

TALLER DE INTRODUCCIÓN A

django

# Virtualenv

## SISTEMA OPERATIVO

### PROYECTO 1

Django 1.8.4  
MySQL-python 1.2.5  
Pillow 2.9.0  
Pyphen 0.9.4  
WeasyPrint 0.26  
Argparse 1.2.1  
Cairocffi 0.7.2  
Cffi 1.5.0  
Cssselect 0.9.1  
Decorator 4.0.4

### PROYECTO 2

Django 1.9  
MySQL-python 1.2.5  
Whoosh 2.7.0  
Argparse 1.2.1  
Decorator 4.0.4  
Django-braces 1.8.1  
Django-tables2 1.0.4  
Ipdb 0.8.1  
Ipython 4.0.0

### PROYECTO N

...

# Virtualenv

```
cd [workspace]
```

```
mkdir tutorial_1
```

```
mkvirtualenv -p python3 tutorial_1
```

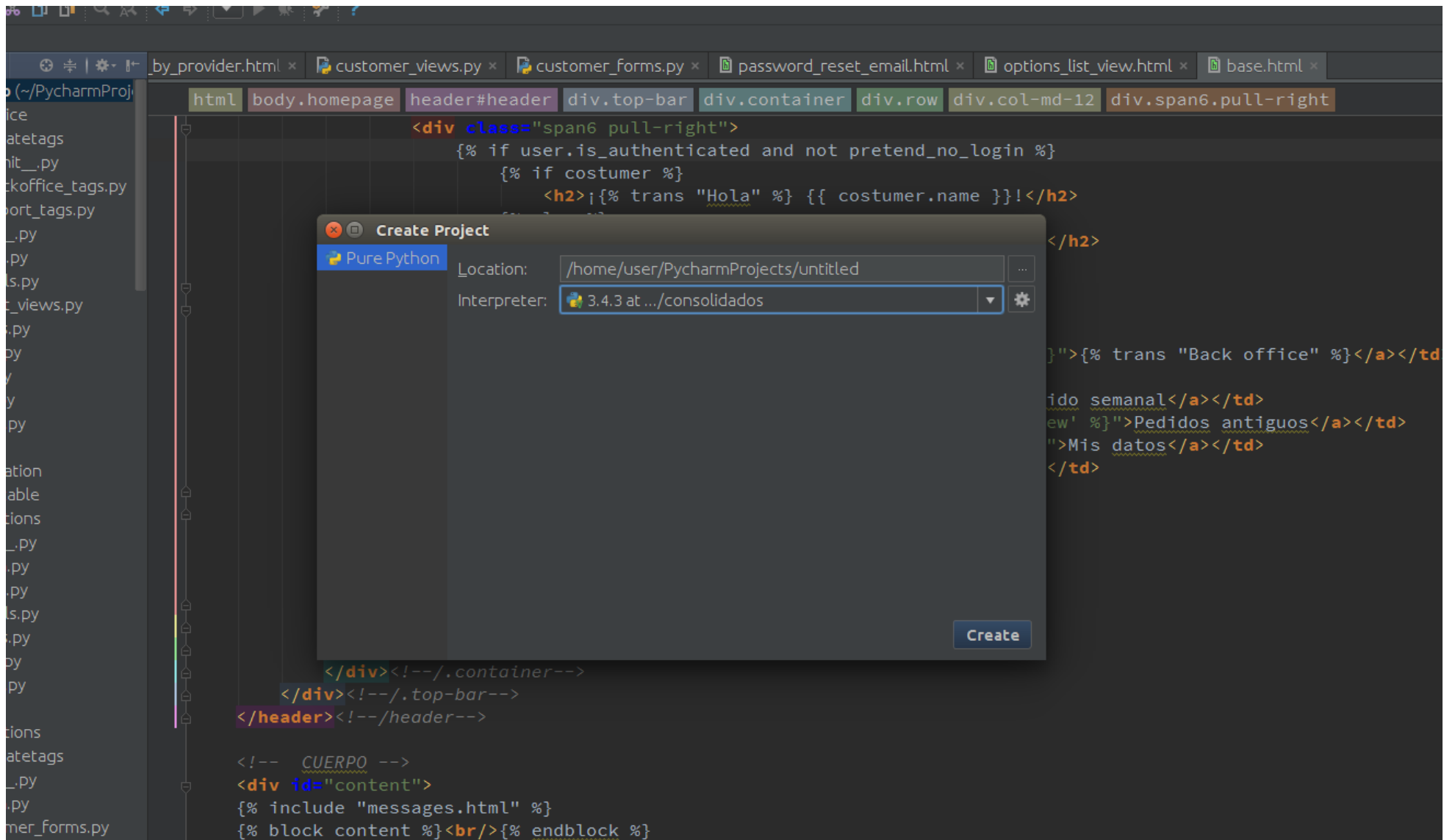
```
pip install django
```

```
django-admin startproject main ./
```

# Levantar servidor de desarrollo

```
python manage.py runserver
```

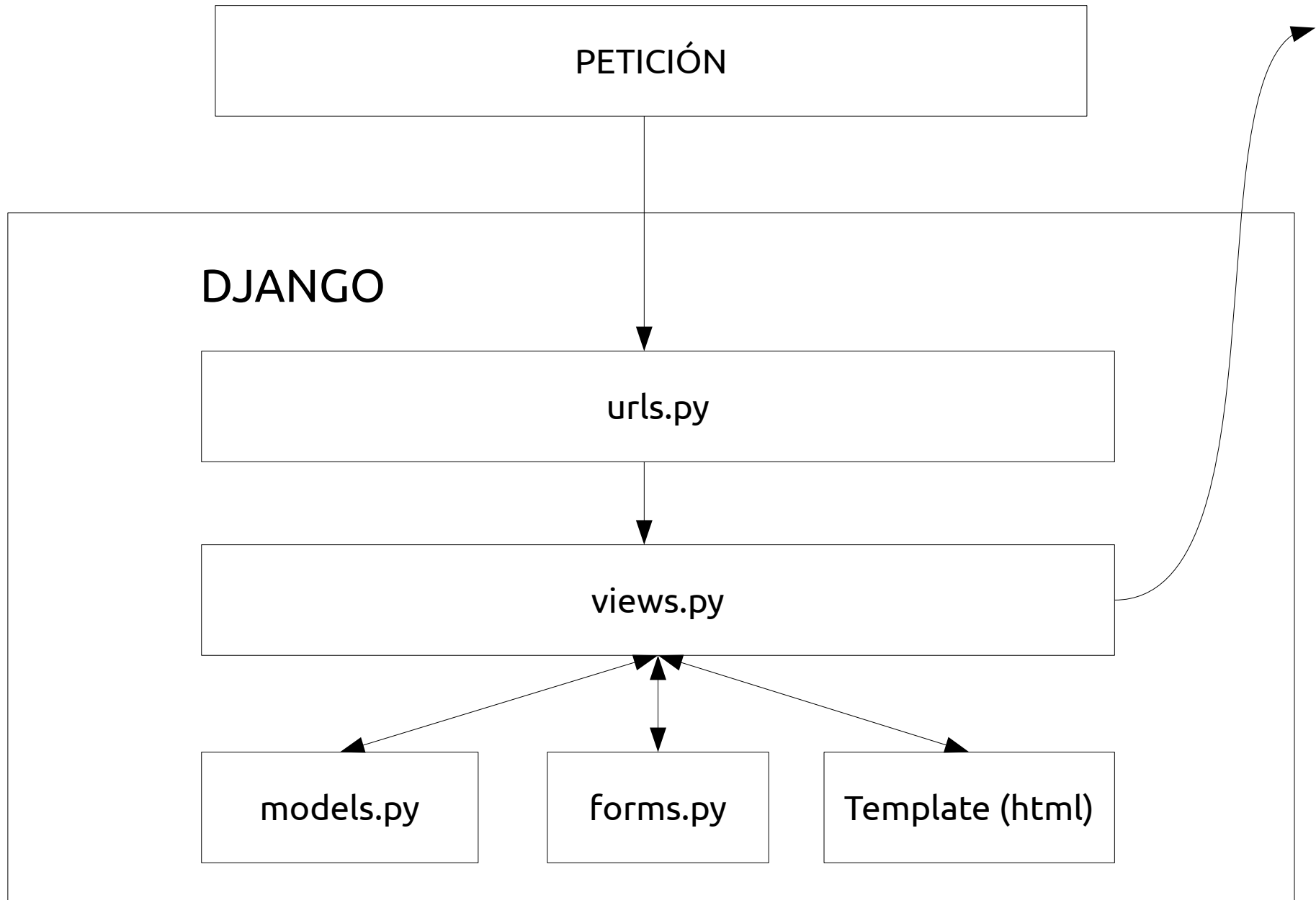
# Crear proyecto en PyCharm



# Ejecutar migraciones iniciales

```
python manage.py migrate
```

# Urls, views, models y templates



# La primera vista

```
# en main/views.py
```

```
from django.views.generic import TemplateView
```

```
class HomeView(TemplateView):  
    template_name = 'home.html'
```



# La primera url

```
# en main/urls.py
```

```
import views
```

```
...
```

```
urlpatterns = [
```

```
    ...
```

```
    url(r'^$', views.HomeView.as_view(), name='HomeView'),  
]
```

# La primera plantilla (template)

```
# creamos directorio main/templates  
# creamos el archivo main/templates/home.html
```

```
<!DOCTYPE html>  
<html>  
<head>  
<meta charset="UTF-8">  
<title>Home del catálogo musical</title>  
<body>Mi primera vista en Django</body>  
</html>
```

# ¿Dónde están las plantillas?

# en main/settings.py

```
TEMPLATES = [  
    {  
        'BACKEND': 'django.template.backends.django.DjangoTemplates',  
        'DIRS': ['templates'],  
        'APP_DIRS': True,  
        'OPTIONS': {  
            'context_processors': [  
                'django.template.context_processors.debug',  
                'django.template.context_processors.request',  
                'django.contrib.auth.context_processors.auth',  
                'django.contrib.messages.context_processors.messages',  
            ],  
        },  
    ],  
]
```

# Plantilla base

# creamos el archivo main/templates/base.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>{% block title %}{% endblock %}</title>
</head>
<body>{% block content %}{% endblock %}</body>
</html>
```

# Adaptamos home.html

```
# editamos main/templates/home.html
```

```
{% extends "base.html" %}
```

```
{% block title %}Home del catálogo musical{% endblock %}
```

```
{% block content %}Mi primera vista en Django{% endblock %}
```

# Creamos una app

```
python manage.py startapp catalog
```

```
# la añadimos a INSTALLED_APPS en main/settings.py
```

```
INSTALLED_APPS = [  
    'django.contrib.admin',  
    'django.contrib.auth',  
    'django.contrib.contenttypes',  
    'django.contrib.sessions',  
    'django.contrib.messages',  
    'django.contrib.staticfiles',  
    'main',  
    'catalog',  
]
```

# El primer modelo

```
# en catalog/models.py
```

```
from django.db import models
```

```
class Band(models.Model):
```

```
    name = models.CharField(max_length=150,  
                             blank=False,  
                             help_text='Nombre del grupo musical')
```

```
    web = models.URLField(blank=True, default="")
```

```
    bio = models.TextField(blank=False, help_text='Biografía del grupo')
```

# Creamos la tabla

```
# creamos la migración  
python manage.py makemigrations
```

```
# ejecutamos las migraciones pendientes  
python manage.py migrate
```



# Url para crear un grupo

```
# añadimos a main/urls.py
...
from catalog import views as catalog_views
...

urlpatterns = [
    ...
    url(r'^band/create/$', catalog_views.BandCreateView.as_view(),
        name='BandCreateView'),
]
```

# Vista para crear un grupo

```
# en catalog/views.py
from django.core.urlresolvers import reverse
from django.views.generic import CreateView
```

```
import forms
import models
```

```
class BandCreateView(CreateView):
    form_class = forms.BandCreateForm
    template_name = 'band/create.html'
    model = models.Band

    def get_success_url(self):
        return reverse('HomeView')
```

# El primer formulario

```
# creamos catalog/forms.py

from django import forms

from catalog import models as catalog_models

class BandCreateForm(forms.ModelForm):
    class Meta:
        model = catalog_models.Band
        fields = ('name',
                  'web',
                  'bio',)
```

# Plantilla band/create.html

```
# creamos directorio catalog/templates/band/  
# creamos el archivo create.html
```

```
{% extends "base.html" %}
```

```
{% block title %}Crear nuevo grupo de música{% endblock %}
```

```
{% block content %}
```

```
<h1>Crear nuevo grupo de música</h1>
```

```
<form method="POST" action="">
```

```
  <table class="table">
```

```
    {{ form.as_table }}
```

```
  </table>
```

```
  <input type="submit" value="Guardar">
```

```
</form>
```

```
{% endblock %}
```

# ¡Qué feo! Usemos Bootstrap

# Añadimos a base.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>{% block title %}{% endblock %}</title>
<link
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css"
rel="stylesheet">
</head>
<body>
<div class="container-fluid">{% block content %}{% endblock %}</div>
</body>
</html>
```

# Cross Site Request Forgery

```
# https://es.wikipedia.org/wiki/Cross-site_request_forgery  
# modificamos catalog/templates/band/create.html
```

```
{% extends "base.html" %}
```

```
{% block title %}Crear nuevo grupo de música{% endblock %}
```

```
{% block content %}
```

```
<h1>Crear nuevo grupo de música</h1>
```

```
<form method="POST" action="">
```

```
    {% csrf_token %}
```

```
    <table class="table">
```

```
        {{ form.as_table }}
```

```
    </table>
```

```
    <input class="btn btn-primary" type="submit" value="Guardar">
```

```
</form>
```

```
{% endblock %}
```

# Url para listar los grupos

# añadimos en urls.py

```
urlpatterns = [  
    url(r'^admin/', admin.site.urls),  
  
    url(r'^$', views.HomeView.as_view(), name='HomeView'),  
  
    url(r'^band/$', catalog_views.BandListView.as_view(), name='BandListView'),  
    url(r'^band/create/$', catalog_views.BandCreateView.as_view(),  
name='BandCreateView'),  
]
```

# Vista para listar grupos

# en catalog/views.py

```
class BandListView(ListView):  
    model = models.Band  
    template_name = 'band/list.html'
```



# Plantilla band/list.html

```
# creamos archivo catalog/templates/band/list.html
```

```
{% extends "base.html" %}
```

```
{% block title %}Lista de grupos{% endblock %}
```

```
{% block content %}
```

```
<h1>Lista de grupos</h1>
```

```
<table class="table">
```

```
<thead><tr><th>Nombre</th><th>Web</th><th>Biografía</th></tr></thead>
```

```
<tbody>
```

```
{% for band in object_list %}
```

```
<tr>
```

```
<td>{{ band.name }}</td>
```

```
<td><a href='{{ band.web }}'>{{ band.web }}</a></td>
```

```
<td>{{ band.bio }}</td>
```

```
</tr>
```

```
{% empty %}
```

```
<tr><td>Aún no hay grupos</td></tr>
```

```
{% endfor %}
```

```
</tbody></table>
```

```
{% endblock %}
```

# Modificamos home.html

```
# modificamos main/templates/home.html
```

```
{% extends "base.html" %}
```

```
{% block title %}Home de mi catálogo{% endblock %}
```

```
{% block content %}
```

```
<a class="btn btn-primary" href="{% url "BandListView" %}">Ver lista de  
grupos</a>
```

```
{% endblock %}
```

# Modificamos list.html

```
# modificamos catalog/templates/band/list.html
```

```
{% extends "base.html" %}
```

```
{% block title %}Lista de grupos{% endblock %}
```

```
{% block content %}
```

```
<h1>Lista de grupos</h1>
```

```
<a class="btn btn-primary" href="{% url 'BandCreateView' %}">Crear nuevo  
grupo</a>
```

```
<br/><br/>
```

```
<table class="table">
```

```
<thead>
```

```
  <tr><th>Nombre</th><th>Web</th><th>Biografía</th><th>Opciones</th></tr>
```

```
</thead>
```

```
...
```

# Modificamos create.html

```
# modificamos catalog/templates/band/create.html

{% extends "base.html" %}

{% block title %}Crear nuevo grupo de música{% endblock %}

{% block content %}
<h1>Crear nuevo grupo de música</h1>
<form method="POST" action="">
  {% csrf_token %}
  <table class="table">
    {{ form.as_table }}
  </table>
  <input class="btn btn-primary" type="submit" value="Guardar">
</form>
{% endblock %}
```

# Url para modificar un grupo

```
# modificamos main/urls.py
```

```
...
```

```
url(r'^band/update/(?P<pk>[0-9]+)/$', catalog_views.BandUpdateView.as_view(),  
    name='BandUpdateView'),
```

```
...
```

# Vista para modificar un grupo

# añadir a catalog/views.py

```
class BandUpdateView(UpdateView):  
    model = models.Band  
    template_name = 'band/create.html'  
    form_class = forms.BandCreateForm  
  
    def get_success_url(self):  
        return reverse('BandListView')
```

# Modificamos list.html

```
# modificamos catalog/templates/list.html
```

```
<table class="table">
<thead>
  <tr><th>Nombre</th><th>Web</th><th>Biografía</th><th>Opciones</th></tr>
</thead>
<tbody>
{% for band in object_list %}
  <tr>
    <td>{{ band.name }}</td>
    <td><a href='{{ band.web }}'>{{ band.web }}</a></td>
    <td>{{ band.bio }}</td>
    <td><a class="btn btn-primary" href='{% url "BandUpdateView" band.pk
%}'>Editar</a></td>
  </tr>
{% empty %}
  <tr><td>Aún no hay grupos</td></tr>
{% endfor %}
</tbody></table>
```

# Url para eliminar grupo

```
# modificamos main/urls.py
```

```
...
```

```
url(r'^band/delete/(?P<pk>[0-9]+)/$', catalog_views.BandDeleteView.as_view(),  
    name='BandDeleteView'),
```

```
...
```



# Vista para eliminar grupo

```
# añadimos a catalog/views.py
```

```
class BandDeleteView(DeleteView):  
    model = models.Band  
    template_name = 'band/delete.html'  
  
    def get_success_url(self):  
        return reverse('BandListView')
```

# Plantilla para eliminar grupo

```
# creamos catalog/templates/band/delete.html
```

```
{% extends "base.html" %}
```

```
{% block title %}Eliminar {{ object.name }}{% endblock %}
```

```
{% block content %}
```

```
<h1>¿De verdad quieres eliminar {{ object.name }}?</h1>
```

```
<form method="POST" action="">
```

```
    {% csrf_token %}
```

```
    <input class="btn btn-primary" type="submit" value="Eliminar">
```

```
</form>
```

```
{% endblock %}
```

# Recordatorio

Documentación oficial:	<a href="http://www.djangoproject.com">http://www.djangoproject.com</a>
Documentación Class Based Views:	<a href="http://ccbv.co.uk/">http://ccbv.co.uk/</a>
Crear virtualenv con python3:	<code>virtualenv -p /usr/bin/python3 [directorio_proyecto]</code>
Entrar en virtualenv:	<code>source [directorio_proyecto]/bin/activate</code>
Crear virtualenv con python3 (virtualenvwrapper):	<code>mkvirtualenv [nombre_virtualenv] -p /usr/bin/python3</code>
Crear virtualenv (virtualenvwrapper):	<code>mkvirtualenv [nombre_virtualenv]</code>
Ver virtualenvs (virtualenvwrapper):	<code>workon</code>
Entrar en virtualenv (virtualenvwrapper):	<code>workon [nombre_virtualenv]</code>
Salir de virtualenv:	<code>deactivate</code>
Instalar un paquete con pip:	<code>pip install [nombre_paquete]</code>
Buscar un paquete con pip:	<code>pip search [lo_que_queramos_buscar]</code>
Crear proyecto:	<code>django-admin startproject [nombre_proy] [directorio]</code>
Crear app:	<code>python manage.py startapp [nombre_app]</code>
Levantar servidor de desarrollo:	<code>python manage.py runserver [puerto_opcional]</code>
Levantar servidor de desarrollo visible en la red:	<code>python manage.py runserver 0.0.0.0:[puerto]</code>
Archivo de configuración de proyecto:	<code>main/settings.py</code>
Hacer migración:	<code>python manage.py makemigrations</code>
Ejecutar migraciones pendientes:	<code>python manage.py migrate</code>