

dtxgen - generate template for LaTeX self-extracting .dtx file

Usage

```
dtxgen [options] basename.[sty|cls]
```

Options

dtxgen recognizes the following options:

<code>-s, --short=...</code>	A short, one-liner, description for the class or package. By default, the string 'A new LaTeX class' or 'A new LaTeX package' will be used.
<code>-n, --name=...</code>	Your name (first name, followed by surname). Alternatively, you can set a default value in the environment variable <code>NAME</code> ; if you do so and still use this option, the option's value will have priority.
<code>-m, --mail=...</code>	Your email address. Alternatively, you can set a default value in the environment variable <code>EMAIL</code> ; if you do so and still use this option, the option's value will have priority.
<code>-c, --class=...</code>	For class templates only: inserts a <code>\LoadClass {...}</code> , so that the new class will start with the properties of the ... class. The default is <code>article</code> .
<code>-d, --date=...</code>	Set the initial version's date. By default, the current date will be used. The date should be entered in the <code>yyyymmdd</code> format, although it will be stored the LaTeX way: <code>yyyy/mm/dd</code> .
<code>-V, --version</code>	Prints the script's version and exits.
<code>-h, --help</code>	Prints help information and exits.

Description

This script is based on the self-extracting model dtx file described by Joseph Wright, see <http://www.texdev.net/2009/10/06/a-model-dtx-file/>.

You can use **dtxgen** when you decide to create a new LaTeX class or style. For example, when you decide to make a class named *myclass*, you could run:

```
dtxgen -n 'your name' -m 'your@email.ad' myclass.cls
```

or, if you have an environment with your name and email address defined in `NAME` and `EMAIL`, you could simply type:

```
dtxgen myclass.cls
```

and you would end up with five files: *myclass.dtx*, *myclass.cls*, *myclass.pdf*, *README*, and *Makefile*.

The resulting *.dtx* file is a template file in which some minimal example content will have to be replaced with real content, but the dates, *basename*, author's name and email address are already in place and, depending on whether you use used a *.cls* or a *.sty* extension in the argument, it is formatted to be either a class or a package source file.

Makefile

The *Makefile* can be used to compile new versions of your work; it contains the following targets:

<code>all</code>	(the default) generate the style or class file, the pdf-documentation, and a <i>README</i> file.
<code>distclean</code>	remove all files that can be regenerated,
<code>clean</code>	same, except the style or class file, the pdf-documentation, and a <i>README</i> file.
<code>inst</code>	install in the user's TeX tree,
<code>install</code>	install in the local TeX tree (uses <code>sudo</code>)
<code>zip</code>	produce a zip file ready for upload to CTAN