Developing Websites using Python and Django

Author: Dr. S. Mukherjee

Date: February 2015

Preface

This is a step-by-step documentation of hands-on development of python projects specially aimed for online interface. This document in addition to development, it also assembles all the necessary installation and access to various software and environments to make the development more efficient.

Table of Contents

Developing Websites using Python and Django

Getting Started - Basic setup:

Assuming Python 2.7 installed the steps are given below. On console:

```
> python –version
Python 2.7

> pip install –U django==1.5.4
[apparently uninstalled Django 1.7 and installed Django 1.5.4]

> python -c "import django; print(django.get_version())"
1.5.4
```

```
c:\Users\SM\Desktop\Activities 2015\Research\Django> python
c:\python27\scripts\django-admin.py startproject mytango_proj

c:\Users\SM\Desktop\Activities 2015\Research\Django> cd my*

...\mytango_proj> python manage.py runserver
Validating models...

0 errors found
February 04, 2015 - 22:02:09
Django version 1.5.4, using settings 'mytango_proj.settings'
Development server is running at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
[04/Feb/2015 22:05:53] "GET / HTTP/1.1" 200 1963

... \mytango_proj> python manage.py runserver 192.168.0.104:5555
Validating models...

0 errors found
February 04, 2015 - 22:27:37
Django version 1.5.4, using settings 'mytango_proj.settings'
Development server is running at http://192.168.0.104:5555/
Quit the server with CTRL-BREAK.
```

Using computer IP and appropriate port we can open even on another computer browser `http://192.168.0.104:5555/` Let us now create an app called Gumbo within the project

```
... \mytango_proj> python manage.py startapp Gumbo
```

Now first tell my Django project about the new application's existence. To do this, need to modify the settings.py file, in the project's configuration directory. In the file found the INSTALLED_APPS tuple. Added the Gumbo application to the end of the tuple, which then looked like the following example:

```
INSTALLED_APPS = (
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.sites',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    # Uncomment the next line to enable the admin:
    # 'django.contrib.admin',
    # Uncomment the next line to enable admin documentation:
    # 'django.contrib.admindocs',
    'Gumbo',
)
```

Next edit view.py file in the Gumbo directory

```
# done by SM on 04.02.2015:
from django.http import HttpResponse

def index(request):
    return HttpResponse("Gumbo says: Bark! Bark! hello world!")
```

With the view created, it is only part of the way to allowing a user to access it. For a user to see this view, we must map a Uniform Resources Locator (URL) to the view as below:

Within the Gumbo application directory, we now need to create a new file called urls.py. The contents of the file will allow to map URLs for my application to specific views. A simple urls.py file is below.

```
from django.conf.urls import patterns, url
from Gumbo import views
urlpatterns = patterns('',
        url(r'^$', views.index, name='index'))
```

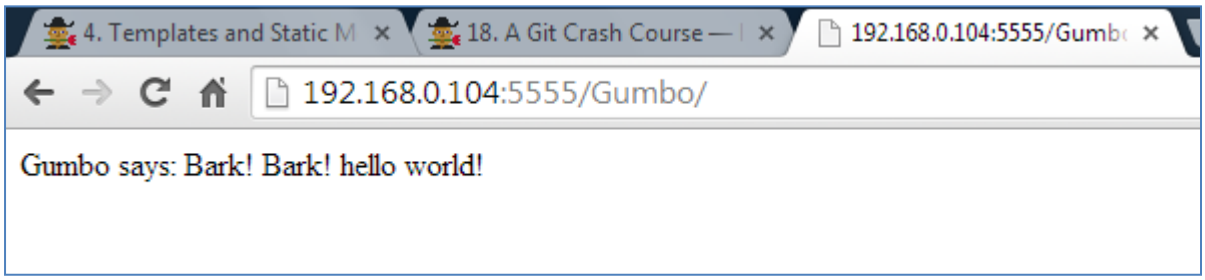
Open the project's urls.py file which is located inside Gumbo directory. As a relative path from your workspace directory, this would be the file. Update the urlpatterns tuple to include on the last line:

```
url(r'^Gumbo/', include('Gumbo.urls')), # SM IS ADDING THIS NEW TUPLE!
```

Now run the server again:

```
... \mytango_proj> python manage.py runserver 192.168.0.104:5555
```

Open [even on another computer] the following address in a browser `http://192.168.0.104:5555/Gumbo`



See reference in http://www.tangowithdjango.com/book/chapters/setup.html

Adding Database:

Edit the settings.py in the project folder to incorporate the database:

```
import os.path
...
BASE_DIR = 'C:\Users\SM\Desktop\Activities 2015\Research\Django\mytango_proj'
DATABASE_PATH=os.path.join(BASE_DIR, 'Gumbo.db')
...
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.sqlite3', # Added 'sqlite3'.
        'NAME': DATABASE_PATH,  # also defined the path at the top.
        }
}
...
TIME_ZONE = 'Asia/Kolkata'  #changed from 'America/Chicago' - optional
```

Then execute the command below for creation & syncing databases

```
... \mytango_proj> python manage.py syncdb
Creating tables ...
Creating table auth_permission
Creating table auth_group_permissions
Creating table auth_group
Creating table auth_user_groups
Creating table auth_user_user_permissions
Creating table auth_user
Creating table django_content_type
Creating table django_session
Creating table django_site

You just installed Django's auth system, which means you don't have any superusers defined.
Would you like to create one now? (yes/no): yes
Username (leave blank to use 'sm'): Saurabh
Email address: stat.mukherjee@gmail.com
Password:[smartgumbo]
Password (again): [smartgumbo]
Superuser created successfully.
Installing custom SQL ...
Installing indexes ...
Installed 0 object(s) from 0 fixture(s)
```

A database called *Gumbo.db* is now created at the project folder. Then make the following edits to models.py in Gumbo folder to create my poll model.

```
from django.db import models

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


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

Then issue the command:

```
... \mytango_proj> python manage.py sql Gumbo
BEGIN;
...
COMMIT;
```

Then uncomment the admin app 'django.contrib.admin', (under INSTALLED_APPS= in settings.py) and run syncdb command again. The table *django_admin_log* is created.

Once done, opened project's urls.py file. This was created in the project configuration directory. Within the file, first uncomment the two lines after the import statements. We also need to ensure that the URL pattern for /admin/ is present, and points to the admin.site.urls module as shown below.

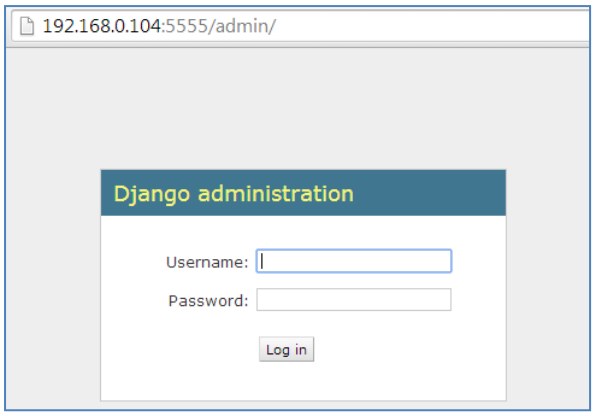
```
from django.conf.urls import patterns, include, url
from django.conf import settings

# Uncomment the next two lines to enable the admin:
from django.contrib import admin # UNCOMMENT THIS LINE
admin.autodiscover() # UNCOMMENT THIS LINE, TOO!

urlpatterns = patterns('',
        url(r'^Gumbo/', include('Gumbo.urls')),
        url(r'^admin/', include(admin.site.urls)), # ADD THIS LINE
        )
```

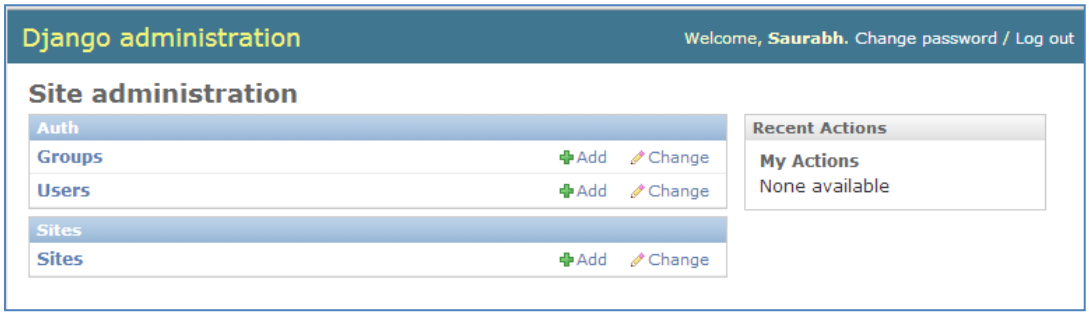
Then we can run server again:

```
...\mytango_proj>python manage.py runserver 192.168.0.104:5555
Validating models...

0 errors found
February 05, 2015 - 12:35:33
Django version 1.5.4, using settings 'mytango_proj.settings'
Development server is running at http://192.168.0.104:5555/
Quit the server with CTRL-BREAK.
```

Now opened the following address on my ipad safari browser http://192.168.0.104:5555/admin to see on my command prmpt and broser as follow:

```
[05/Feb/2015 12:35:53] "GET /Gumbo/ HTTP/1.1" 200 36
[05/Feb/2015 12:36:07] "GET /admin HTTP/1.1" 301 0
[05/Feb/2015 12:36:16] "GET /admin/ HTTP/1.1" 200 1882
[05/Feb/2015 12:36:46] "POST /admin/ HTTP/1.1" 302 0
[05/Feb/2015 12:36:47] "GET /admin/ HTTP/1.1" 200 3299
```

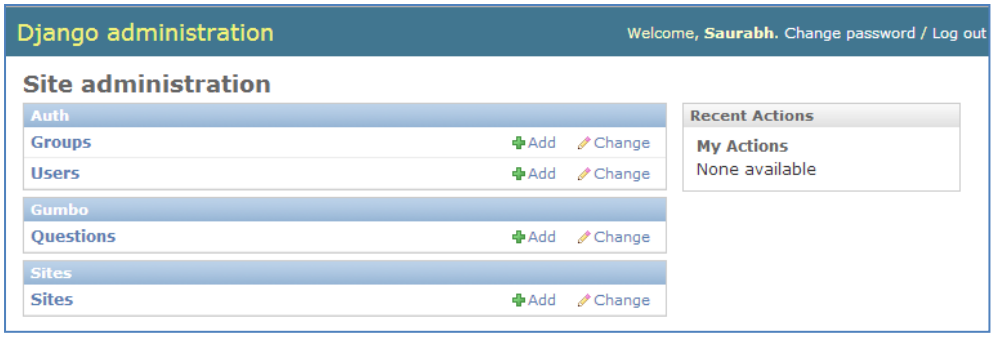


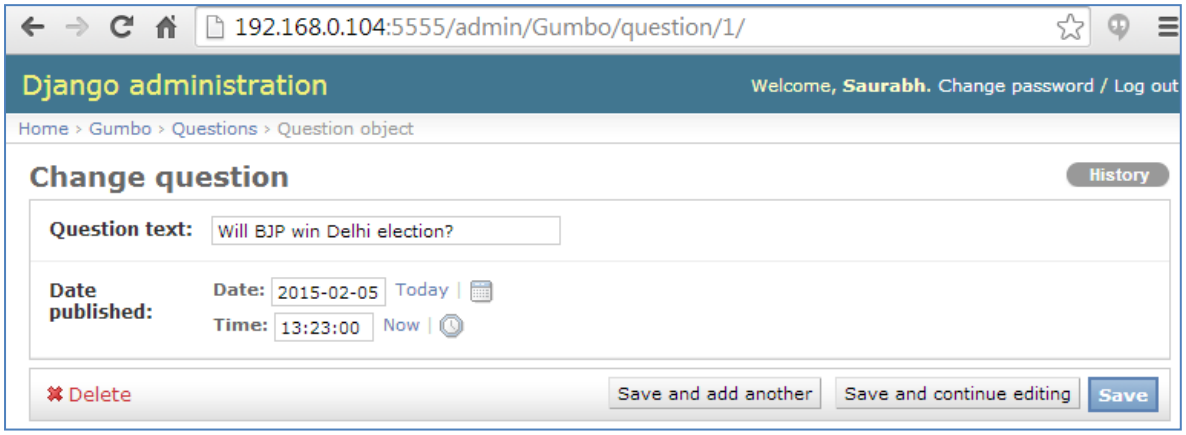


To enhance admin interface we create a new python file in `Gumbo` application directory called `admin.py` and added the following code to the new file.

```
from django.contrib import admin
from Gumbo.models import Question

admin.site.register(Question)
```

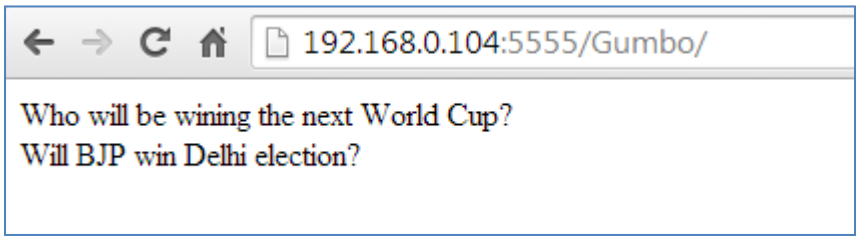
This will *register* the models with the admin interface. If we were to have another model, it would be a trivial case of calling the `admin.site.register()` function, passing the model in as a parameter!





Finally, to get a simple non-admin view of these questions we can revise the views.py in Gumbo folder as below:

```
#def index(request):
#    return HttpResponse("Gumbo says: Bark! Bark! hello world!")

# redfining view to see question database  1st by importing the question class
from Gumbo.models import Question
def index(request):
    qs_list = Question.objects.order_by('-pub_date')[:5]
    output = '<br/>'.join([p.question_text for p in qs_list])
    return HttpResponse(output)
```



Reference: https://docs.djangoproject.com/en/1.7/intro/tutorial01/

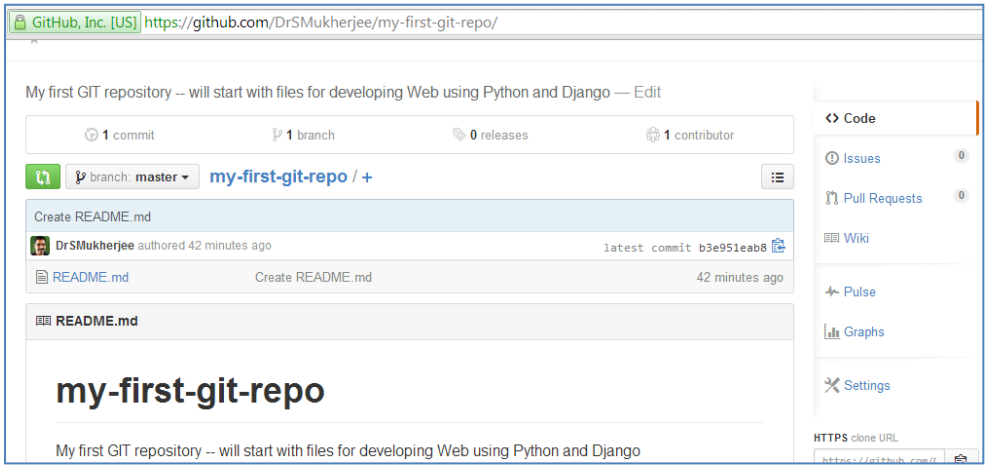
GitHub Set Up

The coding is getting complicated – as I need to change back and forth codes on existing files. So, this is the right time to set-up a version control system. Using GIT for that.

Downlaoded and installed GIT for windows. Already had set-up account.

Username: DrSMukherjee (using my gmail which is made private and so GIT will be using DrSMukherjee@users.noreply.github.com)

Create a new repository:



While logged in on the Github online, go to the shell prompt and use the following to create a clone on desktop

```
SM@TROY /
$ cd ~/Desktop

SM@TROY ~/Desktop
$ cd Acti*

SM@TROY ~/Desktop/Activities 2015
$ cd Res*

SM@TROY ~/Desktop/Activities 2015/Research
$ mkdir SMGit

SM@TROY ~/Desktop/Activities 2015/Research
$ cd SMG*

SM@TROY ~/Desktop/Activities 2015/Research/SMGit
$ git clone https://github.com/DrSMukherjee/my-first-git-repo.git
Cloning into 'my-first-git-repo'...
remote: Counting objects: 3, done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0)
Unpacking objects: 100% (3/3), done.
Checking connectivity... done.

SM@TROY ~/Desktop/Activities 2015/Research/SMGit
$
```

Then

1. Made some changes on the README.md file and pushed it onto GitHub
2. Copied the content of the django project folder into the clone folder SMGit then ***pushed*** the entire structure onto GitHub
3. Remove the .pyc files online at GitHub repository (in Gumbo folder) and then ***pulled*** those changes locally
4. Remove the .pyc files online at GitHub repository (in mytango_proj folder) and then ***pushed*** those changes (removal) online at GitHub

The log of the command prompt is saved for reference.

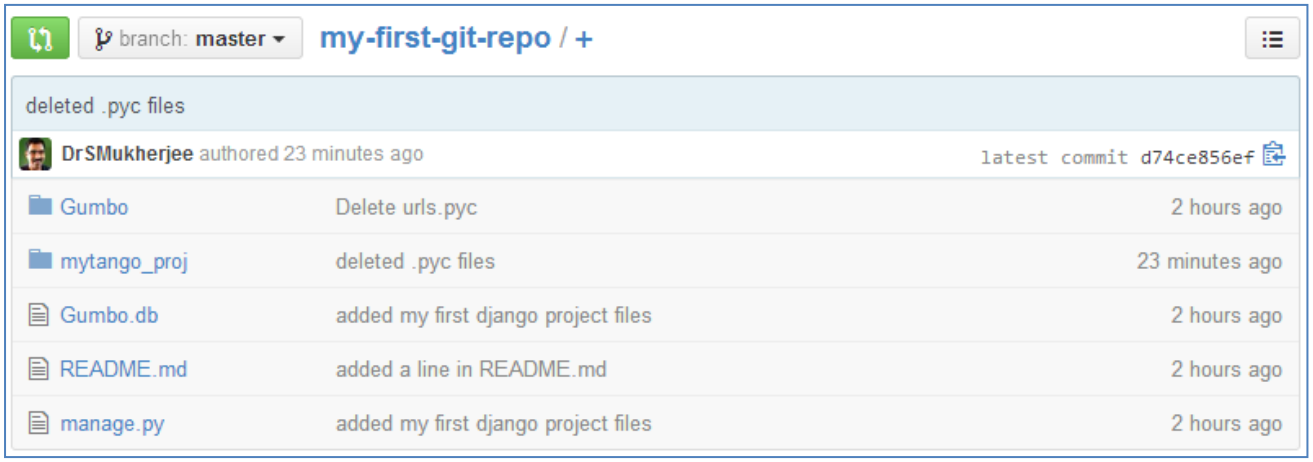
```
        modified:   README.md

no changes added to commit (use "git add" and/or "git commit -a")

SM@TROY ~/Desktop/Activities 2015/Research/SMGit/my-first-git-repo (master)
$ git add README.md

SM@TROY ~/Desktop/Activities 2015/Research/SMGit/my-first-git-repo (master)
$ git status
On branch master
Your branch is up-to-date with 'origin/master'.

Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        modified:   README.md


SM@TROY ~/Desktop/Activities 2015/Research/SMGit/my-first-git-repo (master)
$ git commit -m "added a line in README.md"
[master 8d08007] added a line in README.md
 1 file changed, 1 insertion(+)

SM@TROY ~/Desktop/Activities 2015/Research/SMGit/my-first-git-repo (master)
$ git push origin master
Username for 'https://github.com': DrSMukherjee
Password for 'https://DrSMukherjee@github.com':
Counting objects: 5, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 321 bytes | 0 bytes/s, done.
Total 3 (delta 1), reused 0 (delta 0)
To https://github.com/DrSMukherjee/my-first-git-repo.git
   b3e951e..8d08007  master -> master

SM@TROY ~/Desktop/Activities 2015/Research/SMGit/my-first-git-repo (master)
$ git status
On branch master
Your branch is up-to-date with 'origin/master'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        Gumbo.db
        Gumbo/
        manage.py
        mytango_proj/

nothing added to commit but untracked files present (use "git add" to track)

SM@TROY ~/Desktop/Activities 2015/Research/SMGit/my-first-git-repo (master)
$ git add .
warning: LF will be replaced by CRLF in Gumbo/models.py.
```

```
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in Gumbo/tests.py.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in Gumbo/views.py.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in manage.py.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in mytango_proj/settings.py.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in mytango_proj/urls.py.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in mytango_proj/wsgi.py.
The file will have its original line endings in your working directory.

SM@TROY ~/Desktop/Activities 2015/Research/SMGit/my-first-git-repo (master)
$ git push origin master
Username for 'https://github.com': DrSMukherjee
Password for 'https://DrSMukherjee@github.com':
Everything up-to-date

SM@TROY ~/Desktop/Activities 2015/Research/SMGit/my-first-git-repo (master)
$ git status
On branch master
Your branch is up-to-date with 'origin/master'.

Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        new file:   Gumbo.db
        new file:   Gumbo/__init__.py
        new file:   Gumbo/__init__.pyc
        new file:   Gumbo/admin.py
        new file:   Gumbo/admin.pyc
        new file:   Gumbo/models.py
        new file:   Gumbo/models.pyc
        new file:   Gumbo/tests.py
        new file:   Gumbo/urls.py
        new file:   Gumbo/urls.pyc
        new file:   Gumbo/views.py
        new file:   Gumbo/views.pyc
        new file:   manage.py
        new file:   mytango_proj/__init__.py
        new file:   mytango_proj/__init__.pyc
        new file:   mytango_proj/settings.py
        new file:   mytango_proj/settings.pyc
        new file:   mytango_proj/urls.py
        new file:   mytango_proj/urls.pyc
        new file:   mytango_proj/wsgi.py
        new file:   mytango_proj/wsgi.pyc


SM@TROY ~/Desktop/Activities 2015/Research/SMGit/my-first-git-repo (master)
$ git commit -m "added my first django project files"
[master 52fe718] added my first django project files
warning: LF will be replaced by CRLF in Gumbo/models.py.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in Gumbo/tests.py.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in Gumbo/views.py.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in manage.py.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in mytango_proj/settings.py.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in mytango_proj/urls.py.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in mytango_proj/wsgi.py.
```

```
The file will have its original line endings in your working directory.
 21 files changed, 276 insertions(+)
 create mode 100644 Gumbo.db
 create mode 100644 Gumbo/__init__.py
 create mode 100644 Gumbo/__init__.pyc
 create mode 100644 Gumbo/admin.py
 create mode 100644 Gumbo/admin.pyc
 create mode 100644 Gumbo/models.py
 create mode 100644 Gumbo/models.pyc
 create mode 100644 Gumbo/tests.py
 create mode 100644 Gumbo/urls.py
 create mode 100644 Gumbo/urls.pyc
 create mode 100644 Gumbo/views.py
 create mode 100644 Gumbo/views.pyc
 create mode 100644 manage.py
 create mode 100644 mytango_proj/__init__.py
 create mode 100644 mytango_proj/__init__.pyc
 create mode 100644 mytango_proj/settings.py
 create mode 100644 mytango_proj/settings.pyc
 create mode 100644 mytango_proj/urls.py
 create mode 100644 mytango_proj/urls.pyc
 create mode 100644 mytango_proj/wsgi.py
 create mode 100644 mytango_proj/wsgi.pyc

SM@TROY ~/Desktop/Activities 2015/Research/SMGit/my-first-git-repo (master)
$ git push origin master
Username for 'https://github.com': DrSMukherjee
Password for 'https://DrSMukherjee@github.com':
Counting objects: 25, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (23/23), done.
Writing objects: 100% (24/24), 13.01 KiB | 0 bytes/s, done.
Total 24 (delta 2), reused 0 (delta 0)
To https://github.com/DrSMukherjee/my-first-git-repo.git
   8d08007..52fe718  master -> master

SM@TROY ~/Desktop/Activities 2015/Research/SMGit/my-first-git-repo (master)
$ git pull origin master
remote: Counting objects: 12, done.
remote: Compressing objects: 100% (12/12), done.
remote: Total 12 (delta 8), reused 0 (delta 0)
Unpacking objects: 100% (12/12), done.
From https://github.com/DrSMukherjee/my-first-git-repo
 * branch            master     -> FETCH_HEAD
   52fe718..889c92c  master     -> origin/master
Updating 52fe718..889c92c
Fast-forward
 Gumbo/admin.pyc  | Bin 324 -> 0 bytes
 Gumbo/models.pyc | Bin 981 -> 0 bytes
 Gumbo/urls.pyc   | Bin 386 -> 0 bytes
 Gumbo/views.pyc  | Bin 662 -> 0 bytes
 4 files changed, 0 insertions(+), 0 deletions(-)
 delete mode 100644 Gumbo/admin.pyc
 delete mode 100644 Gumbo/models.pyc
 delete mode 100644 Gumbo/urls.pyc
 delete mode 100644 Gumbo/views.pyc

SM@TROY ~/Desktop/Activities 2015/Research/SMGit/my-first-git-repo (master)
$ git status
On branch master
Your branch is up-to-date with 'origin/master'.

Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        deleted:    mytango_proj/__init__.pyc
```

```
        deleted:    mytango_proj/settings.pyc
        deleted:    mytango_proj/urls.pyc
        deleted:    mytango_proj/wsgi.pyc

no changes added to commit (use "git add" and/or "git commit -a")

SM@TROY ~/Desktop/Activities 2015/Research/SMGit/my-first-git-repo (master)
$ git add --all

SM@TROY ~/Desktop/Activities 2015/Research/SMGit/my-first-git-repo (master)
$ git commit -m "deleted .pyc files"
[master d74ce85] deleted .pyc files
 4 files changed, 0 insertions(+), 0 deletions(-)
 delete mode 100644 mytango_proj/__init__.pyc
 delete mode 100644 mytango_proj/settings.pyc
 delete mode 100644 mytango_proj/urls.pyc
 delete mode 100644 mytango_proj/wsgi.pyc

SM@TROY ~/Desktop/Activities 2015/Research/SMGit/my-first-git-repo (master)
$ git push origin master
Username for 'https://github.com': DrSMukherjee
Password for 'https://DrSMukherjee@github.com':
Counting objects: 5, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 298 bytes | 0 bytes/s, done.
Total 3 (delta 2), reused 0 (delta 0)
To https://github.com/DrSMukherjee/my-first-git-repo.git
   889c92c..d74ce85  master -> master

SM@TROY ~/Desktop/Activities 2015/Research/SMGit/my-first-git-repo (master)
$ ls *
Gumbo.db  README.md  manage.py

Gumbo:
__init__.py  __init__.pyc  admin.py  models.py  tests.py  urls.py  views.py

mytango_proj:
__init__.py  settings.py  urls.py  wsgi.py
```

Latest view at GitHub



Creating and Rendering Graphics

Now added a dynamically created plot on the website:

Basic steps involved editing just two file –

~\Gumbo\views.py: (added 'graph' and imported necessary modules)

```
import StringIO
import PIL
import PIL.Image
from django.http import HttpResponse
from matplotlib import pylab as pl
from Gumbo.models import Question

def index(request):

    qs_list = Question.objects.order_by('-pub_date')[:5]

    output = '<br/>'.join([p.question_text for p in qs_list])
    return HttpResponse(output)

def graph(request):
    x = [1, 2, 3, 4, 5, 6, 7]
    y = [1, 4, 9, 16, 25, 36, 49]
    pl.plot(x, y, linewidth=2)
    pl.xlabel("x axis")
    pl.ylabel("y axis")
    pl.grid(False)
    pl.title("Sample graph")
    # pl.show()

    buff = StringIO.StringIO()
    canvas = pl.get_current_fig_manager().canvas
    canvas.draw()
    graphIMG = PIL.Image.frombytes('RGB', canvas.get width height(), canvas.tostring rgb())
    graphIMG.save(buff, "PNG")
    pl.close()

    return HttpResponse(buff.getvalue(), mimetype="image/png")
```

~\Gumbo\url.py: (added the 2nd url statement)

```
from django.conf.urls import patterns, url
from Gumbo import views

urlpatterns = patterns('',
        url(r'^$', views.index, name='index'),

        url(r'^my graph$', views.graph, name='graph'),
        )
```

Note: using PyCharm IDE – which is making huge differences ☺