

```
django-admin startproject website
```

**mostrar ls del contingut:**

```
website/  
  manage.py  
  website/  
    __init__.py  
    settings.py  
    urls.py  
    wsgi.py
```

```
cd website
```

```
sudo python manage.py runserver 80
```

```
python manage.py startapp imdb
```

```
imdb/  
  __init__.py  
  admin.py  
  apps.py  
  migrations/  
    __init__.py  
  models.py  
  tests.py  
  views.py
```

stw/settings.py:

```
INSTALLED_APPS = [  
    'imdb',  
    ....  
]
```

MODELS:

```
from __future__ import unicode_literals  
  
from django.db import models  
  
class Movie(models.Model):
```

```

name = models.CharField(max_length=75)
description = models.CharField(max_length=300)
cover_img = models.CharField(max_length=300)
def __str__(self):
    return self.name + ':' + self.description

```

```

# CREATE TABLE "imdb_movie" ("id" integer NOT NULL PRIMARY KEY
AUTOINCREMENT, "name" varchar(75) NOT NULL, "description" varchar(300) NOT
NULL, "cover_img" varchar(300) NOT NULL);

```

```

class Comment(models.Model):
    movie = models.ForeignKey(Movie)
    name = models.CharField(max_length=30)
    text = models.CharField(max_length=200)
    def __str__(self):
        return self.name + ':' + self.text

```

```

python manage.py makemigrations imdb //per dir que imdb ha patit
canvis

```

```

python manage.py migrate //per aplicar canvis a la bd

```

## VIEWS:

les vistes poden fer moltes coses, desde generar un arxiu .pdf fins a mostrar alguna cosa per pantalla. Però sempre han de retornar un `HttpResponse` o una excepció.

imdb/views.py:

```

from django.shortcuts import render, get_object_or_404
from django.http import HttpResponse, HttpResponseBadRequest,
HttpResponseRedirect
from .models import Movie, Comment
from .forms import CommentForm

# Create your views here.
def index(request):
    return HttpResponse('<h1>Welcome!</h1>')

def movie_detail(request, movie_id):
    movie = get_object_or_404(Movie, pk=movie_id)
    return render(request, 'movie.html', {'movie':movie})

```

```

def movie_comment(request, movie_id):
    if request.method == 'POST':
        form = CommentForm(request.POST)
        if form.is_valid():
            data = form.cleaned_data
            movie_obj = get_object_or_404(Movie, pk=movie_id)
            comment = Comment(movie=movie_obj, name=data['name'],
text=data['comment'])
            comment.save()
            return HttpResponseRedirect('/imdb/movie/' + movie_id + '/')
        else:
            return HttpResponseRedirect('Fill all the spaces')
    else:
        return HttpResponseRedirect('Only POST')

```

### TEMPLATES:

```

mkdir imdb/static/
posar les fotos a imdb/static/
mkdir imdb/templates/

```

imdb/templates/movie.html:

```

{% load staticfiles %}

<h3>{{ movie.name }}</h3>
<h4>{{ movie.description }}</h4>
<hr>
{% for comment in movie.comment_set.all %}
    <p>{{comment.name}}: {{comment.text}}</p>
{% endfor %}

<form action="/imdb/movie/{{ movie.id }}/comment/" method="post">
    {% csrf_token %}
    <label for="name">Name: </label>
    <input id="name" type="text" name="name">
    <label for="comment">Comment: </label>
    <input id="comment" type="text" name="comment">
    <input type="submit" value="OK">
</form>

```

### FORMS:

imdb/forms.py:

```
from django import forms
```

```
class CommentForm(forms.Form):  
    name = forms.CharField(label='name', max_length=30)  
    comment = forms.CharField(label='comment', max_length=200)
```

### URLS:

imdb/urls.py:

```
from django.conf.urls import url  
from . import views  
urlpatterns = [  
    # /  
    url(r'^$', views.index, name='index'),  
    # /movie/5/  
    url(r'^movie/(?P<movie_id>[0-9]+)/$', views.movie_detail, name='movie'),  
    # /movie/5/comment  
    url(r'^movie/(?P<movie_id>[0-9]+)/comment/$', views.movie_comment,  
name='movie'),  
]
```

stw/urls.py:

```
from django.conf.urls import url, include  
from django.contrib import admin  
  
urlpatterns = [  
    url(r'^admin/', admin.site.urls),  
    url(r'^imdb/', include('imdb.urls')),  
]
```

- 
- 
- Presentar Django for dummies, dir que explicarem els punts més importants per crear una aplicació web en Django. I que anirem fent la nostra aplicació mentre presentem.
  - How to start diapositives (projectes i apps)
  - mkdir projectes
  - django-admin startproject stw // Creem el projecte (carpeta STW)
  - cd stw
  - python manage.py startapp imdb // Creem la app imdb
  - ls -R // mostrem tot el contingut i expliquem una mica per sobre
  - gedit stw/settings.py & // Afegir 'imdb' a installed\_apps
  - Models, explicar diapos
  - gedit imdb/models.py & // mencionar que es creen claus primàries sense posar-les
  - python manage.py makemigrations imdb
  - python manage.py migrate
  - ls // Mostrar la base de dades sqlite i dir que es pot configurar un altre
  - Admin, explicar diapos
  - gedit imdb/admin.py & // Afegir els models al admin
  - python manage.py createsuperuser
  - python manage.py runserver
  - anar a localhost/admin/ i afegir 2 pelis i un comentari
  - Vistes, explicar diapos
  - gedit imdb/views.py & // Afegir les vistes
  - Templates, explicar diapos
  - mkdir imdb/templates/
  - gedit imdb/templates.py/movie.html & // Crear fitxer de template i afegir contingut
  - mkdir idmb/static/
  - posar les dues fotos allà
  - URLs, explicar diapos
  - gedit imdb/urls.py & // Crear fitxer i afegir contingut
  - gedit stw/urls.py & // Afegir lo del include
  - Ensenyar projecte final!
  - Comentar future work