

Tarea 4 - Vistas en Django

En el archivo **polls/views.py** y ponga el siguiente código Python en ella:

polls/views.py

```
from django.http import HttpResponseRedirect

def index(request):
    return HttpResponseRedirect("Hello, world. You're at the polls
index.")
```

Esta es la vista más simple posible en Django. Para llamar la vista, tenemos que asignarla a una URL y para ello necesitamos una URLconf.

Para crear una URLconf en el directorio encuestas, cree un archivo llamado **urls.py**. El directorio de su aplicación debe verse así:

```
polls/
  __init__.py
  admin.py
  apps.py
  migrations/
    __init__.py
  models.py
  tests.py
  urls.py
  views.py
```

Incluya el siguiente código en el archivo **polls/urls.py**:

polls/urls.py

```
from django.urls import path

from . import views

urlpatterns = [
    path('/', views.index, name='index'),
]
```

El siguiente paso es señalar la URLconf raíz en el módulo **polls.urls**. En **mysite/urls.py** añade un import para **django.urls.include** e inserte una **include()** en la lista **urlpatterns** , para obtener:

```
mysite/urls.py

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 hacer referencia a otros URLconfs. Cada vez que Django encuentra **include()** corta cualquier parte de la URL que coincide hasta ese punto y envía la cadena restante a la URLconf incluida para seguir el proceso.

La idea detrás de **include()** es facilitar la conexión y ejecución inmediata de las URLs. Dado que las encuestas están en su propia URLconf (**polls/urls.py**) se pueden ubicar en «/polls/», «/fun_polls /», «/content/polls/» o en cualquier otra ruta raíz , y la aplicación todavía seguirá funcionando.

Cuándo utilizar **include()**

Siempre debe usar **include()** cuando incluye otros patrones de URL. **admin.site.urls** es la única excepción a esto.

Se ha enviado una vista **index** al URLconf. se puede comprobar si funciona, ejecute el servidor

Al ir <http://localhost:8000/polls/> en el navegador, debería estar el texto «*Hello, world. You're at the polls index.*» el cual se definió en la vista **index**.

La **path()** función recibe cuatro argumentos, dos requeridos **route** y **view**; y dos opcionales **kwargs** y **name**. Este es el momento de revisar para que sirven estos argumentos.

argumento **path()**: **route**

route es una cadena que contiene un patrón de URL. Cuando Django procesa una petición comienza por el primer patrón en **urlpatterns** y continua hacia abajo por la lista comparando la URL solicitada con cada patrón hasta encontrar aquel que calza.

Estas expresiones regulares no buscan parámetros GET y POST o el nombre de dominio. Por ejemplo en una petición a la dirección URL **https://www.example.com/myapp/**, la URLconf buscará **myapp/**. En una petición a **https://www.example.com/myapp/?page=3** la URLconf también buscará **myapp/**.

argumento path(): view

Cuando Django encuentra una coincidencia de expresiones regulares llama a la función de la vista especificada con un objeto **HttpRequest** como primer argumento y cualquiera de los valores «capturados» de la ruta como argumentos de palabra clave. Le daremos un ejemplo de esto en un momento.

argumento path(): kwargs

Los argumentos arbitrarios de palabra clave se pueden pasar en un diccionario a la vista destino.

argumento path(): name

Dar un nombre a su URL le permite referirse a ella de forma inequívoca desde otras partes de Django sobre todo desde las plantillas. Esta potente característica le permite realizar cambios globales en los patrones de URL de su proyecto modificando solo un único archivo.