

## Pasos para agregar una función en el archivo `views.py`, llamada `listar`

- a. Agregar un archivo `urls.py` en la aplicación. Permite manipular las direcciones de la aplicación. Relacionarlo con el archivo `urls.py` del proyecto.

```
# urls de la aplicación
from django.urls import path
# se importa las vistas de la aplicación
from . import views

urlpatterns = [
    path('listar', views.listar, name='listar'),
    path("", views.listar, name='listar'),
]

# urls del proyecto
from django.contrib import admin
from django.urls import include, path

urlpatterns = [
    path('admin/', admin.site.urls),
    path('ligapro/', include('campeonato.urls')),
]
```

- b. Incluir en la función las siguientes líneas, que permiten:

- Obtener la información de la base de datos.
- Crear una variable que traslade información desde la vista al template.

```
def listar(request):
    """
```

```
        Listar los registros del modelo Equipo,
        obtenidos de la base de datos.
```

"""

```
# a través del ORM de django se obtiene
# los registros de la entidad; el listado obtenido
# se lo almacena en una variable llamada
# equipos
equipos = Equipo.objects.all()
# se obtiene el número de elementos de la lista
numero_equipos = len(equipos)
# en la variable tipo diccionario llamada informacion_template
# se agregará la información que estará disponible
# en el template
informacion_template = {'equipos': equipos, 'numero_equipos':
numero_equipos}
return render(request, 'listar_equipo.html', informacion_template)
```

- c. Crear una carpeta llamada *templates* en el directorio de la aplicación. Agregar un archivo que se llame, *listar\_equipo.html*. Además, se genera un archivo llamado *principal.html*; en este archivo se originará el esqueleto base o plantilla de los templates.

```
# principal.html
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width"
/>
    {% load static %}
    <link rel="stylesheet" type="text/css" href="{% static
'css/miestilo.css' %}">
    <link rel="stylesheet" type="text/css" href="https://cdn.
datatables.net/1.13.1/css/jquery.dataTables.min.css">
```

```

<script
    src="https://code.jquery.com/jquery-3.6.3.min.js"

    integrity="sha256-pvPw+upLPUjgMXy0G+800xUf+/
    lm1MZjXxxgOcBQBXU="

    crossorigin="anonymous"></script>

<script type="text/javascript" src="https://cdn.datatables.
net/1.13.1/js/jquery.dataTables.min.js"></script>
<title>
Proyecto Ejemplo
</title>
</head>
<body>
<h1>Aplicación Equipos </h1>
<hr>
<section>
{% block contenido %}
{% endblock %}
</section>

<footer>
<p>Loja-Ecuador</p>
<p>{% now "j F Y H:i" %}</p>
</footer>
</body>
</html>

```

- d. Incluir en el archivo *listar\_equipo.html*, las siguientes líneas de código, que permiten visualizar la lista de objetos, que vienen desde la vista.

```

# listar_equipo.html
{% extends "principal.html" %}

```

```

{% block contenido %}
<h3>Lista de Equipos</h3>
<p><b>Número de equipos: ({{numero_equipos}})</b> </p>
<table id="tabla_datos">
<thead>
<tr>
<th>Nombre</th>
<th>Siglas</th>
<th>Estadio</th>
<th>Seguidores</th>
<th>Campeonatos</th>
</tr>
</thead>
<tbody>
{% for equipo in equipos %}
<tr>
<td>{{equipo.nombre}}</td>
<td>{{equipo.siglas}}</td>
<td>{{equipo.estadio}}</td>
<td>{{equipo.seguidores}}</td>
<td>{{equipo.campeonatos}}</td>
</tr>
{% endfor %}
</table>
</tbody>
<script type="text/javascript">
$(document).ready( function () {
$('#tabla_datos').DataTable();
});
</script>
{% endblock %}

```

- e. Ingresar la siguiente dirección: `http://localhost:8000/ligapro/listar`.

Se debe visualizar los registros de la entidad equipo.

### Figura 1.

*Listado de objetos a través de una función del views y un template*

#### Aplicación Equipos

##### Lista de Equipos

Número de equipos: (2)

Show  entries

Search:

Nombre	Siglas	Estadio	Seguidores	Campeonatos
Barcelona	BSC	Isidro Carbo	30000	17

Showing 1 to 1 of 1 entries (filtered from 2 total entries)

Previous  Next

Loja-Ecuador

30 January 2023 07:44

**Nota.**











