

CS 683 Emerging Technologies
Fall Semester, 2006
Doc 8 Django Views & Template
Sep 21, 2006

Copyright ©, All rights reserved. 2006 SDSU & Roger
Whitney, 5500 Campanile Drive, San Diego, CA 92182-7700
USA. OpenContent (<http://www.opencontent.org/opl.shtml>)
license defines the copyright on this document.

References

Django Template Language, <http://www.djangoproject.com/documentation/templates/>

Django Tutorial

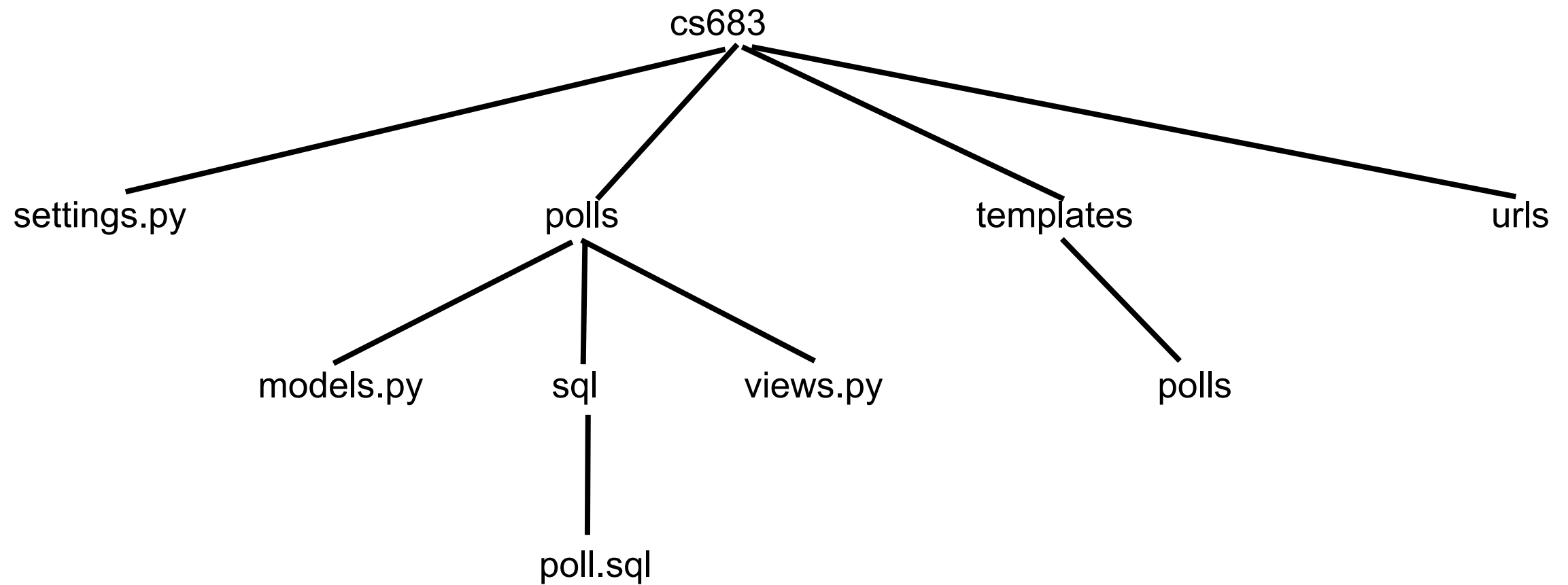
<http://www.djangoproject.com/documentation/tutorial3/>, <http://www.djangoproject.com/documentation/tutorial4/>

Most of the examples in this document are from the above references

Reading

Django Template Language, <http://www.djangoproject.com/documentation/templates/>

Polls Example



cs683/polls/models.py

```
from django.db import models
```

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

```
    def __str__(self):  
        return self.question
```

```
class Admin:  
    pass
```

```
class Choice(models.Model):  
    poll = models.ForeignKey(Poll)  
    choice = models.CharField(maxlength=200)  
    votes = models.IntegerField()
```

```
    def __str__(self):  
        return self.choice
```

```
class Admin:  
    pass
```

urls.py

```
from django.conf.urls.defaults import *

urlpatterns = patterns("",
    (r'^polls/$', 'cs683.polls.views.index'),
    (r'^polls/(?P<poll_id>\d+)/detail.html$', 'cs683.polls.views.detail'),
    (r'^polls/(?P<poll_id>\d+)/results/details$', 'cs683.polls.views.results'),
    (r'^polls/(?P<poll_id>\d+)/vote/$', 'cs683.polls.views.vote'),
)
```

```
urlpatterns = patterns('cs683.polls.views',
    (r'^polls/$', 'index'),
    (r'^polls/(?P<poll_id>\d+)/$', 'detail'),
    (r'^polls/(?P<poll_id>\d+)/results/$', 'results'),
    (r'^polls/(?P<poll_id>\d+)/vote/$', 'vote'),
)
```

Part of cs683/polls/views.py

```
from django.shortcuts import render_to_response
from cs683.polls.models import Choice, Poll

def index(request):
    poll_list = Poll.objects.all().order_by('-pub_date')
    return render_to_response('polls/index.html', {'object_list': poll_list})
```

cs683/templates/polls/poll_list.html

```
{% if object_list %}
    <ul>
        {% for poll in object_list %}
            <li><a href="{{poll.id}}/details.html">{{ poll.question }}</a></li>
        {% endfor %}
    </ul>
{% else %}
    <p>No polls are available.</p>
{% endif %}
```

Template Variables

```
def sample(request):  
    cat = {'a':'mouse'}  
    dog = ['a', 'b']  
    bar = Bar()  
    return render_to_response('sample.html', {'bar':bar, 'cat': cat, 'dog':dog})
```

sample.html

```
{{cat.a}}  
{{dog.1}}  
{{bar.foo}}
```

```
class Bar:  
    def foo(self):  
        return 'sample'
```


Tags

- block
- comment
- cycle
- debug
- extends
- filter
- firstof
- for
- if
- ifchanged
- ifequal
- ifnotequal
- include
- load
- now
- regroup
- spaceless
- ssi
- templatetag
- widthratio

<http://www.djangoproject.com/documentation/templates/>

debug Tag Output

```
{'BaseHTTPServer': , 'Cookie': , 'SocketServer': , 'StringIO': , 'UserDict': , '__builtin__': , '__future__': , '__main__': , '_codecs': , '_locale': , '_random': , '_socket': , '_sre': , '_ssl': , '_weakref': , 'array': , 'atexit': , 'base64': , 'binascii': , 'bisect': , 'cPickle': , 'cStringIO': , 'cgi': , 'codecs': , 'copy': , 'copy_reg': , 'cs683': , 'cs683.books': , 'cs683.books.django': None, 'cs683.books.models': , 'cs683.django': None, 'cs683.polls': , 'cs683.polls.cs683': None, 'cs683.polls.django': None, 'cs683.polls.models': , 'cs683.polls.urls': , 'cs683.polls.views': , 'cs683.settings': , 'cs683.simpleExamples': , 'cs683.simpleExamples.django': None, 'cs683.simpleExamples.models': , 'cs683.urls': , 'datetime': , 'dis': , 'django': , 'django.conf': , 'django.conf.django': None, 'django.conf.global_settings': , 'django.conf.os': None, 'django.conf.sys': None, 'django.conf.urls': , 'django.conf.urls.defaults': , 'django.conf.urls.django': None, 'django.contrib': , 'django.contrib.admin': , 'django.contrib.admin.django': None, 'django.contrib.admin.models': , 'django.contrib.auth': , 'django.contrib.auth.datetime': None, 'django.contrib.auth.django': None, 'django.contrib.auth.middleware': , 'django.contrib.auth.models': , 'django.contrib.contenttypes': , 'django.contrib.contenttypes.django': None, 'django.contrib.contenttypes.models': , 'django.contrib.sessions': , 'django.contrib.sessions.base64': None, 'django.contrib.sessions.cPickle': None, 'django.contrib.sessions.datetime': None, 'django.contrib.sessions.django': None, 'django.contrib.sessions.md5': None, 'django.contrib.sessions.middleware': , 'django.contrib.sessions.models': , 'django.contrib.sessions.random': None, 'django.contrib.sessions.sys': None, 'django.contrib.sites': , 'django.contrib.sites.django': None, 'django.contrib.sites.models': , 'django.core': , 'django.core.cache': , 'django.core.cache.backends': , 'django.core.cache.backends.base': , 'django.core.cache.backends.django': None, 'django.core.cache.backends.simple': , 'django.core.cache.backends.time': None, 'django.core.cache.cgi': None, 'django.core.cache.django': None, 'django.core.django': None, 'django.core.email': None, 'django.core.exceptions': , 'django.core.handlers': , 'django.core.handlers.base': , 'django.core.handlers.django': None, 'django.core.handlers.pprint': None, 'django.core.handlers.sys': None, 'django.core.handlers.wsgi': , 'django.core.mail': , 'django.core.management': , 'django.core.optparse': None, 'django.core.os': None, 'django.core.re': None, 'django.core.rfc822': None, 'django.core.servers': , 'django.core.servers.BaseHTTPServer': None, 'django.core.servers.basehttp': , 'django.core.servers.django': None, 'django.core.servers.os': None, 'django.core.servers.re': None, 'django.core.servers.sys': None, 'django.core.servers.time': None, 'django.core.servers.types': None, 'django.core.servers.urllib': None, 'django.core.sets': None, 'django.core.shutil': None, 'django.core.signals': , 'django.core.smtplib': None, 'django.core.sys': None, 'django.core.textwrap': None, 'django.core.urlresolvers': , 'django.core.validators': , 'django.db': , 'django.db.backends': , 'django.db.backends.datetime': None, 'django.db.backends.sqlite3': , 'django.db.backends.sqlite3.base': , 'django.db.backends.sqlite3.django': None, 'django.db.backends.sqlite3.pysqlite2': None, 'django.db.backends.sqlite3.threading': None, 'django.db.backends.time': None, 'django.db.backends.util': , 'django.db.django': None, 'django.db.models': , 'django.db.models.base': , 'django.db.models.bisect': None, 'django.db.models.django': None, 'django.db.models.fields': , 'django.db.models.fields.datetime': None, 'django.db.models.fields.django': None, 'django.db.models.fields.generic': , 'django.db.models.fields.os': None, 'django.db.models.fields.related': , 'django.db.models.fields.sets': None, 'django.db.models.fields.time': None, 'django.db.models.loading': , 'django.db.models.manager': , 'django.db.models.manipulators': , 'django.db.models.operator': None, 'django.db.models.options': , 'django.db.models.os': None, 'django.db.models.query': , 'django.db.models.re': None, 'django.db.models.related': , 'django.db.models.sets': None, 'django.db.models.signals': , 'django.db.models.sys': None, 'django.db.models.types': None, 'django.db.thread': None, 'django.db.transaction': , 'django.dispatch': , 'django.dispatch.__future__': None, 'django.dispatch.dispatcher': , 'django.dispatch.django': None, 'django.dispatch.errors': , 'django.dispatch.robustapply': , 'django.dispatch.saferef': , 'django.dispatch.traceback': None, 'django.dispatch.types': None, 'django.dispatch.weakref': None, 'django.forms': , 'django.forms.django': None, 'django.http': , 'django.http.Cookie': None, 'django.http.cgi': None, 'django.http.django': None, 'django.http.pprint': None, 'django.http.urllib': None, 'django.middleware': , 'django.middleware.common': , 'django.middleware.django': None, 'django.middleware.doc': , 'django.middleware.md5': None, 'django.middleware.os': None, 'django.shortcuts': , 'django.shortcuts.django': None, 'django.template': , 'django.template.context': , 'django.template.defaultfilters': , 'django.template.defaulttags': , 'django.template.django': None, 'django.template.inspect': None, 'django.template.loader': , 'django.template.loader_tags': , 'django.template.loaders': , 'django.template.loaders.app_directories': , 'django.template.loaders.django': None, 'django.template.loaders.filesystem': , 'django.template.loaders.os': None, 'django.template.pprint': None, 'django.template.random': None, 'django.template.re': None, 'django.template.sys': None, 'django.utils': , 'django.utils.threading_local': , 'django.utils.autoreload': , 'django.utils.cStringIO': None, 'django.utils.cache': , 'django.utils.datastructures': , 'django.utils.datetime': None, 'django.utils.django': None, 'django.utils.functional': , 'django.utils.gettext': None, 'django.utils.html': , 'django.utils.md5': None, 'django.utils.os': None, 'django.utils.re': None, 'django.utils.string': None, 'django.utils.sys': None, 'django.utils.termcolors': , 'django.utils.text': , 'django.utils.thread': None, 'django.utils.threading': None, 'django.utils.time': None, 'django.utils.translation': , 'django.utils.types': None, 'django.views': , 'django.views.debug': , 'django.views.django': None, 'django.views.itertools': None, 'django.views.os': None, 'django.views.re': None, 'email': , 'email.Charset': , 'email.Encoders': , 'email.Errors': , 'email.Header': , 'email.MIMEBase': , 'email.MIMENonMultipart': , 'email.MIMEText': , 'email.Message': , 'email.Utils': , 'email.__future__': None, 'email._compat22': , 'email._parseaddr': , 'email.base64': None, 'email.base64MIME': , 'email.binascii': None, 'email.cStringIO': None, 'email.email': None, 'email.encodings': None, 'email.os': None, 'email.quopri': None, 'email.quopriMIME': , 'email.random': None, 'email.re': None, 'email.socket': None, 'email.string': None, 'email.time': None, 'email.types': None, 'email.uu': None, 'email.warnings': None, 'encodings': , 'encodings.aliases': , 'encodings.ascii': , 'encodings.codecs': None, 'encodings.encodings': None, 'encodings.exceptions': None, 'encodings.types': None, 'errno': , 'exceptions': , 'fcntl': , 'gettext': , 'hmac': , 'imp': , 'inspect': , 'itertools': , 'linecache': , 'locale': , 'math': , 'md5': , 'mimetools': , 'opcode': , 'operator': , 'optparse': , 'os': , 'os.path': , 'posix': , 'posixpath': , 'pprint': , 'pysqlite2': , 'pysqlite2.sqlite': , 'pysqlite2.datetime': None, 'pysqlite2.dbapi2': , 'pysqlite2.pysqlite2': None, 'pysqlite2.time': None, 'quopri': , 'random': , 're': , 'rfc822': , 'sets': , 'settings': , 'shutil': , 'signal': , 'site': , 'smtplib': , 'socket': , 'sre': , 'sre_compile': , 'sre_constants': , 'sre_parse': , 'stat': , 'string': , 'strop': , 'struct': , 'sys': , 'tempfile': , 'textwrap': , 'thread': , 'threading': , 'time': , 'token': , 'tokenize': , 'traceback': , 'types': , 'urllib': , 'uu': , 'warnings': , 'weakref': , 'zipimport': }
```

Generic Views

```
from django.conf.urls.defaults import *
```

```
urlpatterns = patterns("",  
    (r'^polls/$', 'cs683.polls.views.index'),  
)
```

```
from django.conf.urls.defaults import *  
from cs683.polls.models import Poll
```

```
info_dict = {  
    'queryset': Poll.objects.all(),  
}
```

```
urlpatterns = patterns("",  
    (r'^$', 'django.views.generic.list_detail.object_list', info_dict),
```

Template Inheritance

cs683/templates/polls/pollBase.html

```
<html>
  <head>
    <title>
      CS 683 Poll Project
    </title>
  </head>
<body>
  <h1>SDSU CS683 Polls</h1>
  {% block pageContent %}
  {% endblock %}
</body>
```

cs683/templates/polls/pollBase.html

```
{% extends "polls/pollBase.html" %}

{% block pageContent %}

{% if object_list %}
  <ul>
    {% for poll in object_list %}
      <li><a href="{{poll.id}}/details.html">
        {{ poll.question }}</a></li>
    {% endfor %}
  </ul>
{% else %}
  <p>No polls are available.</p>
{% endif %}

{% endblock %}
```

What is (?P<poll_id>\d+)

```
urlpatterns = patterns("
    (r'^polls/$', 'cs683.polls.views.index'),
    (r'^polls/(?P<poll_id>\d+)/detail.html$', 'cs683.polls.views.detail'),
```

What is (?P<poll_id>\d+)

http://127.0.0.1:8000/polls/



(r'^polls/\$', 'cs683.polls.views.index')

cs683.polls.views.index(request)



```
def index(request):  
    poll_list = Poll.objects.all().order_by('-pub_date')  
    return render_to_response('polls/poll_list.html', {'object_list': poll_list})
```

cs683/template/polls/poll_list.html

click on first poll



```
{% if object_list %}  
<ul>  
    {% for poll in object_list %}  
        <li><a href="{{poll.id}}/details.html">{{ poll.question }}</a></li>  
    {% endfor %}  
</ul>
```

http://127.0.0.1:8000/polls/1/details.html

What is (?P<poll_id>\d+)

http://127.0.0.1:8000/polls/1/details.html



```
(r'^polls/(?P<poll_id>\d+)/detail.html$', 'cs683.polls.views.detail'),
```

cs683.polls.views.detail(request, poll_id=1)



```
from django.shortcuts import get_object_or_404, render_to_response
```

```
detail(request, poll_id):
```

```
    poll = get_object_or_404(Poll, pk=poll_id)
```

```
    return render_to_response('polls/poll_detail.html', {'poll': poll})
```

cs683/template/polls/poll_detail.html

cs683/template/polls/poll_detail.html

```
{% extends "polls/pollBase.html" %}
```

```
{% block pageContent %}
```

```
<h1>{{ poll.question }}</h1>
```

```
{% if error_message %}<p><strong>{{ error_message }}</strong></p>{% endif %}
```

```
<form action="/polls/{{ poll.id }}/vote/" method="post">
```

```
{% for choice in poll.choice_set.all %}
```

```
    <input type="radio" name="choice" id="choice{{ forloop.counter }}" value="{{ choice.id }}" />
```

```
    <label for="choice{{ forloop.counter }}">{{ choice.choice }}</label><br />
```

```
{% endfor %}
```

```
<input type="submit" value="Vote" />
```

```
</form>
```

```
{% endblock %}
```


poll_detail.html - Rendered

SDSU CS683 Polls

Do you like Django Reinhardt's music?

- ☐ Yes
- ☐ No
- ☐ Who is Django Reinhard?

Handling the Form

```
def vote(request, poll_id):
    poll = get_object_or_404(Poll, pk=poll_id)
    try:
        selected_choice = poll.choice_set.get(pk=request.POST['choice'])
    except (KeyError, Choice.DoesNotExist):
        # Redisplay the poll voting form.
        return render_to_response('polls/poll_detail.html', {
            'poll': poll,
            'error_message': "You didn't select a choice.",
        })
    else:
        selected_choice.votes += 1
        selected_choice.save()
        # Always return an HttpResponseRedirect after successfully dealing
        # with POST data. This prevents data from being posted twice if a
        # user hits the Back button.
        return HttpResponseRedirect('/polls/%s/results/' % poll.id)
```

cs683/templates/polls/results.html

```
{% extends "polls/pollBase.html" %}
```

```
{% block pageContent %}
```

```
<h1>{{ poll.question }}</h1>
```

```
<ul>
```

```
{% for choice in poll.choice_set.all %}
```

```
    <li>{{ choice.choice }} -- {{ choice.votes }} vote{{ choice.votes|pluralize }}</li>
```

```
{% endfor %}
```

```
</ul>
```

```
{% endblock %}
```