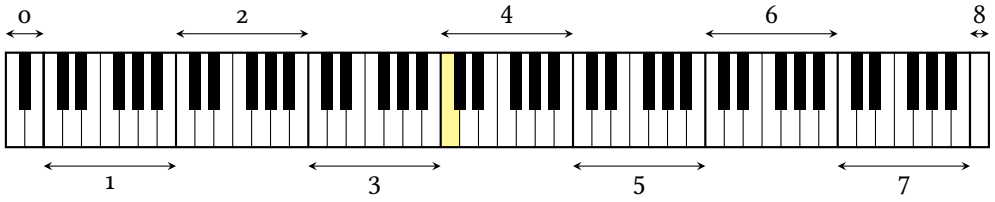


Figure 2: Note value – the number part



5 Adding notes

5.1 Commands for notes

5.1.1 Note values

Every white note is assigned to a 'value', which is the *scientific pitch notation* of that note. These values have two parts: the letter part and the number part:

- The letter part can have seven values: `A`, `B`, `C`, ..., `G`, indicating the name of the note (*do*, *re*, *mi*, ...). (See figure 1).
- The number part is a whole number between 0 to 8, indicating which octave the note is in. (See figure 2).

For example, *Für Elise* by Beethoven starts with an `E5` (a *mi* at the 5th octave).

We will only work with these values. To have black notes in your score, you can use `accidental` to add the accidentals, see more in section 5.4.1.

The package will automatically detect which staff you are using, when you use these values.

5.1.2 Note names

It is very possible that a note will be refered to later in the staff (to add notations to it, etc.). In this package, to refer to notes, we will use note *names*. Just like Ti*k*Z node names, etc. – you can leave the name empty if you want, but you will not be able to communicate with that unnamed note any time later in the document.

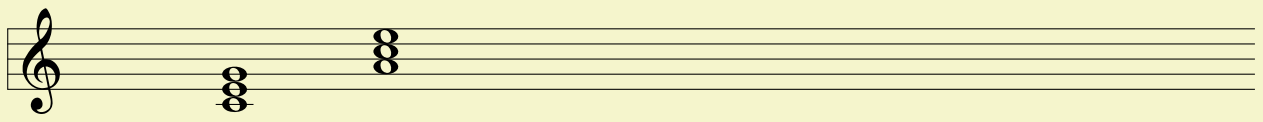
5.1.3 Whole notes

`\tmwhole`[⟨*options*⟩]{⟨*x-pos*⟩}{⟨*note value list*⟩}{⟨*name*⟩}

Add a set of whole notes at x-position ⟨*x-pos*⟩. Each value in the comma-separated list ⟨*note value list*⟩ corresponds to a note. Note that if you want to pass options to some notes, you can prefix [⟨*options*⟩] to the beginning of the note name, e.g. `\tmwhole{4}{[red]C4,E4}{a}`.

⟨*name*⟩ can be left empty, but as in the staff naming, I strongly advise you to find some name for each note set.

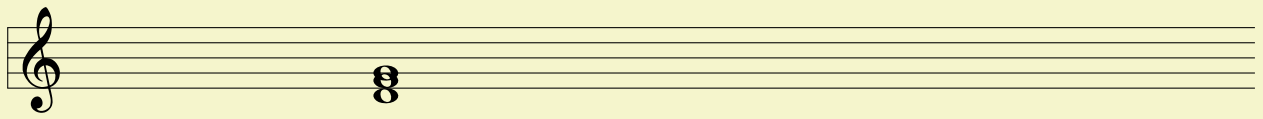
The center of note with note value x will be marked as coordinate (⟨*name*⟩-x). For example, note F5 will be marked as (⟨*name*⟩-F5). Also, the point on the middle line of the staff which is at ⟨*x-pos*⟩ will be marked as (⟨*name*⟩-center).



```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmwhole{3}{C4,E4,G4}{c-major}\tmwhole{5}{A4,C5,E5}{a-minor}
\end{tmstaff}
\end{tmline}
```

5.1.4 Relative positioning

Consider the following example:

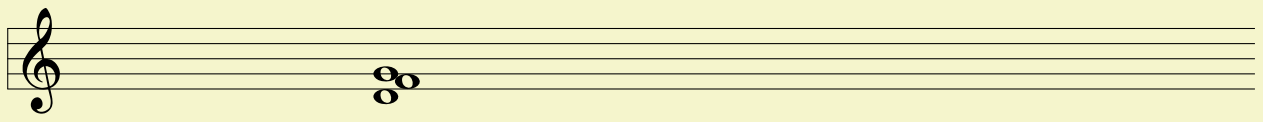


```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmwhole{5}{D4,F4,G4}{}% G7
\end{tmstaff}
\end{tmline}
```

It looks very bad, doesn't it? Note G4 should be shifted a bit to the right. You can achieve this by using the following key:

/tm/relative=⟨*relative position*⟩ (no default, initially center)

Apply relative position to the note. ⟨*relative position*⟩ can be either center, left or right.



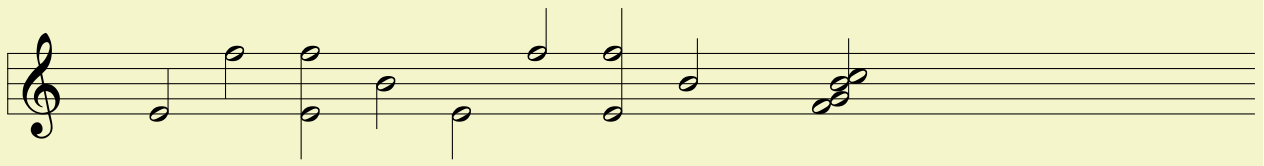
```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmwhole{5}{D4,[relative=right]F4,G4}{}
\end{tmstaff}
\end{tmline}
```

5.1.5 Half notes

\tmhalf[⟨*options*⟩]{⟨*x-pos*⟩}{⟨*note value list*⟩}{⟨*name*⟩}

Add a half note at position ⟨*x-pos*⟩. The stem direction will be automatically determined.

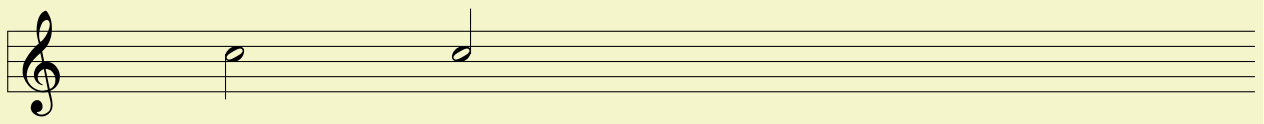
The end of the stem is marked as coordinate (⟨*name*⟩-tail). (This also applies to \tmquarter, \tmeighth and \tmmorethaneighth.)



```
\begin{tmline}%
\begin{tmstaff}{g}{}
  \tmhalf{2}{E4}{}\tmhalf{3}{F5}{}\tmhalf{4}{E4,F5}{}\tmhalf{5}{B4}{}
  \tmhalf[reversed]{6}{E4}{}      \tmhalf[reversed]{7}{F5}{}
  \tmhalf[reversed]{8}{E4,F5}{}   \tmhalf[reversed]{9}{B4}{}
  \tmhalf{11}{[relative=left]F4,B4,G4,[relative=right]C5}{}
\end{tmstaff}%
\end{tmline}
```

5.1.6 Stem direction

You can change the default direction of the stem by using `reversed`, see the following example and section 8.3.

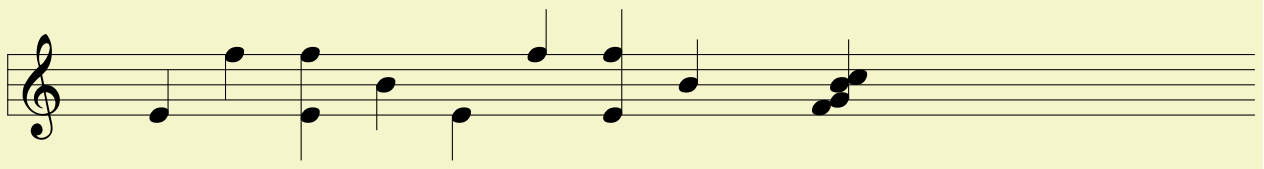


```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmhalf{3}{C5}{}\tmhalf[reversed]{6}{C5}{}
\end{tmstaff}
\end{tmline}
```

5.1.7 Quarter notes

\tmquarter[⟨*options*⟩]{⟨*x-pos*⟩}{⟨*note value list*⟩}{⟨*name*⟩}

Similar to `\tmhalf`.

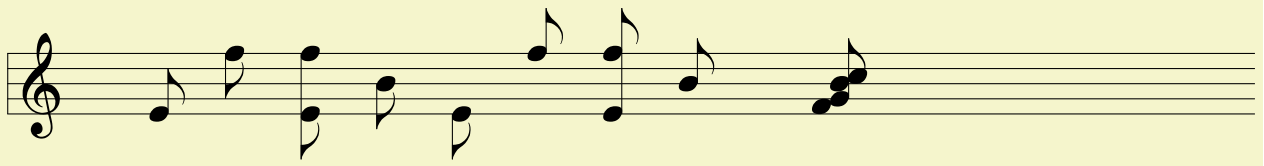


```
\begin{tmline}%
\begin{tmstaff}{g}{}
  \tmquarter{2}{E4}{}\tmquarter{3}{F5}{}\tmquarter{4}{E4,F5}{}\tmquarter{5}{B4}{}
  \tmquarter[reversed]{6}{E4}{}       \tmquarter[reversed]{7}{F5}{}
  \tmquarter[reversed]{8}{E4,F5}{}    \tmquarter[reversed]{9}{B4}{}
  \tmquarter{11}{[relative=left]F4,B4,G4,[relative=right]C5}{}
\end{tmstaff}%
\end{tmline}
```

5.1.8 Eighth notes

\tmeighth[⟨*options*⟩]{⟨*x-pos*⟩}{⟨*note value list*⟩}{⟨*name*⟩}

Similar to `\tmhalf`.



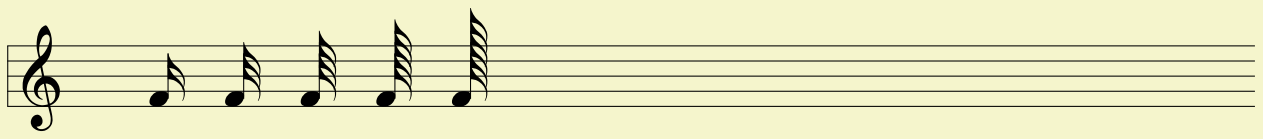
```
\begin{tmline}%
\begin{tmstaff}{g}{}
  \tmeighth {2}{E4}{}\tmeighth{3}{F5}{}\tmeighth{4}{E4,F5}{}\tmeighth{5}{B4}{}
  \tmeighth[reversed]{6}{E4}{}          \tmeighth[reversed]{7}{F5}{}
  \tmeighth[reversed]{8}{E4,F5}{}       \tmeighth[reversed]{9}{B4}{}
  \tmeighth{11}{[relative=left]F4,B4,G4,[relative=right]C5}{}
\end{tmstaff}%
\end{tmline}
```

5.1.9 More than eighth notes

The commands described in this section applies to every notes below the eighth notes, including the sixteenth note (semiquaver), the thirty-second note (demisemiquaver), etc.

\tmmorethaneighth[⟨*options*⟩]{⟨*x-pos*⟩}{⟨*note value list*⟩}{⟨*number of flags*⟩}{⟨*name*⟩}

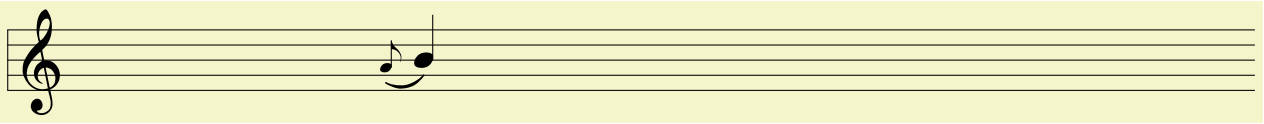
Similar to `\tmhalf`.



```
\begin{tmline}%
\begin{tmstaff}{g}{}
  \tmmorethaneighth{2}{F4}{2}{note-2}
  \tmmorethaneighth{3}{F4}{3}{note-3}
  \tmmorethaneighth{4}{F4}{4}{note-4}
  \tmmorethaneighth{5}{F4}{5}{note-5}
  \tmmorethaneighth{6}{F4}{6}{note-6}
\end{tmstaff}%
\end{tmline}
```

5.1.10 Grace notes

You can use `scale` to have smaller notes:

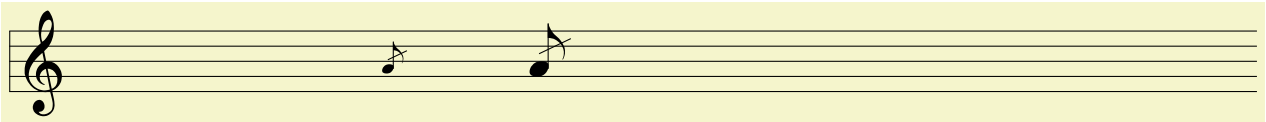


```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmeighth[scale=.6]{5}{A4}{a}\tmquarter[reversed]{5.5}{B4}{b}
  \tmslur[amplitude=1mm]{a}{b}
\end{tmstaff}
\end{tmline}
```

To strike the note, you can use the following key:

/tm/grace strike out=⟨*true or false*⟩ (default `true`)

Strike the grace note.



```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmeighth[scale=.6,grace strike out]{5}{A4}{}\tmeighth[grace strike out]{7}{A4}{}
\end{tmstaff}
\end{tmline}
```

5.2 Beaming

`\begin{`**`tmbeam`**`}[`⟨*options*⟩`]`
⟨*environment contents*⟩
`\end{`**`tmbeam`**`}`

Add a beaming note series. All notes inside this environment are 'beamed' together, and all stems point **upwards**.

`\begin{`**`tmbeam*`**`}[`⟨*options*⟩`]`
⟨*environment contents*⟩
`\end{`**`tmbeam*`**`}`

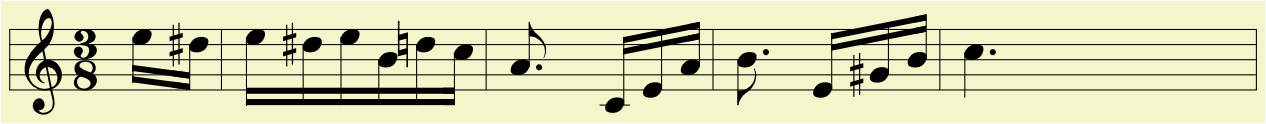
Identical to `{tmbeam}`, only all stems point **downwards**.

All notes to be beamed inside these environments need to be added using the following command (`\tmeighth`, ... will simply not work):

`\tmbeamnote``[`⟨*options*⟩`]{`⟨*x-pos*⟩`}{`⟨*note value*⟩`}{`⟨*number of flags*⟩`}{`⟨*name*⟩`}`

Add a note to the beaming series. If ⟨*number of flags*⟩ is 1, it is an eighth note, if ⟨*number of flags*⟩ is 2, it is a sixteenth note, and so on.

Important note: Because of the algorithm working behind the scene, you *must* give a separate name to each `\tmbeamnote` inside `{tmbeam}` or `{tmbeam*}`. Doing otherwise will result in weird output.



```
\begin{tmline}%
\begin{tmstaff}{g}{p1}
  \tmtimesignature{1}{3}{8}
  \begin{tmbeam*}
    \tmbeamnote{1.75}{E5}{2}{}\tmbeamnote{2.5}{[accidental=sharp]D5}{2}{a}
  \end{tmbeam*}
  \tmbarlineinline{2.8}
  \begin{tmbeam*}
    \tmbeamnote{3.25}{E5}{2}{a}\tmbeamnote{4}{[accidental=sharp]D5}{2}{b}\tmbeamnote{4.5}{E5}{2}{c}
    \tmbeamnote{5}{B4}{2}{d}\tmbeamnote{5.5}{[accidental=natural]D5}{2}{e}\tmbeamnote{6}{C5}{2}{f}
  \end{tmbeam*}
  \tmbarlineinline{6.3}\tmeighth[dot=1]{6.75}{A4}{a}
  \begin{tmbeam}
    \tmbeamnote{8}{C4}{2}{a}\tmbeamnote{8.5}{E4}{2}{b}\tmbeamnote{9}{A4}{2}{c}
  \end{tmbeam}
  \tmbarlineinline{9.3}\tmeighth[dot=1]{9.75}{B4}{a}
  \begin{tmbeam}
    \tmbeamnote{10.75}{E4}{2}{a}\tmbeamnote{11.5}{[accidental=sharp]G4}{2}{b}\tmbeamnote{12}{B4}{2}{c}
  \end{tmbeam}
  \tmbarlineinline{12.3}\tmquarter[dot=1]{12.75}{C5}{a}
\end{tmstaff}%
\tmbarlineendline{p1}{p1}%
\end{tmline}
```

5.3 Commands for rests

This package currently provides support for the following rests:

\tmwholerest[⟨*options*⟩]{⟨*x-pos*⟩}

Add a whole rest at x-position ⟨*x-pos*⟩.

\tmhalfrest[⟨*options*⟩]{⟨*x-pos*⟩}

Add a half rest at x-position ⟨*x-pos*⟩.

\tmquarterrest[⟨*options*⟩]{⟨*x-pos*⟩}

Add a quarter rest at x-position ⟨*x-pos*⟩.

\tmbelowquarterrest[⟨*options*⟩]{⟨*x-pos*⟩}{⟨*number*⟩}

Add a rest at x-position ⟨*x-pos*⟩, whose value is below a quarter rest. The rest has ⟨*number*⟩ 'flags': if ⟨*number*⟩ is 1, it is an eighth rest, if ⟨*number*⟩ is 2, it is a sixteenth rest, and so on... Currently ⟨*number*⟩ must be an integer between 1 and 4.

\tmeighthrest[⟨*options*⟩]{⟨*x-pos*⟩}

Identical to `\tmbelowquarterrest` where ⟨*number*⟩ is 1.

\tmsixteenthrest[⟨*options*⟩]{⟨*x-pos*⟩}

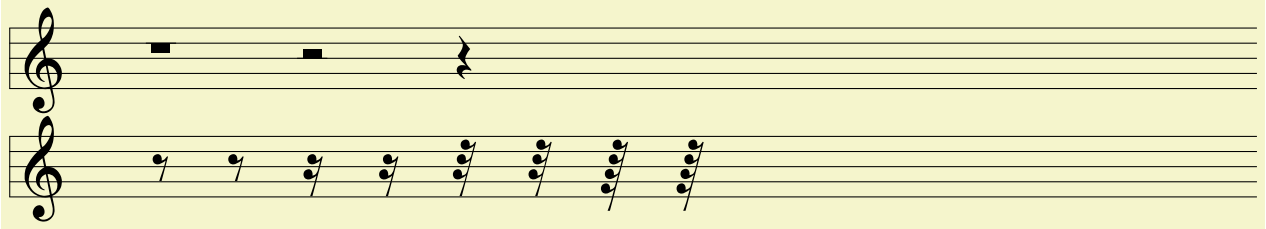
Identical to `\tmbelowquarterrest` where ⟨*number*⟩ is 2.

\tmthirtysecondrest[⟨*options*⟩]{⟨*x-pos*⟩}

Identical to `\tmbelowquarterrest` where ⟨*number*⟩ is 3.

\tmsixtyfourthrest[⟨*options*⟩]{⟨*x-pos*⟩}

Identical to `\tmbelowquarterrest` where ⟨*number*⟩ is 4.



```
\begin{tmline}%
\begin{tmstaff}{g}{}
  \tmwholerest{2}\tmhalfrest{4}\tmquarterrest{6}
\end{tmstaff}%
\begin{tmstaff}{g}{}
  \tmbelowquarterrest{2}{1}\tmeighthrest{3}
  \tmbelowquarterrest{4}{2}\tmsixteenthrest{5}
  \tmbelowquarterrest{6}{3}\tmthirtysecondrest{7}
  \tmbelowquarterrest{8}{4}\tmsixtyfourthrest{9}
\end{tmstaff}%
\end{tmline}
```

5.4 Miscellaneous

5.4.1 Accidentals

/tm/accidental=⟨*options*⟩ (no default, initially `none`)

Change key directory to `/tm/accidental options`. All other keys documented in this section needs to be put inside ⟨*options*⟩ to work.

By default, `/tm/accidental options/none` is executed.

/tm/accidental options/sharp (no value)

Add a sharp accidental to note.

/tm/accidental options/flat (no value)

Add a flat accidental to note.

/tm/accidental options/double sharp (no value)

Add a double sharp accidental to note.

/tm/accidental options/double flat (no value)

Add a double flat accidental to note.

/tm/accidental options/natural (no value)

Add a natural accidental to note.

/tm/accidental options/none (no value)

Doesn't add any accidentals. This is the default behaviour.

Sometimes the accidentals may not be in a good position. You can change their positions by using these transformation keys.

/tm/accidental options/xshift=⟨*length*⟩ (no default, initially `0pt`)

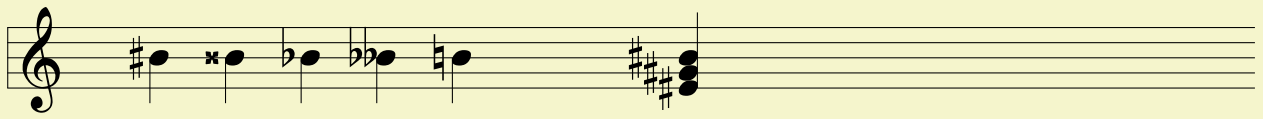
Shift the accidental by ⟨*length*⟩ in the horizontal direction.

/tm/accidental options/yshift=⟨*length*⟩ (no default, initially `0pt`)

Shift the accidental by ⟨*length*⟩ in the vertical direction.

/tm/accidental options/shift=⟨*coordinate*⟩ (no default)

Shift the accidental by ⟨*coordinate*⟩.



```
\begin{tmline}%
\begin{tmstaff}{g}{}
  \tmquarter{2}{[accidental=sharp]B4}{a}\tmquarter{3}{[accidental=double sharp]B4}{b}
  \tmquarter{4}{[accidental=flat]B4}{c} \tmquarter{5}{[accidental=double flat]B4}{d}
  \tmquarter{6}{[accidental=natural]B4}{e}
  \tmquarter{9}{[accidental=sharp]E4,[accidental={sharp,xshift=-2mm}]G4,[accidental={sharp,xshift=-4mm}]B4}{f}
\end{tmstaff}%
\end{tmline}
```

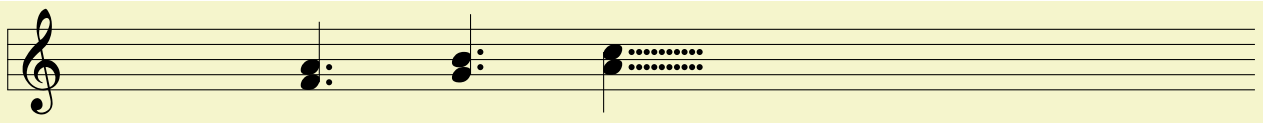
5.4.2 Dots

/tm/dot=⟨*dot options*⟩ (no default)

Change key path to `/tm/dot options`. You have to put all other keys specified in this section inside ⟨*dot options*⟩ for them to work.

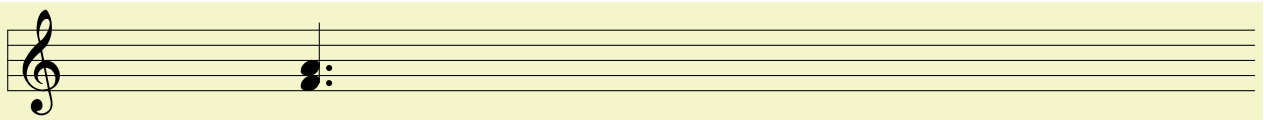
/tm/dot options/number=⟨*number*⟩ (no default, initially `0`)

Add ⟨*number*⟩ dot to note.



```
\begin{tmline}%
\begin{tmstaff}{g}{}
  \tmquarter[dot={number=1}]{4}{F4,A4}{a}\tmquarter[dot={number=1}]{6}{G4,B4}{b}
  \tmquarter[dot={number=10}]{8}{A4,C5}{c}
\end{tmstaff}%
\end{tmline}
```

Obviously, writing `dot={number=x}` is too long for such a job. Therefore, this package helps reduce the number of key strokes by identify any unknown 'key' in `/tm/dot options` that is a one-digit number as the value for `number`. Hence, you can use `dot=x` if you don't want to use transformation keys.



```
\begin{tmline}%
\begin{tmstaff}{g}{}
  \tmquarter[dot=1]{4}{F4,A4}{a}
\end{tmstaff}%
\end{tmline}
```

Like in the accidentals, you also have transformation options.

/tm/dot options/xshift=⟨*length*⟩ (no default, initially `0pt`)

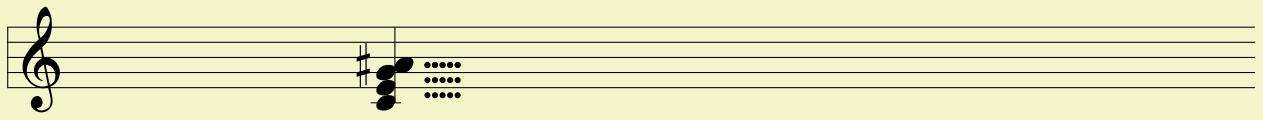
Shift the dots by ⟨*length*⟩ in the horizontal direction. This is probably the most commonly used of the three.

/tm/dot options/yshift=⟨*length*⟩ (no default, initially `0pt`)

Shift the dots by ⟨*length*⟩ in the vertical direction.

/tm/dot options/shift=⟨*coordinate*⟩ (no default)

Shift the dots by ⟨*coordinate*⟩.



```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter[dot={5,xshift=3mm}]{5}{C4,E4,G4,[{accidental=sharp,relative=right}]A4}{a}
\end{tmstaff}
\end{tmline}
```

5.4.3 Articulations

\tmstaccato[⟨*options*⟩]{⟨*note name*⟩}

Add *staccato* to note ⟨*note name*⟩.

\tmtenuto[⟨*options*⟩]{⟨*note name*⟩}

Add *tenuto* to note ⟨*note name*⟩.

\tmaccentabove[⟨*options*⟩]{⟨*note name*⟩}

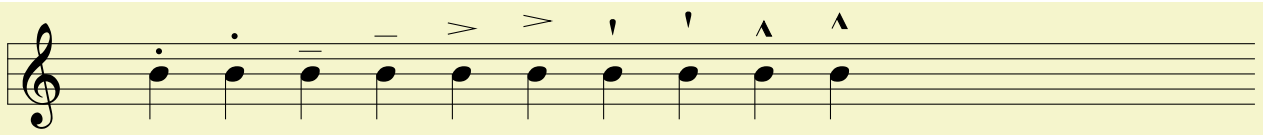
Add an accent to note ⟨*note name*⟩ (one form of *marcato*).

\tmstaccatissimo[⟨*options*⟩]{⟨*note name*⟩}

Add *staccatissimo* to note ⟨*note name*⟩.

\tmmarcato[⟨*options*⟩]{⟨*note name*⟩}

Add *marcato* to note ⟨*note name*⟩.



```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter{2}{B4}{x}\tmstaccato{x}
  \tmquarter{3}{B4}{x}\tmstaccato[line shift=2]{x}
  \tmquarter{4}{B4}{x}\tmtenuto{x}
  \tmquarter{5}{B4}{x}\tmtenuto[line shift=2]{x}
  \tmquarter{6}{B4}{x}\tmaccentabove{x}
  \tmquarter{7}{B4}{x}\tmaccentabove[line shift=1]{x}
  \tmquarter{8}{B4}{x}\tmstaccatissimo{x}
  \tmquarter{9}{B4}{x}\tmstaccatissimo[line shift=1]{x}
  \tmquarter{10}{B4}{x}\tmmarcato{x}
  \tmquarter{11}{B4}{x}\tmmarcato[line shift=1]{x}
\end{tmstaff}
\end{tmline}
```

For fermatas we have the following two commands:

\tmfermataabove[⟨*options*⟩]{⟨*note name*⟩}

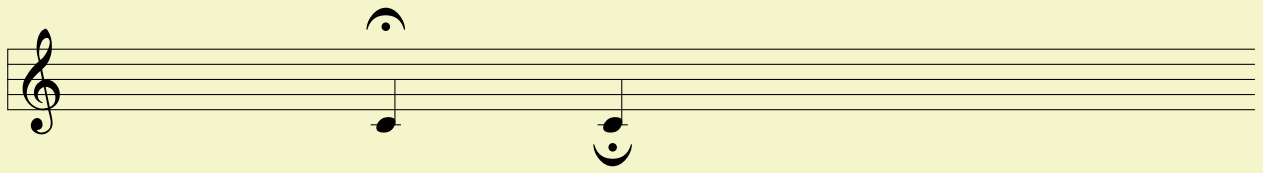
Add an 'above-fermata' to ⟨*note name*⟩.

\tmfermata[⟨*options*⟩]{⟨*note name*⟩}

Alias of `\tmfermataabove`.

\tmfermatabelow[⟨*options*⟩]{⟨*note name*⟩}

Add a 'below-fermata' to ⟨*note name*⟩.



```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter{5}{C4}{x}\tmfermataabove{x}
  \tmquarter{8}{C4}{x}\tmfermatabelow{x}
\end{tmstaff}
\end{tmline}
```

5.4.4 Ornaments

\tmtrill[⟨*options*⟩]{⟨*note name*⟩}

Add a trill to note ⟨*note name*⟩.

\tmturn[⟨*options*⟩]{⟨*note name*⟩}

Add a turn to note ⟨*note name*⟩.

\tmuppermordent[⟨*options*⟩]{⟨*note name*⟩}

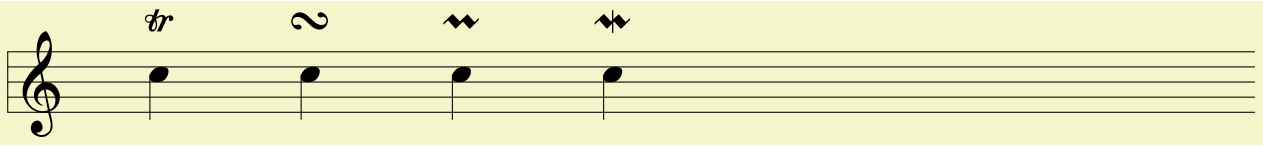
Add a 'upper' mordent to note ⟨*note name*⟩.

\tmlowermordent[⟨*options*⟩]{⟨*note name*⟩}

Add an 'lower' mordent to note ⟨*note name*⟩.

\tmmordent[⟨*options*⟩]{⟨*note name*⟩}

Alias of `\tmuppermordent`.



```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter{2}{C5}{a}\tmquarter{4}{C5}{b}\tmquarter{6}{C5}{c}\tmquarter{8}{C5}{d}
  \tmtrill{a}\tmturn{b}\tmuppermordent{c}\tmlowermordent{d}
\end{tmstaff}
\end{tmline}
```

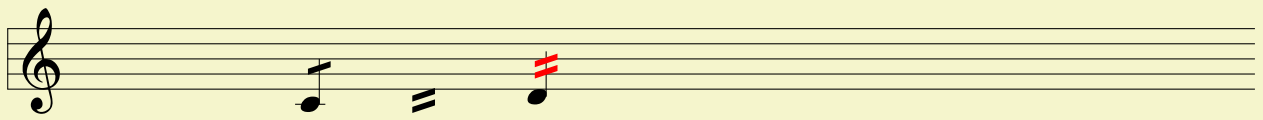
5.4.5 Tremolo

`\tmtremolo`[⟨*options*⟩]{⟨*note*⟩}{⟨*number*⟩}

Add a tremolo to note ⟨*note*⟩.

`\tmtremolocoordinate`[⟨*options*⟩]{⟨*coordinate*⟩}{⟨*number*⟩}

Add a tremolo to coordinate ⟨*coordinate*⟩.



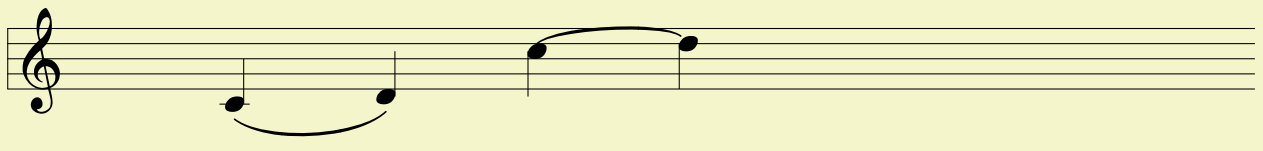
```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter{4}{C4}{a}\tmtremolo{a}{1}\tmquarter{7}{D4}{b}\tmtremolo[red]{b}{2}
  \tmtremolocoordinate{5.5,-.55}{2}
\end{tmstaff}
\end{tmline}
```

6 Lines

6.1 Slur

`\tmslur`[⟨*options*⟩]{⟨*note 1*⟩}[⟨*shift 2*⟩]{⟨*note 2*⟩}

Draw a slur joining ⟨*note 1*⟩ and ⟨*note 2*⟩. The slur will join the *lowest* notes of the two note set, i.e. it will go down and then go up.[2] You can change this direction using `reversed`.

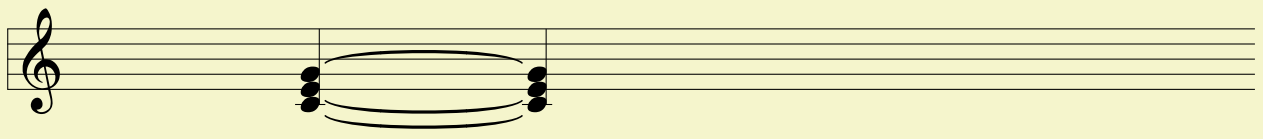


```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter{3}{C4}{a}\tmquarter{5}{D4}{b}\tmslur{a}{b}
  \tmquarter{7}{C5}{a}\tmquarter{9}{D5}{b}
  \tmslur[reversed,amplitude=1.5mm,start yshift=-1mm,end shift={(-1mm,-1mm)}]{a}{b}
\end{tmstaff}
\end{tmline}
```

`\tmslurcoordinate`[⟨*options*⟩]{⟨*coordinate 1*⟩}{⟨*coordinate 2*⟩}

Draw a slur from ⟨*coordinate 1*⟩ to ⟨*coordinate 2*⟩. The slur will go down and then go up.

You can use this command to tie two notes as follows. It is how `\tmtie` is currently working.



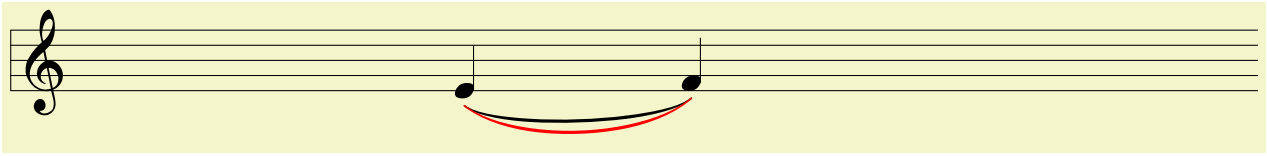
[2] Not being a native speaker, I can't find an appropriate English word for this :).

```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter{4}{C4,E4,G4}{a}\tmquarter{7}{C4,E4,G4}{b}
  \tmslurcoordinate[amplitude=1.5mm,start shift={(2mm,-1.5mm)},end shift={(-2mm,-1.5mm)}]{a-C4}{b-C4}
  \tmslurcoordinate[amplitude=1.5mm,start shift={(2mm,-1.5mm)},end shift={(-2mm,-1.5mm)}]{a-E4}{b-E4}
  \tmslurcoordinate[reversed,amplitude=1.5mm,start shift={(2mm,1.5mm)},end shift={(-2mm,1.5mm)}]{a-G4}{b-G4}
\end{tmstaff}
\end{tmline}
```

Essentially, the slur is drawn using the `calligraphic curved parenthesis` decoration, offered by the spath3 package. You can control the amplitude of this decoration, a.k.a. the height of the slur, by the following key:

/tm/amplitude=⟨*length*⟩ (no default, initially `2.5mm`)

Control the amplitude of the slurs.

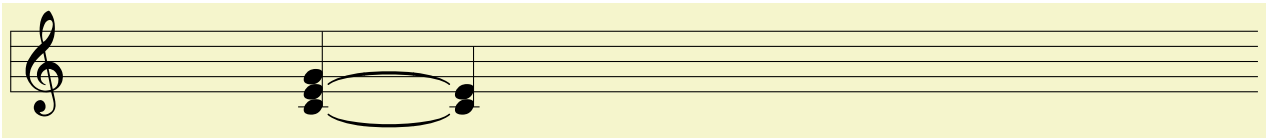


```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter{6}{E4}{a}\tmquarter{9}{F4}{b}
  \tmslur{a}{b}\tmslur[red,amplitude=4mm]{a}{b}
\end{tmstaff}
\end{tmline}
```

6.2 Tying notes

\tmtie[⟨*options*⟩]{⟨*note 1*⟩}{⟨*note 2*⟩}

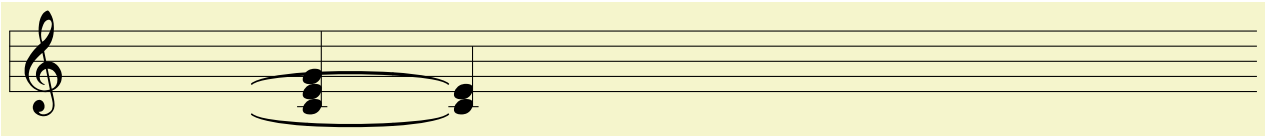
Add a tie between ⟨*note 1*⟩ and ⟨*note 2*⟩.



```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter{4}{C4,E4,G4}{a}\tmquarter{6}{C4,E4}{b}\tmtie{b}{a}
  %\tmtie{a}{b}: Error
\end{tmstaff}
\end{tmline}
```

Important note:

- `\tmtie` is intended to be used with note sets having more than one notes. Of course, with note sets having just one note it still works, but expected behaviour is not guaranteed. In those cases, use `\tmslur` and friends, documented in section 6.1, instead.
- The number of notes in ⟨*note 1*⟩ *must not* be more than that in ⟨*note 2*⟩. So, the order matters – in the example above, you can't use `\tmtie{a}{b}` because that will resulted in error. Of course, if ⟨*note 1*⟩ and ⟨*note 2*⟩ have the same number of notes, which is very usually the case, you can use any order as you want.
- Note that the starting coordinate is always the one having the less x-coordinate, no matter how the notes are ordered in `\tmtie`. In the example above, the starting coordinate is `a`, although it comes after `b`. So `start xshift` (say) will be applied to `a`, not `b`.



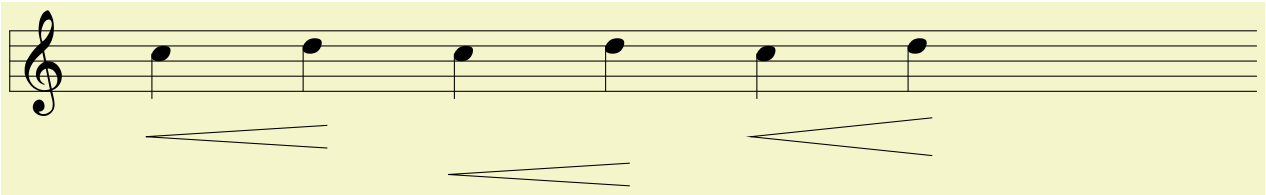
```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter{4}{C4,E4,G4}{a}\tmquarter{6}{C4,E4}{b}\tmtie[start xshift=-1cm]{b}{a}
\end{tmstaff}
\end{tmline}
```

6.3 Crescendo and diminuendo

6.3.1 Crescendo

\tmcrescendohairpin[⟨*options*⟩]{⟨*note 1*⟩}{⟨*note 2*⟩}

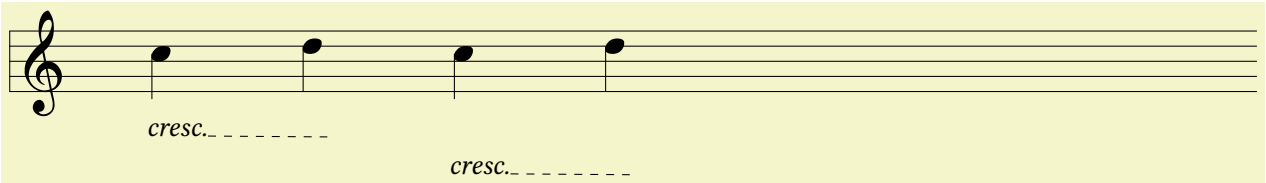
Draw a crescendo hairpin between ⟨*note 1*⟩ and ⟨*note 2*⟩.



```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter{2}{C5}{a} \tmquarter{4}{D5}{b} \tmcrescendohairpin{a}{b}
  \tmquarter{6}{C5}{a} \tmquarter{8}{D5}{b} \tmcrescendohairpin[yshift=-5mm]{a}{b}
  \tmquarter{10}{C5}{a}\tmquarter{12}{D5}{b}\tmcrescendohairpin[cresc dim sep=5mm]{a}{b}
\end{tmstaff}
\end{tmline}
```

\tmcrescendoline[⟨*options*⟩]{⟨*note 1*⟩}{⟨*note 2*⟩}

Draw a crescendo line between ⟨*note 1*⟩ and ⟨*note 2*⟩.




```
\begin{tmline}
  \begin{tmstaff}{g}{}
    \tmquarter{2}{C5}{a}\tmquarter{4}{D5}{b}\tmcrescendoline{a}{b}
    \tmquarter{6}{C5}{a}\tmquarter{8}{D5}{b}\tmcrescendoline[yshift=-5mm]{a}{b}
  \end{tmstaff}
\end{tmline}
```

\tmcrescendo[⟨*options*⟩]{⟨*note 1*⟩}{⟨*note 2*⟩}

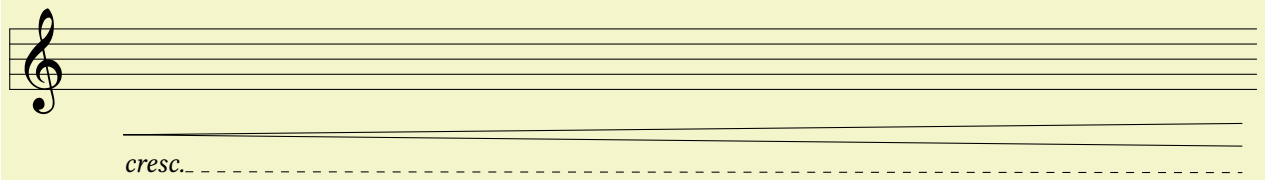
Alias of `\tmcrescendohairpin`.

\tmcrescendohairpincoordinate[⟨*options*⟩]{⟨*coordinate 1*⟩}{⟨*coordinate 2*⟩}

Draw a crescendo hairpin between ⟨*coordinate 1*⟩ and ⟨*coordinate 2*⟩. The coordinates do *not* include parentheses.

\tmcrescendolinecoordinate[⟨*options*⟩]{⟨*coordinate 1*⟩}{⟨*coordinate 2*⟩}

Draw a crescendo line between ⟨*coordinate 1*⟩ and ⟨*coordinate 2*⟩.




```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmcrescendohairpincoordinate{1.5,-1}{\linewidth-2mm,-1}
  \tmcrescendolinecoordinate{1.5,-1.5}{\linewidth-2mm,-1.5}
\end{tmstaff}
\end{tmline}
```

\tmcrescendocoordinate[⟨*options*⟩]{⟨*coordinate 1*⟩}{⟨*coordinate 2*⟩}

Alias of `\tmcrescendohairpincoordinate`.

6.3.2 Diminuendo

All commands are just like in crescendo (section 6.3.1).

\tmdiminuendohairpin[⟨*options*⟩]{⟨*note 1*⟩}{⟨*note 2*⟩}

Add a diminuendo hairpin between ⟨*note 1*⟩ and ⟨*note 2*⟩.

\tmdiminuendoline[⟨*options*⟩]{⟨*note 1*⟩}{⟨*note 2*⟩}

Add a diminuendo line between ⟨*note 1*⟩ and ⟨*note 2*⟩.

\tmdiminuendo[⟨*options*⟩]{⟨*note 1*⟩}{⟨*note 2*⟩}

Alias of `\tmdiminuendohairpin`.

\tmdiminuendohairpincoordinate[⟨*options*⟩]{⟨*coordinate 1*⟩}{⟨*coordinate 2*⟩}

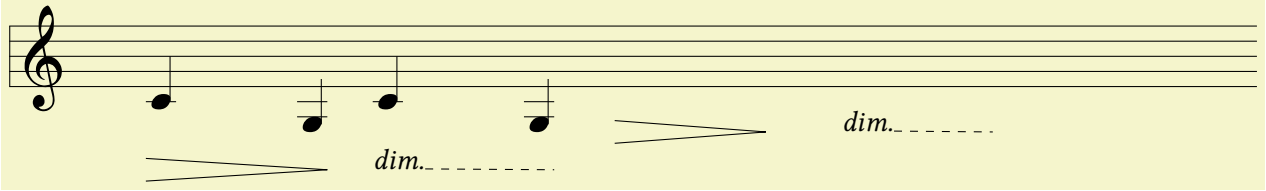
Add a diminuendo hairpin between ⟨*coordinate 1*⟩ and ⟨*coordinate 2*⟩.

\tmdiminuendolinecoordinate[⟨*options*⟩]{⟨*coordinate 1*⟩}{⟨*coordinate 2*⟩}

Add a diminuendo line between ⟨*coordinate 1*⟩ and ⟨*coordinate 2*⟩.

\tmdiminuendocoordinate[⟨*options*⟩]{⟨*coordinate 1*⟩}{⟨*coordinate 2*⟩}

Alias of `\tmdiminuendohairpincoordinate`.



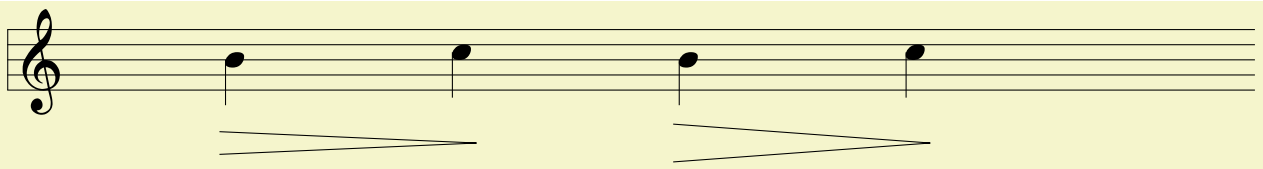

```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter{2}{C4}{a}\tmquarter{4}{G3}{b}\tmdiminuendo{a}{b}
  \tmquarter{5}{C4}{a}\tmquarter{7}{G3}{b}\tmdiminuendoline{a}{b}
  \tmdiminuendocoordinate{8,-1}{10,-1}\tmdiminuendolinecoordinate{11,-1}{13,-1}
\end{tmstaff}
\end{tmline}
```

6.3.3 Customization

You can change the head-width of the crescendo/diminuendo hairpins using the following key:

/tm/cresc dim sep=⟨*length*⟩ (no default, initially `3mm`)

Set the width of the head of the crescendo/diminuendo hairpins.



```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter{3}{B4}{a}\tmquarter{6}{C5}{b}\tmdiminuendo{a}{b}
  \tmquarter{9}{B4}{a}\tmquarter{12}{C5}{b}\tmdiminuendo[cresc dim sep=5mm]{a}{b}
\end{tmstaff}
\end{tmline}
```

6.4 Volta

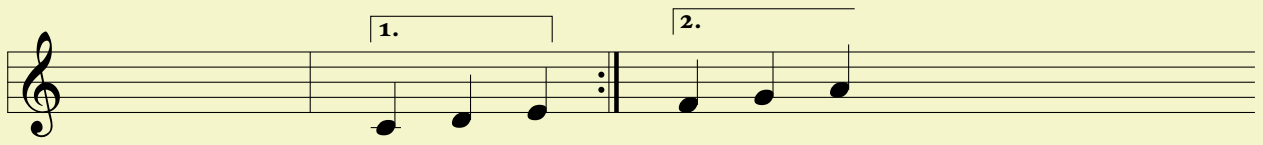
\tmvolta[⟨*options*⟩]{⟨*note 1*⟩}{⟨*note 2*⟩}{⟨*number*⟩}

Draw a volta line between ⟨*note 1*⟩ and ⟨*note 2*⟩.

By default, volta lines are closed. You can 'unclose' it by using

/tm/volta unclosed=⟨*true or false*⟩ (default `true`)

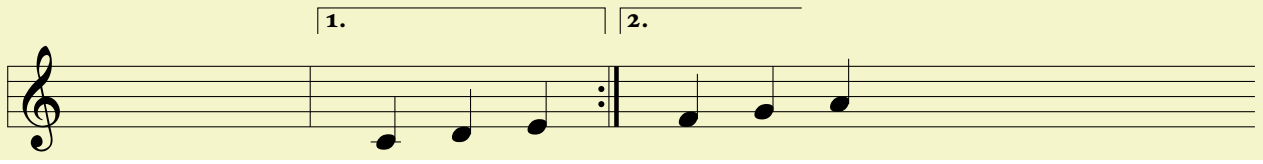
If set to `true`, the volta will be unclosed.




```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmbarlineinline{4}
  \tmquarter{5}{C4}{a}\tmquarter{6}{D4}{}\tmquarter{7}{E4}{b}
  \tmendrepeatbarlineinline{8}
  \tmquarter{9}{F4}{c}\tmquarter{10}{G4}{}\tmquarter{11}{A4}{d}
  \tmvolta{a}{b}{1}\tmvolta[volta unclosed]{c}{d}{2}
\end{tmstaff}
\end{tmline}
```

\tmvoltacoordinate[⟨*options*⟩]{⟨*coordinate 1*⟩}{⟨*coordinate 2*⟩}{⟨*number*⟩}

Draw a volta line between ⟨*coordinate 1*⟩ and ⟨*coordinate 2*⟩.

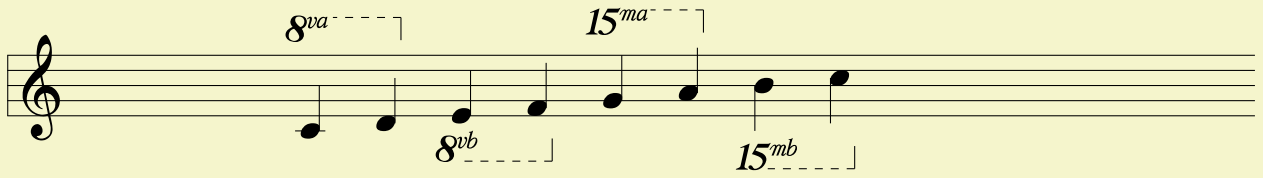


```
\begin{tmline}
  \begin{tmstaff}{g}{}
    \tmbarlineinline{4}
    \tmquarter{5}{C4}{}\tmquarter{6}{D4}{}\tmquarter{7}{E4}{}
    \tmendrepeatbarlineinline{8}
    \tmquarter{9}{F4}{}\tmquarter{10}{G4}{}\tmquarter{11}{A4}{}
    \tmvoltacoordinate{4.1,1}{7.9,1}{1}\tmvoltacoordinate[volta unclosed]{8.1,1}{10.5,1}{2}
  \end{tmstaff}
\end{tmline}
```

6.5 Octave lines

\tmoctave[⟨*options*⟩]{⟨*note 1*⟩}{⟨*note 2*⟩}{⟨*type*⟩}

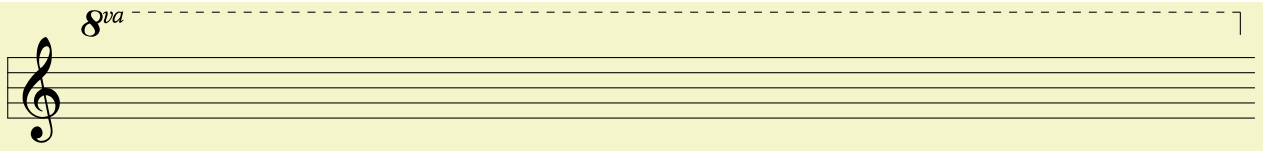
Draw an octave line between ⟨*note 1*⟩ and ⟨*note 2*⟩. ⟨*type*⟩ can be one of the following values: `8va`, `8vb`, `15ma`, `15mb`.




```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter{4}{C4}{a} \tmquarter{5}{D4}{b} \tmoctave{a}{b}{8va}
  \tmquarter{6}{E4}{a} \tmquarter{7}{F4}{b} \tmoctave{a}{b}{8vb}
  \tmquarter{8}{G4}{a} \tmquarter{9}{A4}{b} \tmoctave{a}{b}{15ma}
  \tmquarter{10}{B4}{a}\tmquarter{11}{C5}{b}\tmoctave{a}{b}{15mb}
\end{tmstaff}
\end{tmline}
```

\tmoctavecoordinate[⟨*options*⟩]{⟨*coordinate 1*⟩}{⟨*coordinate 2*⟩}{⟨*type*⟩}

Draw an octave line between ⟨*coordinate 1*⟩ and ⟨*coordinate 2*⟩.




```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmoctavecoordinate{1,1}{\linewidth-2mm,1}{8va}
\end{tmstaff}
\end{tmline}
```

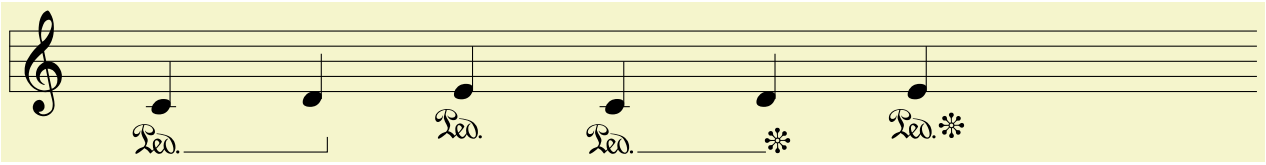
6.6 Pedal lines

\tmpedal[⟨*options*⟩]{⟨*note 1*⟩}{⟨*note 2*⟩}

Draw a pedal line not ended with a star (❋) between ⟨*note 1*⟩ and ⟨*note 2*⟩.

\tmpedalstar[⟨*options*⟩]{⟨*note 1*⟩}{⟨*note 2*⟩}

Draw a pedal line ended with a star between ⟨*note 1*⟩ and ⟨*note 2*⟩



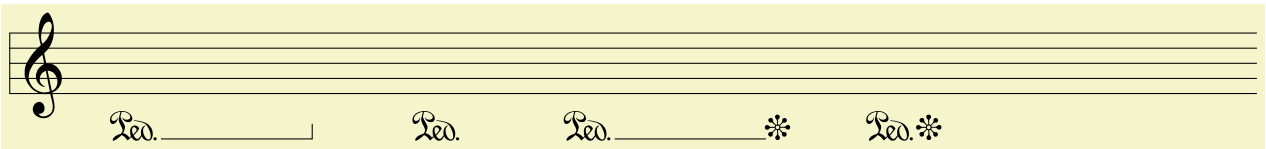
```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter{2}{C4}{a}\tmquarter{4}{D4}{b}\tmpedal{a}{b}
  \tmquarter{6}{E4}{a}\tmpedal{a}{a}
  \tmquarter{8}{C4}{a}\tmquarter{10}{D4}{b}\tmpedalstar{a}{b}
  \tmquarter{12}{E4}{a}\tmpedalstar{a}{a}
\end{tmstaff}
\end{tmline}
```

\tmpedalcoordinate[⟨*options*⟩]{⟨*coordinate 1*⟩}{⟨*coordinate 2*⟩}

Draw a pedal line not ended with a star between ⟨*coordinate 1*⟩ and ⟨*coordinate 2*⟩.

\tmpedalstarcoordinate[⟨*options*⟩]{⟨*coordinate 1*⟩}{⟨*coordinate 2*⟩}

Draw a pedal line ended with a star between ⟨*coordinate 1*⟩ and ⟨*coordinate 2*⟩.

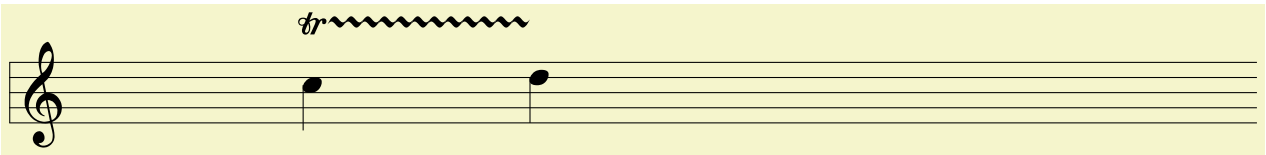


```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmpedalcoordinate{2,-1}{4,-1}\tmpedalcoordinate{6,-1}{6,-1}
  \tmpedalstarcoordinate{8,-1}{10,-1}\tmpedalstarcoordinate{12,-1}{12,-1}
\end{tmstaff}
\end{tmline}
```

6.7 Trill lines

\tmtrillline[⟨*options*⟩]{⟨*note 1*⟩}{⟨*note 2*⟩}

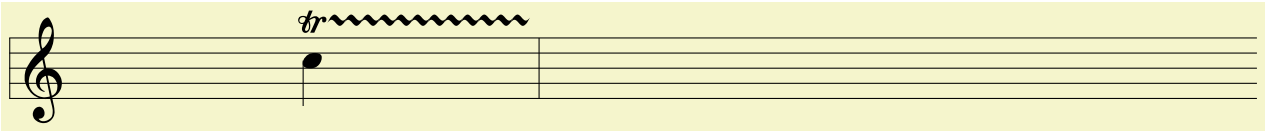
Draw a trill line from ⟨*note 1*⟩ to *the left* of ⟨*note 2*⟩.



```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter{4}{C5}{a}\tmquarter{7}{D5}{b}
  \tmtrillline{a}{b}% Note that this doesn't get over note b.
\end{tmstaff}
\end{tmline}
```

\tmtrilllinecoordinate[⟨*options*⟩]{⟨*coordinate 1*⟩}{⟨*coordinate 2*⟩}

Draw a trill line from ⟨*coordinate 1*⟩ to *the left* of ⟨*coordinate 2*⟩.

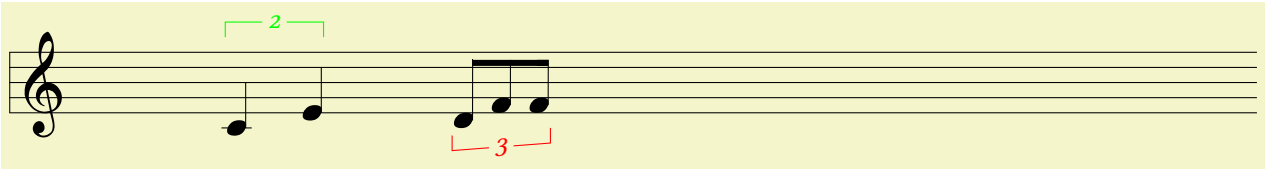


```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter{4}{C5}{}\tmbarlineinline{7}\tmtrilllinecoordinate{4,.6}{7,.6}
\end{tmstaff}
\end{tmline}
```

6.8 Tuplets

\tmtuplets[⟨*options*⟩]{⟨*note 1*⟩}{⟨*note 2*⟩}{⟨*number*⟩}

Draw a ⟨*number*⟩-th tuplet between ⟨*note 1*⟩ and ⟨*note 2*⟩. By default it will be drawn above the notes, but you can use `reversed` to draw them below the notes.




```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter{3}{C4}{a}\tmquarter{4}{E4}{b}
  \begin{tmbeam}
    \tmbeamnote{6}{D4}{1}{c}\tmbeamnote{6.5}{F4}{1}{}\tmbeamnote{7}{F4}{1}{d}
  \end{tmbeam}
  \tmtuplets[green]{a}{b}{2}\tmtuplets[reversed,red]{c}{d}{3}
\end{tmstaff}
\end{tmline}
```

7 Other in-line stuffs

7.1 Clefs

You can add a clef in-line using the following commands:

\tmgclef[⟨*options*⟩]{⟨*x-pos*⟩}

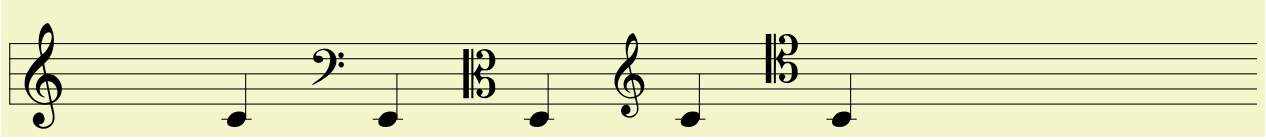
Add a treble clef at position ⟨*x-pos*⟩. The clef will be scaled down a bit as per standards. ⟨*shift*⟩ works like in `\tmkeysignature`.

\tmcclef[⟨*options*⟩]{⟨*x-pos*⟩}

Work like `\tmgclef`, but the clef is the alto clef.

\tmfclef[⟨*options*⟩]{⟨*x-pos*⟩}

Work like `\tmgclef`, but the clef is the bass clef.

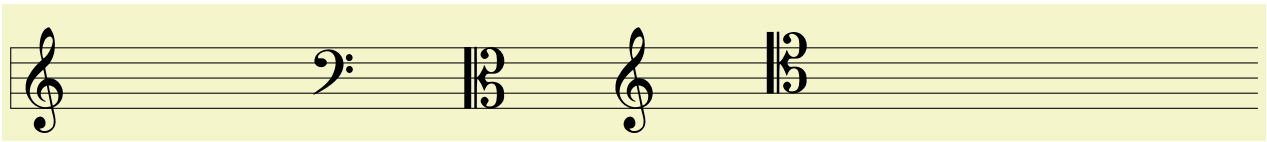


```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter{3}{C4}{}
  \tmfclef{4}\tmquarter{5}{C4}{}
  \tmcclef{6}\tmquarter{7}{C4}{}
  \tmgclef{8}\tmquarter{9}{C4}{}
  \tmcclef[line shift=2]{10}\tmquarter{11}{C4}{}
\end{tmstaff}
\end{tmline}
```

However, sometimes you don't want these clefs to be scaled. In those cases, you can use the following key:

/tm/unscaled=⟨*true or false*⟩ (default `true`)

Unscale the staves drawn by `\tmgclef` and friends.



```
\begin{tmline}
\begin{tmstaff}[unscaled]{g}{}
  \tmfclef{4}\tmcclef{6}\tmgclef{8}\tmcclef[line shift=2]{10}
\end{tmstaff}%
\end{tmline}
```

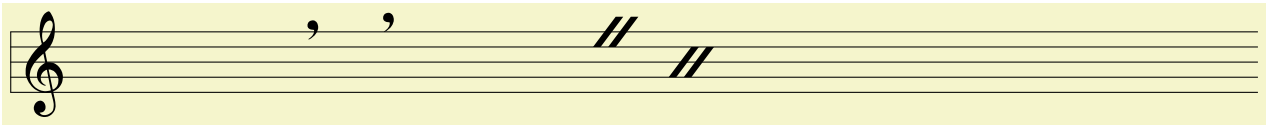
7.2 Breaths

\tmbreath[⟨*options*⟩]{⟨*x-pos*⟩}

Add a breath mark (a comma) to position ⟨*x-pos*⟩.

\tmcaesura[⟨*options*⟩]{⟨*x-pos*⟩}

Add a caesura to position ⟨*x-pos*⟩.

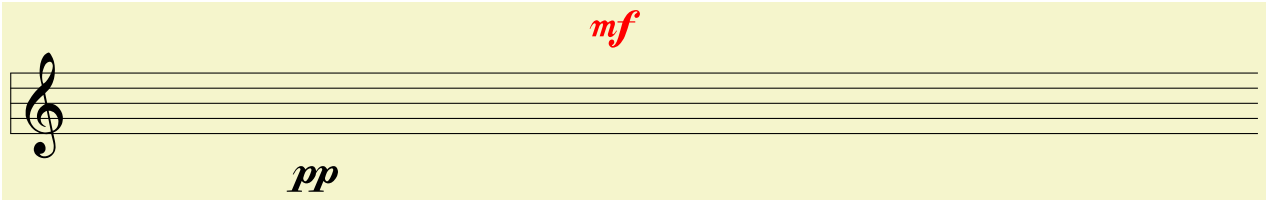


```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmbreath{4}\tmbreath[line shift=1]{5}\tmcaesura{8}\tmcaesura[line shift=-4]{9}
\end{tmstaff}
\end{tmline}
```

7.3 Dynamics

\tmdynamics[⟨*options*⟩]{⟨*coordinate*⟩}{⟨*type*⟩}

Add a dynamics notation to ⟨*coordinate*⟩. ⟨*type*⟩ can be one of the following values: `mp`, `p`, `pp`, `ppp`, `mf`, `f`, `ff`, `fff` and `fp`.



```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmdynamics{4,-1}{pp}\tmdynamics[red]{8,1}{mf}
\end{tmstaff}
\end{tmline}
```

7.4 Arpeggios and glissandi

7.4.1 Arpeggios

\tmarpeggio[⟨*options*⟩]{⟨*note*⟩}

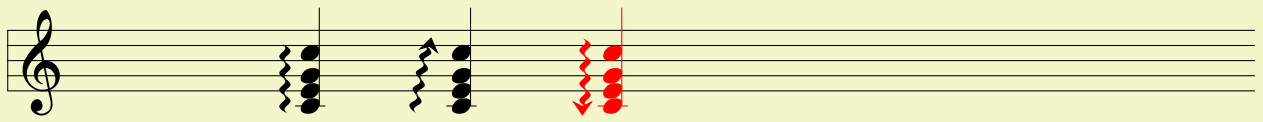
Draw an arpeggio to ⟨*note*⟩. Note that the starting point (for `start xshift` usage...) is the lower point.

\tmarpeggioup[⟨*options*⟩]{⟨*note*⟩}

Draw an 'increasiing' arpeggio to ⟨*note*⟩.

\tmarpeggiodown[⟨*options*⟩]{⟨*note*⟩}

Draw a 'decreasing' arpeggio to ⟨*note*⟩.

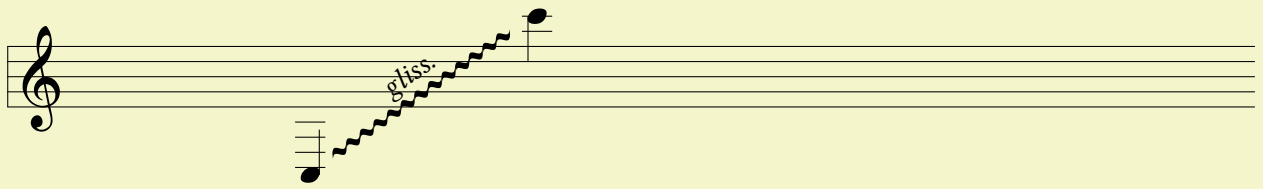


```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter{4}{C4,E4,G4,C5}{a}     \tmarpeggio{a}
  \tmquarter{6}{C4,E4,G4,C5}{a}     \tmarpeggioup[start xshift=-2mm]{a}
  \tmquarter[red]{8}{C4,E4,G4,C5}{a}\tmarpeggiodown[use note color]{a}
\end{tmstaff}
\end{tmline}
```

7.4.2 Glissandi

\tmglissando[⟨*options*⟩]{⟨*note 1*⟩}{⟨*note 2*⟩}

Draw a glissando from the lowest note of ⟨*note 1*⟩ to the highest note of ⟨*note 2*⟩.

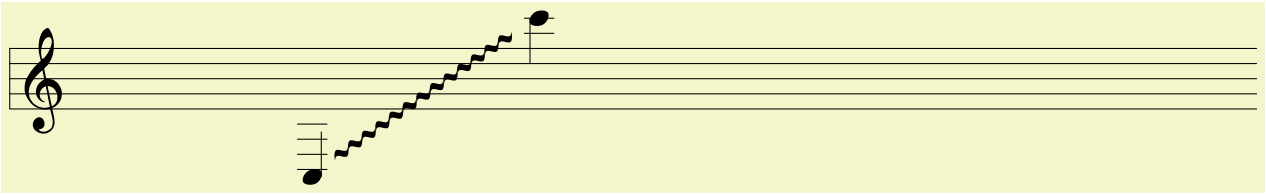



```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter{4}{C3}{a}\tmquarter{7}{C6}{b}\tmglissando{a}{b}
\end{tmstaff}
\end{tmline}
```

By default, the word *gliss.* will be displayed. You can hide that word (e.g. when the two notes are too close) by using this key:

/tm/hide text=⟨*true or false*⟩ (default `true`)

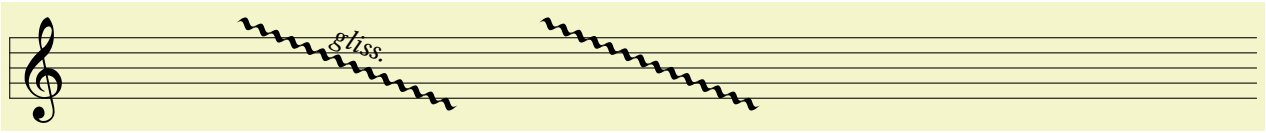
Hide the word *gliss.* in glissando commands.



```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter{4}{C3}{a}\tmquarter{7}{C6}{b}\tmglissando[hide text]{a}{b}
\end{tmstaff}
\end{tmline}
```

\tmglissandocoordinate[⟨*options*⟩]{⟨*coordinate 1*⟩}{⟨*coordinate 2*⟩}

Draw a ‘glissando line’ from ⟨*coordinate 1*⟩ to ⟨*coordinate 2*⟩.




```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmglissandocoordinate{3,.6}{6,-.6}\tmglissandocoordinate[hide text]{7,.6}{10,-.6}
\end{tmstaff}
\end{tmline}
```

7.5 Repeat notations

\tmmeasurerepeat[⟨*options*⟩]{⟨*x-pos*⟩}

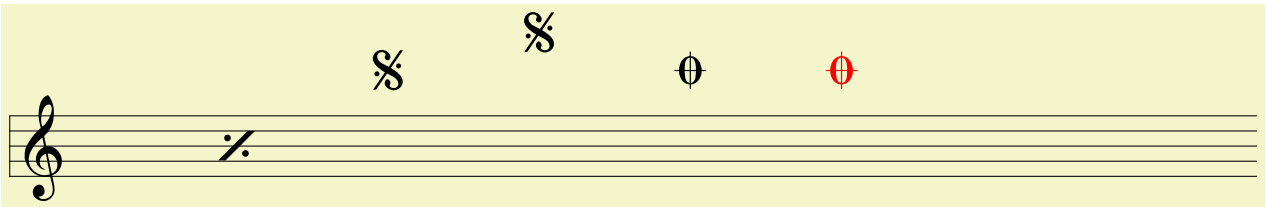
Add a ‘measure-repeat’ notation (𝄎) at ⟨*x-pos*⟩.

\tmsegno[⟨*options*⟩]{⟨*x-pos*⟩}

Add a segno at ⟨*x-pos*⟩.

\tmcoda[⟨*options*⟩]{⟨*x-pos*⟩}

Add a coda at ⟨*x-pos*⟩.



```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmmeasurerepeat{3}\tmsegno{5}\tmsegno[yshift=5mm]{7}\tmcoda{9}\tmcoda[red]{11}
\end{tmstaff}
\end{tmline}
```

8 Customization

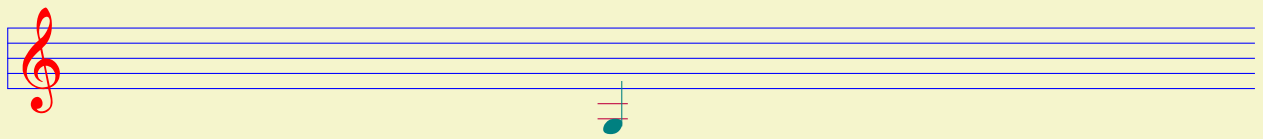
8.1 Color

/tm/line color=⟨*color*⟩ (no default, initially `black`)

Set the color of the lines, including the five main lines in each staff and the additional lines in the notes.

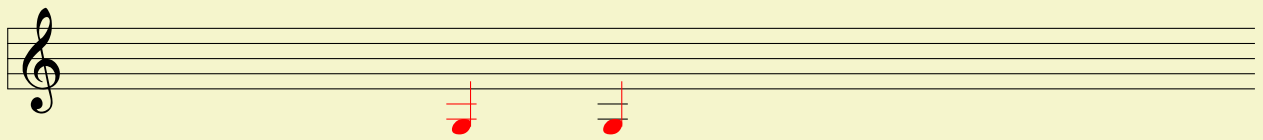
/tm/color=⟨*color*⟩ (no default, initially `black`)

Set the color of everything except the lines, whose color has been handled using `line color`.



```
\begin{tmline}
\begin{tmstaff}[color=red,line color=blue]{g}{}
  \tmquarter[color=teal,line color=purple]{8}{G3}{}
\end{tmstaff}
\end{tmline}
```

You can specify colors in a more compact way. If you want to set `color` and `line color` to the same color ⟨*color*⟩, you can use ⟨*color*⟩ as an option. This works pretty much like the way you use colors in Ti*k*Z. However, keep in mind that *both* `line color` and `color` are affected by specifing this way.

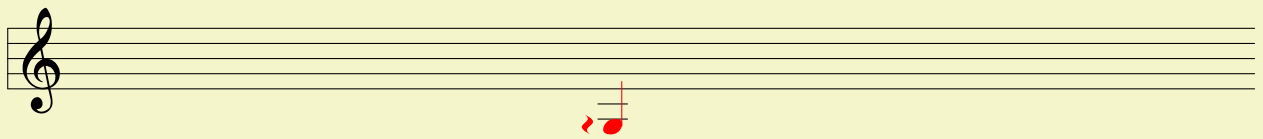


```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter[red]{6}{G3}{}
  \tmquarter[color=red]{8}{G3}{}
\end{tmstaff}
\end{tmline}
```

When adding notations to a note, e.g. when you use `\tmarpeggio` to a note, you might want to set the color of the additional notation to be the same as the color of that note. You can do so using the following key:

/tm/use note color=⟨*true or false*⟩ (default `true`)

If this key is set to `true`, the color of the additional notation is set to the color of its 'parent' note.

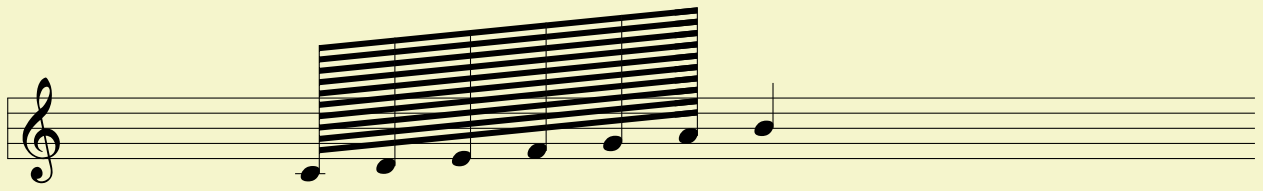


```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter[color=red]{8}{G3}{a}\tmarpeggio[use note color]{a}
\end{tmstaff}
\end{tmline}
```

8.2 Note length

/tm/note length=⟨*length*⟩ (no default, initially 6mm)

Reset the stem length of notes.



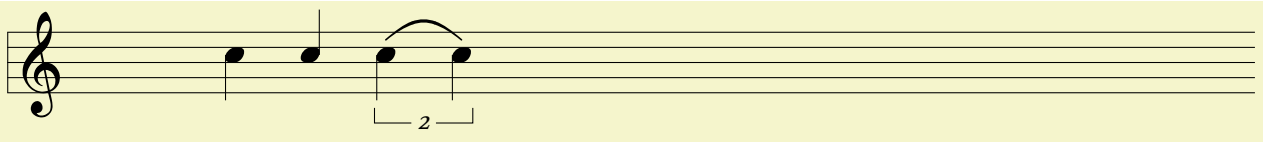
```
\begin{tmline}
\begin{tmstaff}{g}{}
  \begin{tmbeam}[note length=1.7cm]
    \tmbeamnote{4}{C4}{10}{a}\tmbeamnote{5}{D4}{10}{b}\tmbeamnote{6}{E4}{10}{c}
    \tmbeamnote{7}{F4}{10}{d}\tmbeamnote{8}{G4}{10}{e}\tmbeamnote{9}{A4}{10}{f}
  \end{tmbeam}
  \tmquarter[reversed]{10}{B4}{}
\end{tmstaff}
\end{tmline}
```

8.3 Reversing

/tm/reversed=⟨*true or false*⟩ (default true)

This key can alter the way some notations look:

- If this is applied to `\tmhalf`, `\tmquarter` and friends, this will change the direction of the stem. Note that this key does nothing when being applied to a beamed environment (see more in sectioin 5.1).
- If this key is applied to `\tmslur`, the direction of the slur will be reversed (see more in section 6.1).
- If this key is applied to `\tmtuplets`, it will change the position of the tuplet in relative to the notes (see more in section 6.8).




```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter{3}{C5}{a}\tmquarter[reversed]{4}{C5}{b}
  \tmquarter{5}{C5}{c}\tmquarter{6}{C5}{d}\tmslur[reversed]{c}{d}\tmtuplets[reversed]{c}{d}{2}
\end{tmstaff}
\end{tmline}
```

8.4 Transformations

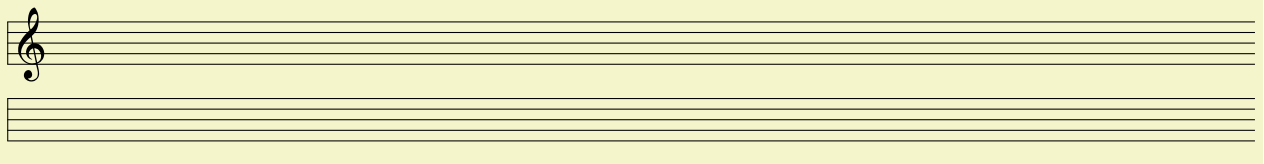
8.4.1 Scaling

/tm/scale=⟨*number*⟩ (no default, initially 1)

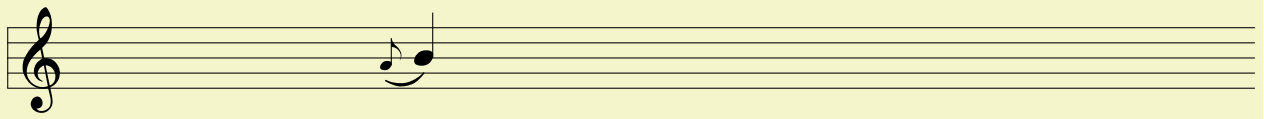
If applied to `{tmstaff}` or `{tmstaff*}`, scale the whole staff.

If applied to `{tmline}`, well, scale the whole line.

If applied to an unbeamed note, scale that note. You can, therefore, use `scale=0.7` (say) to get a grace note. **Note:** This feature is still experimental.



```
\begin{tmline}[scale=0.7]
\begin{tmstaff}{g}{}\end{tmstaff}%
\begin{tmstaff*}{}\end{tmstaff*}%
\end{tmline}%
```



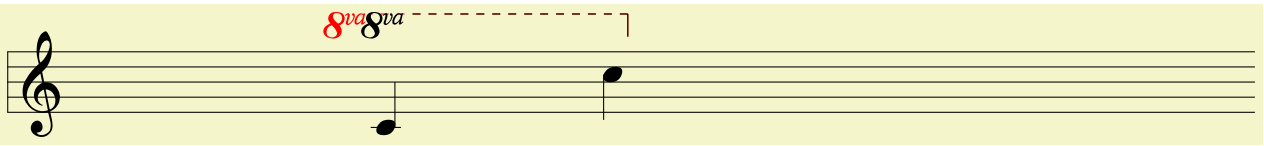
```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmeighth[scale=.6]{5}{A4}{a}\tmquarter[reversed]{5.5}{B4}{b}
  \tmslur[amplitude=1mm]{a}{b}
\end{tmstaff}
\end{tmline}
```

8.4.2 Shifting for lines

These applies to the lines command, see section 6 and `\tmbrace` and `\tmbracket`.

/tm/start xshift=⟨*length*⟩ (no default, initially `0pt`)

Shift the starting point of the line by ⟨*length*⟩ in the horizontal direction.




```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter{5}{C4}{a}\tmquarter{8}{C5}{b}
  \tmoctave[red,start xshift=-5mm]{a}{b}{8va}
  \tmoctave{a}{b}{8va} % For comparison
\end{tmstaff}
\end{tmline}
```

/tm/start yshift=⟨*length*⟩ (no default, initially `0pt`)

Shift the starting point of the line by ⟨*length*⟩ in the vertical direction.

/tm/start shift=⟨*coordinate*⟩ (no default)

Shift the starting point of the line by ⟨*coordinate*⟩.

/tm/end xshift=⟨*length*⟩ (no default, initially `0pt`)

Shift the ending point of the line by ⟨*length*⟩ in the horizontal direction.

/tm/end yshift=⟨*length*⟩ (no default, initially `0pt`)

Shift the ending point of the line by ⟨*length*⟩ in the vertical direction.

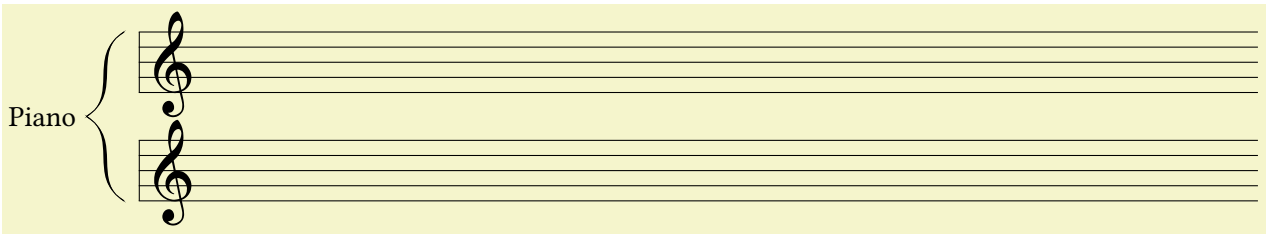
/tm/end shift=⟨*coordinate*⟩ (no default)

Shift the ending point of the line by ⟨*coordinate*⟩.

8.4.3 Brace-specific shifting

/tm/brace middle xshift=⟨*length*⟩ (no default, initially `0pt`)

Shift the middle point of the brace by ⟨*length*⟩ in the horizontal direction.




```
\begin{tmline}[staff offset=1.7cm]
\begin{tmstaff}{g}{a}\end{tmstaff}%
\begin{tmstaff}{g}{b}\end{tmstaff}%
\tmbrace[brace middle xshift=-3mm]{a}{b}{Piano}%
\end{tmline}
```

/tm/brace middle yshift=⟨*length*⟩ (no default, initially `0pt`)

Shift the middle point of the brace by ⟨*length*⟩ in the vertical direction.

/tm/brace middle shift=⟨*coordinate*⟩ (no default)

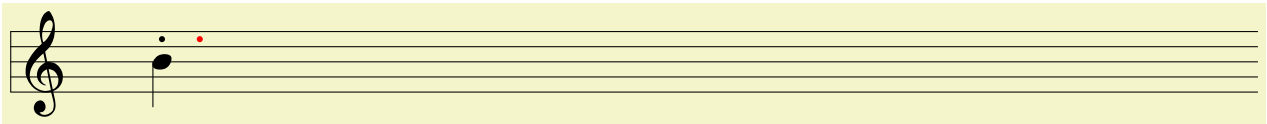
Shift the middle point of the brace by ⟨*coordinate*⟩.

8.4.4 Shifting for others

Keep in mind that shifting only works with some commands, like the articulations or the accidentals. Things whose coordinates are already specified, e.g. `\tmwhole`, may or may not be affected by these keys.

/tm/xshift=⟨*length*⟩ (no default, initially `0pt`)

Shift the object by ⟨*length*⟩ in the horizontal direction. In the lines this is a shorthand for `start xshift` and `end xshift`.



```
\begin{tmline}
\begin{tmstaff}{g}{}
  \tmquarter{2}{B4}{x}\tmstaccato[red,xshift=5mm]{x}\tmstaccato{x}
\end{tmstaff}
\end{tmline}
```

/tm/yshift=⟨*length*⟩ (no default, initially `0pt`)

Shift the object by ⟨*length*⟩ in the horizontal direction. In the lines this is a shorthand for `start yshift` and `end yshift`

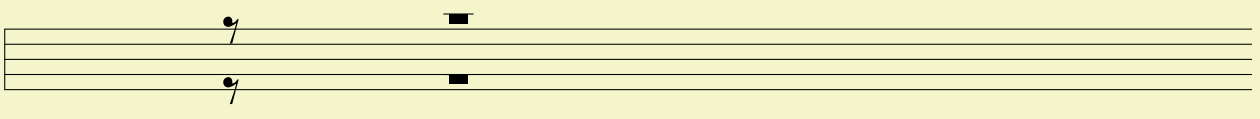
/tm/shift=⟨*coordinate*⟩ (no default)

Shift the object by ⟨*coordinate*⟩ in the horizontal direction. In the lines this is a shorthand for `start shift` and `end shift`.

/tm/line shift=⟨*number*⟩ (no default, initially `0`)

Shift the object by ⟨*number*⟩ 'note' lines. For example, if an accidental is displayed for a `D4` note, `line shift=1` will make it being displayed for a (possibly imaginary) `E4` (*not* `F4`!) note at that position. This is effectively `yshift` where ⟨*length*⟩ is set to ⟨*number*⟩ × ⟨*half of line sep*⟩.

See the following example for more information about how `line shift` works:



```
\begin{tmline}
\begin{tmstaff*}{}
  \tmeighthrest[line shift=4]{3}\tmeighthrest[line shift=-4]{3}
  \tmwholerest[line shift=4]{6} \tmwholerest[line shift=-4]{6}
\end{tmstaff*}
\end{tmline}
```

9 Outside the lines

9.1 Commands

\tmlittlehalf[⟨*options*⟩]

Give 𝅗𝅥. This is intended to be used in tempo text.

\tmlittlehalfdot[⟨*options*⟩]

Give 𝅗𝅥. (`\tmlittlehalf` with a dot).

\tmlittlequarter[⟨*options*⟩]

Give ♩.

\tmlittlequarterdot[⟨*options*⟩]

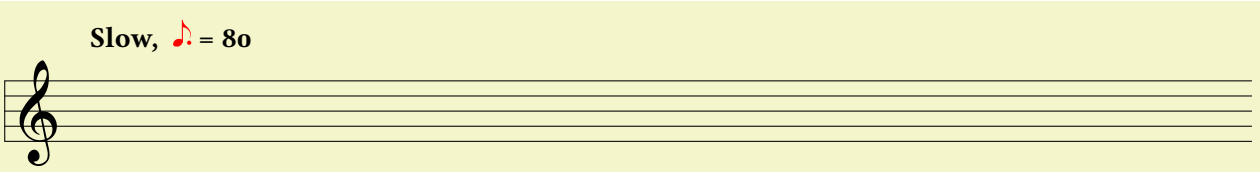
Give ♩. (`\tmlittlequarter` with a dot).

\tmlittleeighth[⟨*options*⟩]

Give ♪.

\tmlittleeighthdot[⟨*options*⟩]

Give ♪. (`\tmlittleeighth` with a dot).




```
\begin{tmline}
\begin{tmstaff}{g}{}
  \path (1,1) node[right,font=\bfseries] {Slow, \tmlittleeighthdot[red]\ = 80};
\end{tmstaff}
\end{tmline}
```

9.2 Pics

To construct a music line, this package defines many different Ti*k*Z pics. You can use them to define commands like in section 9.1.

9.2.1 Clefs

Pic type `tm-g-clef`

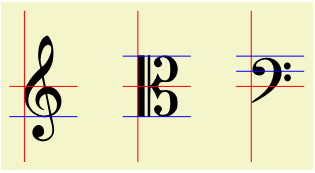
Draw a treble clef at normal size. When you use `\tmgclef` with `unscaled=false`, the clef is scaled down to 0.8 times its normal size.

Pic type `tm-c-clef`

Draw an alto clef at normal size.

Pic type `tm-f-clef`

Draw a bass clef at normal size.



```
\begin{tikzpicture}
  \begin{scope}
    \pic {tm-g-clef};
    \draw[red,ultra thin] (-.2,0) -- (.7,0) (0,1) -- (0,-1);
    \draw[blue,ultra thin] (-.2,-.4) -- (.7,-.4);
  \end{scope}
  \begin{scope}[xshift=1.5cm]
    \pic {tm-c-clef};
    \draw[red,ultra thin] (-.2,0) -- (.7,0) (0,1) -- (0,-1);
    \draw[blue,ultra thin] (-.2,-.4) -- (.7,-.4) (-.2,.4) -- (.7,.4);
  \end{scope}
  \begin{scope}[xshift=3cm]
    \pic {tm-f-clef};
    \draw[red,ultra thin] (-.2,0) -- (.7,0) (0,1) -- (0,-1);
    \draw[blue,ultra thin] (-.2,.2) -- (.7,.2) (-.2,.4) -- (.7,.4);
  \end{scope}
\end{tikzpicture}
```

The bounding boxes of `tm-c-clef` and `tm-f-clef` is set to the same height of `tm-g-clef` so that the distances between staves of different clefs can be equally positioned.

9.2.2 Note heads

We have three main versions: the whole note, the half note and the quarter note. Each main version has three different sub-versions for three different values of `relative`.

Whole note pics:

Pic type `tm-whole-note-center`

Draw a whole note.

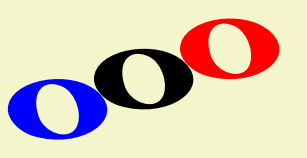
Pic type `tm-whole-note-left`

Draw a whole note.

Pic type `tm-whole-note-right`

Draw a whole note.

The difference of the three pics:



```
\begin{tikzpicture}[scale=4,transform shape]
  \path (0,0)   pic       {tm-whole-note-center}
        (0,.1)  pic[red]  {tm-whole-note-right}
        (0,-.1) pic[blue] {tm-whole-note-left};
\end{tikzpicture}
```

Half note pics:

Pic type `tm-half-note-head-center`

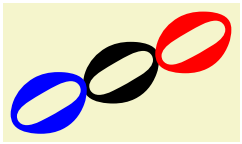
Draw a half note head.

Pic type `tm-half-note-head-left`

Draw a half note head.

Pic type `tm-half-note-head-right`

Draw a half note head.



```
\begin{tikzpicture}[scale=4,transform shape]
\path (0,0)   pic       {tm-half-note-head-center}
      (0,.1)  pic[red]  {tm-half-note-head-right}
      (0,-.1) pic[blue] {tm-half-note-head-left};
\end{tikzpicture}
```

Quarter note pics:

Pic type `tm-quarter-note-head-center`

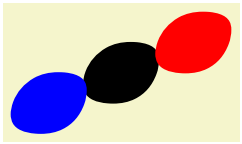
Draw a quarter note head.

Pic type `tm-quarter-note-head-left`

Draw a quarter note head.

Pic type `tm-quarter-note-head-right`

Draw a quarter note head.



```
\begin{tikzpicture}[scale=4,transform shape]
\path (0,0)   pic       {tm-quarter-note-head-center}
      (0,.1)  pic[red]  {tm-quarter-note-head-right}
      (0,-.1) pic[blue] {tm-quarter-note-head-left};
\end{tikzpicture}
```

9.2.3 Flags

Pic type `tm-note-flag-up`

Draw a flag for ‘up’-stems.

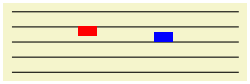
Pic type `tm-note-flag-down`

Draw a flag for ‘down’-stems.

```
\begin{tikzpicture}[scale=2,transform shape]
  \draw[red] (0,-.5) -- (0,0);
  \pic at (0,0) {tm-note-flag-up};
  \draw[red] (1,0) -- (1,-.5);
  \pic at (1,-.5) {tm-note-flag-down};
\end{tikzpicture}
```

9.2.4 Rests

Whole rest and half rest can be easily drawn with a simple rectangle, so there are no pics for them. Internally this is how they are drawn when `\tmwholerest` and `\tmhalfrest` are executed:

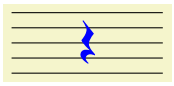


```
\begin{tikzpicture}
  \foreach \i in {-.4,-.2,0,.2,.4} \draw (0,\i) -- (3,\i);
  \fill[red] (.875,.08) rectangle ++ (.25,.12);
  \fill[blue] (1.875,0) rectangle ++ (.25,.12);
\end{tikzpicture}
```

Other rests:

Pic type `tm-quarter-note-rest`

Draw a quarter rest.



```
\begin{tikzpicture}
  \foreach \i in {-.4,-.2,0,.2,.4} \draw (0,\i) -- (2,\i);
  \pic[blue] at (1,0) {tm-quarter-note-rest};
\end{tikzpicture}
```

For eighth rest and below, the pic name is `tm-`⟨*number*⟩`-note-rest`, where ⟨*number*⟩ is the number of 'flags' in the rest notation. So `tm-1-note-rest` is the eighth rest, and so on. Currently ⟨*number*⟩ must be either `1`, 2, 3 or 4.

Pic type `tm-1-note-rest`

Draw an eighth rest.

Pic type `tm-2-note-rest`

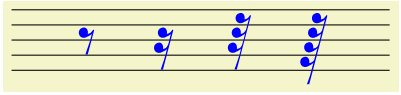
Draw an sixteenth rest.

Pic type `tm-3-note-rest`

Draw an thirty-second rest.

Pic type `tm-4-note-rest`

Draw an sixty-fourth rest.



```
\begin{tikzpicture}
  \foreach \i in {-.4,-.2,0,.2,.4} \draw (0,\i) -- (5,\i);
  \foreach \i in {1,2,3,4} \pic[blue] at (\i,0) {tm-\i-note-rest};
\end{tikzpicture}
```

9.2.5 Numbers

The pics draw musical numbers, taken from the music font *Maestro*. Digit ⟨*x*⟩ has a pic named `tm-number-`⟨*x*⟩. By default, these pics are `4mm` high.

Pic type `tm-number-0`

Draw number 0. **0**

Pic type `tm-number-1`

Draw number 1. **1**

Pic type `tm-number-2`

Draw number 2. **2**

Pic type `tm-number-3`

Draw number 3. **3**

Pic type `tm-number-4`

Draw number 4. **4**

Pic type `tm-number-5`

Draw number 5. **5**

Pic type `tm-number-6`

Draw number 6. **6**

Pic type `tm-number-7`

Draw number 7. **7**

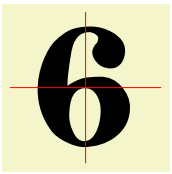
Pic type `tm-number-8`

Draw number 8. **8**

Pic type `tm-number-9`

Draw number 9. **9**

Position in relative to the origin:



```
\begin{tikzpicture}
  \pic[scale=4] at (0,0) {tm-number-6};
  \draw[ultra thin,red] (0,1) -- (0,-1) (-1,0) -- (1,0);
\end{tikzpicture}
```

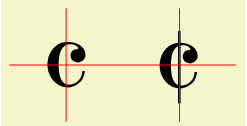
9.2.6 Other time signature notations

Pic type `tm-common-time`

Draw common time notation. **𝐂**

Pic type `tm-alla-breve-time`

Draw alla breve time notation. **𝐂**



```
\begin{tikzpicture}[scale=1.5,transform shape]
  \path (0,0) pic {tm-common-time} (1,0) pic {tm-alla-breve-time};
  \draw[ultra thin,red] (-.5,0) -- (1.5,0);
  \foreach \i in {0,1} \draw[ultra thin,red] (\i,.5) -- (\i,-.5);
\end{tikzpicture}
```

9.2.7 Accidentals

Pic type `tm-sharp`

Draw a sharp notation.

Pic type `tm-flat`

Draw a flat notation.

Pic type `tm-natural`

Draw a natural notation.

Pic type `tm-double-sharp`

Draw a double sharp notation.

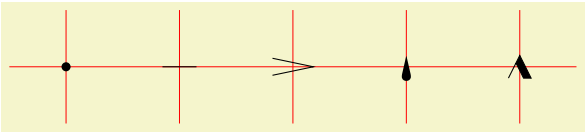
Pic type `tm-double-flat`

Draw a double flat notation.

```
\begin{tikzpicture}[scale=1.5,transform shape]
  \foreach \i in {0,1,2,3,4} \draw[ultra thin,red] (\i,-.5) -- (\i,.5);
  \draw[ultra thin,red] (-.5,0) -- (4.5,0);
  \path (0,0) pic {tm-sharp} (1,0) pic {tm-flat} (2,0) pic {tm-natural}
    (3,0) pic {tm-double-sharp} (4,0) pic {tm-double-flat};
\end{tikzpicture}
```

9.2.8 Articulations

Only fermata notations are drawn using pics. Other articulations are all drawn using normal Ti*k*Z commands. This is how those articulations are drawn internally:



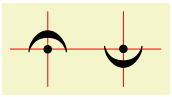
```
\begin{tikzpicture}[scale=1.5,transform shape]
  \draw[line width=.1pt,red] (-.5,0) -- (4.5,0);
  \foreach \i in {0,1,2,3,4} \draw[line width=.1pt,red] (\i,.5) -- (\i,-.5);
  \fill[shift={(0,0)}] (0,0) circle (.4mm);%staccato
  \draw[shift={(1,0)}] (-.15,0) -- (.15,0);%tenuto
  \draw[shift={(2,0)}] (-.18,.075) -- (.18,0) -- (-.18,-.075);%accent above
  \fill[shift={(3,0)},rounded corners=.5pt]
    (0,.1) -- (-.04,-.075) to[out=-90,in=-90,looseness=2] (.04,-.075) -- cycle;%staccatissimo
  \draw[shift={(4,0)},fill]
    (-.1,-.1) -- (0,.1) -- (.1,-.1) -- (.033333,-.1) -- (-.033333,.0333333);%marcato
\end{tikzpicture}
```

Pic type `tm-fermata-above`

Draw a ‘fermata above’ notation.

Pic type `tm-fermata-below`

Draw a ‘fermata below’ notation.



```
\begin{tikzpicture}
  \draw[line width=.1pt,red] (-.5,0) -- (1.5,0) (0,-.5) -- (0,.5) (1,-.5) -- (1,.5);
  \path (0,0) pic {tm-fermata-above} (1,0) pic {tm-fermata-below};
\end{tikzpicture}
```

9.2.9 Ornaments

Pic type `tm-trill`

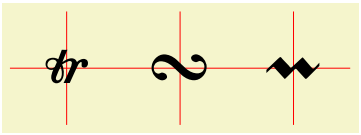
Draw a trill.

Pic type `tm-turn`

Draw a turn.

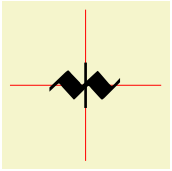
Pic type `tm-mordent`

Draw a mordent.



```
\begin{tikzpicture}[scale=1.5,transform shape]
  \draw[line width=.1pt,red] (-.5,0) -- (2.5,0);
  \foreach \i in {0,1,2} \draw[line width=.1pt,red] (\i,.5) -- (\i,-.5);
  \path (0,0) pic {tm-trill} (1,0) pic {tm-turn} (2,0) pic {tm-mordent};
\end{tikzpicture}
```

That `tm-mordent` is the ‘upper’ mordent. To have the ‘lower’ version, this is how the package is drawing internally:

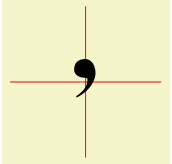


```
\begin{tikzpicture}[scale=2,transform shape]
  \draw[line width=.1pt,red] (-.5,0) -- (.5,0) (0,-.5) -- (0,.5);
  \path (0,0) pic {tm-mordent};
  \draw[line width=1pt] (0,-.15) -- (0,.15);
\end{tikzpicture}
```

9.2.10 Breath marks

Pic type `tm-breath-mark`

Draw a breath mark.



```
\begin{tikzpicture}[scale=2,transform shape]
  \draw[ultra thin,red] (-.5,0) -- (.5,0) (0,-.5) -- (0,.5);
  \pic at (0,0) {tm-breath-mark};
\end{tikzpicture}
```

The caesura is not drawn using a pic. This is how it is drawn:

```
\begin{tikzpicture}[scale=2,transform shape]
  \foreach \i in {-.4,-.2,0,.2,.4} \draw (-.5,\i) -- (.5,\i);
  \fill[purple]
    (-.3,.2) -- (-.2,.2) -- (.1,.6) -- (0,.6) -- cycle
    (-.1,.2) -- (0,.2) -- (.3,.6) -- (.2,.6) -- cycle;
\end{tikzpicture}
```

9.2.11 Lines-related notations

Currently the following pics are defined:

Pic type `tm-8va`

Draw a 8va notation.

Pic type `tm-8vb`

Draw a 8vb notation.

Pic type `tm-15ma`

Draw a 15ma notation.

Pic type `tm-15mb`

Draw a 15mb notation.

```
\foreach \i in {8va,8vb,15ma,15mb} {\tikz\pic{tm-\i};\quad}
```

Pic type `tm-ped`

Draw the stylised *ped* used in pedal lines.

Pic type `tm-ped-star`

Draw the star used in pedal lines.

```
\begin{tikzpicture}
  \draw (0,0) pic {tm-ped} -- (3,0) pic {tm-ped-star};
\end{tikzpicture}
```

9.2.12 Dynamics

Pic type `tm-dynamics-mf`

Draw dynamics notation `mf`.

Pic type `tm-dynamics-f`

Draw dynamics notation `f`.

Pic type `tm-dynamics-ff`

Draw dynamics notation `ff`.

Pic type `tm-dynamics-fff`

Draw dynamics notation `fff`.

Pic type `tm-dynamics-mp`

Draw dynamics notation `mp`.

Pic type `tm-dynamics-p`

Draw dynamics notation `p`.

Pic type `tm-dynamics-pp`

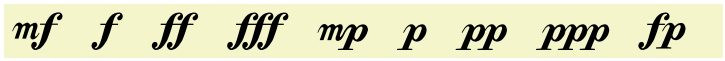
Draw dynamics notation `pp`.

Pic type `tm-dynamics-ppp`

Draw dynamics notation `ppp`.

Pic type `tm-dynamics-fp`

Draw dynamics notation `fp`.



```
\foreach \i in {mf,f,ff,fff,mp,p,pp,ppp,fp} {\tikz\pic{tm-dynamics-\i};\quad}
```

9.2.13 Repeat notations

Pic type `tm-measure-repeat`

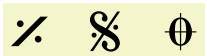
Draw 𝄎

Pic type `tm-segno`

Draw a segno.

Pic type `tm-coda`

Draw a coda.



```
\begin{tikzpicture}
  \path (0,0) pic {tm-measure-repeat} (1,0) pic {tm-segno} (2,0) pic {tm-coda};
\end{tikzpicture}
```

Index