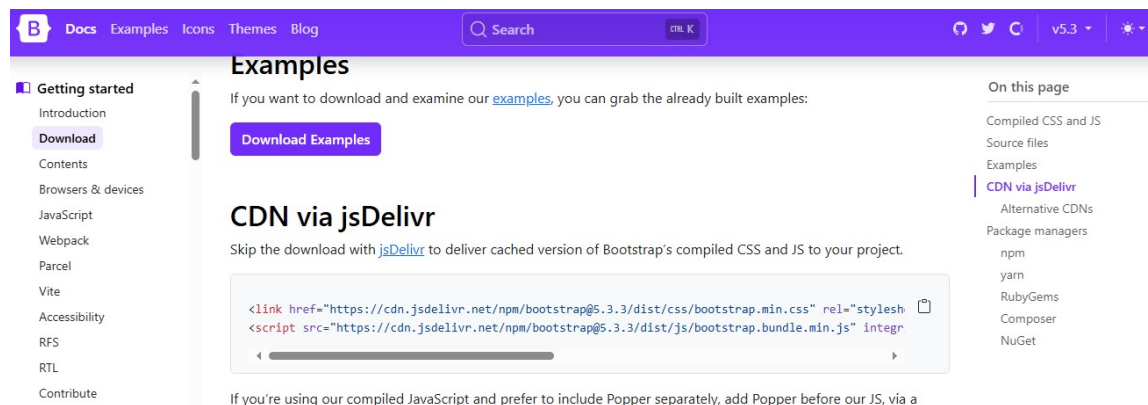


1. Estilos utilizando la nube

Lo que tenemos hasta ahora no tiene visualmente un aspecto agradable. Vamos a ver cómo podemos incorporarlo.

Vamos a irnos Bootstrap (framework de css, lo buscamos en google), a la sección Docs y luego a la sección Download. Nos vamos a la parte de CDN dónde nos encontraremos los links, tanto del css como del javascripts que nos apartarán los estilos.



Copiamos ambos enlaces con la precaución de que el enlace de css, el primero de ellos lo vamos a copiar en nuestro index.html, en el bloque de css y el otro enlace lo copiaremos en el bloque de script. Lo copiamos en el index.html solamente porque el resto de página extienden del él.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>{%block title%} Blog de autores y libros {%endblock title%}</title>
  {%block css%}
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"
  rel="stylesheet" integrity="sha384-QwTKZyjpPEjISv5WaRU90FeRpok6YctnYmDr5pNlyT2bRjXh0JMhY6hW+ALEwIH"
  crossorigin="anonymous">
  {%endblock css%}

</head>
<body>
  {%block menu%}
  <ul>
    <li><a href="/autor/">Autores</a></li>
    <li><a href="/libro/">Libros</a></li>
  </ul>
  {%endblock menu%}
  {%block content%}
  {%endblock content%}
  {%block footer%}
  {%endblock footer%}
  {%block js%}
```

```

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"
integrity="sha384-YvpcrYf0tY3lHB60NNkmXc5s9fDVZLESaAA55NDz0xhy9GkcIdsIK1eN7N6jIeHz"
crossorigin="anonymous"></script>
{%endblock js%}
</body>
</html>

```

Si comprobamos nuestras páginas la apariencia es ya distinta.



Para cambiar aún más la apariencia hay que aplicar algunas de las clases en las etiquetas directamente, usando los estilos que importamos que ha en los enlaces que hemos copiado de Bootstrap.

Si quisiéramos aplicar algún estilo por ejemplo en el listado de los libros, buscaríamos algún estilo que nos venga bien en el archivo bootstrap.css, y colocaríamos la clase en la etiqueta correspondiente:

Ejemplo:

```

.btn-secondary {
    color: #fff;
    background-color: #6c757d;
}

```

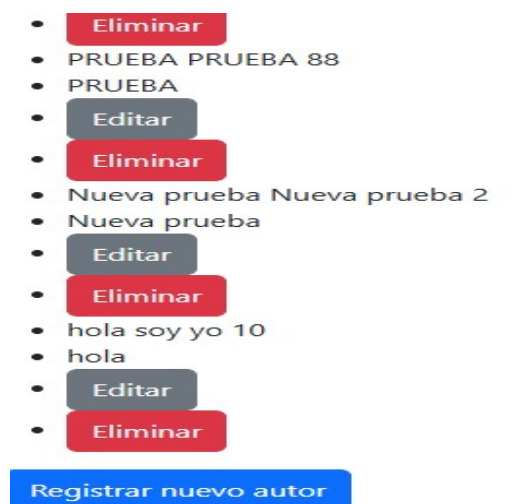
```
border-color: #6c757d;

}

{% extends 'index.html' %}
{% block title%} Listado de Autores {%endblock title%}

{%block content%}
<ul>
    {% for aut in authors %}
        <li>{{ aut }}</li>
        <li>{{ aut.name}}</li>      <!-- Lo que contiene la variable autor es una lista -->
        <li><a class="btn btn-secondary" href="/update-autor/{{aut.id}}">Editar</a></li>
        <li><a class="btn btn-danger" href="/delete-autor/{{aut.id}}">Eliminar</a></li>
    {% endfor %}
</ul>
<div><a class="btn btn-primary" href="/new-autor/">Registrar nuevo autor</a></div>

{%endblock content%}
```



De esa forma podríamos aplicar los estilos en todo el proyecto. Habría que coger las clases de Bootstrap que más nos interesen para cada etiqueta.

2.Estilos utilizando archivos locales

Otra opción sería tener todos estos archivos en nuestro proyecto. Para ellos nos crearemos una carpeta static y dentro, dos más una para los archivos css y otras para los javascript (css y js)

Ahora lo que haremos es crear en primer lugar un archivo dentro de css con el nombre bootstrap.min.css con todo el contenido de lo que hay en el link:

<https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css>

Igualmente hacemos dentro de la carpeta js. Creamos un fichero con el nombre bootstrap.min.js con el código que aparece en el enlace:

<https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js>

Para desvincular de la nube, quitamos los enlaces que habíamos puesto en los bloques css y js de la pagina index.html

Ahora hay que decirle a Django que tire de éstos archivos que hemos creado en nuestra aplicación. Hay que crear un tag, dónde se le dice a Django que debe cargar archivos estático. Al comienzo del index.html, incluimos la línea:

```
{% load static %}
```

Así le decimos la ruta dónde irá a buscar los archivos estáticos. Además, en el index.html hay que indicar dentro del bloque css y del js lo siguiente:

```
{%block css%}  
  <link rel="stylesheet" href="{% static 'css/bootstrap.min.css' %}">  
{%endblock css%}
```

```
{%block js%}  
  <script src="{% static 'js/bootstrap.min.js' %}"></script>  
{%endblock js%}
```

Y por último hay que irse a setting.py y crear la variable STATICFILES_DIRS, una variable que va a contener el directorio dónde Django va a ir a buscar los archivos estáticos.

```
STATIC_URL = 'static/'  
STATICFILES_DIRS = [ BASE_DIR/ 'stactic' ] #Basedir es la base del directorio de nuestro proyecto.  
Está definida más arriba, dentro del fichero setting.py. Así, de esta forma estamos indicando un  
subdirectorio dentro de nuestro proyecto
```

Si vuelvo a cargar la página, me vuelve a aparecer los estilos, pero ésta vez tirando de aquellos que tengo dentro de mi carpeta.

