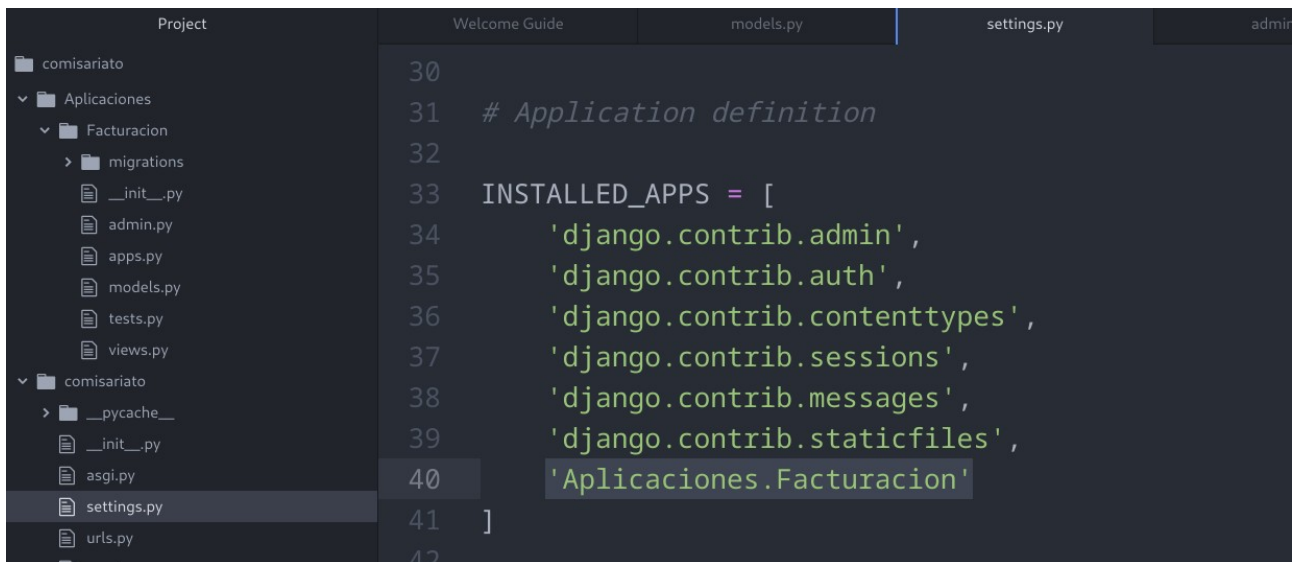
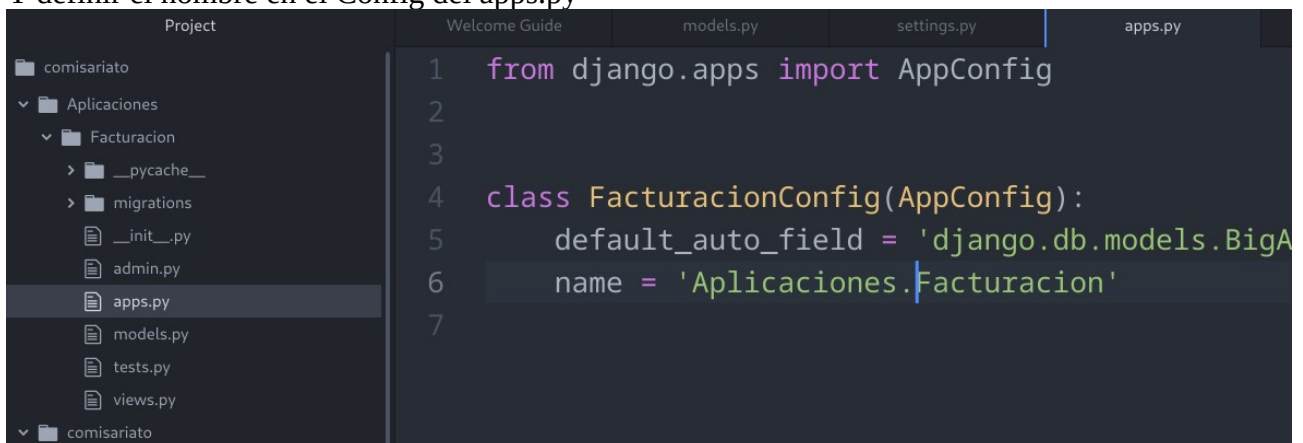


1. Crear Proyecto  
\$ django-admin startproject comisariato
2. Entrar a la carpeta **comisariato**  
\$ cd comisariato
3. Crear una carpeta para las aplicaciones y entrar a ella  
\$ mkdir Aplicaciones  
\$ cd Aplicaciones
4. Crear una aplicación  
\$ django-admin startapp Facturacion
5. Registrar la carpeta de la nueva aplicación en settings.py de la carpeta comisariato



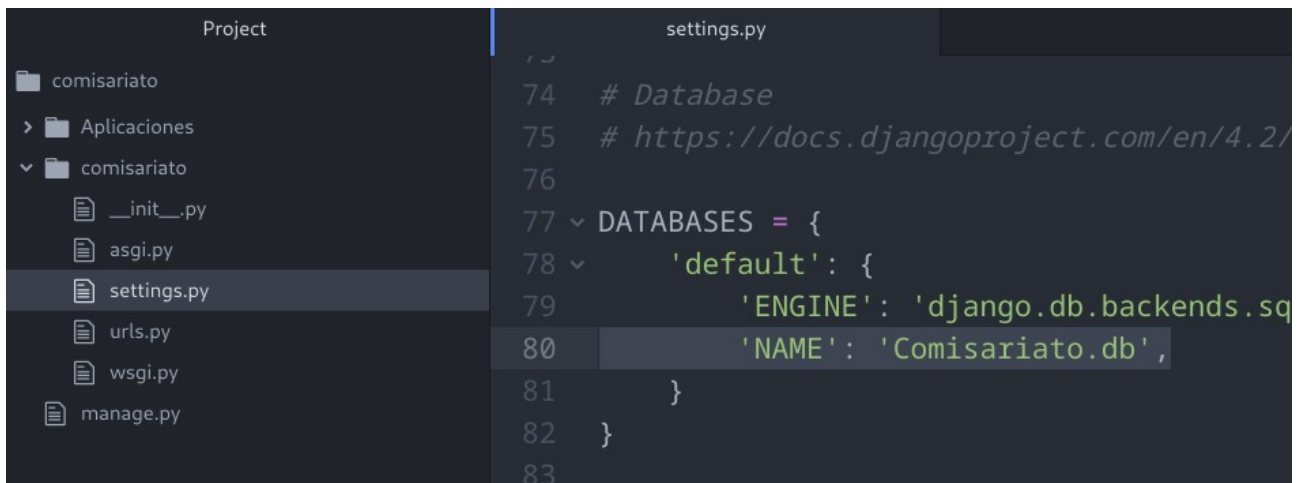
```
30
31 # Application definition
32
33 INSTALLED_APPS = [
34     'django.contrib.admin',
35     'django.contrib.auth',
36     'django.contrib.contenttypes',
37     'django.contrib.sessions',
38     'django.contrib.messages',
39     'django.contrib.staticfiles',
40     'Aplicaciones.Facturacion'
41 ]
42
```

Y definir el nombre en el Config del apps.py



```
1 from django.apps import AppConfig
2
3
4 class FacturacionConfig(AppConfig):
5     default_auto_field = 'django.db.models.BigAutoField'
6     name = 'Aplicaciones.Facturacion'
7
```

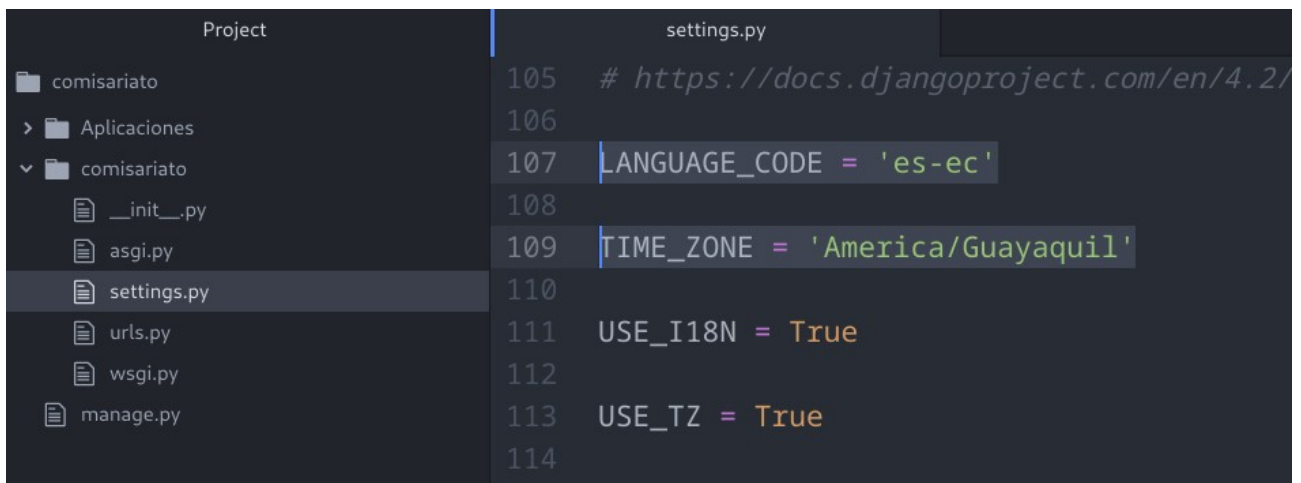
6. Configurar una base de datos con SQLite3



```
Project
├── comisariato
│   ├── Aplicaciones
│   └── comisariato
│       ├── __init__.py
│       ├── asgi.py
│       ├── settings.py
│       ├── urls.py
│       ├── wsgi.py
│       └── manage.py
└── settings.py

settings.py
74 # Database
75 # https://docs.djangoproject.com/en/4.2/
76
77 DATABASES = {
78     'default': {
79         'ENGINE': 'django.db.backends.sqlite3',
80         'NAME': 'Comisariato.db',
81     }
82 }
83
```

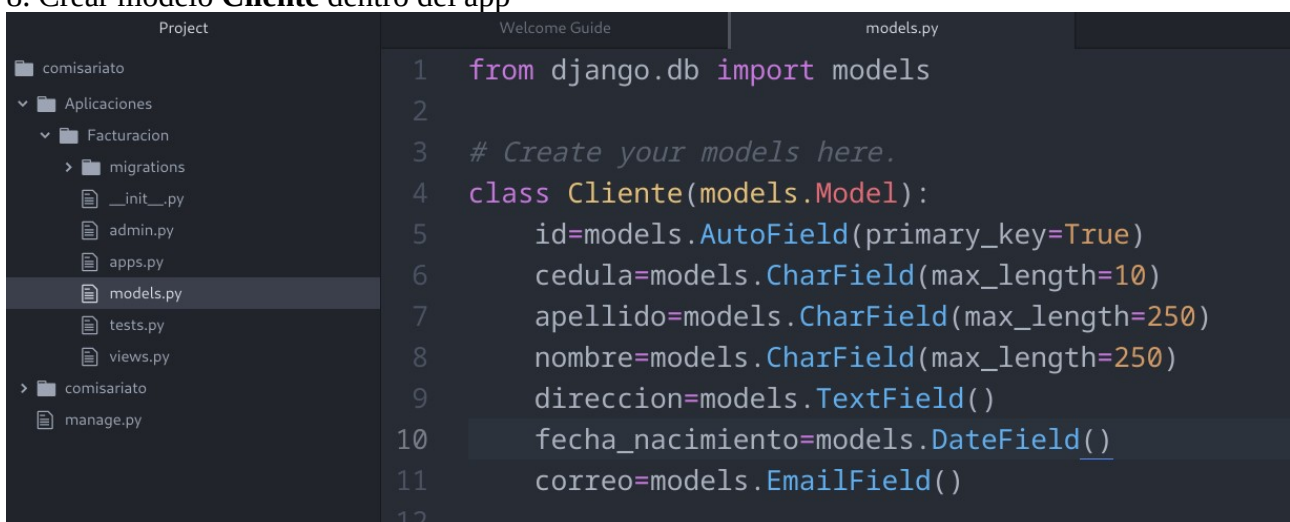
7. Definimos el idioma y la zona horaria:



```
Project
├── comisariato
│   ├── Aplicaciones
│   └── comisariato
│       ├── __init__.py
│       ├── asgi.py
│       ├── settings.py
│       ├── urls.py
│       ├── wsgi.py
│       └── manage.py
└── settings.py

settings.py
105 # https://docs.djangoproject.com/en/4.2/
106
107 LANGUAGE_CODE = 'es-ec'
108
109 TIME_ZONE = 'America/Guayaquil'
110
111 USE_I18N = True
112
113 USE_TZ = True
114
```

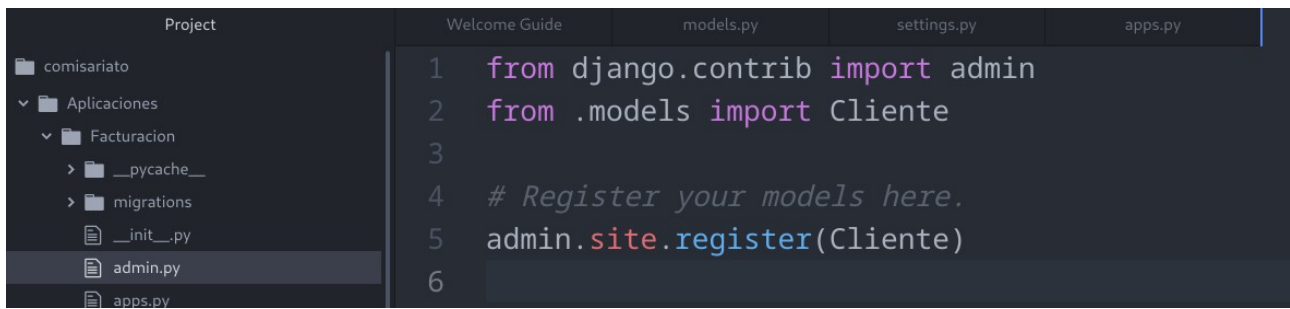
8. Crear modelo **Cliente** dentro del app



```
Project
├── comisariato
│   ├── Aplicaciones
│   └── Facturacion
│       ├── migrations
│       ├── __init__.py
│       ├── admin.py
│       ├── apps.py
│       ├── models.py
│       ├── tests.py
│       └── views.py
└── comisariato
    ├── manage.py
    └── models.py

models.py
1 from django.db import models
2
3 # Create your models here.
4 class Cliente(models.Model):
5     id=models.AutoField(primary_key=True)
6     cedula=models.CharField(max_length=10)
7     apellido=models.CharField(max_length=250)
8     nombre=models.CharField(max_length=250)
9     direccion=models.TextField()
10    fecha_nacimiento=models.DateField()
11    correo=models.EmailField()
12
```

## 9. Activar el panel de administración de Django en admin.py del app

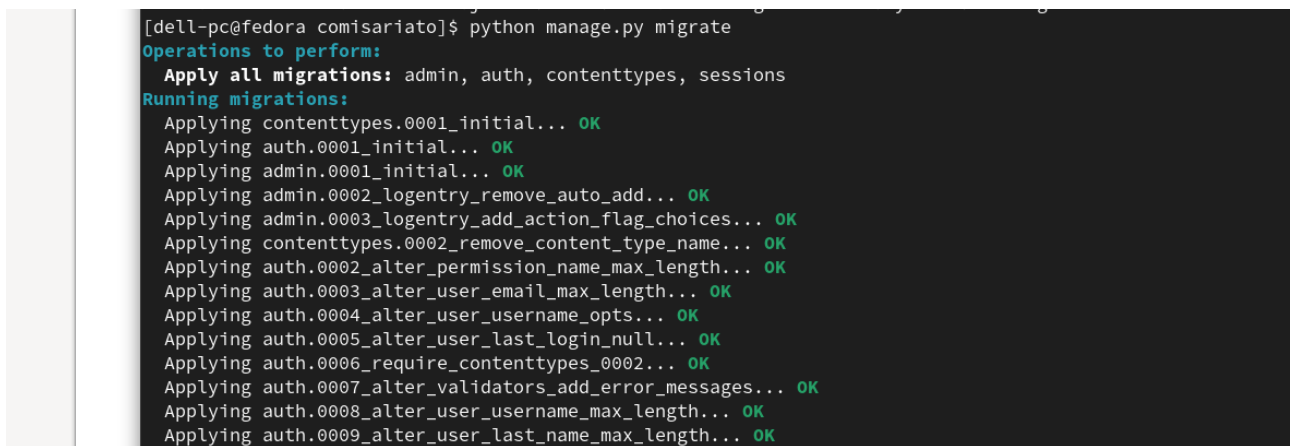


```
Project Welcome Guide models.py settings.py apps.py
comisariato
  Aplicaciones
    Facturacion
      __pycache__
      migrations
      __init__.py
      admin.py
      apps.py

1 from django.contrib import admin
2 from .models import Cliente
3
4 # Register your models here.
5 admin.site.register(Cliente)
6
```

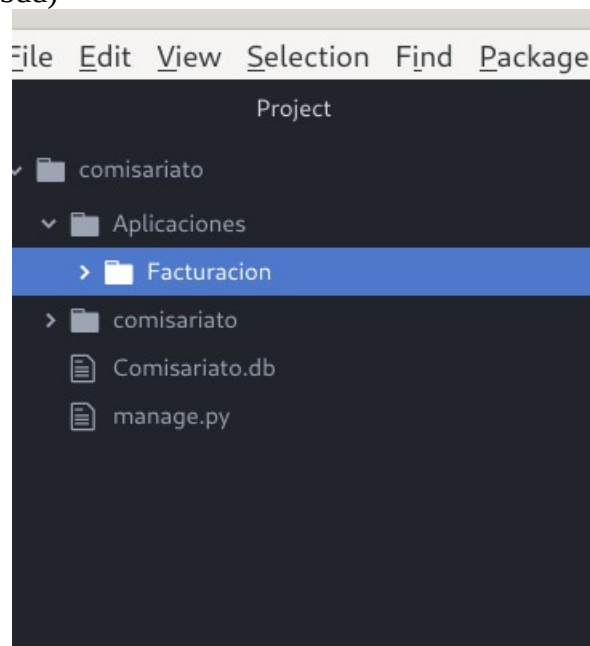
## 10. En la carpeta del proyecto ejecutar:

\$ python manage.py migrate



```
[dell-pc@fedora comisariato]$ python manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
```

## Migracion Inicial (Crea la bdd)



11. Crear la tabla Cliente en la bdd correspondiente mediante:

\$ python manage.py makemigrations

```
dell-pc@fedora:~/Documentos/septimo2324/comisariato
[dell-pc@fedora comisariato]$ python manage.py makemigrations
Migrations for 'Facturacion':
  Aplicaciones/Facturacion/migrations/0001_initial.py
    - Create model Cliente
[dell-pc@fedora comisariato]$
```

12. Ejecutar la migracion con:

\$ python manage.py migrate

13. Crear super Usuario

\$ python manage.py createsuperuser

```
dell-pc@fedora:~/Documentos/septimo2324/comisariato
[dell-pc@fedora comisariato]$ python manage.py makemigrations
Migrations for 'Facturacion':
  Aplicaciones/Facturacion/migrations/0001_initial.py
    - Create model Cliente
[dell-pc@fedora comisariato]$ python manage.py createsuperuser

You have 1 unapplied migration(s). Your project may not work properly until you apply the migrations for app(s): Facturacion.
Run 'python manage.py migrate' to apply them.
Nombre de usuario (leave blank to use 'dell-pc'): renequis
Dirección de correo electrónico: renequisaguano@gmail.com
Password:
Password (again):
Esta contraseña es demasiado corta. Debe contener al menos 8 caracteres.
Esta contraseña es demasiado común.
Esta contraseña es completamente numérica.
Bypass password validation and create user anyway? [y/N]: y
Superuser created successfully.
[dell-pc@fedora comisariato]$
```

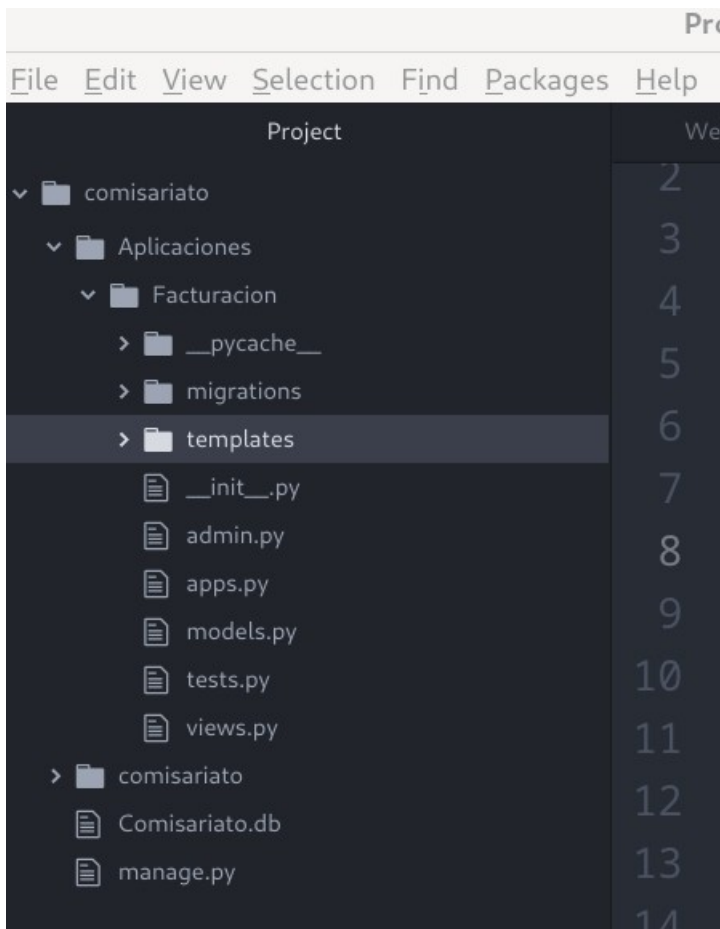
14. Correr el servidor

\$ python manage.py runserver

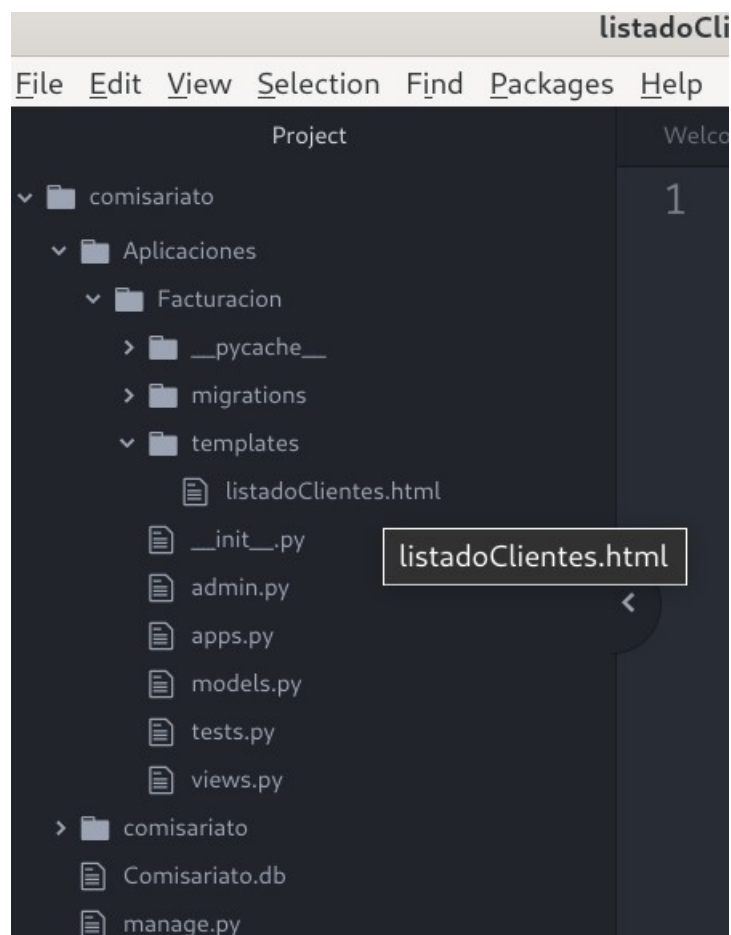




16. Crear carpeta para **templates** (Todo con minúscula)



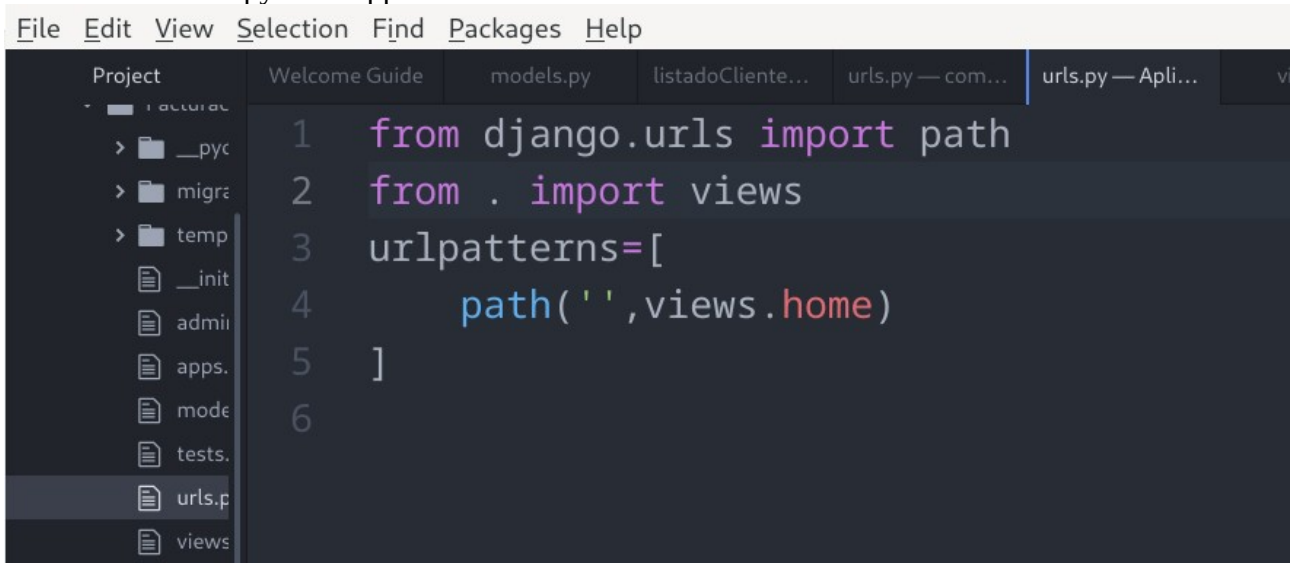
Crear archivo listadoCLientes.html





## 17. Configurar redireccionamiento

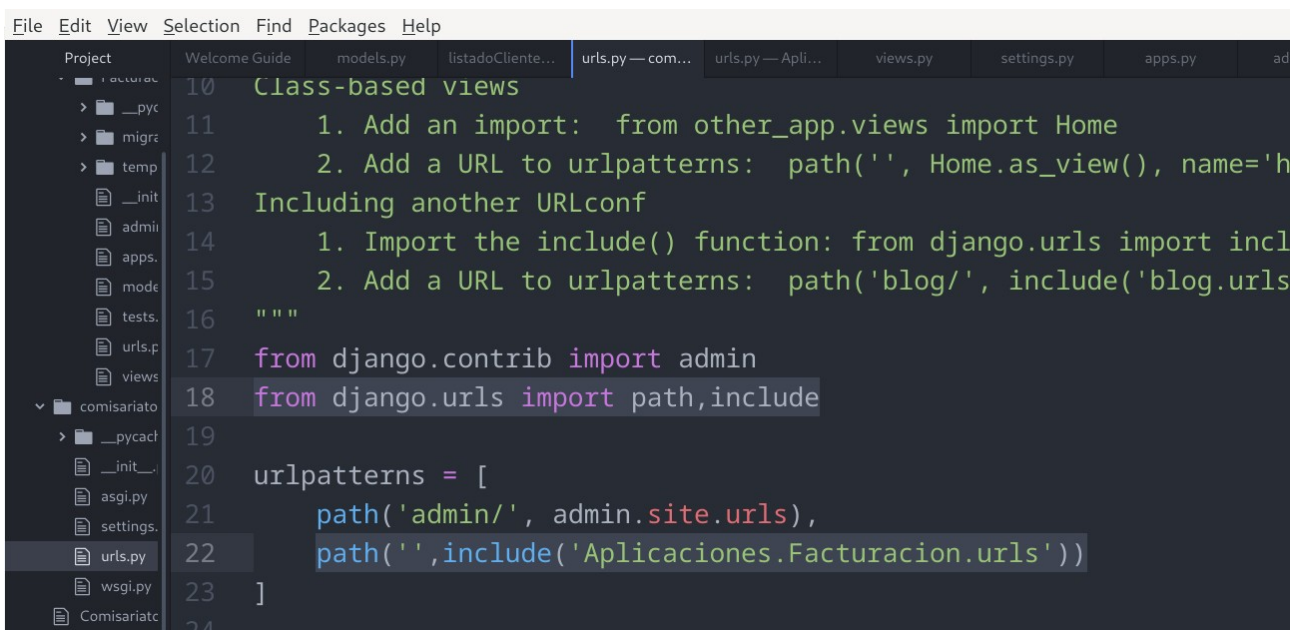
Crear archivo urls.py en el app



The screenshot shows a code editor with a menu bar (File, Edit, View, Selection, Find, Packages, Help) and a project explorer on the left. The project explorer shows a folder named 'Facturacion' with subfolders '\_\_pyc', 'migraciones', and 'templates', and files '\_\_init\_\_.py', 'admin.py', 'apps.py', 'models.py', 'tests.py', 'urls.py', and 'views.py'. The 'urls.py' file is selected and its content is displayed in the editor:

```
1 from django.urls import path
2 from . import views
3 urlpatterns=[
4     path('',views.home)
5 ]
6
```

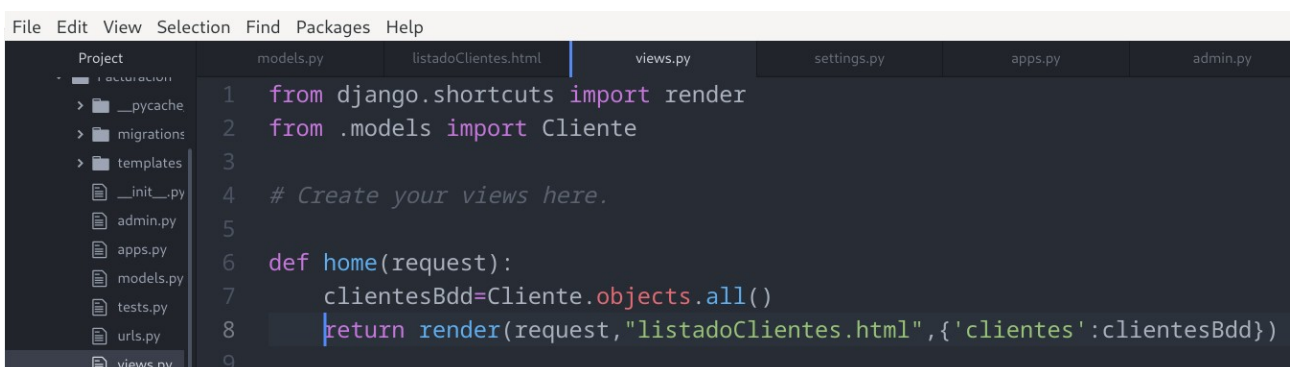
## 18. Configurar el archivo views.py del app



The screenshot shows a code editor with a menu bar (File, Edit, View, Selection, Find, Packages, Help) and a project explorer on the left. The project explorer shows a folder named 'comisariato' with subfolders '\_\_pycache\_\_' and 'Comisariato', and files '\_\_init\_\_.py', 'asgi.py', 'settings.py', 'urls.py', 'wsgi.py', and 'Comisariato.py'. The 'views.py' file is selected and its content is displayed in the editor:

```
10 Class-based views
11 1. Add an import: from other_app.views import Home
12 2. Add a URL to urlpatterns: path('', Home.as_view(), name='h
13 Including another URLconf
14 1. Import the include() function: from django.urls import incl
15 2. Add a URL to urlpatterns: path('blog/', include('blog.urls
16 """
17 from django.contrib import admin
18 from django.urls import path,include
19
20 urlpatterns = [
21     path('admin/', admin.site.urls),
22     path('',include('Aplicaciones.Facturacion.urls'))
23 ]
24
```

## 19. Consulta de Datos

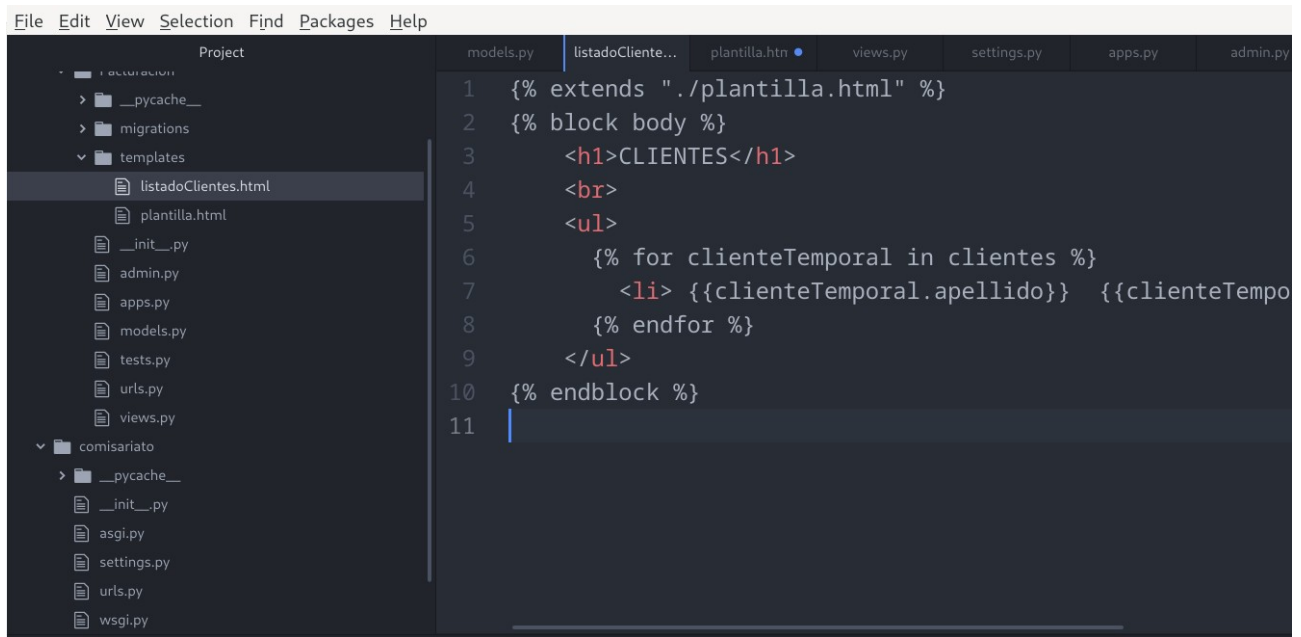


The screenshot shows a code editor with a menu bar (File, Edit, View, Selection, Find, Packages, Help) and a project explorer on the left. The project explorer shows a folder named 'Facturacion' with subfolders '\_\_pycache\_\_', 'migrations', and 'templates', and files '\_\_init\_\_.py', 'admin.py', 'apps.py', 'models.py', 'tests.py', 'urls.py', and 'views.py'. The 'views.py' file is selected and its content is displayed in the editor:

```
1 from django.shortcuts import render
2 from .models import Cliente
3
4 # Create your views here.
5
6 def home(request):
7     clientesBdd=Cliente.objects.all()
8     return render(request,"listadoClientes.html",{ 'clientes':clientesBdd})
9
```

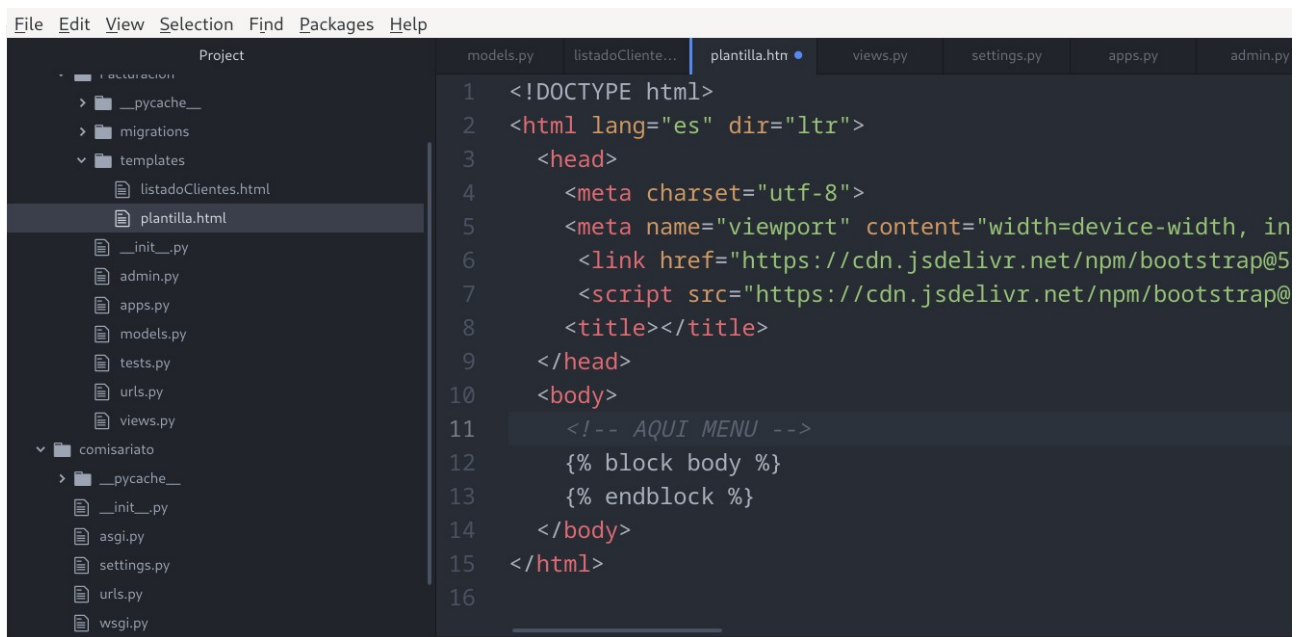


## 20. Mostrar Datos en Pantalla con Jinja2



```
File Edit View Selection Find Packages Help
Project
- folder
  > folder __pycache__
  > folder migrations
  v folder templates
    listadoClientes.html
    plantilla.html
  __init__.py
  admin.py
  apps.py
  models.py
  tests.py
  urls.py
  views.py
v folder comisariato
  > folder __pycache__
  __init__.py
  asgi.py
  settings.py
  urls.py
  wsgi.py
models.py listadoCliente... plantilla.htr views.py settings.py apps.py admin.py
1 {% extends "../plantilla.html" %}
2 {% block body %}
3     <h1>CLIENTES</h1>
4     <br>
5     <ul>
6         {% for clienteTemporal in clientes %}
7             <li> {{clienteTemporal.apellido}} {{clienteTempo
8             {% endfor %}
9         </ul>
10 {% endblock %}
11
```

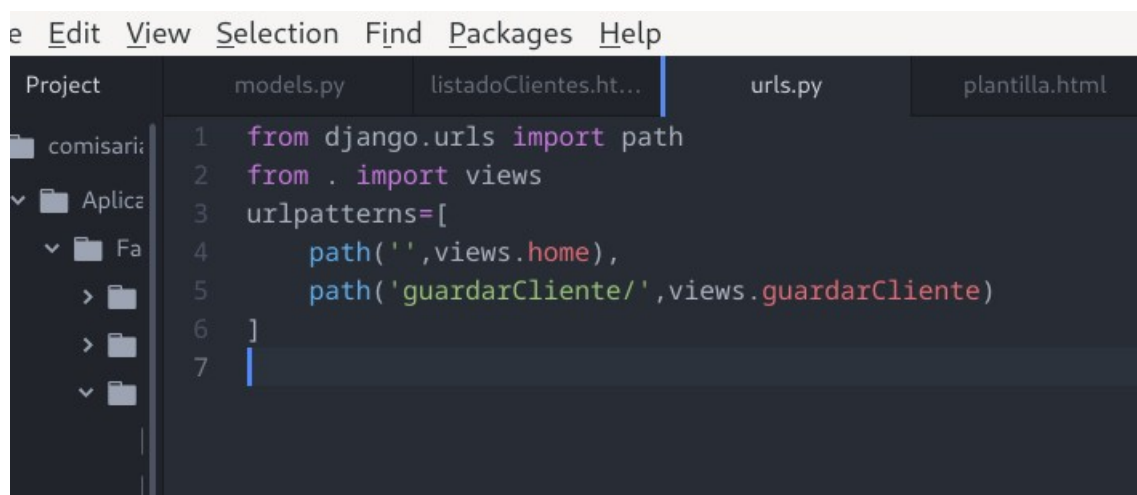
## 21. Generar el archivo plantilla.html reutilizable



The screenshot shows a code editor with a dark theme. The left sidebar displays a project structure with folders like 'templates' and 'comisariato', and files like 'listadoClientes.html' and 'plantilla.html'. The main editor area shows the content of 'plantilla.html', which is an HTML template. The code includes a DOCTYPE declaration, an HTML tag with language and direction attributes, a head section with meta tags for charset and viewport, and a body section with a comment indicating where a menu should be placed. The code is as follows:

```
1 <!DOCTYPE html>
2 <html lang="es" dir="ltr">
3   <head>
4     <meta charset="utf-8">
5     <meta name="viewport" content="width=device-width, in
6     <link href="https://cdn.jsdelivrivr.net/npm/bootstrap@5
7     <script src="https://cdn.jsdelivrivr.net/npm/bootstrap@
8     <title></title>
9   </head>
10  <body>
11    <!-- AQUÍ MENU -->
12    {% block body %}
13    {% endblock %}
14  </body>
15 </html>
16
```

## 22. Vista Crear Cliente



The screenshot shows a code editor with a dark theme. The left sidebar displays a project structure with folders like 'comisari...' and 'Aplica...', and files like 'urls.py' and 'plantilla.html'. The main editor area shows the content of 'urls.py', which is a Django URL configuration file. The code includes imports for 'path' from 'django.urls' and 'views' from the current directory, and a list of URL patterns. The code is as follows:

```
1 from django.urls import path
2 from . import views
3 urlpatterns=[
4     path('',views.home),
5     path('guardarCliente/',views.guardarCliente)
6 ]
7
```

File Edit View Selection Find Packages Help

Project models.py listadoClientes.ht... urls.py plantilla.html views.py settings.py apps.py

```
1 from django.shortcuts import render, redirect
2 from .models import Cliente
3
4 # Create your views here.
5
6 def home(request):
7     clientesBdd=Cliente.objects.all()
8     return render(request,"listadoClientes.html",{ 'clientes':clientesBdd})
9
10 def guardarCliente(request):
11     cedula=request.POST["cedula"]
12     apellido=request.POST["apellido"]
13     nombre=request.POST["nombre"]
14     direccion=request.POST["direccion"]
15     fecha_nacimiento=request.POST["fecha_nacimiento"]
16     correo=request.POST["correo"]
17     cliente=Cliente.objects.create(cedula=cedula,apellido=apellido,
18                                   nombre=nombre,direccion=direccion,fecha_nacimiento=fecha_nacimiento,correo=correo)
19     return redirect('/')
20
```

comisari...

Aplic...

Fa...

comis...

