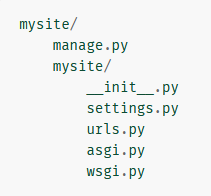
**Django project en Windows**

Para iniciar un Proyecto en django

<https://argentinaenpython.com/django-girls/tutorial/django_start_project/>

>django-admin.exe startproject carga .

Estructura



Para correr el servidor de desarrollo

> py manage.py runserver

O probar

$ python manage.py runserver

Para cambiar el puerto

> py manage.py runserver 8080

Para cambiar el puerto y la ip

>py manage.py runserver 0:8000

**Creando una aplicación**

> py manage.py startapp datos

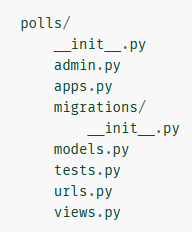
**Escribiendo la primera vista**

from django.http import HttpResponse

def index(request):

return HttpResponse("Hello, world. You're at the polls index.")

Esto pertenece a una vista en django. Para llamar la vista, es necesario mapearlo a una URL, y para esto necesitamos un URLConf, en la app datos, se crea un archivo urls.py



from django.urls import path

from . import views

urlpatterns = [

path('', views.index, name='index'),

]

Trabajar sobre urls.py del proyecto

from django.contrib import admin

from django.urls import include, path

urlpatterns = [

path('polls/', include('polls.urls')),

path('admin/', admin.site.urls),

]

La función include() permite referenciar otras URLconf. Cada vez que Django encuentra include(), corta cualquier parte de la URL que coincida hasta ese punto y envía la cadena restante a la URLconf incluida para su posterior procesamiento.

**Creando modelos**

Filosofía

Un modelo es la fuente única y definitiva de información sobre tus datos. Contiene los campos y comportamientos esenciales de los datos que estás almacenando. Django sigue el principio DRY. El objetivo es definir tu modelo de datos en un lugar y derivar automáticamente cosas de él.

Esto incluye las migraciones - a diferencia de Ruby On Rails, por ejemplo, las migraciones se derivan completamente de tu archivo de modelos, y son esencialmente un historial que Django puede recorrer para actualizar tu esquema de base de datos para que coincida con tus modelos actuales.