Django Reference Sheet

Project Setup

Create A Project's Directory Structure
```
django-admin.py startproject projectname
```
(this command needs to be issued in the directory you w ant the project to be in)

Customize the DB and other setting in the *projectname*/settings/main.py file

Initialize the DB
```
django-admin.py init
```
(this command should be issued in your *projectname* directory)

Set your DJANGO_SETTING_MODULE environment variable
Unix:
```
export DJANGO_SETTINGS_MODULE=projectname.settings.main
```
Windows:
```
set DJANGO_SETTINGS_MODULE=myproject.settings.main
```
(you'll probably w ant to set these permanently in your ~/.bashrc file in Unix or your my computer properties in Window s)

More Details: http://www.djangoproject.com/documentation/tutorial1/

App Setup

Create an App's Directory Structure
```
django-admin.py startapp appname
```
(this command should be issued in your *projectname/apps* directory)

Setup App's models in the *projectname*/apps/*appname*/models/*appname*.py file

Add the App to the INSTALLED_APPS in *projectname*/settings/main.py

Install the App in the DB
```
django-admin.py install appname
```
(this command should be issued in your *projectname* directory)

More Details:
http://www.djangoproject.com/documentation/tutorial1/#creating-models

Django Admin SQL Commands

All the following commands output sql specific to the *appname* specified in the command and should be run from with the *projectname* directory.

```
django-admin.py sqlinitialdata appname
```
Outputs the initial-data inserts required for Django's admin framework.

```
django-admin.py sql appname
```
Outputs the table creation statements

```
django-admin.py sqlall appname
```
Outputs both the sqlinitialdata and sql data

```
django-admin.py sqlindexes appname
```
Outputs the the create index statements

```
django-admin.py sqlclear appname
```
Outputs the the drop table statements and the delete statements for deleting the "initialdata".

```
django-admin.py startapp appname
```
(this should be issued in your *projectname/apps* directory)

More Details:
http://www.djangoproject.com/documentation/tutorial1/#activating-models

Template Syntax

There are two types of syntax:
`{% something %}` - For tags which include blocks and functional items like loops, if statements, and includes.

`{{ variable }}` - For variable insertion where "variable" is replaced with it's value.

URL Dispatcher Sample Config Files

Sample *projectname*/settings/urls/main.py file
```
from django.conf.urls.defaults import *
urlpatterns = patterns('',
    (r'^weblog/',        include('django_website.apps.blog.urls.blog')),
    (r'^documentation/', include('django_website.apps.docs.urls.docs')),
    (r'^comments/',      include('django.contrib.comments.urls.comments')),
    (r'',                include('django.conf.urls.flatfiles')),
)
```

Sample *projectname*/apps/blog/urls/main.py file
```
from django.conf.urls.defaults import *
info_dict = {
    'app_label': 'blog',
    'module_name': 'entries',
}
urlpatterns = patterns('django.views.generic.date_based.',
    (r'^(?P<year>\d{4})/(?P<month>[a-z]{3})/(?P<day>\w{1,2})/$', 'archive_day', info_dict),
    (r'^(?P<year>\d{4})/(?P<month>[a-z]{3})/$', 'archive_month', info_dict),
    (r'^(?P<year>\d{4})/$', 'archive_year', info_dict),
    (r'^/?$', 'archive_index', info_dict),
)
```

Italicized Text – Placeholder names for your object(s) • `Courier Text` – Commands you type in

add - Adds the arg to the value
addslashes - Adds slashes, for passing strings to JavaScript.
Capfirst - Capitalizes the first character of the value
center - Centers the value in a field of a given width
cut - Removes all values of arg from the given string
date - Formats a dateto the given format using "now" format
default - If value is unavailable, use given default
dictsort - Takes a list of dicts, returns that list sorted by the property given in the argument.
dictsortreversed - Takes a list of dicts, returns that list sorted in reverse order by the property given in the argument.
divisibleby - Returns true if the value is divisible by the argument
escape - Escapes a string's HTML
filesizeformat - Format the value like a 'human-readable' file.
First - Returns the first item in a list
fix_ampersands - Replaces ampersands with & entities
floatformat - Displays a floating point number as 34.2 (with one decimal places) - but only if there is a decimal point.
get_digit - Given a whole number, returns the requested digit of it, where 1 is the right-most digit, 2 is the second-right-most digit, etc. Returns the original value for invalid input,
join - Joins a list with a string, like Python's str.join(list)
length - Returns the length of the value - useful for lists
length_is - Returns a boolean of whether the value's length is the argument
linebreaks - Converts newlines into <p> and
s
linebreaksbr - Converts newlines into
s
linenumbers - Displays text with line numbers
ljust - Left-aligns the value in a field of a given width, Argument: field size
lower - Converts a string into all lowercase
make_list - Returns the value turned into a list.
phone2numeric - Takes a phone number and converts it in to its numerical equivalent
pluralize - Returns 's' if the value is not 1.
pprint - A wrapper around pprint.pprint -- for debugging.

random - Returns a random item from the list
removetags - Removes a space separated list of [X]HTML tags from the output
rjust - Right-aligns the value in a field of a given width, Argument: field size
slice - Returns a slice of the list.
slugify - Converts to lowercase, removes non-alpha chars and converts spaces to hyphens
stringformat - Formats the variable according to the argument, a string formatting specifier. This specifier uses Python string formating syntax, with the exception that the leading "%" is dropped.
striptags - Strips all [X]HTML tags
time - Formats a time according to the given format (same as the now tag).
timesince - Formats a date as the time since that date (i.e. "4 days, 6 hours")
title - Converts a string into titlecase
truncatewords - Truncates a string after a certain number of words. Argument: Number of words to truncate after.
unordered_list - Recursively takes a self-nested list and returns an HTML unordered list -- WITHOUT opening and closing <ul> tags.
upper - Converts a string into all uppercase
urlencode - Escapes a value for use in a URL
urlize - Converts URLs in plain text into clickable links
urlizetrunc - Converts URLs into clickable links, truncating URLs to the given character limit. Argument: Length to truncate URLs to.
wordcount - Returns the number of words
wordwrap - Wraps words at specified line length. Argument: number of words to wrap the text at.
yesno - Given a string mapping values for true, false and (optionally) None, returns one of those strings according to the value. Sample argument "yeah,no,maybe" returns yeah for true, no for false, and maybe for none.

Template Tag Reference

block – Defines an area that the content can be inserted into.
`{% block content %}Default Content{% endblock %}`
comment - `{% comment %}{% endcomment %}`
cycle – Rotates through items
`<tr class="{% cycle red,blue as colors %}">..</tr>`
`<tr class="{% cycle colors %}">...</tr>`
debug – Outputs a bunch of debug info `{% debug %}`
extends – Inherits from another template
`{% extends "base" %}`
(leave off the .html for the template name)
filter – Applies the specified filter(s) to the content in between
`{% filter lower %}Lowercase this{% endfilter %}`
firstOf – Outputs the first true variable, or nothing if all are false.
`{% firstof var1 var2 var3 %}`
for – Loop through list item
`{% for athlete in athlete_list %}`
`    {{ athlete.name }}`
`{% endfor %}`
forloop.counter = current loop iteration starting at 1
forloop.counter0 = current loop iteration starting at 0
forloop.first = True if first time through loop
forloop.last = True if this is the list time through the loop
forloop.parentloop = Parentloop counter (???)

if – Evaluates for true or false
`{% if athlete %}{{ athlete }}{% endif %}`
(if statements can have "not" and "or", but are not allow ed to have "and")
ifchanged – Outputs it's content if different from last loop.
`{% ifchanged %}{{ name }}{% endifchanged %}`
(code w ould be w ithin for loop)
ifnotequal – Checks for equality
`{% ifnotequal v1 v2 %}hi{% endifnotequal %}`
load – loads a custom tag set `{% load comments %}`
now – Outputs current date `{% now "jS F Y H:i" %}`
(formatting matches php's date function for the most part)
regroup –Regroup a list of like objects by an attribute
`{% regroup people by gender as grouped %}`
ssi – Includes a file
`{% ssi /home/html/side.html parsed %}`
(if the option passed "parsed it w ill be treated as a template)
templatetag – Used to escape template tags
`{% templatetag openblock %}`
(arguments open/closeblock and open/closevariable)
widthratio – Calculates the ratio of a given value to a maximum value, and then applies that ratio to a constant.
`{% widthratio this_value max_value 100 %}`

Italicized Text – Placeholder names for your object(s)   •   `Courier Text` – Commands you type in

Sample Template

```
{% extends "base" %}
{% block title %}{{ title|title }}{% endblock %}
{% block intro %}
    Default Intro Copy that can be replaced by templates extending this one
{% endblock %}
{% block content %}
   {% for entry in blog_entries %}
      {% ifchanged %}<h1>{{ entry.pub_date|date"F, y" }}</h1>{% endifchanged %}
      <h2>{{ entry.title }}</h2> <p>
      {{ entry.body|escape|urlizetrunc:"40"|linebreaks}}</p>
   {% endfor %}

{% endblock %}
```

Additional Template Resources:
http://www.djangoproject.com/documentation/templates/
http://www.djangoproject.com/documentation/tutorial3/

General Django Resources:
Django Website - http://www.djangoproject.com/
Django User Group/Mailing List - http://groups-beta.google.com/group/django-users
Django Developers Group/Mailing List - http://groups-beta.google.com/group/django-developers
Django IRC - irc://irc.freenode.net/django
IRC Logs - http://loglibrary.com/show_page/latest/179

Most of the Content in this reference sheet was gathered from the http://www.djangoproject.com/ website.

Document Version: .1.1
The current document will always be available at: http://www.dobbes.com/

License:
Creative Commons Attribution-NonCommercial-ShareAlike 2.5 License
http://creativecommons.org/licenses/by-nc-sa/2.5/

Italicized Text – Placeholder names for your object(s)   •   `Courier Text` – Commands you type in