Paths

* DjangoPlay/mysite – root directory
* Manage.py cmd utility for interacting with project
* Mysite/mysite – actual python project, import anything in here (ex. Mysite.urls)
* Mysite/\_\_init\_\_.py – empty file, tells python that directory should be considered a python project
* Mysite/settings.py – settings/configurations
* Mysite/urls.py – url declarations, table of contents
* Mysite/wsgi.py – entry point for wsgi compatible web servers

Commands

* Django-admin startproject <project\_name>
* Python manage.py runserver // <http://127.0.0.1:8000/>
* Python manage.py startapp <app name> // create an app, in same directory as manage.py
* Ctrl – c // break
* <http://127.0.0.1:8000/polls/> // polls page
* Python manage.py migrate // looks at INSTALLED\_APPS and creates any necessary database tables according to the database settings in settings.py
* Python manage.py makemigrations polls //store changes in models as a migration
* Python manage.py sqlmigrate polls 0001 // takes migration names and returns SQL
* Python manage.py shell // use this instead of python because it uses the proper paths

Other

* In mysite/settings.py, INSTALLED\_APPS
  + [**django.contrib.admin**](https://docs.djangoproject.com/en/2.0/ref/contrib/admin/#module-django.contrib.admin) – The admin site. You’ll use it shortly.
  + [**django.contrib.auth**](https://docs.djangoproject.com/en/2.0/topics/auth/#module-django.contrib.auth) – An authentication system.
  + [**django.contrib.contenttypes**](https://docs.djangoproject.com/en/2.0/ref/contrib/contenttypes/#module-django.contrib.contenttypes) – A framework for content types.
  + [**django.contrib.sessions**](https://docs.djangoproject.com/en/2.0/topics/http/sessions/#module-django.contrib.sessions) – A session framework.
  + [**django.contrib.messages**](https://docs.djangoproject.com/en/2.0/ref/contrib/messages/#module-django.contrib.messages) – A messaging framework.
  + [**django.contrib.staticfiles**](https://docs.djangoproject.com/en/2.0/ref/contrib/staticfiles/#module-django.contrib.staticfiles) – A framework for managing static files.
* View: type of web page that serves a specific function. Example: blog homepage, index page, detail page, results page
* URLconfs: maps url patterns to veiws