



Django Level I - Intro... 74%
(/m/7/3832)

Intro and Overview (/m/

Django as MVC (/m/

Installing Django (/m/

Big Picture (/m/

Getting Started (/m/

Routes and Controllers (/m/

Views vs Templates (/m/

Templates and Static Files (/m/

Time Display (/m/

Session Data and Forms (/m/

Random Word Generator (/m/

Quiz (/m/

Survey Form (/m/

Context Objects Re-explained (/m/

Route Parameters (/m/

Quiz II (/m/

Django setup summary sh... (/m/

Disappearing Ninjas

Django Ninja Gold

Django Level II - App... 0%
(/m/7/3834)

Django Extras • 0%
(/m/7/3843)

Django Level III - Ad... 0%
(/m/7/3844)

Here is a crib sheet of the process of creating a Django application from scratch. This is a summary learned up to this point. It's a good idea to be very familiar with this process, and this sheet is what you've gotten a good grasp of the concepts they describe.

Install

macOS:

```
> virtualenv djangoEnv
> source djangoEnv/bin/activate
(djangoEnv)>pip install django
```

PC:

```
> python -m virtualenv djangoEnv
> call djangoEnv/scripts/activate
(djangoEnv)>pip install django
```

Create project

```
# 'main' is the project name used below
django-admin startproject main
```

The project has been created, inspect the file & folders created as a result:

- Folder: main
 - File: manage.py <= used to run the application.
 - Folder: main <= contains the files specific to our app.
 - File: __init__.py
 - File: settings.py
 - File: urls.py
 - File: wsgi.py

A few more config items to complete the setup:

- Navigate into the main directory that you just created.

```
> cd main
```

- Make a new apps directory

```
> mkdir apps
```

- Navigate into apps

```
> cd apps
```

- Create an __init__.py file. This just let's Python know that this folder is viewable by Python can look in here and access the code!

Bash: (Mac, Linux) =>

```
touch __init__.py
```

PCs: (Command Prompt) =>

```
nul> __init__.py <-- You can disregard the Access Denied message
```

Start the new app.

```
> python ../manage.py startapp first_app
```

The app 'first_app' has been created, let's look at the updated file/folder structure

- Folder: main



BACK TO TRACKS

Django Level I - Intro... 74%
(/m/7/3832)

Intro and Overview (/m/

Django as MVC (/m/

Installing Django (/m/

Big Picture (/m/

Getting Started (/m/

Routes and Controllers (/m/

Views vs Templates (/m/

Templates and Static Files (/m/

Time Display (/m/

Session Data and Forms (/m/

Random Word Generator (/m/

Quiz (/m/

Survey Form (/m/

Context Objects Re-explained (/m/

Route Parameters (/m/

Quiz II (/m/

Django setup summary sh... (/m/

Disappearing Ninjas

Django Ninja Gold

Django Level II - App... 0%
(/m/7/3834)

Django Extras • 0%
(/m/7/3843)

Django Level III - Ad... 0%
(/m/7/3844)

- Folder: apps <= created by our steps above
 - File: __init__.py <= created by our steps above
 - Folder: first_app <= created as a result to the call to 'startapp'
 - File: __init__.py
 - File: admin.py
 - File: apps.py
 - File: models.py <= The 'M' in MVC
 - File: tests.py
 - File: views.py <= The CONTROLLER in our MVC pattern
- File: manage.py <= used to run the application.
- Folder: main <= contains the files specific to our app.
 - File: __init__.py
 - File: settings.py
 - File: urls.py
 - File: wsgi.py

Open the settings.py file and add a line to your INSTALLED_APPS:

```
INSTALLED_APPS = [
    'apps.first_app', ### <= added this line!
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
]
```

Open the urls.py file, add the line to import include and add a line to include all of the url routes routes.

```
from django.conf.urls import url, include # Notice we added include
from django.contrib import admin
urlpatterns = [
    url(r'^', include('apps.first_app.urls')) # And now we use include to pull in our first_app.urls...
]
```

Now, create the urls.py file in our 'first_app' folder so that the final file/folder structure looks like

- Folder: main
 - Folder: apps <= created by our steps above
 - File: __init__.py <= created by our steps above
 - Folder: first_app <= created as a result to the call to 'startapp'
 - File: __init__.py
 - File: admin.py
 - File: apps.py
 - File: models.py <= The 'M' in MVC
 - File: tests.py
 - File: views.py <= The CONTROLLER in our MVC pattern
 - urls.py <= the file that will handle the routes of our app
 - File: manage.py <= used to run the application.
 - Folder: main <= contains the files specific to our app.
 - File: __init__.py
 - File: settings.py
 - File: urls.py
 - File: wsgi.py

Add some routes to our urls.py file!



BACK TO TRACKS

Django Level I - Intro...	74%
(/m/7/3832)	
Intro and Overview	(/m/
Django as MVC	(/m/
Installing Django	(/m/
Big Picture	(/m/
Getting Started	(/m/
Routes and Controllers	(/m/
Views vs Templates	(/m/
Templates and Static Files	(/m/
Time Display	(/m/
Session Data and Forms	(/m/
Random Word Generator	(/m/
Quiz	(/m/
Survey Form	(/m/
Context Objects Re-explained	(/m/
Route Parameters	(/m/
Quiz II	(/m/
Django setup summary sh...	(/m/
Disappearing Ninjas	
Django Ninja Gold	
Django Level II - App..	0%
(/m/7/3834)	
Django Extras	0%
(/m/7/3843)	
Django Level III - Ad...	0%
(/m/7/3844)	

In Django we set up our routes this way:

```
from django.conf.urls import url
from . import views
# ^ So we can call functions from our routes!
urlpatterns = [
    url(r'^$', views.index)
]
```

The Flask equivalent:

```
@app.route('/')
def index():
```

Finally, run the app:

```
cd ..
python manage.py runserver
```

Adding views & static files

- Folder: main
 - Folder: apps <= created by our steps above
 - File: __init__.py <= created by our steps above
 - Folder: first_app <= created as a result to the call to 'startapp'
 - Folder: templates
 - Folder: first_app
 - Files: *.html
 - Folder: static
 - Folder: first_app
 - Folder: css
 - Folder: js
 - Folder: images
 - File: __init__.py
 - File: admin.py
 - File: apps.py
 - File: models.py <= The 'M' in MVC
 - File: tests.py
 - File: views.py <= The CONTROLLER in our MVC pattern
 - urls.py <= the file that will handle the routes of our app
 - File: manage.py <= used to run the application.
 - Folder: main <= contains the files specific to our app.
 - File: __init__.py
 - File: settings.py
 - File: urls.py
 - File: wsgi.py

Example of a Controller action ('View' in Django:)

```
# views.py
...
def index(request):
    return render(request, "first_app/index.html")
```

Implement sessions in our app:

```
# Need to be in same directory as manage.py file
> python manage.py makemigrations
> python manage.py migrate
```

Context**On the server:**

```
from django.shortcuts import render, HttpResponseRedirect, redirect
def show_ninja(request, ninja):
    #ninja got passed in through the url parameter!
    context = {'myninja' : ninja}
    return render(request, '/myproject/showmyninja.html', context)
```

In our templates:

2/20/2017

BACK TO TRACKS

Django Level I - Intro...74%
(/m/7/3832)

Intro and Overview (/m/)

Django as MVC (/m/)

Installing Django (/m/)

Big Picture (/m/)

Getting Started (/m/)

Routes and Controllers (/m/)

Views vs Templates (/m/)

Templates and Static Files (/m/)

Time Display (/m/)

Session Data and Forms (/m/)

Random Word Generator (/m/)

Quiz (/m/)

Survey Form (/m/)

Context Objects Re-explained (/m/)

Route Parameters (/m/)

Quiz II (/m/)

Django setup summary sh... (/m/)

Disappearing Ninjas

Django Ninja Gold

Django Level II - App...0%
(/m/7/3834)

Django Extras • 0%
(/m/7/3843)

Django Level III - Ad...0%
(/m/7/3844)

Django Level I - Introduction

<!-- From inside /myproject/showmyninja.html -->{{myninja}}</div><div><h3>Route Parameters</h3><p>In our urls.py file:</p><pre># Inside apps/appname/urls.py might look like this:
from django.conf.urls import url
from . import views
urlpatterns = [
 url(r'^/en/(?P<djangoversion>[0-9]\.[0-9])/topics/http/urls/\$', views.index)
]</pre><p>In our views.py file:</p><pre>from django.shortcuts import render, HttpResponseRedirect, redirect
Create your views here.
def index(request, djangoversion):
 print(djangoversion) # will result in a string e.g. "1.9"
 print("hello, I am your first request")
 return HttpResponseRedirect('hello')</pre></div><div><div>Privacy Policy</div><div>To report</div></div></div>

http://learn.codingdojo.com/m/7/3832/29055

4/4