Examples of Code Listings in LaTeX

Andrew Pinkham

June 28, 2013

Abstract

This project is meant to help jump start writers looking to display source code in LaTeX documents. The project provides a set of TeX files and a PDF of the output of said TeX files. The intention is that a reader may view the various code listing demonstrations in the PDF, and then read, learn, or simply copy the LaTeX necessary to display their code in the same manner. Basic knowledge of LaTeX is assumed.

We display code using the following methods: `verbatim`, `listings` (not to be confused with `listing`), and `minted`.

To avoid a monolithic LaTeX document and make perusing the code simpler, each subsection is it's own TeX file, as is the preamble.

Note that all code is licensed under the Simplified BSD License, a copy of which is included with the project code, found on github.

Contents

List of listings

1 Basic Tools and Techniques for Listing Code

1.1 `verbatim` Environment

Here is an example that uses LaTeX's `verbatim` environment.

```
from django.db import models

class Poll(models.Model):
    question = models.CharField(max_length=200)
    pub_date = models.DateTimeField('date published')

class Choice(models.Model):
    poll = models.ForeignKey(Poll)
    choice_text = models.CharField(max_length=200)
    votes = models.IntegerField(default=0)
```

Note that the quotes used are standard programming quotes, as opposed to LaTeX's backtick-quote combination. This is possible because of the `upquote` package, included in the preamble of the document.

LaTeX further supplies the `\verb` command, allowing for inline formatting.

On a related note, note that the `\verb` command can only be used in captions if the entire command is preceded by the `\cprotect` command, included in the `cprotect` package, which you can read more about on the documentation provided by CTAN. An example is demonstrated in the caption of listing 1.1 on page 4.

Finally, text in chapter titles (or section, subsection, etc) are formatted with `\texttt` in this document, as the use of `\verb` in these areas appears to conflict with the `hyperref` package. We could use `\texttt` in captions as well, but that would defeat our purpose of demonstrating all the tools.

1.2 `listings` Package

The following uses the `listings` package to create an environment.

Listing 1: `models.py` from Django Tutorial using Listings

```
1  from django.db import models
2
3  class Poll(models.Model):
4      question = models.CharField(max_length=200)
5      pub_date = models.DateTimeField('date published')
6
7  class Choice(models.Model):
8      poll = models.ForeignKey(Poll)
9      choice_text = models.CharField(max_length=200)
10     votes = models.IntegerField(default=0)
```

On top of an environment, the `listings` package also provides a way to format inline text using `\lstinline{}`, replacing the `verb` command. Note that the settings for `\lstinline{}` must be set in the options of `\lstset{}` in the preamble. For more on the `listings` package, please see the wikibook on the subject, or else the documentation provided by CTAN. Note that while not discussed here, it is possible to create a list of listings (a table of contents of all the listings).

1.3 `Minted` Package

The last environment I'll demonstrate is the `minted` package. It takes code in the environment, and runs it through Python's `pygments` library. The resulting colored LaTeX syntax is placed in a `verbatim` environment.

```
from django.db import models

class Poll(models.Model):
    question = models.CharField(max_length=200)
    pub_date = models.DateTimeField('date published')

class Choice(models.Model):
    poll = models.ForeignKey(Poll)
    choice_text = models.CharField(max_length=200)
    votes = models.IntegerField(default=0)
```

While `minted` does not provide an inline formatting command, it does provide a shortcut command `mint` for short code snippets.

```
fib = lambda n: n if n < 2 else fib(n-1) + fib(n-2)
```

The `minted` environment also provides the ability to be inserted into listings, with labels and captions.

```
from django.db import models

class Poll(models.Model):
    question = models.CharField(max_length=200)
    pub_date = models.DateTimeField('date published')

class Choice(models.Model):
    poll = models.ForeignKey(Poll)
    choice_text = models.CharField(max_length=200)
    votes = models.IntegerField(default=0)
```

Listing 1.1: `models.py` from Django Tutorial using Minted

2 Custom Environments and Macros

This section will demonstrate various macros defined in the preamble that you can use to make your document clearer.

2.1 Aside

While not for code, per se, this aside environment demonstrates the use of the `mdframed` package.

Aside - Lorem Ipsum Example

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.