### La página de inicio de sesión

Primero implementaremos una página de inicio de sesión. Utilizaremos la vista login predeterminada que proporciona Django, por lo que el patrón de URL para esta aplicación es un poco diferente. Crea un nuevo archivo *urls.py* en el directorio *ll\_project/accounts/* y añádele lo siguiente:

**accounts/urls.py**

"""Defines URL patterns for accounts."""  
  
from django.urls import path, include  
  
app\_name = 'accounts'  
urlpatterns = [  
 # Include default auth urls.  
 path('', include('django.contrib.auth.urls')),  
]

Importamos la función path, y luego importamos la función include para poder incluir algunas URL de autenticación por defecto que Django ha definido. Estas URL predeterminadas incluyen patrones de URL con nombre, como 'login' y 'logout'. Establecemos la variable app\_name en 'accounts' para que Django pueda distinguir estas URL de las que pertenecen a otras aplicaciones. Incluso las URL predeterminadas proporcionadas por Django, cuando se incluyan en el archivo *urls.py* de la app accounts, serán accesibles a través del espacio de nombres accounts.

El patrón de la página de acceso coincide con la URL *http://localhost:8000/accounts/login/*. Cuando Django lee esta URL, la palabra *accounts* indica a Django que busque en *accounts/urls.py*, y *login* le indica que envíe las peticiones a la vista por defecto de Django login.

#### La plantilla de inicio de sesión

Cuando el usuario solicite la página de inicio de sesión, Django utilizará una función de vista por defecto, pero aún tenemos que proporcionar una plantilla para la página. Las vistas de autenticación por defecto buscan plantillas dentro de una carpeta llamada *registration*, así que tendremos que crear esa carpeta. Dentro del directorio *ll\_project/accounts/*, crea un directorio llamado *templates*; dentro de éste, crea otro directorio llamado *registration*. Aquí está la plantilla *login.html*, que debería guardarse en *ll\_project/accounts/templates/registration*:

**login.html**

{% extends 'learning\_logs/base.html' %}  
  
{% block content %}  
  
❶ {% if form.errors %}  
 <p>Your username and password didn't match. Please try again.</p>  
 {% endif %}  
  
❷ <form action="{% url 'accounts:login' %}" method='post'>  
 {% csrf\_token %}  
❸ {{ form.as\_div }}  
  
❹ <button name="submit">Log in</button>  
 </form>  
  
{% endblock content %}

Esta plantilla amplía *base.html* para garantizar que la página de inicio de sesión tenga el mismo aspecto que el resto del sitio. Ten en cuenta que una plantilla de una aplicación puede heredar de una plantilla de otra aplicación.

Si se establece el atributo errors del formulario, mostramos un mensaje de error ❶, informando de que la combinación de nombre de usuario y contraseña no coincide con nada almacenado en la base de datos.

Queremos que la vista de inicio de sesión procese el formulario, así que establecemos el argumento action como la URL de la página de inicio de sesión ❷. La vista de inicio de sesión envía un objeto form a la plantilla, y nos toca a nosotros mostrar el formulario ❸ y añadir un botón de envío ❹.

#### El ajuste LOGIN\_REDIRECT\_URL

Una vez que un usuario se registra correctamente, Django necesita saber dónde enviarlo. Esto lo controlamos en el archivo de configuración.

Añade el siguiente código al final de *settings.py*:

**settings.py**

--snip--  
# My settings.  
LOGIN\_REDIRECT\_URL = 'learning\_logs:index'

Con todos los ajustes por defecto en *settings.py*, es útil marcar la sección en la que vamos a añadir nuevos ajustes. La primera configuración nueva que añadiremos es LOGIN\_REDIRECT\_URL, que indica a Django a qué URL debe redirigir después de un intento de inicio de sesión con éxito.

#### Enlace a la página de inicio de sesión

Vamos a añadir el enlace de inicio de sesión a *base.html* para que aparezca en todas las páginas. No queremos que el enlace aparezca cuando el usuario ya haya iniciado sesión, así que lo anidaremos dentro de una etiqueta {% if %}:

**base.html**

<p>  
 <a href="{% url 'learning\_logs:index' %}">Learning Log</a> -  
 <a href="{% url 'learning\_logs:topics' %}">Topics</a> -  
❶ {% if user.is\_authenticated %}  
❷ Hello, {{ user.username }}.  
 {% else %}  
❸ <a href="{% url 'accounts:login' %}">Log in</a>  
 {% endif %}  
</p>  
  
{% block content %}{% endblock content %}

En el sistema de autenticación de Django, cada plantilla tiene un objeto user disponible que siempre tiene un atributo is\_authenticated establecido: el atributo es True si el usuario está conectado y False si no lo está. Este atributo te permite mostrar un mensaje a los usuarios autenticados y otro a los no autenticados.

Aquí mostramos un saludo a los usuarios conectados ❶. Los usuarios autenticados tienen un atributo adicional username, que utilizamos para personalizar el saludo y recordar al usuario que está conectado ❷. Para los usuarios que no se han autenticado, mostramos un enlace a la página de inicio de sesión ❸.

#### Utilizar la página de inicio de sesión

Ya hemos configurado una cuenta de usuario, así que vamos a iniciar sesión para ver si la página funciona. Ve a *http://localhost:8000/admin/*. Si todavía has iniciado sesión como administrador, busca un enlace de **cierre de sesión** en la cabecera y haz clic en él.

Cuando hayas cerrado la sesión, ve a *http://localhost:8000/accounts/login/*. Deberías ver una página de inicio de sesión similar a la que se muestra en [la Figura 19-4](#figure19-4). Introduce el nombre de usuario y la contraseña que configuraste anteriormente, y deberías volver a la página de inicio. La cabecera de la página de inicio debería mostrar un saludo personalizado con tu nombre de usuario.

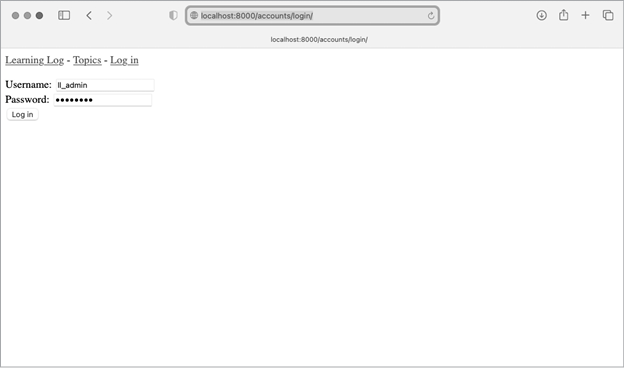


Figura 19-4: Página de inicio de

sesión

[anterior](c19_8.html)[Subtema 9 de 23: (Ver todo)](c19.html)[siguiente](c19_10.html)