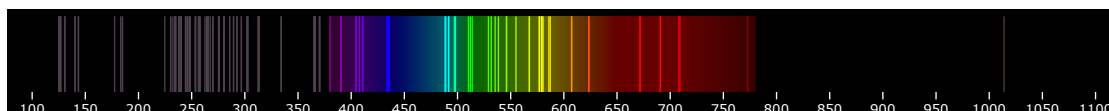


Manual for pgf-spectra 3.0.2

Hugo Gomes
hugo.parelho@gmail.com

16th June 2025



```
\pgfspectra[element=Hg,axis,axis step=50,begin=100,end=1100,back=visible40,gamma=.6,line width=.5pt]
```

Abstract

The purpose of this package is to draw the spectrum of elements in a simple way. It's based on the package *pst-spectra*, but with some extra options. It relies on pgf/TikZ to draw the desired spectrum, continuous or discrete. As in *pst-spectra* there is data available for the spectra of 98 elements and their ions. It also allows the user to draw a spectrum with their own custom data.

The lines data extends from Extreme UV to Near IR ($10\text{ nm} \leq \lambda \leq 4000\text{ nm}$). See section *The lines data* below for more information.

It is possible to redshift the lines of a spectrum, by directly entering the redshift value or the velocity and the angle to compute the redshift value (Doppler Redshift).

Spectral lines data can be presented in a table or exported to a file.

This package also provides color conversion (correlated color temperature), shadings for use with TikZ and/or PGFPLOTS and color maps for use with PGFPLOTS.

Contents

Installation and usage	1
What's new	2
In version 3.0.2	2
In version 3.0.1a	2
In version 3.0.1	2
In version 3.0.0	2
In version 2.2.0	3
In version 2.1.1	3
In version 2.1.0	3
In version 2.0.0	4
The lines data	6
Data based on pst-spectra	6
Data based on NIST	6
Commands overview	7
\pgfspectra	7
\wcolor{wavelength}	8
\pgfspectraStyle[options]	8
\pgfspectraStyleReset	9
\pgfspectrashade[<h v>](start,end){name}	9
\usepgfspectralibrary{name(s)}	9
The options for \pgfspectra	10
~> use visible shading	10
~> width	10
~> height	10
~> element	10
~> charge	11
~> Imin	11
~> relative intensity	11
~> relative intensity threshold	11
~> line intensity	12
~> gamma	13
~> brightness	13
~> back	14
~> backIRUV	14
~> IRcolor	14
~> UVcolor	15
~> lines	15
~> line width	15
~> begin	16
~> end	16
~> absorption	16
~> axis	17
~> axis step	17
~> axis ticks	17
~> axis unit	17
~> axis unit precision	18
~> axis label	19
~> axis label text	19
~> axis label position	19
~> axis color	19
~> axis font	20

↪ axis font color	20
↪ label	20
↪ label position	20
↪ label font	20
↪ label font color	21
↪ label before text	21
↪ label after text	21
↪ redshift	21
↪ show redshift value	22
Libraries	23
data	24
\pgfspectradata[name of data set]{keys}	24
The options for \pgfspectradata	24
↪ name of data set	24
↪ precision	24
↪ unit	25
\pgfspectratable[table options]{name(s) of data set(s)}	25
The options for \pgfspectratable	25
↪ title	25
↪ back color	27
↪ data back color	27
↪ text color	27
↪ width	28
↪ elements column width	28
\pgfspectrawrite[filename]{name(s) of data set(s)}	28
pgfplots	30
\pgfspectraplotmap[< h>]{name}	30
\pgfspectraplotshade[options]{name}	31
The options for \pgfspectraplotshade	32
↪ shade begin	32
↪ shade end	32
↪ shade opacity	33
↪ shade opacity color	34
↪ logarithmic	34
tempercolor	36
\tempercolor{Kelvin}	36
rainbow	37
\pgfspectrarainbow[tikz options](rainbow options){radius}	37
The options for \pgfspectrarainbow	38
↪ rainbow start	38
↪ rainbow knock out	38
↪ rainbow fade	39
↪ rainbow transparency	39
↪ rainbow background	39
Examples	41
Using \pgfspectra	41
Using \pgfspectrashade in TikZ	44
Using \pgfspectraplotshade and \pgfspectraplotmap with PGFPLOTS	45
Using \pgfspectrarainbow	62
Alphabetical list of available options	64
Recommendations and known issues	67

Installation and usage

pgf-spectra is placed under the terms of the L^AT_EX Project Public License, version 1.3 or later (<http://www.latex-project.org/lppl.txt>). pgf-spectra loads and only requires the package `TikZ`.

You need to put the style file (pgf-spectra.sty) in a location where PDF^LA_TE_X can find them. According to the TDS conventions this may be a subdirectory named tex/latex/pgfspectra/ or tex/latex/misc/ in your (site specific) installation tree (insert your appropriate directory delimiter instead of /, if needed).

If you are using PDF^LA_TE_X, just can simply include the style file without any option via the `\usepackage` command, `\usepackage{pgf-spectra}`

It can also be loaded with *one option* to select the data source:
`\usepackage[option]{pgf-spectra}`

For more detailed information see section [The lines data](#).

What's new

► In version 3.0.2

- Fixed incorrect interaction of `ith`. See <https://tex.stackexchange.com/questions/746103/problem-with-pgf-spectra-and-use-comma/746104#746104> for more details.

► In version 3.0.1a

- Removed lines data outside the range $10\text{ nm} - 4000\text{ nm}$ (H, He, Be, C, Si, Kr, Sn, I) in LSE data file, which generate a package warning, e.g. «Package pgf-spectra Warning: invalid wavelength (4051.1602nm). The wavelength must be lesser or equal to 4000nm (NIR)... on input line 27».
- Fixed chemical symbol label width producing incorrect alignment on consecutive lines for the spectrum of some elements, e.g.

```
\pgfspectraStyle[axis,begin=400,end=700,axis step=20,axis ticks=9,
axis color=lightgray,axis font color=black,label,width=.88\textwidth,height=0.7cm]
\noindent\pgfspectra[element=Cl]
\noindent\pgfspectra[element=Rn]
```

► In version 3.0.1

- Fixed a small bug dealing with the `axis label/axis label text` background color, which occurred when setting a *visible background* with the `back` key, e.g.:
`\pgfspectra[element=H,axis label text={wavenumber in nanometer},back=visible50]`

► In version 3.0.0

- In the `\pgfspectra` command, syntax for the key *lines* has been updated: it is now possible to specify a continuous range using *start wavelength to end wavelength*, for instance, `lines={380 to 500}` or `lines={380 to 500,633}` or `lines={633,555,380 to 500}`.
- New keys for `\pgfspectra`:
 - `axis label`
 - `axis label text`
 - `axis label position`
- New command `\usepgfspectralibrary` for loading pgf-spectra libraries.
- New library `data` to load spectral lines data, display it in a table or export it to a file. This library provides the commands:
 - `\pgfspectradata` to load the desired data.
 - `\pgfspectratable` displays loaded data in a table.
 - `\pgfspectrawrite` writes loaded data to a file.
- Some commands from previous versions have been moved to new libraries:
 - `\pgfspectraplotshade` and `\pgfspectraplotmap` moved to new library `pgfplots`.
 - `\tempercolor` moved to new library `tempercolor`.
 - `\pgfspectrarainbow` moved to new library `rainbow`.

► In version 2.2.0

- LSE data *renormalized* to lower values within \TeX capabilities. This prevents Dimension too large errors and subsequent errors in spectrum drawing when using the **relative intensity** key.
- Added maximum intensities in IR, visible an UV ranges for both data sources (NIST and LSE). Now, when using the **relative intensity** key, the interval of wavelengths is detected and the respective maximum intensity is used. For example, if the spectrum is within the visible range, say between 400 and 700 nanometers, the maximum visible intensity will be used. The same is true for ranges only in IR or UV ranges. But, if the spectrum to be drawn is in more than one region – for example, from 300 to 1000 or from 400 to 1000 or from 300 to 700 nanometers – the maximum intensity that will be used is defined as the maximum intensity of all data.
- Fixed some typos in the manual.

► In version 2.1.1

- Code rewrote for the command `\pgfspectraplotshade`, supporting two new keys:
 - shade begin
 - logarithmic

► In version 2.1.0

- The continuous visible region is now drawn via *TikZ* shading, improving a little bit the speed of the whole process.
- Minor fix: the width of the emission/absorption lines are now correctly drawn.
- New keys for `\pgfspectra`:
 - use visible shading
 - backVIS
 - axis unit
 - axis unit precision
- New color conversion command, which converts a temperature in Kelvin to the correspondent rgb color, based on correlated color temperature:
 - `\tempercolor{temperature in Kelvin}`
- New commands that provides shadings to use in *TikZ*:
 - `\pgfspectrashade[<h|v>](start,end){name}`
 - `\pgfspectrarainbow[tikz options](rainbow options){radius}`

The *TikZ* keys that affect the rainbow are:

 - * color
 - * opacity
 - * scope fading

The specific rainbow options are:

- * rainbow fade
- * rainbow start
- * rainbow knock out
- * rainbow background
- * rainbow transparency
- New command that provides a shading to use in PGFPLOTS:
 - `\pgfspectraplotshade[options]{name}` with the following specific keys
 - * shade end
 - * shade opacity
 - * shade opacity color
- New command that builds a color map to use in PGFPLOTS:
 - `\pgfspectraplotmap[<|h>]{name}`

► In version 2.0.0

- The package can now be loaded with one of the following options:
 - `\usepackage[NIST]{pgf-spectra}` (**default**)
 - `\usepackage[LSE]{pgf-spectra}`
- Range of spectral window from 10 nm to 4000 nm (previous version from 380 nm to 780 nm).
- Added the lines data outside the visible range for the 98 elements.
- No more dependency on the package `ifthen` (code rewritten with the `\ifx` \TeX primitive).
- Setting/disabling global options to draw the spectra's with the new commands:
 - `\pgfspectraStyle[options]`
 - `\pgfspectraStyleReset`
- New keys:
 - axis ticks
 - backIRUV (only for emission spectrum)
 - IRcolor (for emission lines and for background in absorption spectrum)
 - UVcolor (for emission lines and for background in absorption spectrum)
 - redshift
 - show redshift value
- The issues with the zooming of the pdf viewer sometimes introducing blank lines in the spectra have been fixed:



The rendition should now be working for every zoom (I hope!):

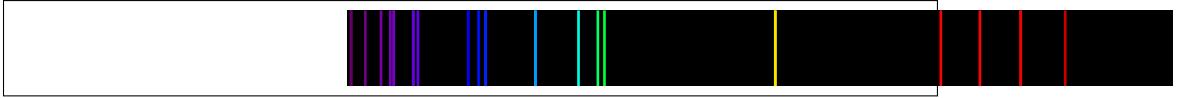


Many thanks to *Daniel García's* suggestion to solve this problem!

- Fixed the problem when putting the spectra inside any horizontal $\text{T}_\text{E}\text{X}$ box, like `\makebox`, `\mbox` or `\hbox`.

For instance, the code `\makebox[\textwidth][c]{\pgfspectra[element=He]}`:

- had as a result in the previous version (version 1.0):



- and will result in version 2.0.0 at:



The lines data

There are two data sets available for drawing the spectra: one based in the previous version, whose data was initially obtained from the package `pst-spectra` and the other obtained from `NIST`.

In both cases are included the lines for 98 elements, from hydrogen ($Z = 1$) to einsteinium ($Z = 99$), except for francium ($Z = 87$). For each element there are lines between 10 nm and 4000 nm (obtained from the referred pages at February 2021).

Data based on `pst-spectra`

This set was obtained from <http://cdsarc.u-strasbg.fr/viz-bin/Cat?VI/16>

According to the information on the page the listed lines are based on "Line Spectra of the Elements", Joseph Reader and Charles H. Corliss CRC Handbook of Chemistry and Physics. This book refers that «The table contains the outstanding spectral lines of neutral (*I*) and singly ionized (*II*) atoms of the elements from hydrogen through plutonium ($Z = 1 - 94$); selected strong lines from doubly ionized (*III*), triply ionized (*IV*), and quadruply ionized (*V*) atoms are also included.»

Note: `pst-spectra` documentation refers "*Line Spectra of the Elements from the Astronomical Data Center of NASA*" as the source material, but I'm assuming the original source is "*Line Spectra of the Elements*", Joseph Reader and Charles H. Corliss CRC Handbook of Chemistry and Physics, obtained from <http://cdsarc.u-strasbg.fr/viz-bin/Cat?VI/16>.

To use this data set load the package `pgf-spectra` with the option `LSE` (acronym to Line Spectra of the Elements):

```
\usepackage[LSE]{pgf-spectra}
```

Number of lines provided: 46031 (see file `pgf-spectraDataLSE.pdf`)

Data based on `NIST`

This set was obtained from <https://physics.nist.gov/PhysRefData/Handbook/Tables/findinglist.htm>

According to the information on the page the listed lines «includes data for the neutral and singly-ionized atoms».

Note: **This set is loaded by default.** Because the data to search is slightly smaller (only neutral and singly-ionized atoms) the time consumption when building the spectra could be a bit lower.

To use this data set load the package `pgf-spectra` without options:

```
\usepackage{pgf-spectra}
```

Number of lines provided: 11980 (see file `pgf-spectraDataNIST.pdf`);

Commands overview

The *main* commands, those always available, are:

- `\pgfspectra` or `\pgfspectra[options list]`
- `\wcolor{wavelength}`
- `\pgfspectraStyle[options]`
- `\pgfspectraStyleReset`
- `\pgfspectrashade[<h|v>](start,end){name}`
- `\usepgfspectralibrary{name(s)}`

The commands available after loading libraries are:

- `\pgfspectradata[name of data set]{keys}` (*data* library)
- `\pgfspectratable[table options]{name(s) of data set(s)}` (*data* library)
- `\pgfspectrawrite[filename]{name(s) of data set(s)}` (*data* library)
- `\pgfspectraplotshade[options]{name}` (*pgfplots* library)
- `\pgfspectraplotmap[<l|h>]{name}` (*pgfplots* library)
- `\tempercolor{Kelvin}` (*tempercolor* library)
- and finally *just for fun* a command that draws a rainbow, `\pgfspectrarainbow[tikz options](rainbow options){radius}` (*rainbow* library)

► Utilization of `\pgfspectra`

This command is used without options to draw the visible continuous spectrum:

```
\pgfspectra
```

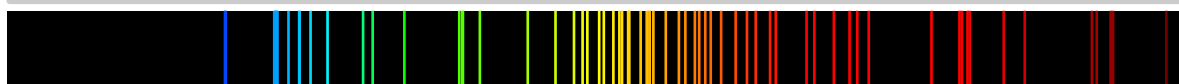


When using options, a continuous or discrete spectra in the visible region can be drawn, for instance:

```
\pgfspectra[width=.5\textwidth,height=1.25cm]
```



```
\pgfspectra[width=\textwidth,element=Ne]
```



The options available for `\pgfspectra` are described in section [The options for `\pgfspectra`](#).

► Utilization of `\wcolor{wavelength}`

A command to convert a wavelength from 380 to 780 nanometres (or other value in the range $10\text{ nm} \leq \lambda \leq 4000\text{ nm}$) to the respective color available as 'wcolor':

```
\tikz{\foreach \x in {380,400,...,780}{\wcolor{\x}
  \draw[fill=wcolor] (.03*\x,0) rectangle ++(.6,.5)
  node[midway,font=\tiny\bfseries,text=black!50] {\x};}}
```



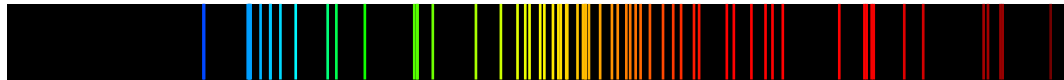
```
\tikz{\foreach \x/\y in {10/0,100/1,500/2,1000/3,2000/4,3000/5,4000/6}{\wcolor{\x}
  \draw[fill=wcolor] (\y,0) rectangle ++(1,.5)
  node[midway,font=\tiny\bfseries,text=black!50] {\x};}}
```



► Utilization of `\pgfspectraStyle[options]`

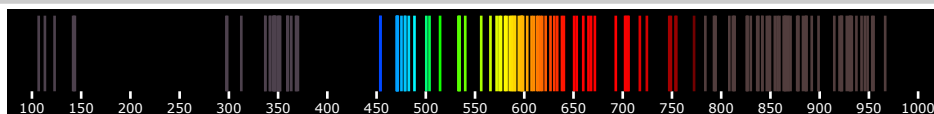
Use this command to set the global style of all the subsequent drawn spectra:

```
\pgfspectra[element=Ne] (before defining the global style)
```

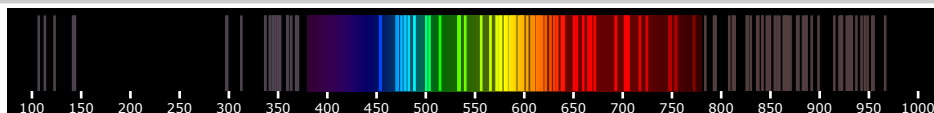


```
\pgfspectraStyle[width=.75\textwidth,axis,begin=100,end=1000,axis step=50]
```

```
\pgfspectra[element=Ne] (after defining the global style)
```

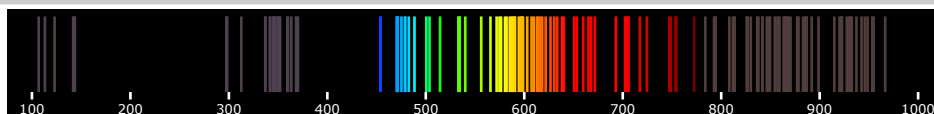


```
\pgfspectra[element=Ne,back=visible40,gamma=.6] (adding other options)
```



Note you can locally override the settings defined in the global style:

```
\pgfspectra[element=Ne,axis step=100] (overriding a global option)
```

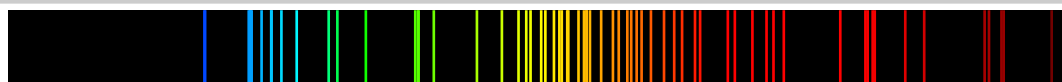


► Utilization of `\pgfspectraStyleReset`

Used to reset all the options of the spectra to their default values:

```
\pgfspectraStyleReset
```

```
\pgfspectra[element=Ne]
```



► Utilization of `\pgfspectrashade[<h|v>](start,end){name}`

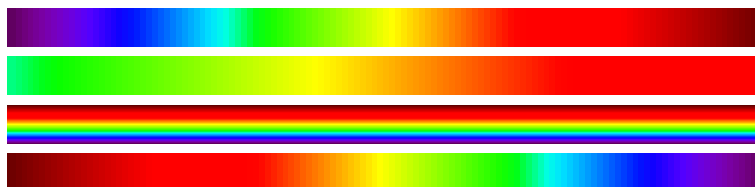
This command builds and makes available a **h**orizontal or a **v**ertical shading, between the 'start' and 'end' wavelengths (in nanometres), to use in TikZ pictures with the provided 'name'.

Note that, in this command, the 'start' wavelength needs to be smaller than the 'end' wavelength and is in the visible region: $\lambda_{\text{start}} < \lambda_{\text{end}}$ and $380 \leq \lambda \leq 780$.

The optional parameter takes the value **h** or **v** and has the default value of **h**.

```
\pgfspectrashade(380,780){myShadeA}
\pgfspectrashade(500,700){myShadeB}
\pgfspectrashade[v](380,780){myShadeC}
```

```
\tikz{\fill[shading=myShadeA] (0,0) rectangle (10,.5);}
\\ [3pt]\tikz{\fill[shading=myShadeB] (0,0) rectangle (10,.5);}
\\ [3pt]\tikz{\fill[shading=myShadeC] (0,0) rectangle (10,.5);}
\\ [3pt]\tikz{\fill[shading=myShadeA,shading angle=180] (0,0) rectangle (10,.5);}
```



► Utilization of `\usepgfspectralibrary{name(s)}`

Use this command to load pgf-spectra libraries anywhere in the document. The `name(s)` is a single library `name` or a comma separated list of library `names`.

As of now, available libraries are `data`, `pgfplots`, `tempercolor` and `rainbow`.

```
% Loading data library
\usepgfspectralibrary{data}
% Loading data and tempercolor libraries
\usepgfspectralibrary{data,tempercolor}
```

The options for `\pgfspectra`

For the commands `\pgfspectra` and `\pgfspectraStyle` there are a set of options available to draw the spectrum as described below.

The list of options is of the form `'key'` or `'key=value'` separated by commas.

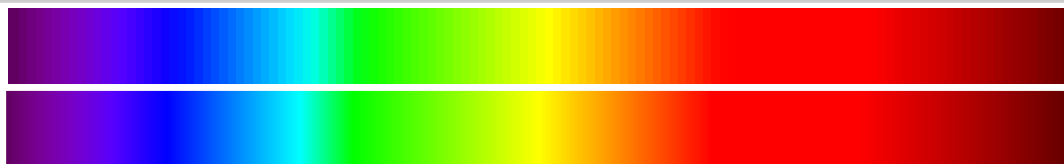
use visible shading

default: *true*

The visible region of the spectra is drawn using a *TikZ* shading instead of line by line, resulting in a faster drawing of that region. When set to `'false'` the visible region is drawn line by line: this value could be useful for printers that tend to be problematic when printing the shadings.

(new in v2.1.0)

```
\pgfspectra
\\ \pgfspectra[use visible shading=false]
```



width

default: *0.9\textwidth*

Sets the width of the spectrum.

```
\pgfspectra[width=10cm]
```



height

default: *1cm*

Sets the height of the spectrum.

```
\pgfspectra[height=40pt]
```



element

default: *NONE*

A single chemical symbol of an element or a list of chemical symbols.

```
\pgfspectra[element=H]
```



```
\pgfspectra[element={H,He}]
```



charge

default: 0

The charge of the *particle* to draw the spectrum. Use 'all' to get all available lines for the element, i.e, for the atom and all the positive ions that exist in the database. For LSE data: a value between 0 and 4; all other values are processed as 'all'. For NIST data: 0 or 1. All other values are processed as 'all'.

```
\pgfspectra[element=He]
```



```
\pgfspectra[element=He,charge=1]
```



```
\pgfspectra[element=He,charge=2]
```

Element "He" with charge "2" have no lines to display.

```
\pgfspectra[element=He,charge=all]
```

**Imin**

default: 0

The minimum intensity of the lines to put in the spectrum. Value from 0 to 1.

```
\pgfspectra[element=He,Imin=.5]
```



```
\pgfspectra[element=He,Imin=.05]
```

**relative intensity**default: *false*

Draws the lines respecting the intensity of the observed spectrum.

```
\pgfspectra[element=He,relative intensity]
```

**relative intensity threshold**

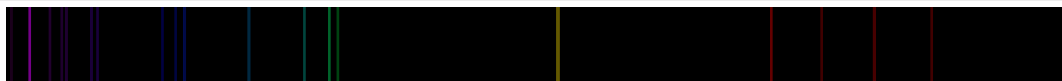
default: 0.25

Sets the minimum intensity for the lines in the spectrum when using relative intensities. When set to 0.25 a line with real intensity 0 will have a spectral intensity of 0.25 and a line with intensity equal to the max intensity observed in that spectrum will have an intensity in the computed spectrum of 1, assuming of course, an overall intensity in the range between 0 and 1.

```
\pgfspectra[element=He,relative intensity,relative intensity threshold=0]
```



```
\pgfspectra[element=He,relative intensity,relative intensity threshold=.25]
```



```
\pgfspectra[element=He,relative intensity,relative intensity threshold=.5]
```



```
\pgfspectra[element=He,relative intensity,relative intensity threshold=.75]
```



```
\pgfspectra[element=He,relative intensity,relative intensity threshold=1]
```



In fact, setting the relative intensity threshold to 1 is equivalent to the spectrum without relative intensities:

```
\pgfspectra[element=He]
```



line intensity

default: 100

Draws all the lines with the specified intensity between 0 and 100 (as a percentage of the maximum intensity).

```
\pgfspectra[element=He,line intensity=0]
```



```
\pgfspectra[element=He,line intensity=50]
```



```
\pgfspectra[element=He,line intensity=100]
```



```
\pgfspectra[element=He]
```



gamma

default: 0.8

Gamma color correction: any positive value.

`\pgfspectra[gamma=.1]``\pgfspectra[gamma=.8]``\pgfspectra[gamma=1]``\pgfspectra[gamma=2]``\pgfspectra[gamma=5]``\pgfspectra[gamma=10]`**brightness**

default: 1

Brightness color correction as in the CMYK color model. Value between 0 and 1. Zero stands for black and one for the maximum bright. *This option only works for the continuous component of the spectra, to change the "brightness" of spectral lines use the option 'line intensity'.*

`\pgfspectra[brightness=.1]``\pgfspectra[brightness=.5]``\pgfspectra[brightness=1]`

backdefault: *black*

Sets the background color of the spectrum. Only useful when there are spectral lines. Some shorthand are defined to put the visible region in the background: `'visible5'`, `'visible10'`, `'visible15'`, ... , `'visible100'`.

Note: this labels combined with the `'brightness'` option makes it possible to achieve other values on the background, since the visible amount (5%,10%,...) is multiplied by the value of brightness.

```
\pgfspectra[element=He,back=white]
```



```
\pgfspectra[element=He,back=black!50]
```



```
\pgfspectra[element=He,back=visible50]
```



```
\pgfspectra[element=He,back=visible50,brightness=.26]
```

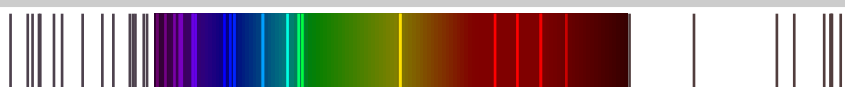
**backIRUV**default: *black*

Sets the background color, *only for the emission spectrum*, outside the visible region

($10nm \leq \lambda < 380nm$ and $780nm < \lambda \leq 4000nm$)

(new in v2.0.0)

```
\pgfspectra[element=He,back=visible50,begin=100,end=1000,backIRUV=white]
```

**IRcolor**default: *rgb(0.3157,0.2373,0.2373)*

Sets the color for emission lines and for background in absorption spectrum in the IR region

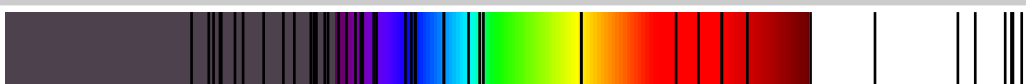
($780nm < \lambda \leq 4000nm$)

(new in v2.0.0)

```
\pgfspectra[element=He,back=visible50,begin=100,end=1000,IRcolor=white]
```



```
\pgfspectra[element=He,begin=100,end=1000,IRcolor=white,absorption]
```



UVcolordefault: `rgb(0.3,0.2568,0.3)`Sets the color for emission lines and for background in absorption spectrum in the UV region ($10\text{nm} \leq \lambda < 380\text{nm}$)*(new in v2.0.0)*`\pgfspectra[element=He,back=visible50,begin=100,end=1000,UVcolor=white]``\pgfspectra[element=He,begin=100,end=1000,UVcolor=white,absorption]`**lines**default: `{}`A comma separated list of wavelengths and/or a range of wavelengths – *start to end* – in the interval $[10; 4000]\text{nm}$ *(Interval updated in v2.2.0; Syntax updated in v3.0.0)*`\pgfspectra[lines={400,500,550,700}]``\pgfspectra[lines={200,205,400,500,550,700,850,950,980},begin=100,end=1000]``\pgfspectra[lines={380 to 500}]``\pgfspectra[lines={420 to 500, 550, 600 to 630, 700, 720}]`**line width**default: `1pt`

The width of each individual line in the spectrum.

`\pgfspectra[line width=2pt]``\pgfspectra[line width=2pt,element=He]`

begin

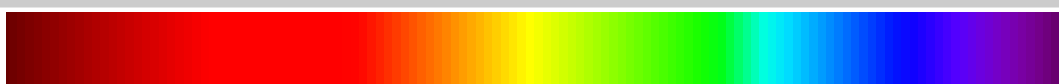
default: 380

The starting wavelength in nanometres of the spectrum ($10 \leq \lambda \leq 4000$).*(Interval updated in v2.2.0)*`\pgfspectra[begin=500]`**end**

default: 740

The finishing wavelength in nanometres of the spectrum ($10 \leq \lambda \leq 4000$).*(Interval updated in v2.2.0)*`\pgfspectra[end=500]`

Remark: *it's obviously possible to set 'begin' and 'end' at the same time and if desired change the order of the wavelengths.*

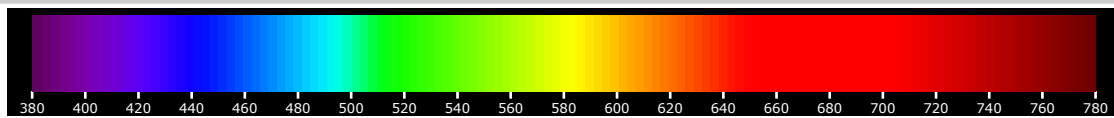
`\pgfspectra[begin=500,end=700]``\pgfspectra[begin=700,end=500]``\pgfspectra[begin=780,end=380]``\pgfspectra[begin=780,end=380,element=He]`**absorption**default: *false*

Draws the absorption spectrum instead of the emission one.

`\pgfspectra[element=H,absorption]``\pgfspectra[element={H,He},absorption]`

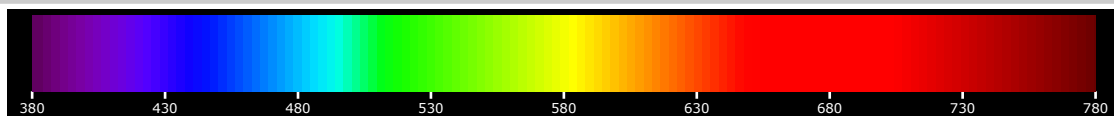
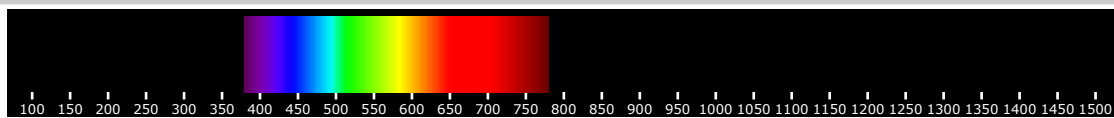
axisdefault: *false*

By default draws a nanometric axis below the spectrum.
In v2.1.0 is now possible to *change the unit* of the axis.

`\pgfspectra[axis]`**axis step**

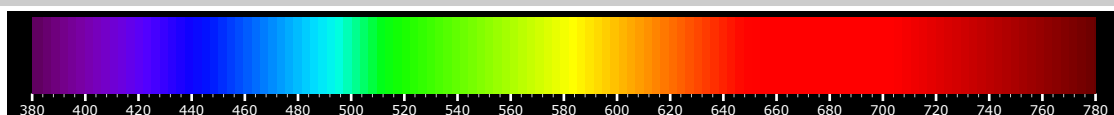
default: 20

The increment (in nanometres) to use in the axis scale.

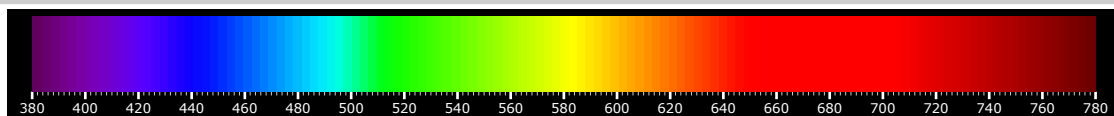
`\pgfspectra[axis,axis step=50]``\pgfspectra[axis,axis step=50,begin=100,end=1500]`**axis ticks**

default: 0

The number of minor ticks between two consecutive ticks in the axis.

(new in v2.0.0)`\pgfspectra[axis,axis ticks=4]`

Keep in mind, if you desire to divide two consecutive ticks into 10 equal parts set `'axis ticks=9'`:

`\pgfspectra[axis,axis ticks=9]`**axis unit**default: *nm*

Sets the unit to use in the displayed values of wavelengths in the axis below the spectrum.
Available units are:

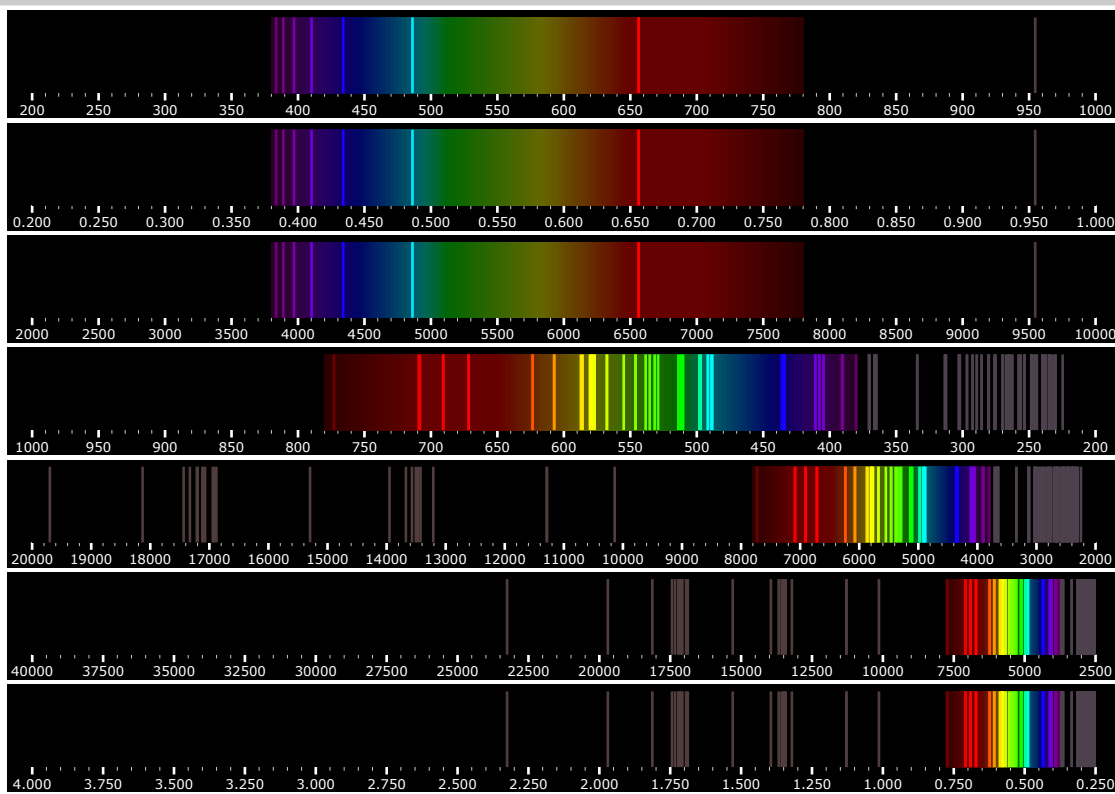
- nanometre (nm): `axis unit=nm`
- micrometre (μm): `axis unit=micron`
- angstrom (\AA): `axis unit=A`

(new in v2.1.0)

```

\pgfspectra[element=H,begin=200,end=1000,axis,axis step=50,axis
ticks=4,back=visible40]
\\ \pgfspectra[element=H,begin=200,end=1000,axis,axis step=50,axis
ticks=4,axis unit=micron,back=visible40]
\\ \pgfspectra[element=H,begin=200,end=1000,axis,axis step=50,axis
ticks=4,axis unit=A,back=visible40]
\\ \pgfspectra[element=Hg,begin=1000,end=200,axis,axis step=50,axis
ticks=4,back=visible40]
\\ \pgfspectra[element=Hg,begin=2000,end=200,axis,axis step=100,axis
ticks=4,axis unit=A,back=visible40]
\\ \pgfspectra[element=Hg,begin=4000,end=250,axis,axis step=250,axis
ticks=4,axis unit=A,back=visible40]
\\ \pgfspectra[element=Hg,begin=4000,end=250,axis,axis step=250,axis
ticks=4,axis unit=micron,back=visible40]

```

**axis unit precision**

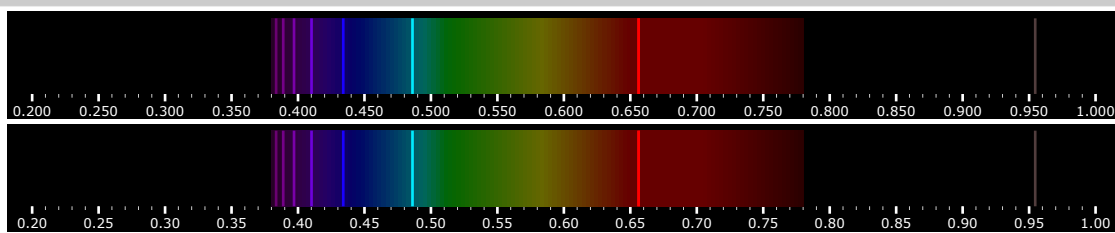
default: 3

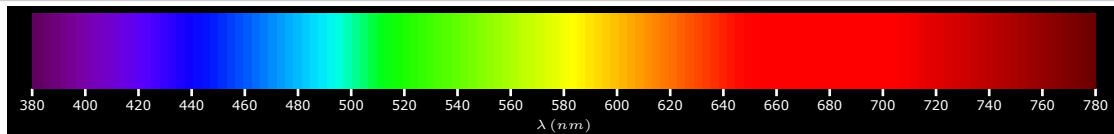
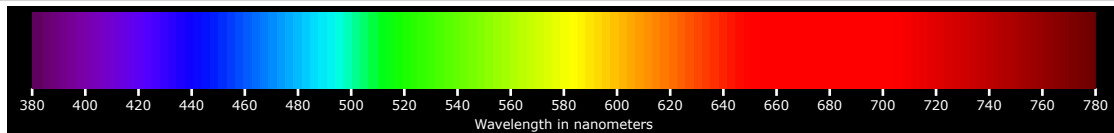
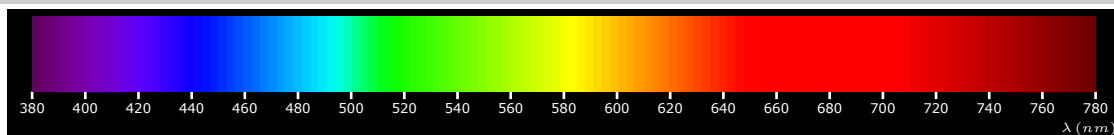
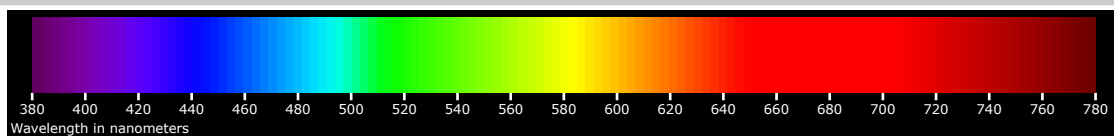
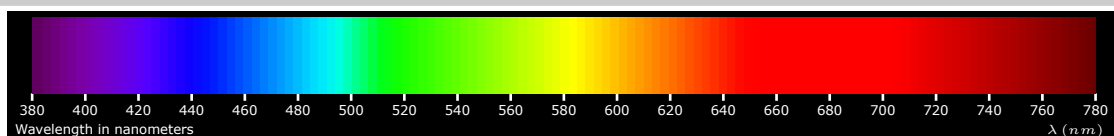
Sets the precision of the displayed wavelenghts in the axis below the spectrum. *(new in v2.1.0)*

```

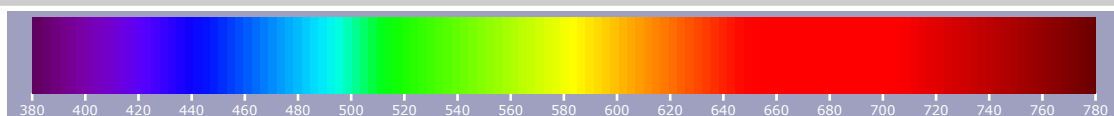
\pgfspectra[element=H,begin=200,end=1000,axis,axis step=50,axis ticks=4,axis
unit=micron,back=visible40]
\\ \pgfspectra[element=H,begin=200,end=1000,axis,axis step=50,axis
ticks=4,axis unit=micron,axis unit precision=2,back=visible40]

```



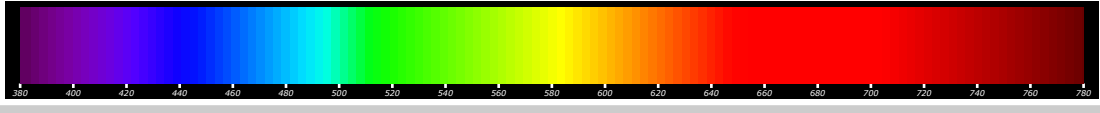
axis labeldefault: *false*Shows or hides (default) the axis label below it. When this key is set to *true*, the `axis` key is also set to *true*. *(new in v3.0.0)*`\pgfspectra[axis label=true]`**axis label text**default: `{}`If not empty, shows the axis label with the given text below it. In this case, the `axis label` key is set to *true*. *(new in v3.0.0)*`\pgfspectra[axis label text={Wavelength in nanometers}]`**axis label position**default: *center*Sets the position of the axis label to *left* or *center* or *right*. *(new in v3.0.0)*`\pgfspectra[axis label=true,axis label position=right]``\pgfspectra[axis label text={Wavelength in nanometers},axis label position=left]``\pgfspectra[axis label text={\hbox to .925\linewidth{\ Wavelength in nanometers\hfillλ(nm)$}},axis label position=left]`**axis color**default: *black*

The color of the axis.

`\pgfspectra[axis,axis color=red!50!green!50!blue!50]`

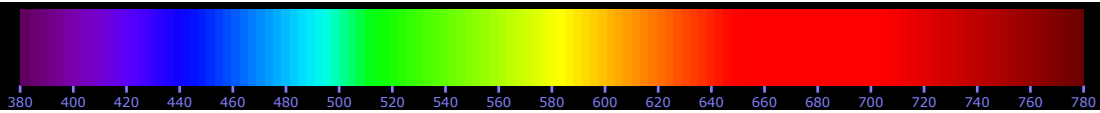
axis font default: `\tiny`
 The font specs to use in the axis.

```
\pgfspectra[axis,axis font=\fontsize{3}{3}\itshape\selectfont]
```



axis font color default: `white`
 The color of the font used in the axis.

```
\pgfspectra[axis,axis font color=blue!50!white]
```



label default: `false`
 Puts a label for the spectrum.

```
\pgfspectra[label]
```



```
\pgfspectra[label,element=He]
```



label position default: `west`
 Sets the position of the label according to:

north west	north	north east
west	<i>spectrum</i>	east
south west	south	south east

```
\pgfspectra[label,label position=east,element=He]
```



label font default: `\bfseries\small`
 The font specs for the label.

```
\pgfspectra[label,label font=\footnotesize\itshape,element=He]
```



label font colordefault: *black*

The color of the font used in the label.

```
\pgfspectra[label,label font color=blue!50!white,element=He]
```

He

**label before text**

default: {}

Inserts text before the value stored in the label: if chemical symbols were provided, the label has them stored, otherwise it is empty.

```
\pgfspectra[label,label before text=text\ ,element=He]
```

text He



Remark: The `_` is to insert a space between the text entered by user and the text stored in label.

label after text

default: {}

Inserts text after the value stored in the label: if chemical symbols were provided, the label has them stored, otherwise it is empty.

```
\pgfspectra[label,label after text=\ text,element=He]
```

He text

**redshift**

default: {}

Redshift (or blueshift) the spectral lines:

The redshift value (z) is *defined* as $1 + z = \lambda_{obs} / \lambda_E$ which leaves the observed wavelength to $\lambda_{obs} = (1 + z)\lambda_E$, given the emitted wavelength of the source (λ_E).

- Use `'redshift=<numeric value>'` to directly enter the redshift value
- or use `'redshift={D=<numeric value 1>/<numeric value 2>}'` to compute the Relativistic Doppler redshift with $\bar{v} = \text{<numeric value 1>}$ and $\theta = \text{<numeric value 2>}^\circ$.

The Relativistic Doppler redshift ($1 + z$) is calculated accordingly:

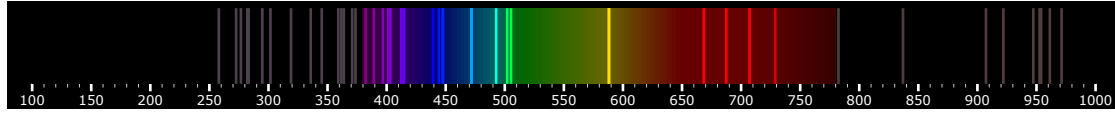
$$1 + z = \frac{1 + \bar{v} \cos \theta}{\sqrt{1 - \bar{v}^2}} \quad \bar{v} = \frac{v}{c}$$

where \bar{v} is the *normalized velocity* (in units of the speed of light in vacuum, c) of the emitter and θ is the angle between the direction of relative motion and the direction of emission in the observer's frame (zero angle is directly away from the observer). So, if the source of light is moving away from an observer, then redshift occurs ($z > 0$), but, if the source moves towards the observer, then blueshift occurs ($z < 0$).

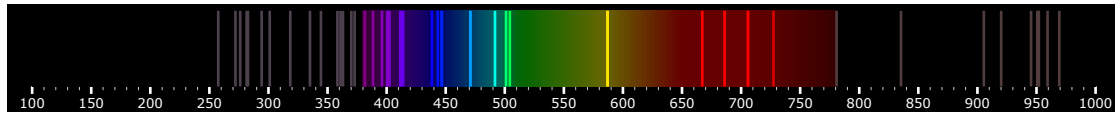


(new in v2.0.0)

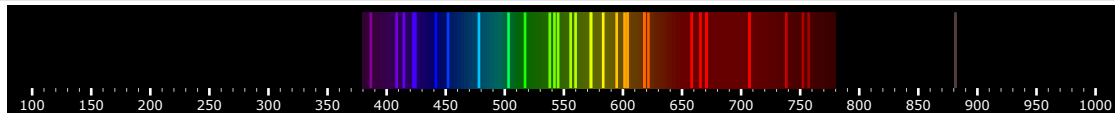
```
\pgfspectra[element=He,back=visible40,gamma=.6,axis,axis step=50,axis ticks=4,begin=100,end=1000,redshift={D=.001/0}]
```



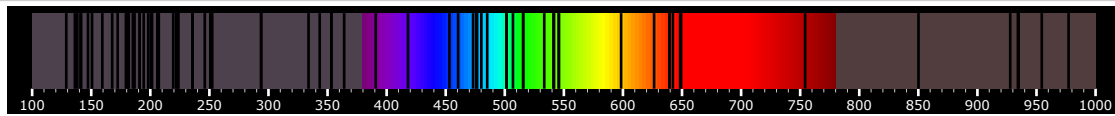
```
\pgfspectra[element=He,back=visible40,gamma=.6,axis,axis step=50,axis ticks=4,begin=100,end=1000,redshift={D=.001/180}]
```



```
\pgfspectra[element=He,back=visible40,gamma=.6,axis,axis step=50,axis ticks=4,begin=100,end=1000,redshift=.5]
```



```
\pgfspectra[element=He,absorption,gamma=.6,axis,axis step=50,axis ticks=4,begin=100,end=1000,redshift=-.5]
```



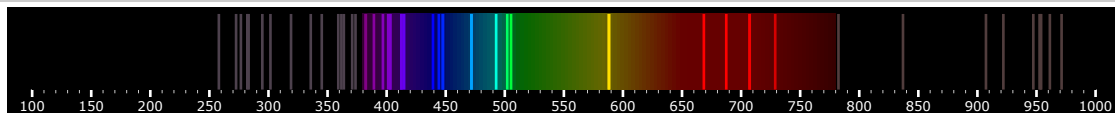
show redshift value

default: *false*

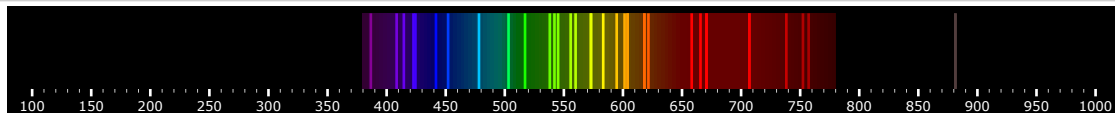
Writes the value of the redshift (left below the spectrum).

(*new in v2.0.0*)

```
\pgfspectra[element=He,back=visible40,gamma=.6,axis,axis step=50,axis ticks=4,begin=100,end=1000,redshift={D=.001/0},show redshift value]
```



```
\pgfspectra[element=He,back=visible40,gamma=.6,axis,axis step=50,axis ticks=4,begin=100,end=1000,redshift=.5,show redshift value]
```



Libraries

In this part the library packages are documented. They provide additional commands to extend the capabilities provided by this package out of the box. The libraries are not loaded by default since many users will not need them.

The available libraries are:

- **data** library:
 - provides the commands to display and/or export to a file the spectral lines of selected elements.
- **pgfplots** library:
 - provides the commands to create a plot shade or a plot map to use with PGFPLOTS.
- **tempercolor** library:
 - provides the command to convert a temperature (in Kelvin) into its correlated color.
- **rainbow** library:
 - provides the command to draw a rainbow.

► data Library

pgf-spectra Library **data**

USAGE: `\usepgfspectralibrary{data}`

This library implements a command that allows the user to extract the spectral lines data used by `\pgfspectra`. After that using one of the other two commands provided it is possible to display the data in a table or export it to a file.

The commands are:

- `\pgfspectradata[name of data set]{keys}`
- `\pgfspectratable[table options]{name(s) of data set(s)}`
- `\pgfspectrawrite[filename]{name(s) of data set(s)}`

► Utilization of `\pgfspectradata[name of data set]{keys}`

This command loads spectral lines data of the element or elements specified with the **element** key. Note that this key is exactly the same as the one used in `\pgfspectra`. This command also shares the **charge**, **Imin**, **redshift**, **begin** and **end** keys with `\pgfspectra`, but has its own **precision** and **unit** keys.

`\pgfspectradata` does not produce any *visible* output, but it does the *work* needed by `\pgfspectratable` and `\pgfspectrawrite` commands. The optional **name of data set** argument makes that *name* available for use with `\pgfspectratable` or `\pgfspectrawrite`.

See [utilization of `\pgfspectratable`](#) and [utilization of `\pgfspectrawrite`](#) for examples of the application of this command.

► The options for `\pgfspectradata`

name of data set

A name formed of valid letters.

default: `{}`

(new in v3.0.0)

```
\pgfspectradata[dataH]{element=H}
```

This command does not generate any visible output, but internally stores the following data:

```
H(+0):383.5,388.9,397.0,410.2,434.0,486.1,656.3;
```

precision

Sets the precision of the data internally stored.

default: `1`

(new in v3.0.0)

```
\pgfspectradata{element=H,precision=2}
```

This command does not generate any visible output, but internally stores the following data:

```
H(+0):383.54,388.90,397.01,410.17,434.05,486.13,656.27,656.29;
```

unit default: {}
 If empty, the `axis unit` key value of `\pgfspectra` is used, otherwise the `unit` key value is used.
 This key, like `axis unit` key, can have the values `nm` for nanometers, `micron` for micrometers or `A` for angstroms. *(new in v3.0.0)*

```
\pgfspectradata{element=H,unit=A}
```

This command does not generate any visible output, but internally stores the following data:

```
H(+0):3835,3889,3970,4102,4340,4861,6563;
```

```
\pgfspectraStyle[axis unit=micron,begin=100]
\pgfspectradata{element=H}
```

This command does not generate any visible output, but internally stores the following data:

```
H(+0):0.1026,0.1216,0.3835,0.3889,0.3970,0.4102,0.4340,0.4861,0.6563;
```

```
\pgfspectraStyleReset
```

► Utilization of `\pgfspectratable[table options]{name(s) of data set(s)}`

This command displays the loaded data in a table. Both arguments are optional.

When used without arguments, the last *unnamed* data loaded is used:

```
\pgfspectradata{element=Ne}% loads the spectral lines of neon
\\ \pgfspectratable% displays the table..
```

SPECTRAL LINES OF ELEMENT	
wavelength in <i>nm</i>	
Ne	453.8 454.0 470.4 470.9 471.0 471.2 471.5 475.3 478.9 479.0 482.7 488.5 500.5 503.8 514.5 533.1 534.1 534.3 540.1 556.3 565.7 571.9 574.8 576.4 580.4 582.0 585.2 587.3 588.2 590.2 590.6 594.5 596.5 597.5 597.6 598.8 603.0 607.4 609.6 612.8 614.3 616.4 618.2 621.7 626.6 630.5 633.4 638.3 640.2 650.7 653.3 659.9 665.2 667.8 671.7 692.9 702.4 703.2 705.1 705.9 717.4 724.5 747.2 748.9 753.6 754.4 772.5

The `table options` keys are `title`, `back color`, `data back color`, `text color`, `width` and `elements column width`.

The optional `name(s) of data set(s)` argument is either a single name or a comma separated list of names previously defined via `\pgfspectradata`.

► The options for `\pgfspectratable`

title default: `<REDSHIFTED >SPECTRAL LINES OF ELEMENT(S)`
`<\\ redshift value>`
`\\ wavelength in unit</relative intensity>`

Sets the title in the table header. *(new in v3.0.0)*

```
\pgfspectradata{element=H}
\\ \pgfspectratable
```

SPECTRAL LINES OF ELEMENT
wavelength in *nm*

H 383.5 388.9 397.0 410.2 434.0 486.1 656.3

```
\pgfspectradata{element={H,He}}
\\ \pgfspectratable
```

SPECTRAL LINES OF ELEMENTS
wavelength in *nm*

H 383.5 388.9 397.0 410.2 434.0 486.1 656.3

He 382.0 388.9 396.5 400.9 402.6 412.1 414.4 438.8 443.8 447.1 447.2 471.3
492.2 501.6 504.8 587.6 667.8 686.7 706.5 706.6 728.1

```
\pgfspectradata[Hrs]{element=H,redshift={D=0.01/180}}
\\ \pgfspectratable{Hrs}
```

REDSHIFTED SPECTRAL LINES OF ELEMENT
Relativistic Doppler redshift: $z=-0.01$ ($v=0.01c$; $\theta=180^\circ$)
wavelength in *nm*

H 379.7 385.0 393.0 406.1 429.7 481.2 649.7

```
\pgfspectradata{element=H,relative intensity}
\\ \pgfspectratable
```

SPECTRAL LINES OF ELEMENT
wavelength in *nm*/relative intensity

H 383.5/0.27 388.9/0.28 397.0/0.28 410.2/0.31 434.0/0.38 486.1/0.58
656.3/0.75 656.3/1.00

```
% this example requires the LSE data loaded
% use \usepackage[LSE]{pgf-spectra} in the document preamble
```

```
\pgfspectradata[dataOI]{element=O}
\\ \pgfspectradata[dataOII]{element=O,charge=1}
\\ \pgfspectradata[dataOIII]{element=O,charge=2}
\\ \pgfspectradata[dataOIV]{element=O,charge=3}
\\ \pgfspectratable[title={Oxygen spectral lines in the visible region\\ Wavelength in
nanometers\\ $\lambda$, (nm)$}]{dataOI}
\\ \pgfspectratable[title={Spectral lines of oxygen and its ions in the visible region\\
Wavelength in nanometers}]{dataOI,dataOII,dataOIII,dataOIV}
```

Oxygen spectral lines in the visible region
Wavelength in nanometers
 λ (*nm*)

O 382.3 394.7 394.8 395.5 423.3 436.8 532.9 533.0 533.1 543.5 543.6 543.7
557.7 595.8 595.9 599.5 604.6 610.6 615.6 615.7 615.8 625.7 626.2 636.6
637.4 645.4 645.6 660.5 665.4 700.2 715.7 725.4 725.5 747.6 747.7 747.9
748.1 770.7 777.2 777.4 777.5

Spectral lines of oxygen and its ions in the visible region												
Wavelength in nanometers												
O	382.3	394.7	394.8	395.5	423.3	436.8	532.9	533.0	533.1	543.5	543.6	543.7
	557.7	595.8	595.9	599.5	604.6	610.6	615.6	615.7	615.8	625.7	626.2	636.6
	637.4	645.4	645.6	660.5	665.4	700.2	715.7	725.4	725.5	747.6	747.7	747.9
	748.1	770.7	777.2	777.4	777.5							
O ⁺	380.3	391.2	391.9	395.4	397.3	398.2	407.0	407.2	407.6	408.4	408.7	408.9
	409.7	410.5	411.9	413.3	414.6	415.3	418.5	419.0	425.4	427.5	430.4	431.7
	433.7	434.6	434.9	436.7	439.6	441.5	441.7	444.8	445.2	446.5	446.6	446.8
	446.9	459.1	459.6	460.9	463.9	464.2	464.9	465.1	466.2	467.6	469.9	470.5
	492.5	494.3										
O ²⁺	396.2	559.2										
O ³⁺	no lines!											

back colordefault: *black!10*Sets the background color of the table header, of the *element(s)* column, of the table border lines and of the lines between rows.*(new in v3.0.0)*`\pgfspectratable[back color=blue!10]`

SPECTRAL LINES OF ELEMENT												
wavelength in <i>nm</i> /relative intensity												
H	383.5/0.27	388.9/0.28	397.0/0.28	410.2/0.31	434.0/0.38	486.1/0.58						
	656.3/0.75	656.3/1.00										

data back colordefault: *white*

Sets the background color of each data row.

(new in v3.0.0)`\pgfspectratable[data back color=black!5]`

SPECTRAL LINES OF ELEMENT												
wavelength in <i>nm</i> /relative intensity												
H	383.5/0.27	388.9/0.28	397.0/0.28	410.2/0.31	434.0/0.38	486.1/0.58						
	656.3/0.75	656.3/1.00										

text colordefault: *black*

Sets the color of all text in the table.

(new in v3.0.0)`\pgfspectratable[text color=blue!50!black]`

SPECTRAL LINES OF ELEMENT												
wavelength in <i>nm</i> /relative intensity												
H	383.5/0.27	388.9/0.28	397.0/0.28	410.2/0.31	434.0/0.38	486.1/0.58						
	656.3/0.75	656.3/1.00										

width

Sets the total width of the table.

default: `\linewidth`*(new in v3.0.0)*`\begin{center}\pgfspectratable[width=.75\linewidth]\end{center}`

SPECTRAL LINES OF ELEMENT	
wavelength in <i>nm</i> /relative intensity	
H	383.5/0.27 388.9/0.28 397.0/0.28 410.2/0.31 434.0/0.38 486.1/0.58 656.3/0.75 656.3/1.00

elements column widthSets the width of the *element(s)* column.default: `2.5em`*(new in v3.0.0)*`\pgfspectratable[elements column width=.1\linewidth]`

SPECTRAL LINES OF ELEMENT	
wavelength in <i>nm</i> /relative intensity	
H	383.5/0.27 388.9/0.28 397.0/0.28 410.2/0.31 434.0/0.38 486.1/0.58 656.3/0.75 656.3/1.00

 Utilization of `\pgfspectrawrite[filename]{name(s) of data set(s)}`

This command **writes the provided data set(s)** to a file making the data available for use at any time, and in any context.

Both arguments are optional:

- if no **filename** is given, the name `spectradata.tex` is used, otherwise the *filename* is formed by any character recognized by the operating system (this means, for instance, that `_` character can be used), followed by an optional extension – `tex`, `csv`, `txt` or `dat`. If the extension is not present, the `tex` extension is appended to the given name.
- if no **name(s) of data set(s)** is given, the last *unnamed* data loaded will be used.

For example, using just `\pgfspectrawrite` writes, in the current working directory, the file `spectradata.tex` with the following contents, which were the last *unnamed* data loaded:

The contents of `spectradata.tex`

```
% Generated by pgf-spectra @ 16 June 2025
% Spectral lines of element(s) and/or their ion(s) from 380 to 780 nanometers
% Data structure:
% chemical symbol,charge,lines/relative intensity
H,0,383.5/0.27,388.9/0.28,397.0/0.28,410.2/0.31,434.0/0.38,486.1/0.58,656.3/0.75,656.3/1.00
```

`\pgfspectrawrite[myfile.csv]` writes, in the current working directory, the file `myfile.csv` with the following contents, which were the last *unnamed* data loaded:

The contents of `myfile.csv`

```
# Generated by pgf-spectra @ 16 June 2025
# Spectral lines of element(s) and/or their ion(s) from 380 to 780 nanometers
# Data structure:
chemical symbol,charge,lines/relative intensity
H,0,383.5/0.27,388.9/0.28,397.0/0.28,410.2/0.31,434.0/0.38,486.1/0.58,656.3/0.75,656.3/1.00
```


`\pgfspectrawrite[oxygen.dat]{dataOI,dataOII,dataOIII,dataOIV}` writes, in the current working directory, the file `oxygen.dat` with the following contents:

The contents of **oxygen.dat**

```
% Generated by pgf-spectra @ 16 June 2025
% Spectral lines of element(s) and/or their ion(s) from 380 to 780 nanometers
% Data structure:
% chemical symbol,charge,lines
O,0,382.3,394.7,394.8,395.5,423.3,436.8,532.9,533.0,533.1,543.5,543.6,543.7,557.7,595.8,595.9,599.5,604.6,610.6,615.6,615.7,
615.8,625.7,626.2,636.6,637.4,645.4,645.6,660.5,665.4,700.2,715.7,725.4,725.5,747.6,747.7,747.9,748.1,770.7,777.2,777.4,777.5
O,+1,380.3,391.2,391.9,395.4,397.3,398.2,407.0,407.2,407.6,408.4,408.7,408.9,409.7,410.5,411.9,413.3,414.6,415.3,418.5,419.0,
425.4,427.5,430.4,431.7,433.7,434.6,434.9,436.7,439.6,441.5,441.7,444.8,445.2,446.5,446.6,446.8,446.9,459.1,459.6,460.9,463.9,
464.2,464.9,465.1,466.2,467.6,469.9,470.5,492.5,494.3
O,+2,396.2,559.2
O,+3,no lines!
```

`\pgfspectrawrite[Hredshift.txt]{Hrs}` writes, in the current working directory, the file `Hredshift.txt` with the following contents:

The contents of **Hredshift.txt**

```
% Generated by pgf-spectra @ 16 June 2025
% Redshifted spectral lines of element(s) and/or their ion(s) from 380 to 780 nanometers
% Relativistic Doppler redshift: z=-0.01 (v=0.01c ; theta=180 degrees)
% Data structure:
% chemical symbol,charge,lines
H,0,379.7,385.0,393.0,406.1,429.7,481.2,649.7
```

► pgfplots Library

pgf-spectra Library `pgfplots`

USAGE: `\usepgfspectralibrary{pgfplots}`

This library implements two commands that define the necessary *material* needed to be used with the PGFLOTS package, referring to the colors provided by `\pgfspectra`. The commands are:

- `\pgfspectraplotmap[<|h>]{name}`
- `\pgfspectraplotshade[options]{name}`

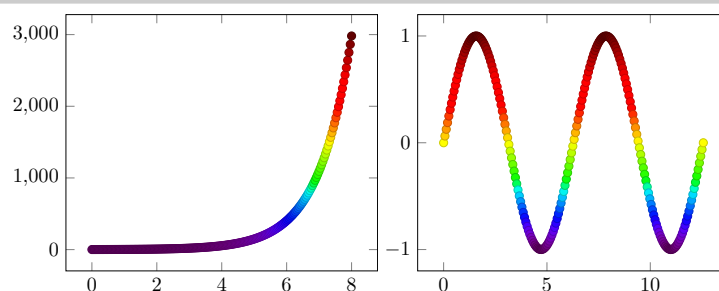
The commands – `\pgfspectrashade`, `\pgfspectraplotshade` and `\pgfspectraplotmap` – were inspired in the [TeX - LaTeX Stack Exchange](#) questions, [Filling optical spectrum curve with color gradient](#) and [How to create a electromagnetic spectrum using pgfplots package \(together with colormaps\)](#), which were referred by Stefan Pinnow, as examples, in a features request for the `pgf-spectra` package.

► Utilization of `\pgfspectraplotmap[<|h>]{name}`

This command builds and makes available a **low** or **high resolution** color map in the wavelength range from 380 nm to 780 nm to use in PGFLOTS with the provided '`name`':

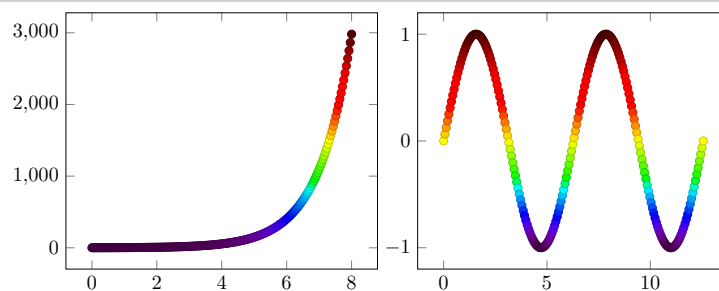
`\pgfspectraplotmap{myColorMap}`% low resolution (default value for optional parameter)

```
\begin{tikzpicture}
\begin{axis}[colormap name=myColorMap]
\addplot+[scatter,only marks,domain=0:8,samples=200] {exp(x)};
\end{axis}
\end{tikzpicture}
\begin{tikzpicture}
\begin{axis}[colormap name=myColorMap]
\addplot+[scatter,only marks,domain=0:4*pi,samples=200] {sin(deg(x))};
\end{axis}
\end{tikzpicture}
```



```
\pgfspectraplotmap[h]{myColorMapH}% high resolution ('h' value in optional parameter)
```

```
\begin{tikzpicture}
\begin{axis}[colormap name=myColorMapH]
\addplot+[scatter,only marks,domain=0:8,samples=200] {exp(x)};
\end{axis}
\end{tikzpicture}
\begin{tikzpicture}
\begin{axis}[colormap name=myColorMapH]
\addplot+[scatter,only marks,domain=0:4*pi,samples=200] {sin(deg(x))};
\end{axis}
\end{tikzpicture}
```



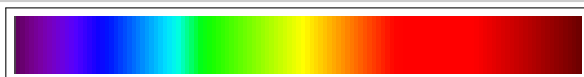
Actually using high or low resolution produces the same effect on plot. The difference resides on the number of colors available to the 'color of colormap' feature. For more information see [Using \pgfspectraplotshade and \pgfspectraplotmap with PGFPLOTS](#).

► Utilization of `\pgfspectraplotshade[options]{name}`

This command, without any options, builds and makes available a shading in the wavelength range from 380 nm to 780 nm to use in PGFPLOTS with the provided 'name'.

```
\pgfspectraplotshade{myPlotShadeA}
```

```
\fbox{\tikz{\fill[shading=myPlotShadeA] (0,0) rectangle (7.5,.75);}}
```



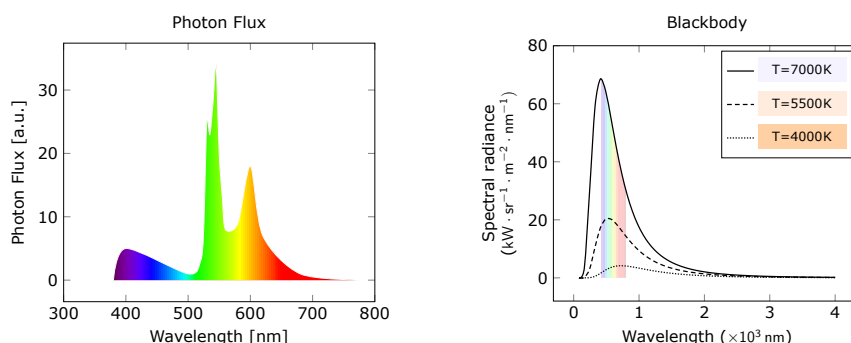
The optional argument can receive specific options for the shade or `\pgfspectra` options:

```
\pgfspectraplotshade[shade begin=0,shade end=4000,IRcolor=white,UVcolor=white,
shade opacity=.2,gamma=.6]{myPlotShadeB}
```

```
\fbox{\tikz{\fill[shading=myPlotShadeB] (0,0) rectangle (7.5,.75);}}
```



The specific options available are `shade end`, `shade opacity` and `shade opacity color`. See [The options for \pgfspectraplotshade](#) for detailed information on using these options. When used in PGFPLOTS it's possible to do plots like:



For these plots and other ones see [Using \pgfspectraplotshade and \pgfspectraplotmap with PGFPLOTS](#).

► The options for `\pgfspectraplotshade`

The command `\pgfspectraplotshade` creates a shade to use with the `\addplot` command provided by the PGFPLOTS package. The shade starts at `shade begin` and finishes at `shade end`. The shading could be adjusted using the following options:

- `shade begin`
- `shade end`
- `shade opacity`
- `shade opacity color`
- `logarithmic`

shade begin

default: 380

This value determines the start wavelength of the computed shading. It should be set equal to the minimum value of the plotted data. It could be different from the inferior limit of the domain provided to the plot (see the [PGFPLOTS package documentation](#) for more information). The range of accepted values goes from 0 nm to `(shade end-1)`. *(new in v2.1.1)*

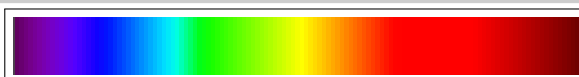
shade end

default: 780

This value determines the end wavelength of the computed shading. It should be set equal to the maximum value of the plotted data and could be different from the superior limit of the domain provided to the plot. The range of accepted values goes from `(shade begin+1)` to 16000 nm. *(new in v2.1.0)*

```
\pgfspectraplotshade{shadeDefault}
```

```
\fbox{\tikz{\fill[shading=shadeDefault] (0,0) rectangle (7.5,.75);}}
```



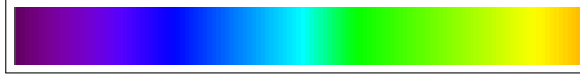
```
\pgfspectraplotshade[shade begin=600]{shadeBegin600}
```

```
\fbox{\tikz{\fill[shading=shadeBegin600] (0,0) rectangle (7.5,.75);}}
```



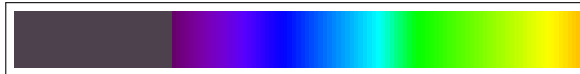
```
\pgfspectraplotshade[shade end=600]{shadeEnd600}
```

```
\fbox{\tikz{\fill[shading=shadeEnd600] (0,0) rectangle (7.5,.75);}}
```



```
\pgfspectraplotshade[shade begin=300,shade end=600]{shade300to600}
```

```
\fbox{\tikz{\fill[shading=shade300to600] (0,0) rectangle (7.5,.75);}}
```



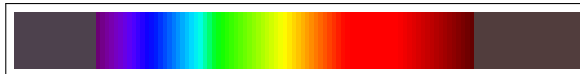
```
\pgfspectraplotshade[shade begin=600,shade end=900]{shade600to900}
```

```
\fbox{\tikz{\fill[shading=shade600to900] (0,0) rectangle (7.5,.75);}}
```



```
\pgfspectraplotshade[shade begin=300,shade end=900]{shade300to900}
```

```
\fbox{\tikz{\fill[shading=shade300to900] (0,0) rectangle (7.5,.75);}}
```



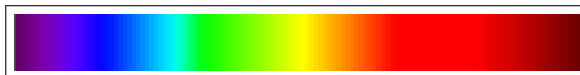
shade opacity

default: 1

The opacity of the computed shade. '0' stands for 0% and the shading is totally transparent; '1' stands for 100% and the shading isn't transparent at all. *(new in v2.1.0)*

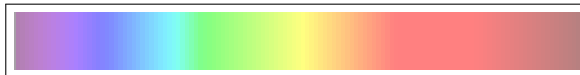
```
\pgfspectraplotshade{shadeDefault}
```

```
\fbox{\tikz{\fill[shading=shadeDefault] (0,0) rectangle (7.5,.75);}}
```



```
\pgfspectraplotshade[shade opacity=.5]{shadeOpacity50}
```

```
\fbox{\tikz{\fill[shading=shadeOpacity50] (0,0) rectangle (7.5,.75);}}
```



```
\pgfspectraplotshade[shade opacity=0]{shadeOpacity0}
```

```
\fbox{\tikz{\fill[shading=shadeOpacity0] (0,0) rectangle (7.5,.75);}}
```



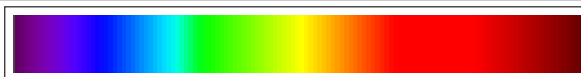
shade opacity colordefault: *white*

The background color of the computed shading. Only visible when **shade opacity** is lesser than 1.

(new in v2.1.0)

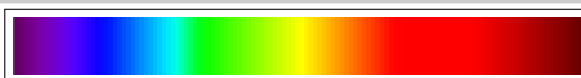
```
\pgfspectraplotshade{shadeDefault}
```

```
\fbox{\tikz{\fill[shading=shadeDefault] (0,0) rectangle (7.5,.75);}}
```



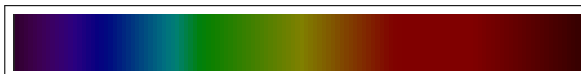
```
\pgfspectraplotshade[shade opacity color=black]{shadeOpacityBlack}
```

```
\fbox{\tikz{\fill[shading=shadeOpacityBlack] (0,0) rectangle (7.5,.75);}}
```



```
\pgfspectraplotshade[shade opacity color=black, shade opacity=.5]{shadeOpacityBlack50}
```

```
\fbox{\tikz{\fill[shading=shadeOpacityBlack50] (0,0) rectangle (7.5,.75);}}
```

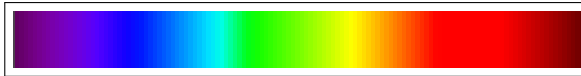
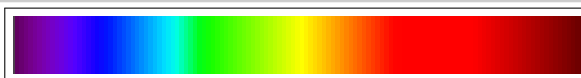
**logarithmic**default: *false*

When set to true the shading is build in a logarithmic scale. The smaller wavelengths are *wided* and the longer ones are *shortened* in the displayed region.

(new in v2.1.1)

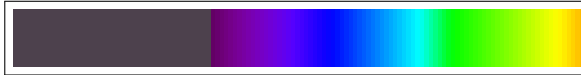
```
\pgfspectraplotshade[logarithmic]{logshadeDefault}
```

```
\fbox{\tikz{\fill[shading=shadeDefault] (0,0) rectangle (7.5,.75);}}
\\ \fbox{\tikz{\fill[shading=logshadeDefault] (0,0) rectangle (7.5,.75);}}
```



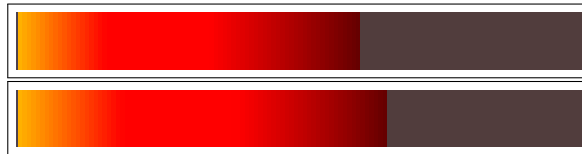
```
\pgfspectraplotshade[logarithmic,shade begin=300,shade end=600]{logshade300to600}
```

```
\fbox{\tikz{\fill[shading=shade300to600] (0,0) rectangle (7.5,.75);}}
\\ \fbox{\tikz{\fill[shading=logshade300to600] (0,0) rectangle (7.5,.75);}}
```



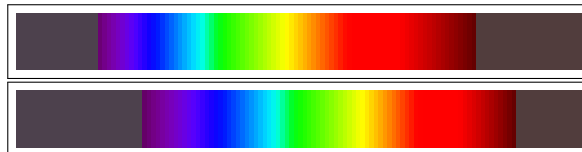
```
\pgfspectraplotshade[logarithmic,shade begin=600,shade end=900]{logshade600to900}
```

```
\fbox{\tikz{\fill[shading=shade600to900] (0,0) rectangle (7.5,.75);}}
\\ \fbox{\tikz{\fill[shading=logshade600to900] (0,0) rectangle (7.5,.75);}}
```



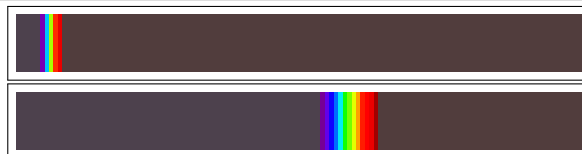
```
\pgfspectraplotshade[logarithmic,shade begin=300,shade
end=900]{logshade300to900}
```

```
\fbox{\tikz{\fill[shading=shade300to900] (0,0) rectangle (7.5,.75);}}
\\ \fbox{\tikz{\fill[shading=logshade300to900] (0,0) rectangle (7.5,.75);}}
```



```
\pgfspectraplotshade[logarithmic,shade begin=10,shade
end=10000]{logshade10to10000}
```

```
\fbox{\tikz{\fill[shading=shade10to10000] (0,0) rectangle (7.5,.75);}}
\\ \fbox{\tikz{\fill[shading=logshade300to900] (0,0) rectangle (7.5,.75);}}
```



► tempercolor Library

pgf-spectra Library **tempercolor**

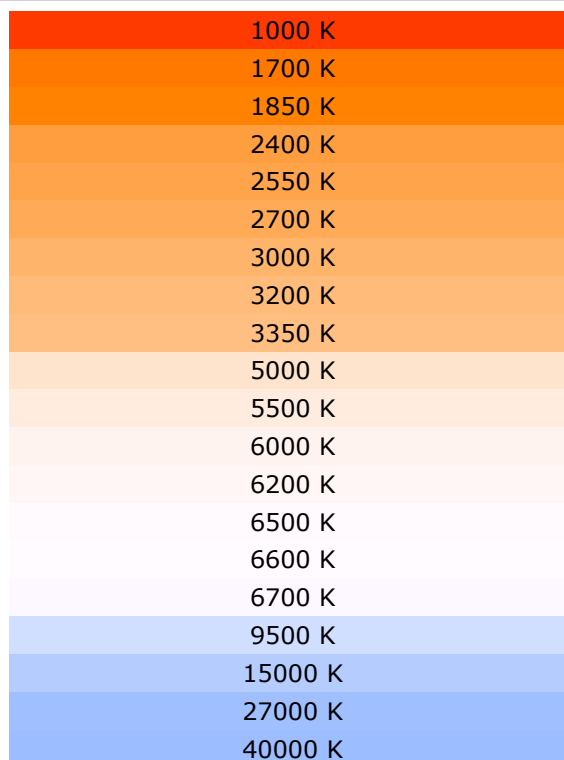
USAGE: `\usepgfspectralibrary{tempercolor}`

This library provides the command `\tempercolor{Kelvin}` that uses the CIE 1964 10-degree color matching function to convert a given temperature, in Kelvin ($1000\text{ K} \leq T \leq 40000\text{ K}$), to the respective correlated color. For more information on the implemented algorithm, please see:

- <https://tannerhelland.com/2012/09/18/convert-temperature-rgb-algorithm-code.html>
 - <https://www.zombieprototypes.com/?p=210>
 - <https://github.com/neilbartlett/color-temperature>
-

► Utilization of `\tempercolor{Kelvin}`

```
\foreach \T in {1000,1700,1850,2400,2550,2700,3000,3200,%
  3350,5000,5500,6000,6200,6500,6600,6700,9500,15000,%
  27000,40000}
{\tempercolor{\T}\tikz{
  \fill[tempercolor,font=\small] (0,0) rectangle (7.5,.5) %
  node[midway] {\color{black}\T K};}\ \ [-1pt]%
}%
```



► rainbow Library

pgf-spectra Library **rainbow**

USAGE: `\usepgfspectralibrary{rainbow}`

This library provides the command `\pgfspectrarainbow[tikz options](rainbow options){radius}` that allows you to draw a rainbow with or without options.

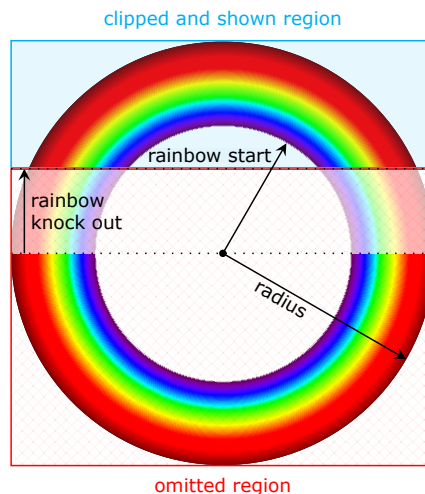
► Utilization of `\pgfspectrarainbow[tikz options](rainbow options){radius}`

Without options this command draws a rainbow with the specified radius:

```
\pgfspectrarainbow{2cm}
```



The rainbow is designed with the following schema resulting in a clipped and shown region:



The options available could be specific options for the rainbow or *common* TikZ options:

- the rainbow specific options:
 - rainbow start
 - rainbow knock out
 - rainbow fade
 - rainbow transparency
 - rainbow background
- the TikZ options: any option known by TikZ and/or TikZ libraries.

 ▶ The options for `\pgfspectrarainbow`

For the command `\pgfspectrarainbow` there are a set of options that control the rainbow drawn.

The specific rainbow options are:

- `rainbow start`
- `rainbow knock out`
- `rainbow fade`
- `rainbow transparency`
- `rainbow background`

Some *TikZ* keys that affect the rainbow are:

- `'color'`
- `opacity`
- `scope fading`

The default rainbow drawn is:

```
\pgfspectrarainbow{2cm}
```


rainbow start

 default: `.6`

The fraction from which the rainbow colors begin, relative to the center of a circle with radius 1. This value should be in the interval $[0,1]$. *(new in v2.1.0)*

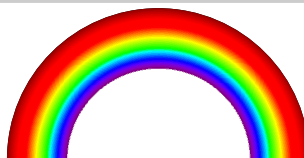
```
\pgfspectrarainbow(rainbow start=.8){2cm}% the rainbow colors starts at 1.6cm
\hspace{1cm}%
\pgfspectrarainbow(rainbow start=.4){2cm}% the rainbow colors starts at .8cm
```


rainbow knock out

 default: `.4`

The relative distance from the half-circle base to perform the clip. This value should be in the interval $[-1,1]$. *(new in v2.1.0)*

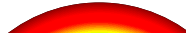
```
\pgfspectrarainbow(rainbow knock out=0){2cm}% the full half circle
```



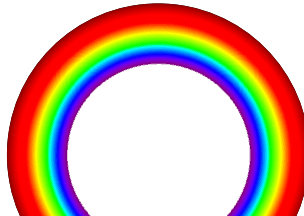
```
\pgfspectrarainbow(rainbow knock out=.4){2cm}% the default value
```



```
\pgfspectrarainbow(rainbow knock out=.8){2cm}% only 80% of the half circle is shown
```



```
\pgfspectrarainbow(rainbow knock out=-.4){2cm}% «extending» the half-circle
```

**rainbow fade**

default: {}

Applies a scope fading in the clipped region (requires loading the TikZ fadings library). For more information about the fadings see the TikZ manual. *(new in v2.1.0)*

```
%\usetikzlibrary{fadings}
...
\pgfspectrarainbow(rainbow fade=south){2cm}
\hspace{1cm}%
\pgfspectrarainbow(rainbow fade=west){2cm}
```

rainbow transparency

default: 0

The overall transparency of the rainbow. '0' (0%) stands for the fill colors in the rainbow without transparency ; '1' (100%) represents a totally transparent rainbow. *(new in v2.1.0)*

```
\pgfspectrarainbow(rainbow transparency=.5){2cm}
```

**rainbow background**

default: white

The background color below the rainbow (only visible with transparency). *(new in v2.1.0)*

```
\pgfspectrarainbow(rainbow background=blue){2cm}
\hspace{1cm}%
\pgfspectrarainbow(rainbow background=blue,rainbow transparency=.5){2cm}
```



Some of the TikZ keys that affect the rainbow:

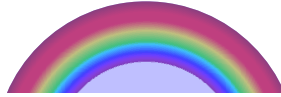
```
\pgfspectrarainbow[blue]{2cm}% Setting only the fill color only takes no effect
```



```
\pgfspectrarainbow[blue,scope fading=south]{2cm}
```



```
\pgfspectrarainbow[blue,opacity=.5]{2cm}
```



More examples in [Using \pgfspectrarainbow](#).

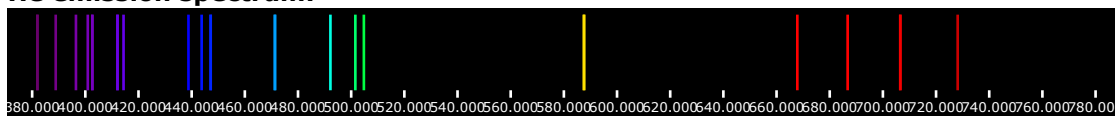
Examples

► Using `\pgfspectra`

Here are some examples for drawing some *eventually useful* spectra:

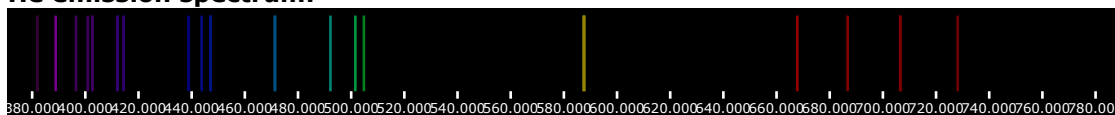
```
\pgfspectra[element=He,axis,label,label position=north west,
label after text=\ emission spectrum:]
```

He emission spectrum:



```
\pgfspectra[element=He,axis,label,label position=north west,label after text=
\ emission spectrum:,relative intensity,relative intensity threshold=.5]
```

He emission spectrum:

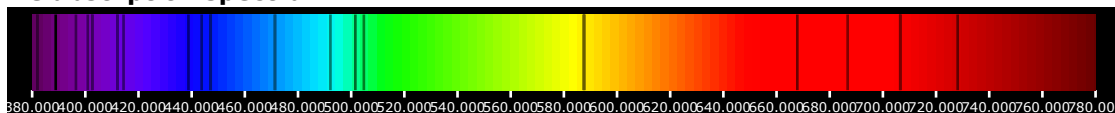


```
\pgfspectra[element=He,charge=all,line intensity=50,Imin=.05]
```



```
\pgfspectra[element=He,absorption,axis,label,label position=north west,label after
text=\ absorption spectrum:,relative intensity,relative intensity threshold=.5]
```

He absorption spectrum:



```
\pgfspectra[element=He,charge=all,absorption,line intensity=50]
```



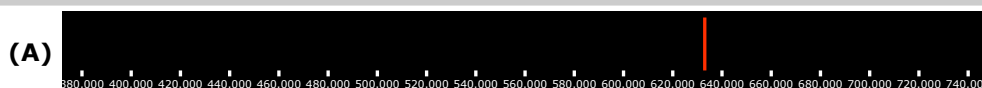
```
\pgfspectra[element=He,charge=all,relative intensity,back=visible75,gamma=2]
```



When the lines are manually inserted it's possible to use 'label before text' only with personalized text. In the next three examples 'label before text' is used to make labels for a multiple choice problem, omitting evidently the type of luminous font.

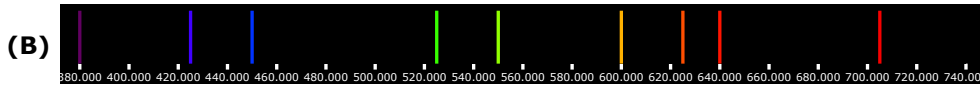
✓ Laser He-Ne

```
\pgfspectra[height=.7cm,end=740,lines={633},line
width=1.25pt,width=.75\linewidth,label,axis,label before text=(A),
axis font=\fontsize{4pt}{6pt}\selectfont]
```



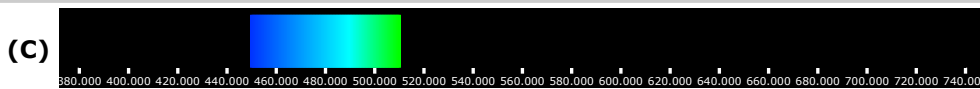
✓ Fluorescent lamp

```
\pgfspectra[height=.7cm,end=740,lines={380,425,450,525,550,600,625,640,705},
line width=1.25pt,width=.75\linewidth,label,axis,label before text=(B),
axis font=\fontsize{4pt}{6pt}\selectfont]
```



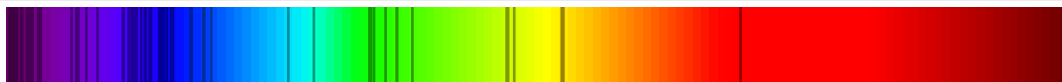
✓ Blue LED

```
\pgfspectra[height=.7cm,end=740,lines={450 to 510},
line width=1.25pt,width=.75\linewidth,label,axis,label before text=(C),
axis font=\fontsize{4pt}{6pt}\selectfont]
```



✓ Sun like spectrum

```
\pgfspectra[element={H,Fe,Mg,Na},absorption,line intensity=40,Imin=.05]
```



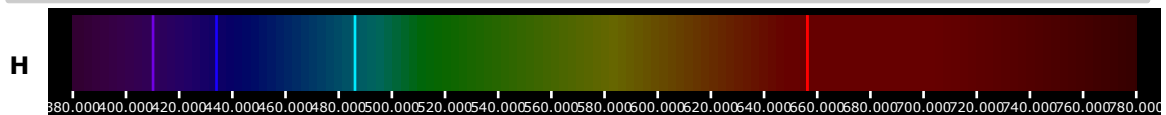
✓ Sirius like spectrum

```
\pgfspectra[element={H,He},absorption,line intensity=40,Imin=.05]
```

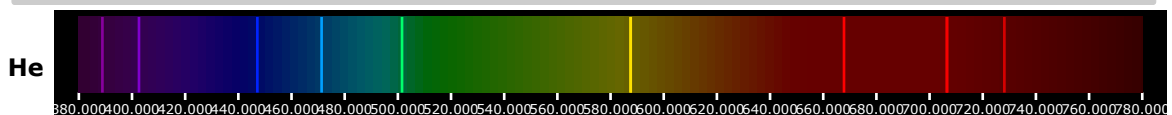


✓ "Classical" emission spectra of elements:

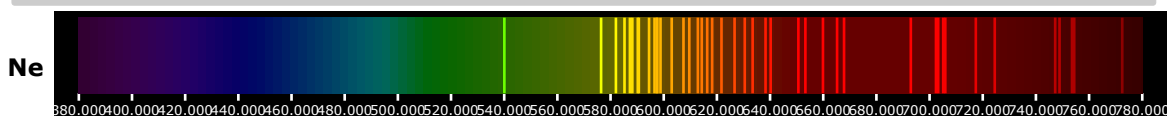
```
\pgfspectra[element=H,back=visible40,gamma=.6,label,axis,Imin=.05]
```



```
\pgfspectra[element=He,back=visible40,gamma=.6,label,axis,Imin=.05]
```

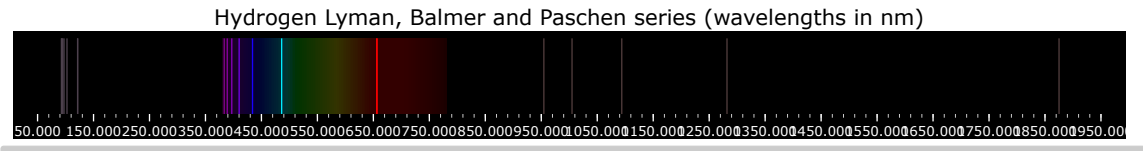


```
\pgfspectra[element=Ne,back=visible40,gamma=.6,label,axis,Imin=.05]
```



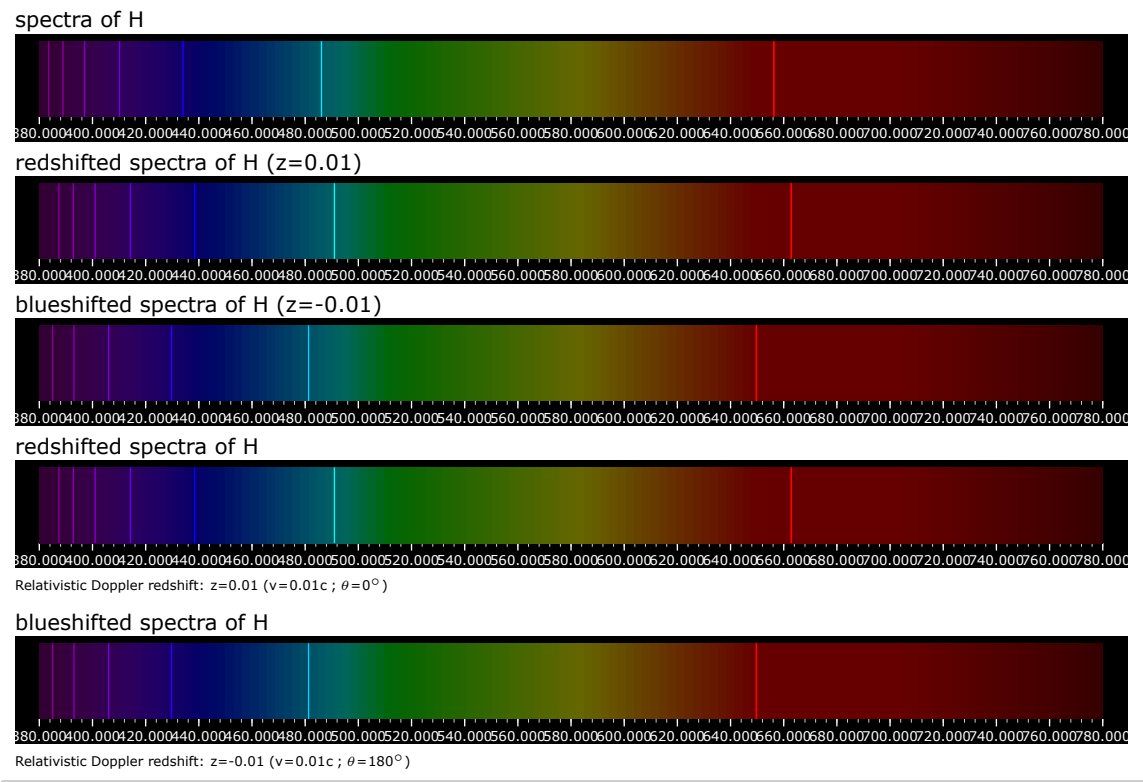
- ✓ Series of hydrogen:

```
\pgfspectra[element=H,line width=.5pt,begin=50,end=1950,axis,axis
step=100,axis ticks=4,back=visible40,gamma=.6,brightness=.5,label,label
position=north,label font=\footnotesize,label after text={ydrogen Lyman, Balmer
and Paschen series (wavelengths in nm)}]
```



- ✓ Redshifted & Blueshifted lines of hydrogen using the \foreach statement:

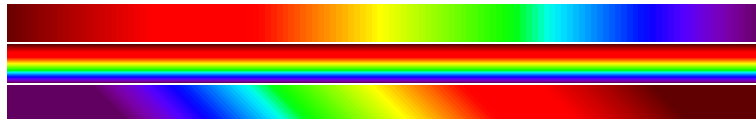
```
\pgfspectraStyle[axis,axis ticks=4,back=visible40,gamma=.6,line width=.5pt]
\pgfspectra[element=H,label,label position=north west,label
font=\footnotesize,label before text={spectra of \ }]
\foreach \SQ/\z/\shift in {H/0.01/redshifted,H/-0.01/blueshifted}{
  \pgfspectra[element=\SQ,label,label position=north west,label
font=\footnotesize,label before text={\shift\ spectra of \ },label after
text={\ (z=\z)},redshift=\z]
}
\foreach \SQ/\z/\shift in {H/{D=0.01/0}/redshifted,H/{D=0.01/180}/blueshifted}{
  \pgfspectra[element=\SQ,label,label position=north west,label
font=\footnotesize,label before text={\shift\ spectra of \ },redshift=\z,show
redshift value]
}
```



► Using `\pgfspectrashade` in TikZ

Obviously, the *normal* TikZ keys used to control the shadings apply to the shading generated via `\pgfspectrashade`:

```
\pgfspectrashade(380,780){myShadeA}
\\ \tikz{\fill[shading=myShadeA,shading angle=180] (0,0) rectangle (10,.5);}
\\ \tikz{\fill[shading=myShadeA,shading angle=90] (0,0) rectangle (10,.5);}
\\ \tikz{\fill[shading=myShadeA,shading angle=45] (0,0) rectangle (10,.5);}
```



Providing an opacity to the drawing and applying a shade works well too:

```
\pgfspectrashade(380,780){myShadeA}
• on black background:
  \\ \tikz{\fill[shading=myShadeA,opacity=.5] (0,0) rectangle (10,.5);}
• on white background:
  \\ \tikz{\fill[white,shading=myShadeA,opacity=.5] (0,0) rectangle (10,.5);}
• on red background:
  \\ \tikz{\fill[red,shading=myShadeA,opacity=.5] (0,0) rectangle (10,.5);}
```

- on black background:



- on white background:



- on red background:



The gamma in the generated shade (via `\pgfspectrashade`) could be modified using the `'gamma'` key of `\pgfspectra` set by the command `\pgfspectraStyle`:

```
\pgfspectrashade(380,780){myShadeA}
\tikz{\fill[myShadeA] (0,0) rectangle (10,.5);}
```



```
\pgfspectraStyle[gamma=2]
\pgfspectrashade(380,780){myShadeGammaII}
\tikz{\fill[myShadeGammaII] (0,0) rectangle (10,.5);}
```



```
\pgfspectraStyle[gamma=10]
\pgfspectrashade(380,780){myShadeGammaX}
\tikz{\fill[myShadeGammaX] (0,0) rectangle (10,.5);}
\pgfspectraStyleReset
```



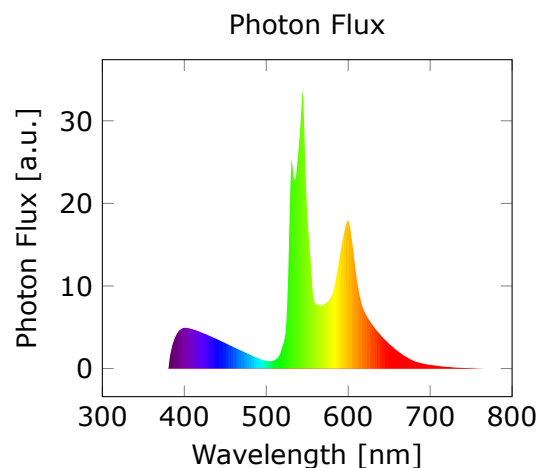
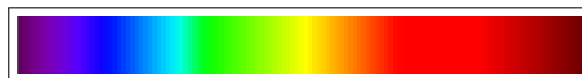
► Using `\pgfspectraplotshade` and `\pgfspectraplotmap` with PGFPLOTS

The command `\pgfspectraplotshade` is designed to build a shading to use with PGFPLOTS. Next examples show a few possibilities of how it could be used regarding two *sources*: a source of light and their photon flux and the blackbody spectral radiance.

In order to correctly make the filling between the path at axis and the plotted curve, the path should begin at 'shade begin' and end at 'shade end':
`\path[name path=axis] (shade begin,0) – (shade end,0);`

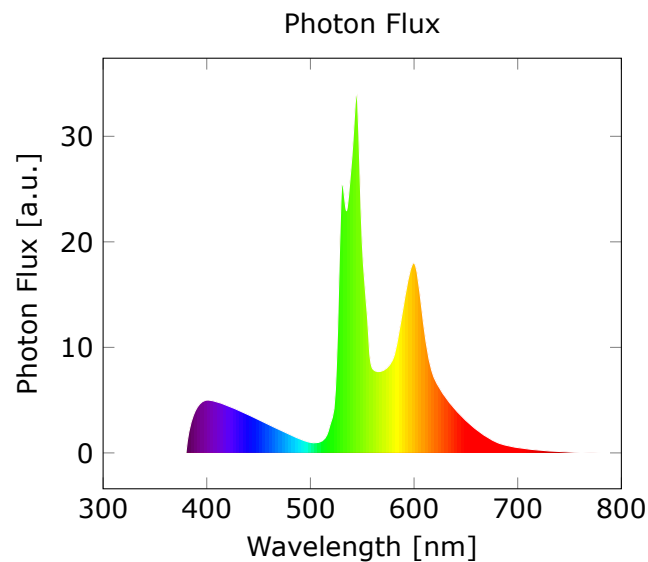
```
\pgfspectraplotshade{visiblespectrum}% default shading [380;780]nm
```

```
\makebox[\linewidth][c]{%
\fbbox{\tikz{\fill[shading=visiblespectrum] (0,0) rectangle (7.5,.75);}}%
}%
\\ [10pt]\begin{tikzpicture}
\begin{axis}[%
title= Photon Flux,%
xlabel={Wavelength [nm]},%
ylabel={Photon Flux [a.u.]},%
xmin=300,%
xmax=800,%
]%
\addplot[smooth, name path=spectrum,white] plot[] coordinates{%
( 380, 0 ) (400,5) ( 500, 1 ) ( 520, 3 ) ( 525, 8 ) ( 530, 25 )
( 535, 23 ) ( 540, 28 ) ( 545, 34 ) ( 550, 20 ) ( 555, 13 )
( 560, 8 ) ( 580, 9 ) ( 600, 18 ) ( 620, 7 ) ( 680, 1 ) ( 780, 0 )
};
\path[name path=axis] (380,0) -- (780,0);
\addplot+ [thick,shading=visiblespectrum]
fill between[of=spectrum and axis];
\end{axis}
\end{tikzpicture}%
```



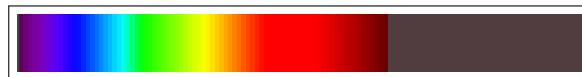
The above example could be obtained with the following *improved* code, based on a suggestion made by Stefan Pinnow:

```
\begin{tikzpicture}
  \pgfmathsetmacro{\xmin}{300}
  \pgfmathsetmacro{\xmax}{800}
  \pgfmathsetmacro{\shbegin}{380}
  \pgfmathsetmacro{\shend}{780}
  \pgfspectraplotshade[shade begin=\shbegin,shade end=\shend]{visiblespectrum}
  \begin{axis}[
    title=Photon Flux,
    xlabel={Wavelength in nm},
    ylabel={Photon Flux in a.u.},
    xmin=\xmin,
    xmax=\xmax,
  ]
    \addplot [smooth, name path=spectrum,white] coordinates {
      (380,0) (400,5) (500,1) (520,3) (525,8) (530,25)
      (535,23) (540,28) (545,34) (550,20) (555,13)
      (560,8) (580,9) (600,18) (620,7) (680,1) (780,0)
    };
    \path [name path=axis] (\shbegin,0) -- (\shend,0);
    \addplot+ [thick,shading=visiblespectrum]
      fill between [of=spectrum and axis];
  \end{axis}
\end{tikzpicture}
```

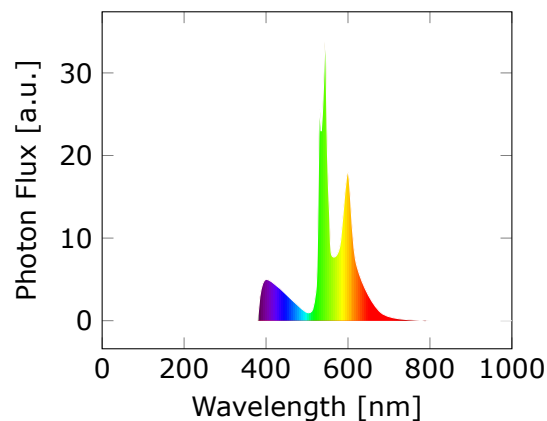


```
\pgfspectraplotshade[shade end=1000]{visiblespectrum}
```

```
\makebox[\linewidth][c]{%
\fbbox{\tikz{\fill[shading=visiblespectrum] (0,0) rectangle (7.5,.75);}}%
}%
\\ [10pt]\begin{tikzpicture}
\begin{axis}[%
title= Photon Flux,%
xlabel={Wavelength [nm]},%
ylabel={Photon Flux [a.u.]},%
xmin=0,%
xmax=1000,%
]
\addplot[smooth, name path=spectrum,white] plot[] coordinates{%
( 380, 0 ) (400,5) ( 500, 1 ) ( 520, 3 ) ( 525, 8 ) ( 530, 25 )
( 535, 23 ) ( 540, 28 ) ( 545, 34 ) ( 550, 20 ) ( 555, 13 )
( 560, 8 ) ( 580, 9 ) ( 600, 18 ) ( 620, 7 ) ( 680, 1 ) ( 780, 0 )
( 800, 0 ) ( 900, 0 ) ( 1000, 0 )
};
\path[name path=axis] (380,0) -- (1000,0);
\addplot+ [thick,shading=visiblespectrum]
fill between[of=spectrum and axis];
\end{axis}
\end{tikzpicture}%
```

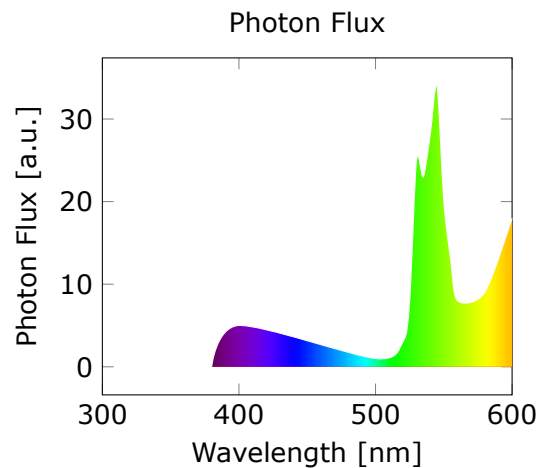
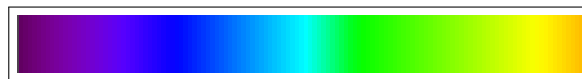


Photon Flux



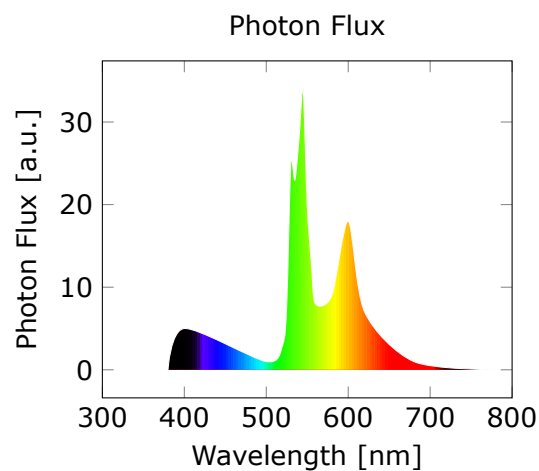
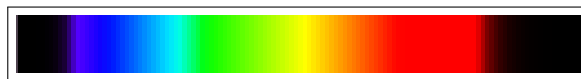
```
\pgfspectraplotshade[shade end=600]{visiblespectrum}
```

```
\makebox[\linewidth][c]{%
\fbbox{\tikz{\fill[shading=visiblespectrum] (0,0) rectangle (7.5,.75);}}%
}%
\\ [10pt]\\ \begin{tikzpicture}
\begin{axis}[%
title= Photon Flux,%
xlabel={Wavelength [nm]},%
ylabel={Photon Flux [a.u.]},%
xmin=300,%
xmax=600,%
]
\addplot[smooth, name path=spectrum,draw=none] plot[] coordinates{%
( 380, 0 ) (400,5) ( 500, 1 ) ( 520, 3 ) ( 525, 8 ) ( 530, 25 )
( 535, 23 ) ( 540, 28 ) ( 545, 34 ) ( 550, 20 ) ( 555, 13 )
( 560, 8 ) ( 580, 9 ) ( 600, 18 )
};
\path[name path=axis] (380,0) -- (600,0);
\addplot+ [thick,shading=visiblespectrum]
fill between[of=spectrum and axis];
\end{axis}
\end{tikzpicture}%
```



```
\pgfspectraplotshade[gamma=10]{visiblespectrumGammaX}
```

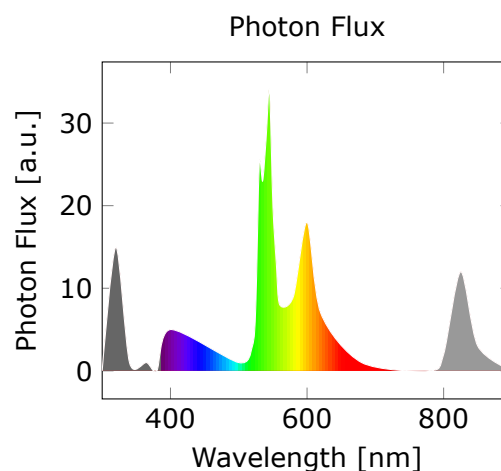
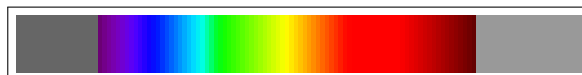
```
\makebox[\linewidth][c]{%
\fbbox{\tikz{\fill[shading=visiblespectrumGammaX] (0,0) rectangle (7.5,.75);}}%
}%
\\ [10pt]\\ \begin{tikzpicture}
\begin{axis}[%
title= Photon Flux,%
xlabel={Wavelength [nm]},%
ylabel={Photon Flux [a.u.]},%
xmin=300,%
xmax=800,%
]
\addplot[smooth, name path=spectrum,white] plot[] coordinates{%
( 380, 0 ) (400,5) ( 500, 1 ) ( 520, 3 ) ( 525, 8 ) ( 530, 25 )
( 535, 23 ) ( 540, 28 ) ( 545, 34 ) ( 550, 20 ) ( 555, 13 )
( 560, 8 ) ( 580, 9 ) ( 600, 18 ) ( 620, 7 ) ( 680, 1 ) ( 780, 0 )
};
\path[name path=axis] (380,0) -- (780,0);
\addplot+ [thick,shading=visiblespectrumGammaX]
fill between[of=spectrum and axis];
\end{axis}
\end{tikzpicture}%
```



Note: when setting the color for IR or UV within `\pgfspectraplotshade` make sure it doesn't end with '`!<number>`' like '`black!40`'; use '`black!40!white`' instead.

```
\pgfspectraplotshade[IRcolor=black!40!white,UVcolor=black!60!white,shade begin=300,shade end=900]{visiblespectrumIRUV}
```

```
\makebox[\linewidth][c]{%
\fbbox{\tikz{\fill[shading=visiblespectrumIRUV] (0,0) rectangle (7.5,.75);}}%
}%
\\ [10pt]\\ \begin{tikzpicture}
\begin{axis}[%
title= Photon Flux,%
xlabel={Wavelength [nm]},%
ylabel={Photon Flux [a.u.]},%
xmin=300,%
xmax=900,%
]
\addplot[smooth, name path=spectrum,white] plot[] coordinates{%
( 300 , 0 ) (320, 15) (340, 1) (365, 1)
( 380, 2.5 ) (400,5) ( 500, 1 ) ( 520, 3 ) ( 525, 8 ) ( 530, 25 )
( 535, 23 ) ( 540, 28 ) ( 545, 34 ) ( 550, 20 ) ( 555, 13 )
( 560, 8 ) ( 580, 9 ) ( 600, 18 ) ( 620, 7 ) ( 680, 1 ) ( 780, 0 )
( 800, 2 ) (825, 12) (850, 3) ( 900, 0 )
};
\path[name path=axis] (300,0) -- (900,0);
\addplot+ [thick,shading=visiblespectrumIRUV]
fill between[of=spectrum and axis];
\end{axis}
\end{tikzpicture}%
```



For the *blackbody spectral radiance*, the Planck's distribution is used with the values:

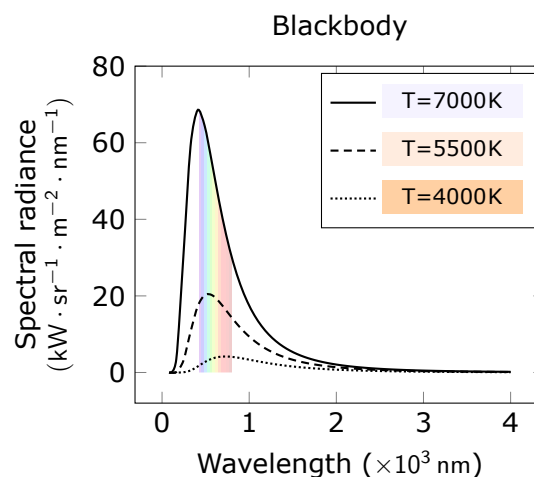
- $c = 3 \times 10^{14}$ microns \cdot s⁻¹ – speed of light
- $h = 6.626 \times 10^{22}$ kg \cdot microns² \cdot s⁻¹ – Planck constant
- $k_B = 1.38 \times 10^{-11}$ kg \cdot microns² \cdot s⁻² \cdot K⁻¹ – Boltzmann constant
- λ – wavelength (microns)
- T – temperature (K)
- Planck distribution: $B_\lambda = 2hc^2 \frac{1}{\lambda^5 e^{\frac{hc}{\lambda k_B T} - 1}}$ (kW \cdot sr⁻¹ \cdot m⁻¹ \cdot nm⁻¹)

The legend of the plots is created with the following definition:

```
\def\myentry#1{\tempercolor{#1}%
\tikz{\fill[tempercolor] (0,-.5pt) rectangle (40pt,.5pt)
node[midway,font=\footnotesize,anchor=mid] {\color{black} T=#1\hspace{.1ex}K};}}
```

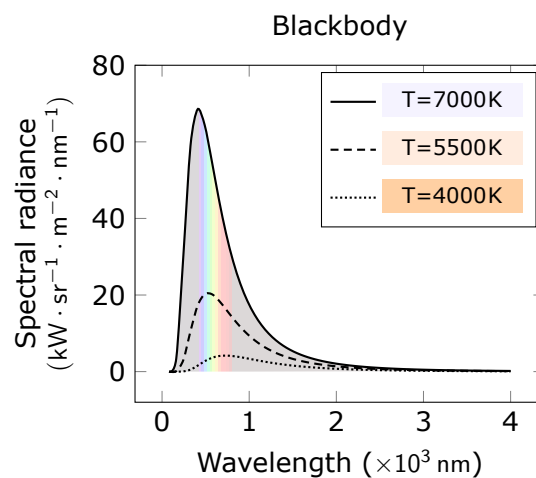
```
\pgfspectraplotshade[shade begin=0,shade end=4000,IRcolor=white,UVcolor=white,
gamma=.6,shade opacity=.2]{BBody}
```

```
\makebox[\linewidth][c]{%
\fbbox{\tikz{\fill[shading=BBody] (0,0) rectangle (7.5,.75);}}%
}%
\\ [10pt]\\ \begin{tikzpicture}
\begin{axis}[title=Blackbody,xlabel={Wavelength (\mathsf{\times 10^3}\,nm)}],%
ylabel={Spectral radiance\mathsf{(kW\cdot sr^{-1}\cdot m^{-2}\cdot nm^{-1})}},%
ylabel style={align=center},ymax=80,domain=0:4]%
\addplot[smooth, name path=spectrum,black,samples=50,thick] plot[]
{119.268/(x^5*(exp(14404/(x*7000))-1))};\addlegendentry{\myentry{7000}}%
\addplot[smooth,black,samples=50,densely dashed,thick] plot[]
{119.268/(x^5*(exp(14404/(x*5500))-1))};\addlegendentry{\myentry{5500}}%
\addplot[smooth,black,samples=50,densely dotted,thick] plot[]
{119.268/(x^5*(exp(14404/(x*4000))-1))};\addlegendentry{\myentry{4000}}%
\path[name path=axis] (axis cs:0,0) -- (axis cs:1,0);
\addplot+ [white,shading=BBody] fill between[of=spectrum and axis];
\end{axis}\end{tikzpicture}%
```



```
\pgfspectraplotshade[shade begin=0,shade end=4000,shade opacity=.2,
gamma=.6]{BBody}
```

```
\makebox[\linewidth][c]{%
\fbbox{\tikz{\fill[shading=BBody] (0,0) rectangle (7.5,.75);}}%
}%
\\ [10pt]\\ \begin{tikzpicture}
\begin{axis}[title=Blackbody,xlabel={Wavelength ( $\mathsf{\times 10^3}$ ,nm)}],%
ylabel={Spectral radiance\
 $\mathsf{\text{kW}\cdot\text{sr}^{-1}\cdot\text{m}^{-2}\cdot\text{nm}^{-1}}$ },%
ylabel style={align=center},ymax=80,domain=0:4]%
\addplot[smooth, name path=spectrum,black,samples=50,thick] plot[]
{119.268/(x^5*(exp(14404/(x*7000))-1))};\addlegendentry{\myentry{7000}}%
\addplot[smooth,black,samples=50,densely dashed,thick] plot[]
{119.268/(x^5*(exp(14404/(x*5500))-1))};\addlegendentry{\myentry{5500}}%
\addplot[smooth,black,samples=50,densely dotted,thick] plot[]
{119.268/(x^5*(exp(14404/(x*4000))-1))};\addlegendentry{\myentry{4000}}%
\path[name path=axis] (axis cs:0,0) -- (axis cs:1,0);
\addplot+ [white,shading=BBody] fill between[of=spectrum and axis];
\end{axis}\end{tikzpicture}%
```

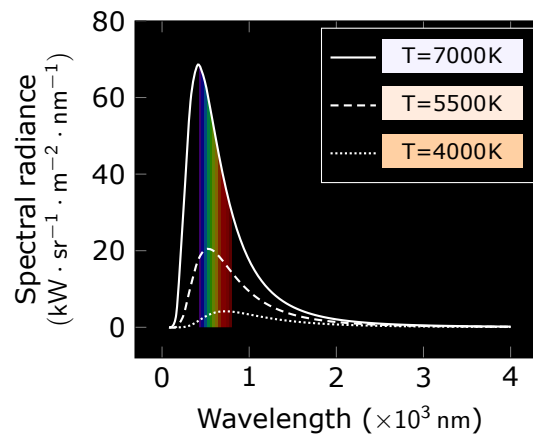



```
\pgfspectraplotshade[shade begin=0,shade end=4000,IRcolor=black,UVcolor=black,
gamma=.6,shade opacity=.5,shade opacity color=black]{BBody}
```

```
\makebox[\linewidth][c]{%
\fbbox{\tikz{\fill[shading=BBody] (0,0) rectangle (7.5,.75);}}%
}%
\\ [10pt]\\ \begin{tikzpicture}
\begin{axis}[
axis background/.style={fill=black},%
legend style={fill=black,draw=white},%
title=Blackbody,xlabel={Wavelength ( $\times 10^3$  nm)},%
ylabel={Spectral radiance\
 $\text{kW} \cdot \text{sr}^{-1} \cdot \text{m}^{-2} \cdot \text{nm}^{-1}$ },%
ylabel style={align=center},ymax=80,domain=0:4]%
\addplot[smooth, name path=spectrum,black,samples=50,thick] plot[]
{119.268/(x^5*(exp(14404/(x*7000))-1))};\addlegendentry{\myentry{7000}}%
\addplot[smooth,black,samples=50,densely dashed,thick] plot[]
{119.268/(x^5*(exp(14404/(x*5500))-1))};\addlegendentry{\myentry{5500}}%
\addplot[smooth,black,samples=50,densely dotted,thick] plot[]
{119.268/(x^5*(exp(14404/(x*4000))-1))};\addlegendentry{\myentry{4000}}%
\path[name path=axis] (axis cs:0,0) -- (axis cs:1,0);
\addplot+ [black,shading=BBody] fill between[of=spectrum and axis];
\end{axis}\end{tikzpicture}%
```



Blackbody

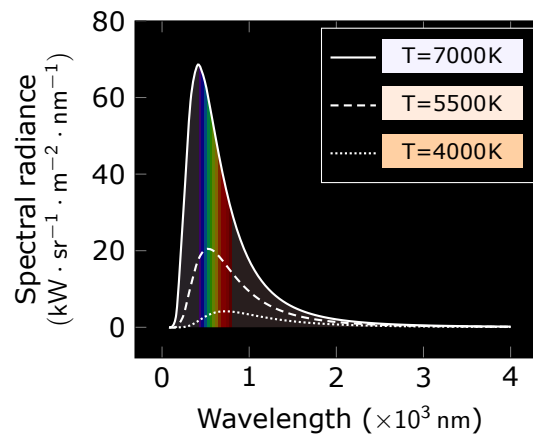


```
\pgfspectraplotshade[shade begin=0,shade end=4000,shade opacity=.5,
gamma=.6,shade opacity color=black]{BBody}
```

```
\makebox[\linewidth][c]{%
\fbbox{\tikz{\fill[shading=BBody] (0,0) rectangle (7.5,.75);}}%
}%
\\ [10pt]\\ \begin{tikzpicture}
\begin{axis}[
axis background/.style={fill=black},%
legend style={fill=black,draw=white},%
title=Blackbody,xlabel={Wavelength ( $\times 10^3$  nm)},%
ylabel={Spectral radiance\
 $\text{kW} \cdot \text{sr}^{-1} \cdot \text{m}^{-2} \cdot \text{nm}^{-1}$ },%
ylabel style={align=center},ymax=80,domain=0:4]%
\addplot[smooth, name path=spectrum,black,samples=50,thick] plot[]
{119.268/(x^5*(exp(14404/(x*7000))-1))};\addlegendentry{\myentry{7000}}%
\addplot[smooth,black,samples=50,densely dashed,thick] plot[]
{119.268/(x^5*(exp(14404/(x*5500))-1))};\addlegendentry{\myentry{5500}}%
\addplot[smooth,black,samples=50,densely dotted,thick] plot[]
{119.268/(x^5*(exp(14404/(x*4000))-1))};\addlegendentry{\myentry{4000}}%
\path[name path=axis] (axis cs:0,0) -- (axis cs:1,0);
\addplot+ [black,shading=BBody] fill between[of=spectrum and axis];
\end{axis}\end{tikzpicture}%
```



Blackbody

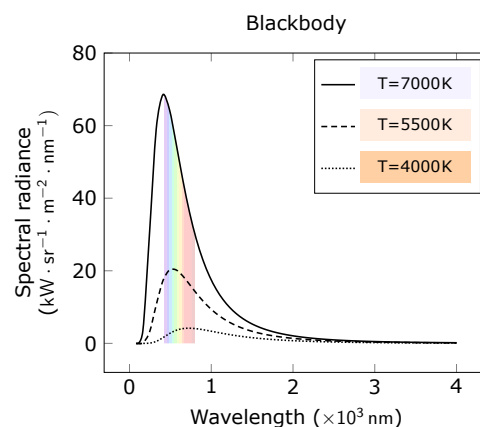


The above examples could be obtained with a much functional and prettier code, also proposed by Stefan Pinnow:

```

\begin{tikzpicture}[
  /pgf/declare function={
    BlackBodySpectralRadiance(\x,\T) = 119.268/(x^5*(exp(14404/(\x*\T))-1));
  },
]
  \pgfspectraplotshade[
    shade begin=0,
    shade end=4000,
    IRcolor=white,
    UVcolor=white,
    gamma=.6,
    shade opacity=.2,
  ]{BBody}
  \def\myentry#1{\tempercolor{#1}%
    \tikz{\fill [tempercolor] (0,-.5pt) rectangle (40pt,.5pt)
      node [midway,font=\footnotesize,anchor=mid]
        {\color{black} $T = #1\,\mathrm{K}$};}%
  }
  \begin{axis}[
    title=Blackbody,
    xlabel={Wavelength in $\mathrm{10^3\,nm}$},
    ylabel={%
      Spectral radiance in\
      $\mathrm{kW\cdot sr^{-1}\cdot m^{-2}\cdot nm^{-1}}$},
    ylabel style={align=center},
    ymax=80,
    cycle list name=linestyles,
    domain=0:4,
    samples=51,
    smooth,
  ]
  \pgfplotsinvokeforeach{7000,5500,4000}{
    \addplot+ [thick,name path=spectrum-#1] {BlackBodySpectralRadiance(x,#1)};
    \addlegendentry{\myentry{#1}}
  }
  \path [name path=axis] (axis cs:0,0) -- (axis cs:4,0);
  \addplot [shading=BBody] fill between [of=spectrum-7000 and axis];
  \end{axis}
\end{tikzpicture}

```



The `logarithmic` option of the `\pgfspectraplotshade` command could be used as a possible solution for the [TeX - LaTeX Stack Exchange](#) question, [How to create a electromagnetic spectrum using pgfplots package \(together with colormaps\)](#).

Filling optical spectrum curve with color gradient (first answer)

The original code lines that was replaced, in this possible answer, are commented.

```

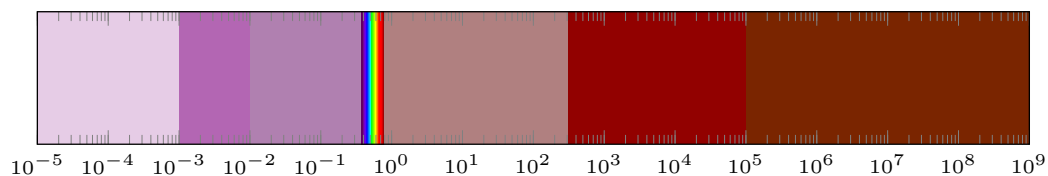
\documentclass[12pt]{article}
\usepackage[dvipsnames,table]{xcolor}
\usepackage{siunitx} % SI-units
\usepackage{pgf-spectra}
\usepackage{pgfplots}
\usepgfplotslibrary{units} % to add units easily to axis
\usepgfplotslibrary{fillbetween} % to fill inbetween curves
\usepgfplotslibrary{colormaps} % to create colormaps
\pgfplotsset{width=12.2cm, height=7cm}
\pgfplotsset{compat=newest} %(making it only compatalbe with
    %new releases of pgfplots)
%\pgfdeclarehorizontalshading{visiblelight}{50bp}{
%color(0.000000000000000bp)=(violet);
%color(8.333333333333333bp)=(blue);
%color(16.666666666666670bp)=(cyan);
%color(25.000000000000000bp)=(green);
%color(33.333333333333330bp)=(yellow);
%color(41.666666666666670bp)=(orange);
%color(50.000000000000000bp)=(red)
%}%
%
% make the horizontal shading and set the UV and IR colors -->
%\pgfspectraStyle[gamma=.6]% uncomment to change the gamma
\wlcolor{380}\colorlet{UV}{wlcolor}%
\wlcolor{780}\colorlet{IR}{wlcolor}%
\pgfspectraplotshade[logarithmic, UVcolor=UV]{logvisiblelight}
\pgfspectraplotshade{visiblelight}
%\pgfspectraStyleReset% uncomment to reset the style
\begin{document}
\begin{tikzpicture}[fill between/on layer={axis grid}]
\begin{axis}[
    xlabel={Wavelength},
    xticklabel style = {font=\tiny,yshift=0.2ex},
    xmin=10^-5,
    xmax=10^9,
    x unit=\si{\micro\meter},
    xmode=log,
    ymin=0,
    ymax=1,
    height=3cm,
    yticklabels={},
    ytick=\empty,
    legend cell align=left,
    legend style={at={(0.85,-0.77)},anchor=north}
]

```

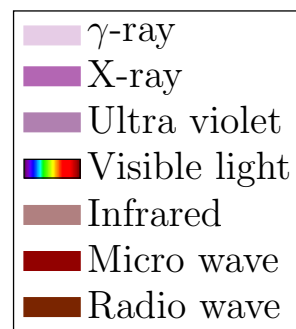
```

\addplot[draw=none, name path=start, forget plot] coordinates{(10^-5,0)(10^-5,1)};
\addplot[draw=none, name path=gamma, forget plot] coordinates{(10^-3,0)(10^-3,1)};
\addplot[draw=none, name path=xrays, forget plot] coordinates{(10^-2,0)(10^-2,1)};
%\addplot[draw=none, name path=uv, forget plot] coordinates{(0.4,0)(0.4,1)};
\addplot[draw=none, name path=uv, forget plot] coordinates{(0.38,0)(0.38,1)};
%\addplot[draw=none, name path=visible, forget plot] coordinates{(0.7,0)(0.7,1)};
\addplot[draw=none, name path=visible, forget plot] coordinates{(0.78,0)(0.78,1)};
\addplot[draw=none, name path=ir, forget plot] coordinates{(10^2.5,0)(10^2.5,1)};
\addplot[draw=none, name path=microwave, forget plot] coordinates{(10^5,0)(10^5,1)};
\addplot[draw=none, name path=radiowave, forget plot] coordinates{(10^9,0)(10^9,1)};
\addplot[violet!20, area legend] fill between[of=start and gamma];
\addlegendentry{\gamma-ray}
\addplot[violet!60, area legend] fill between[of=gamma and xrays];
\addlegendentry{X-ray}
%\addplot[violet, area legend] fill between[of=xrays and uv];
\addplot[UV!50,area legend] fill between[of=xrays and uv];
\addlegendentry{Ultra violet}
\addplot[shading=visiblelight, area legend] fill between[of=uv and visible];
\addlegendentry{Visible light}%makes the correct legend (not logarithmic)
\addplot[shading=logvisiblelight,forget plot] fill between[of=uv and visible];
%\addplot[red, area legend] fill between[of=visible and ir];
\addplot[IR!50,area legend] fill between[of=visible and ir];
\addlegendentry{Infrared}
%\addplot[Bittersweet, area legend] fill between[of=ir and microwave];
\addplot[IR!50!Bittersweet, area legend] fill between[of=ir and microwave];
\addlegendentry{Micro wave}
\addplot[Brown, area legend] fill between[of=microwave and radiowave];
\addlegendentry{Radio wave}
\end{axis}
\end{tikzpicture}
\end{document}

```



Wavelength [μm]



Filling optical spectrum curve with color gradient (second answer)

The original code lines that was replaced, in this possible answer, are commented and the code without changes was omitted.

```

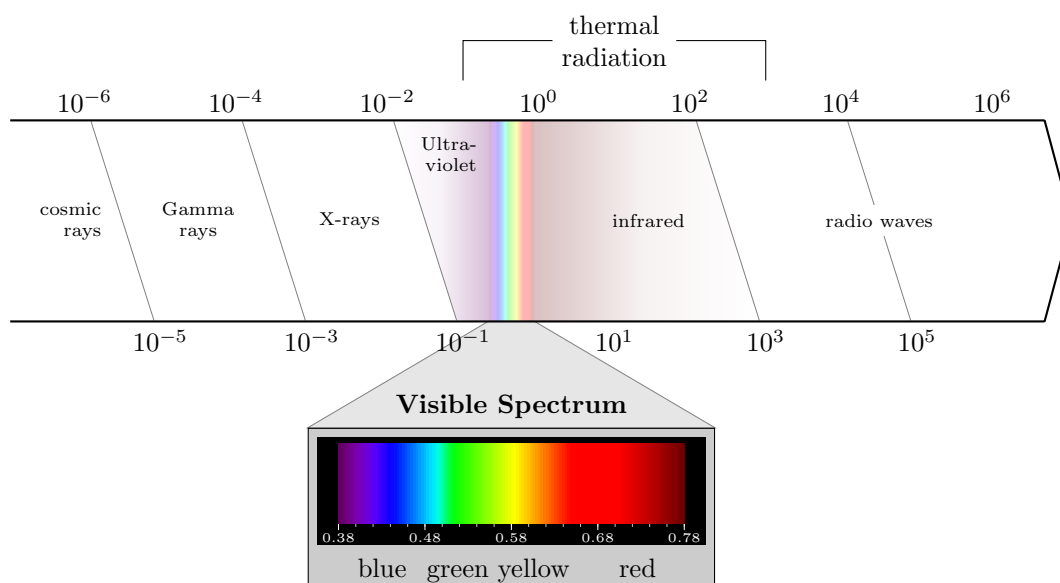
\documentclass{article}
\usepackage{tikz}
\usetikzlibrary{calc, positioning, shapes, backgrounds, fit, arrows}
\usepackage{pgf-spectra}
\usepackage{siunitx}
\usepackage{contour}
\begin{document}
\pgfdeclarehorizontalshading{visiblelight}{50bp}{%
    %https://tex.stackexchange.com/a/348492/120853
%   color(0bp)=(violet!25);
%   color(8.33bp)=(blue!25);
%   color(16.67bp)=(cyan!25);
%   color(25bp)=(green!25);
%   color(33.33bp)=(yellow!25);
%   color(41.5bp)=(orange!25);
%   color(50bp)=(red!25)
%}%
%
% make the horizontal shading and set the UV and IR colors -->
\pgfspectraStyle[gamma=.6]% uncomment to change the gamma
\wlcOLOR{380}\colorlet{UV}{wlcOLOR}%
\wlcOLOR{780}\colorlet{IR}{wlcOLOR}%
\pgfspectraplotshade[logarithmic,shade opacity=.3]{visiblelight}%
\pgfspectraStyleReset% uncomment to reset the style
\begin{tikzpicture}[%
    raylabel/.style={font=\scriptsize}
]
% ... code omitted ... %
% On background layer so already drawn arrow and scale lines cover it up nicely
\begin{scope}[on background layer]
    \node[
        inner sep=0pt,
        outer sep=0pt,
        %fit={{[xshift=-2.2em]WAVELENGTH_0|-ARROW.after tail}
        %([xshift=-2.2em]WAVELENGTH_1|-ARROW.before tail)}, shading=visiblelight]
        fit={{[xshift=-1.9em]WAVELENGTH_0|-ARROW.after tail}
        ([xshift=-3em]WAVELENGTH_1|-ARROW.before tail)}, shading=visiblelight}%
        (SMALL_VISIBLE_LIGHT) {}];
    \shade[
        left color=white,
        %right color=violet!25,
        right color=UV!25,
        %middle color=violet!5,
        middle color=UV!5,
        outer sep=0pt
    ]
    % ... code omitted ... %
    \shade[
        %left color=red!25,
        left color=IR!25,
        right color=white,
        %middle color=red!5,
        middle color=IR!5,
        outer sep=0pt,
    ]
    % ... code omitted ... %
\end{scope}
\end{tikzpicture}

```

```

% Some labels can be drawn automatically at the designated label coordinates:
\foreach [count=\i] \label in {
  {Gamma\\rays},
  {X-rays},
  {},%Skip this one
  {infrared}
}
  {
    \node[raylabel, align=center] at (LABEL_\i) {\label};
  }
% These do not fit the loop and are drawn manually:
\node[raylabel, align=right, anchor=north] at
  ([yshift=-1em,xshift=-2.5pt]$(WAVELENGTH_-2)!0.45!(WAVELENGTH_0)$)
  {Ultra-\\violet};
\node[raylabel, fill=white] at (CONNECTION_6) {radio waves};
\node[raylabel, left=0.1em of CONNECTION_1, align=right] {cosmic\\rays};
\node[
  draw,
  fill=black!20,
  below=4em of SMALL_VISIBLE_LIGHT,
  align=center,
  label=above:{\textbf{Visible Spectrum}}
] (FULL_VISIBLE_LIGHT) {%
%\pgfspectra[width=13em,height=3em]\\
%\pgfspectra[width=13em,height=3em,axis,axis unit=micron,axis step=100,
  axis ticks=4,axis unit precision=2]\\%
  %\pgfspectra also has a builtin axis which of course much better than
  %this terrible approach, but it is in nanometer
  {\num{0.38} \hfill\num{0.48} \hfill\num{0.58}%
    \hfill \num{0.68} \hfill\num{0.78}}\\
};
% ... code omitted ... %
\end{tikzpicture}
\end{document}

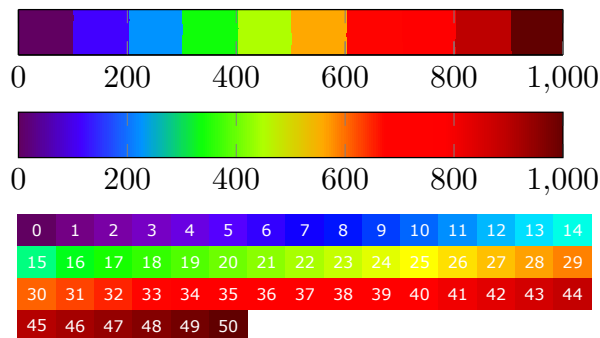
```



Next examples show possible usage of color maps feature of PGFPLOTS with the color map build with the `\pgfspectraplotmap` command:

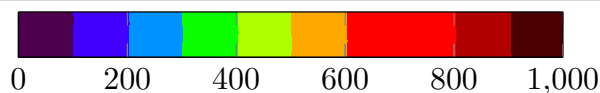
```
\pgfspectraplotmap{myColorMap}% default resolution (51 colors)
```

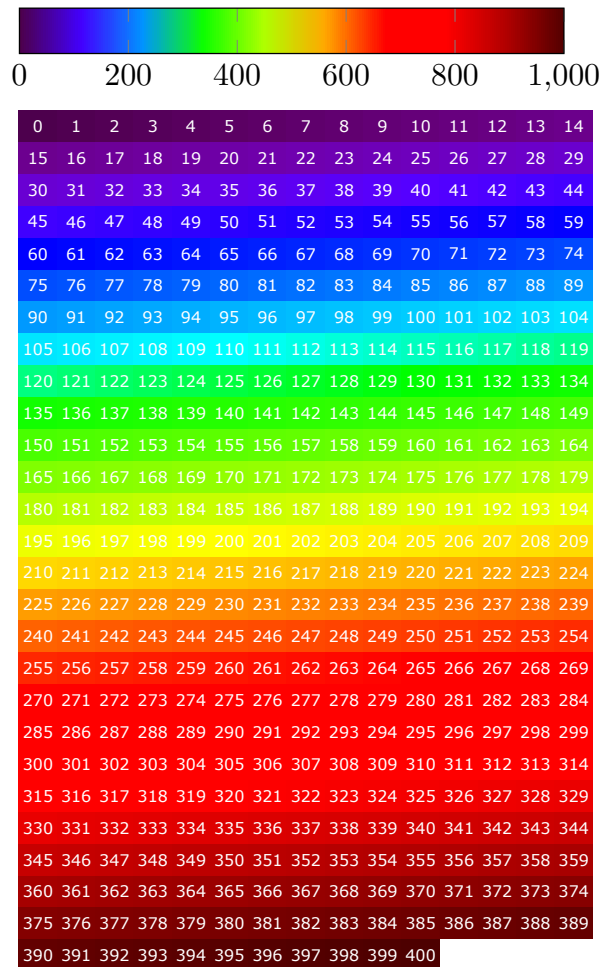
```
\pgfplotscolorbardrawstandalone[colormap={example}%
  {samples of colormap=(10 of myColorMap)},
colorbar horizontal,colormap access=const]
\\ \pgfplotscolorbardrawstandalone[colormap={example}%
  {samples of colormap=(10 of myColorMap)},
colorbar horizontal,colormap access=map]
% --- code improved by Stefan Pinnow --->
\begin{tikzpicture}
  \foreach \i [
    evaluate=\i as \x using {int(mod(\i,15))},
    evaluate=\i as \y using {floor(\i/15)},
  ] in {0,...,\pgfplotscolormaplastindexof{myColorMap}}{
    \fill [index of colormap={\i of myColorMap}]
      (\x*12pt,-\y*10pt) rectangle ++(12pt,10pt)
      node [inner sep=0pt,midway,font=\tiny,text=white] {\i};
  }
\end{tikzpicture}
```



```
\pgfspectraplotmap[h]{myColorMap}% high resolution (401 colors)
% color(0) -> 380nm color(1) -> 381nm ... color(60) -> 380+60=440nm ...
% ... color(400) -> 780nm
```

```
\pgfplotscolorbardrawstandalone[colormap={example}%
  {samples of colormap=(10 of myColorMapH)},
colorbar horizontal,colormap access=const]
\\ \pgfplotscolorbardrawstandalone[colormap={example}%
  {samples of colormap=(10 of myColorMapH)},
colorbar horizontal,colormap access=map]
% --- code improved by Stefan Pinnow --->
\begin{tikzpicture}
  \foreach \i [
    evaluate=\i as \x using {int(mod(\i,15))},
    evaluate=\i as \y using {floor(\i/15)},
  ] in {0,...,\pgfplotscolormaplastindexof{myColorMap}}{
    \fill [index of colormap={\i of myColorMap}]
      (\x*12pt,-\y*10pt) rectangle ++(12pt,10pt)
      node [inner sep=0pt,midway,font=\tiny,text=white] {\i};
  }
\end{tikzpicture}
```





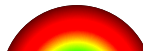
► Using `\pgfspectrarainbow`

Here are some examples of rainbows:

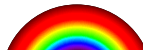
```
\pgfspectrarainbow{1cm}
```



```
\pgfspectrarainbow(rainbow start=0){1cm}
```



```
\pgfspectrarainbow(rainbow start=.4){1cm}
```



```
\pgfspectrarainbow(rainbow start=.8){1cm}
```



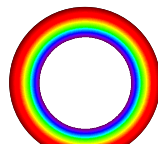
```
\pgfspectrarainbow(rainbow knock out=.8){1cm}
```



```
\pgfspectrarainbow(rainbow knock out=0){1cm}
```



```
\pgfspectrarainbow(rainbow knock out=-.8){1cm}
```



```
\pgfspectrarainbow(rainbow transparency=.5){1cm}
```



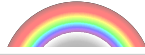
```
\pgfspectrarainbow(rainbow background=white){1cm}
```



```
\pgfspectrarainbow(rainbow background=blue,rainbow transparency=.5){1cm}
```



```
\pgfspectrarainbow(rainbow background=black,rainbow transparency=.5){1cm}
```



```
\pgfspectrarainbow(rainbow background=white,rainbow transparency=.5){1cm}
```



```
\pgfspectrarainbow(rainbow fade=south){1cm}
```



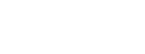
```
\pgfspectrarainbow(rainbow fade=north){1cm}
```



```
\pgfspectrarainbow[white,path fading=south]{1cm}
```



```
\pgfspectrarainbow[white](rainbow fade=south){1cm}
```



```
\pgfspectrarainbow[orange,path fading=west](rainbow fade=south){1cm}
```



```
\pgfspectrarainbow[blue,xslant=.1,opacity=.2]{1cm}
```



Alphabetical list of available options

\pgfspectra

key	description	type	default	value(s)
absorption	minimum intensity for the lines in the spectrum when using their relative intensities	boolean	false	{true, false}
axis	show or hide the axis	boolean	false	{true, false}
axis color	color of the axis	color	black	any named color or user defined color
axis font	font of the axis labels	font commands	{\tiny}	T _E X font commands
axis font color	color of the axis labels	color	white	any named color or user defined color
axis label	shows or hides the axis label below it	boolean	false	{true, false}
axis label position	sets the position of the axis label	text	center	left or center or right
axis label text	if not empty, shows the axis label with the given text below it	text	{}	
axis step	interval in nanometres between two major axis ticks	integer	20	[0; end-begin]nm
axis ticks	number of minor ticks	integer	0	{0,1,2,3,...}
axis unit	unit of the axis labels	text	nm	nm or micron or A
axis unit precision	number of significant digits (for values in nanometres) shown in axis labels	integer	3	{0,1,2,3,...}
back	spectrum background color	color	black	any named color or user defined color
backIRUV	IR and UV emission lines color in emission spectrum or background color of IR and UV regions in absorption spectrum	color	black	any named color or user defined color
backVIS	visible region background color in emission spectrum or emission lines color in absorption spectrum	color	black	any named color or user defined color
begin	first wavelength, in nanometres	integer	380	[10;4000]nm
brightness	brightness color correction as in the CMYK color model	decimal	1	[0;1]
charge	charge of the element(s)	integer	0	LSE Data: {0,1,2,3,4} NIST Data: {0,1}
element	chemical symbol of one element or comma sparated list of chemical symbols elements	text	NONE	H to Es except Fr
end	last wavelength, in nanometres	integer	780	[10;4000]nm
gamma	gamma color correction at the edges of the visible region	decimal	0.8	[0;1]
height	spectrum height	length	1cm	up to maximum T _E X dimension (16384pt)
Imin	minimum intensity of the lines	decimal	0	[0;1]
IRcolor	IR emission lines color in emission spectrum or background color of IR region in absorption spectrum	color	rgb(.3157,.2373,.2373)	any named color or user defined color
label	show or hide the axis labels	boolean	false	{true, false}
label after text	extra text to place after the label of the spectrum	text	{}	
label before text	extra text to place before the label of the spectrum	text	{}	
label font	font of the spectrum label	font commands	{\bfseries\small}	
label font color	color of the font of the spectrum label	color	black	any named color or user defined color
label position	position of the label of the spectrum	text	{west}	{west, north west, north, north east, east, south east, south, south west}

\pgfspectra (continuation)

key	description	type	default	value(s)
line intensity	draw all lines with the same intensity value	integer	100	{0,1,2,...,99,100}
line width	width of each line drawn in the spectrum	length	1pt	up to maximum \TeX dimension (16384pt)
lines	number or comma sparated list of numbers	integer or decimal	{}	[10;4000]nm
redshift	computes and draws the redshifted (or blueshifted) lines	text	{}	numeric value or {D=numeric value between [0;1]/any angle value in degrees}
relative intensity	draws the lines using their relative intensities	boolean	false	{true, false}
relative intensity threshold	all lines with intensity	decimal	0.25	[0;1]
show redshift value	show or hide the redshift value	boolean	false	{true, false}
use visible shading	visible region is drawn using a shading (instead of line by line)	boolean	true	{true, false}
UVcolor	UV emission lines color in emission spectrum or background color of UV region in absorption spectrum	color	rgb(.3,.2568,.3)	any named color or user defined color
width	spectrum width	length	{0.9\textwidth}	up to maximum \TeX dimension (16384pt)

\pgfspectradata

key	description	type	default	value(s)
precision	numerical precision of the data internally stored	integer	1	non negative integer
unit	unit of the spectral lines data	text	{}	nm or micron or A
begin	first wavelength, in nanometres	integer	380	[10;4000]nm
charge *	charge of the element(s)	integer	0	LSE Data: {0,1,2,3,4} NIST Data: {0,1}
element *	chemical symbol of one element or comma sparated list of chemical symbols elements	text	NONE	H to Es except Fr
end *	last wavelength, in nanometres	integer	780	[10;4000]nm
Imin *	minimum intensity of the lines	decimal	0	[0;1]
redshift *	numerical precision of the data internally stored	text	{}	numeric value or {D=numeric value between [0;1]/any angle value in degrees}

* options shared with `\pgfspectra`

\pgfspectratable

key	description	type	default	value(s)
back color	background color of the table header, of the <i>element(s)</i> column, of the table border lines and of the lines between rows	color	black!10	any named color or user defined color
data back color	the background color of each data row	color	white	any named color or user defined color
elements column width	the width of the <i>element(s)</i> column	length	2.5em	up to maximum \TeX dimension (16384pt)
text color	the color of all text in the table	color	black	any named color or user defined color
title	the title in the table header	text	<REDSHIFTED> SPECTRAL LINES OF ELEMENT(S) ..	
width	the total width of the table	length	\linewidth	up to maximum \TeX dimension (16384pt)

\pgfspectraplotshade

key	description	type	default	value(s)
shade begin	first wavelength, in nanometres	integer	380	[0;15999]nm
shade end	last wavelength, in nanometres	integer	780	[1;16000]nm
shade opacity	opacity of the computed shade	decimal	1	[0;1]
shade opacity color	background color of the computed shading	color	white	any named color or user defined color
logarithmic	the shading is build in a logarithmic scale	boolean	false	{true, false}

\pgfspectrarainbow

key	description	type	default	value(s)
rainbow background	background color below the rainbow	color	white	any named color or user defined color
rainbow fade	scope fading in the clipped region	text	{}	any named <i>TikZ</i> fading or user defined fading
rainbow knock out	relative distance from the half-circle base to perform the clip	decimal	.4	[-1;1]
rainbow start	fraction from which the rainbow colors begin, relative to the center of a circle with radius 1	decimal	.6	[0;1]
rainbow transparency	overall transparency of the rainbow	decimal	0	[0;1]

Recommendations and known issues

The code could be a bit slow, so if there are many spectra to draw, the time consumption to get them could be high. In that case it's preferable to compile individual spectrum via the *preview* package, for later inclusion with `\includegraphics{<filename>.pdf}`:

```
1 % <filename>.tex
2 \documentclass{article}
3 \usepackage{pgf-spectra}
4 \usepackage[active,tightpage]{preview}
5 \PreviewEnvironment{tikzpicture}
6 \setlength\PreviewBorder{1pt}%
7 %%%%%%%%%%%
8 \begin{document}
9 \pgfspectra[element=H,width=15cm]
10 \end{document}
```

Hint for TeX 'limits':

If tex capacity exceeded when running...

«! TeX capacity exceeded, sorry [main memory size=2000001].»

just make a `\newpage` at the point of origin of the message (ejecting the page releases the TeX memory!)

Index

COMMANDS

<code>\pgfspectra</code>	7
<code>\pgfspectraStyle</code>	8
<code>\pgfspectraStyleReset</code>	9
<code>\pgfspectrashade</code>	9
<code>\usepgfspectralibrary</code>	9
<code>\wcolor</code>	8

LIBRARIES

data	24
The options for <code>\pgfspectradata</code>	24
The options for <code>\pgfspectratable</code>	25
<code>\pgfspectradata</code>	24
<code>\pgfspectratable</code>	25
<code>\pgfspectrawrite</code>	28
pgfplots	30
The options for <code>\pgfspectraplotshade</code>	32
<code>\pgfspectraplotmap</code>	30
<code>\pgfspectraplotshade</code>	31
rainbow	37
The options for <code>\pgfspectrarainbow</code> ..	38
<code>\pgfspectrarainbow</code>	37
tempercolor	36
<code>\tempercolor</code>	36

OPTIONS

<code>\pgfspectra</code>	
absorption	16
axis	16
axis color	19
axis font	20
axis font color	20
axis label	19
axis label position	19
axis label text	19
axis step	17
axis ticks	17
axis unit	17
axis unit precision	18
back	13
backIRUV	14
begin	15
brightness	13
charge	11
element	10
end	16
gamma	13
height	10
Imin	11
IRcolor	14
label	20
label after text	21
label before text	21
label font	20
label font color	21
label position	20
line intensity	12

line width	15
lines	15
redshift	21
relative intensity	11
relative intensity threshold	11
show redshift value	22
use visible shading	10
UVcolor	15
width	10

`\pgfspectradata`

name of data set	24
precision	24
unit	24

`\pgfspectraplotshade`

logarithmic	34
shade begin	32
shade end	32
shade opacity	33
shade opacity color	33

`\pgfspectrarainbow`

rainbow background	39
rainbow fade	39
rainbow knock out	38
rainbow start	38
rainbow transparency	39

`\pgfspectratable`

back color	27
data back color	27
elements column width	28
text color	27
title	25
width	27