

Creación de plantillas en Django2

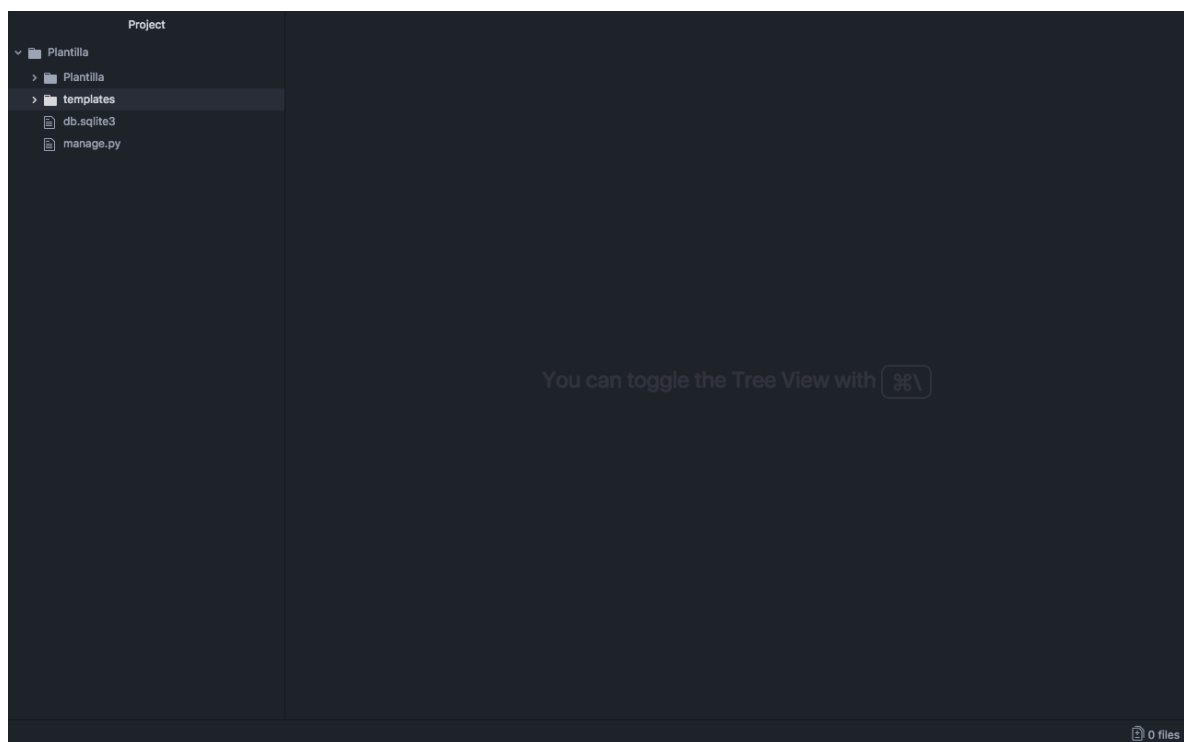
José Santorcuato
Chile 2017

Ingresar a ambiente y ejecutar

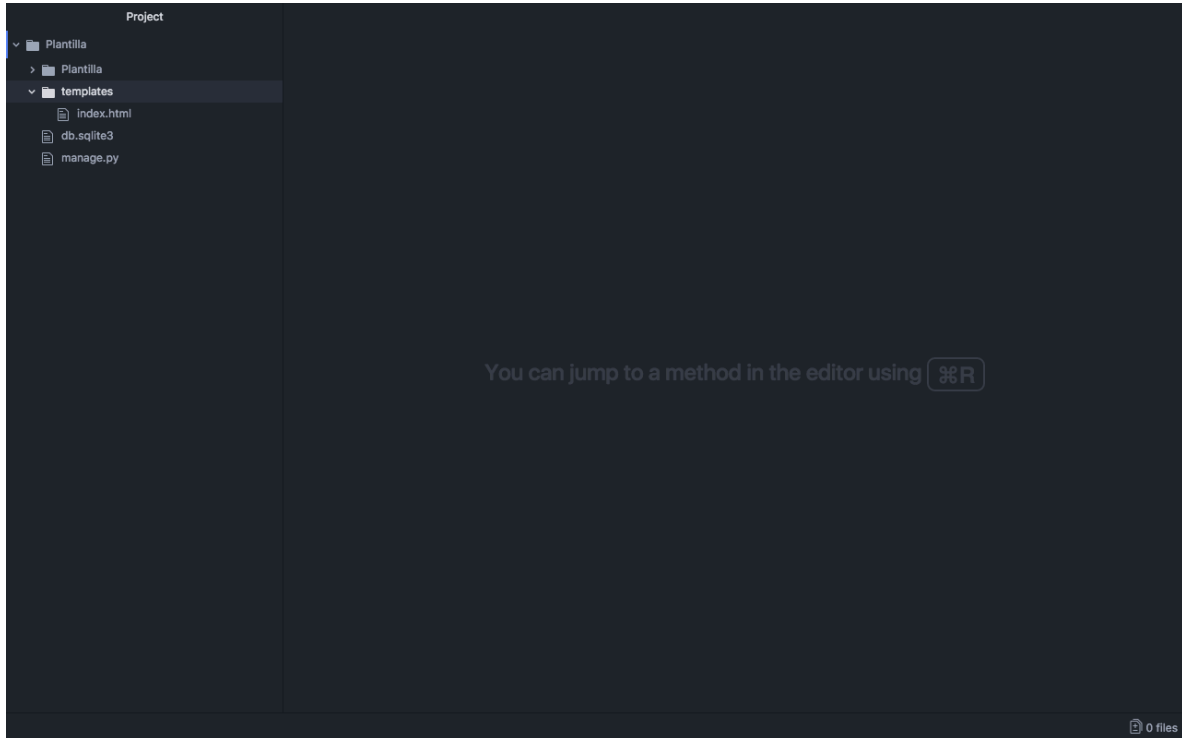
Crearemos un proyecto

```
django-admin startproject Plantilla
```

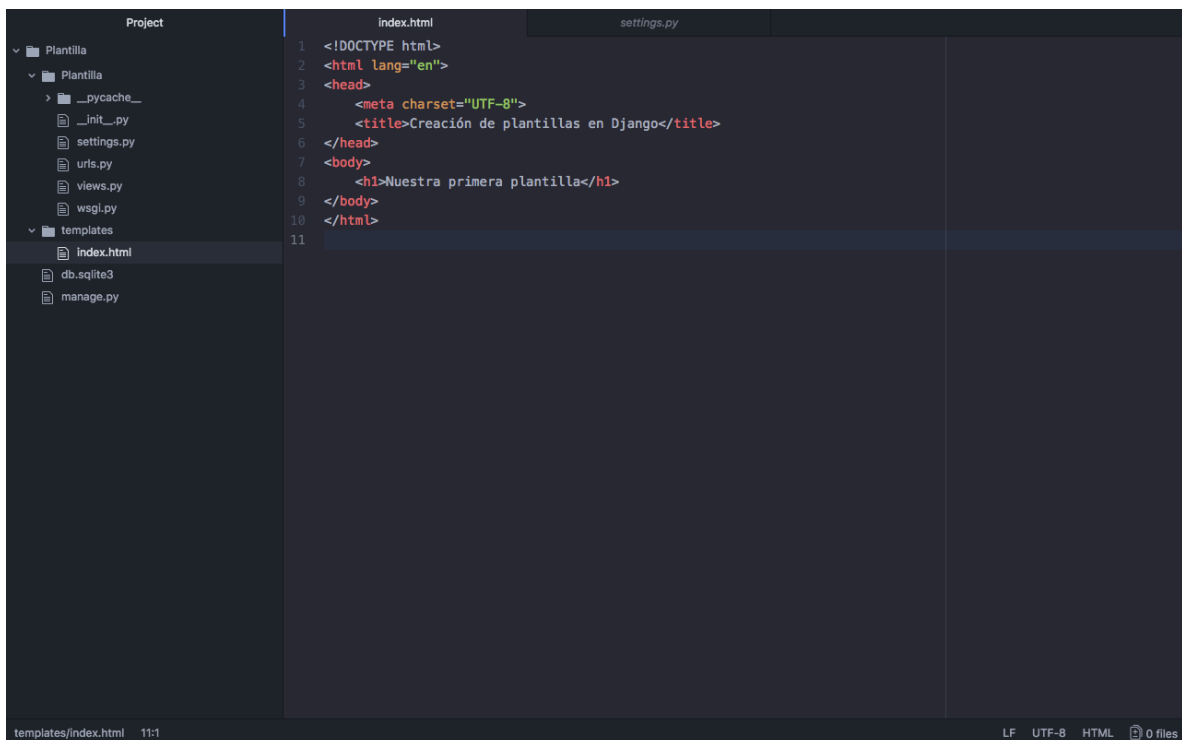
A continuación, creamos un carpeta llamada templates, deberías quedar con esta estructura



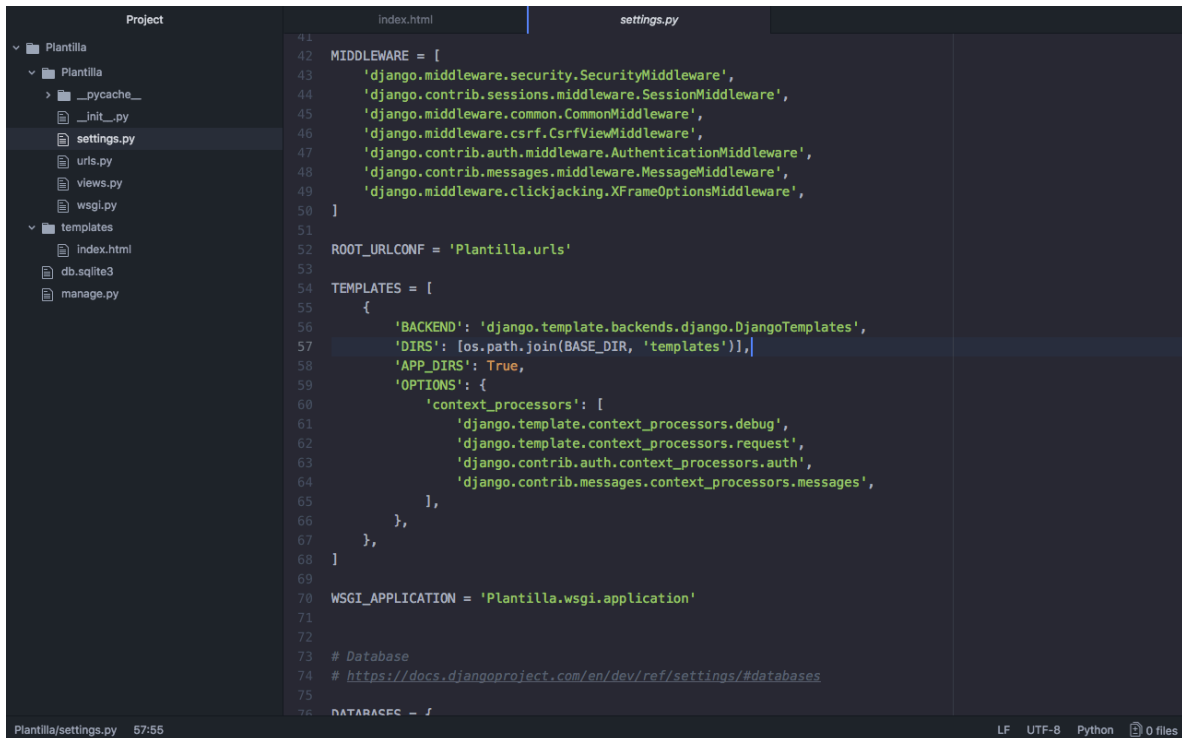
En el interior de la carpeta templates agregamos un archivo llamado index.html



Escribimos los siguiente en index.html



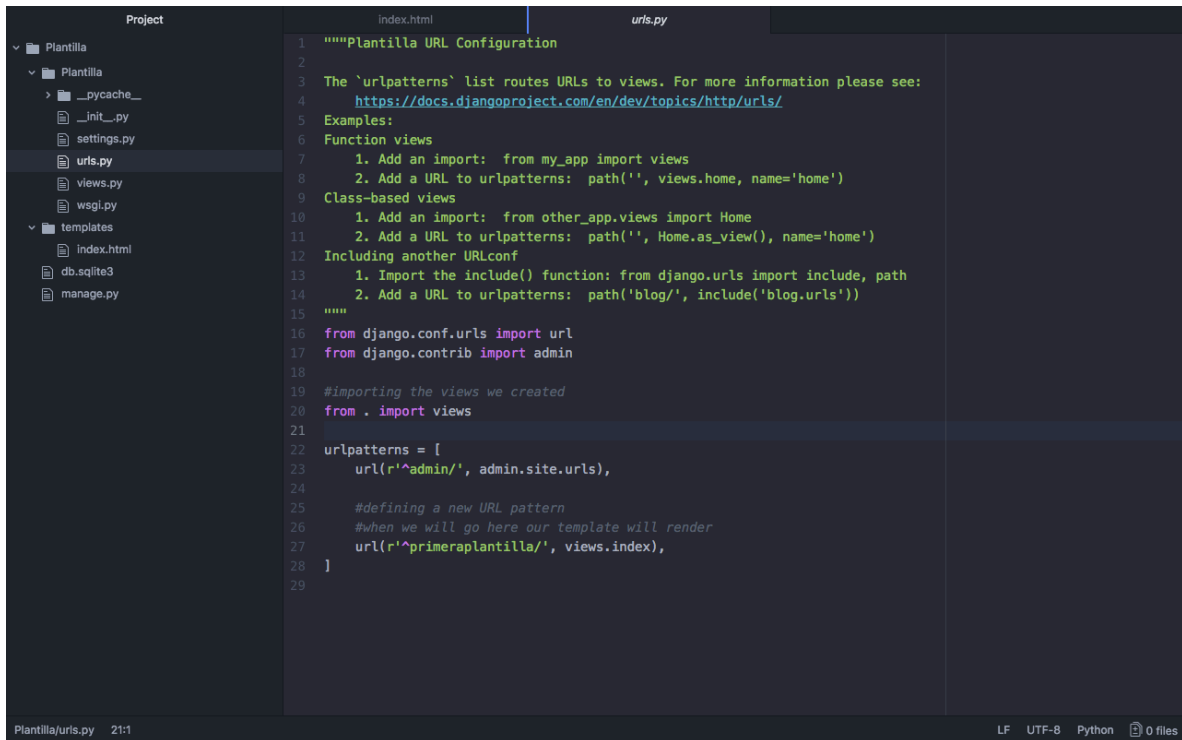
En setttings.py agregamos en DIRS la carpeta que acabamos de crear como ruta para las plantillas



The screenshot shows a code editor with a project named 'Plantilla'. The file explorer on the left shows the project structure, including a 'templates' directory. The main editor displays the 'settings.py' file, which is configured for Django. The 'DIRS' setting is updated to include the 'templates' directory. The 'context_processors' list is also shown.

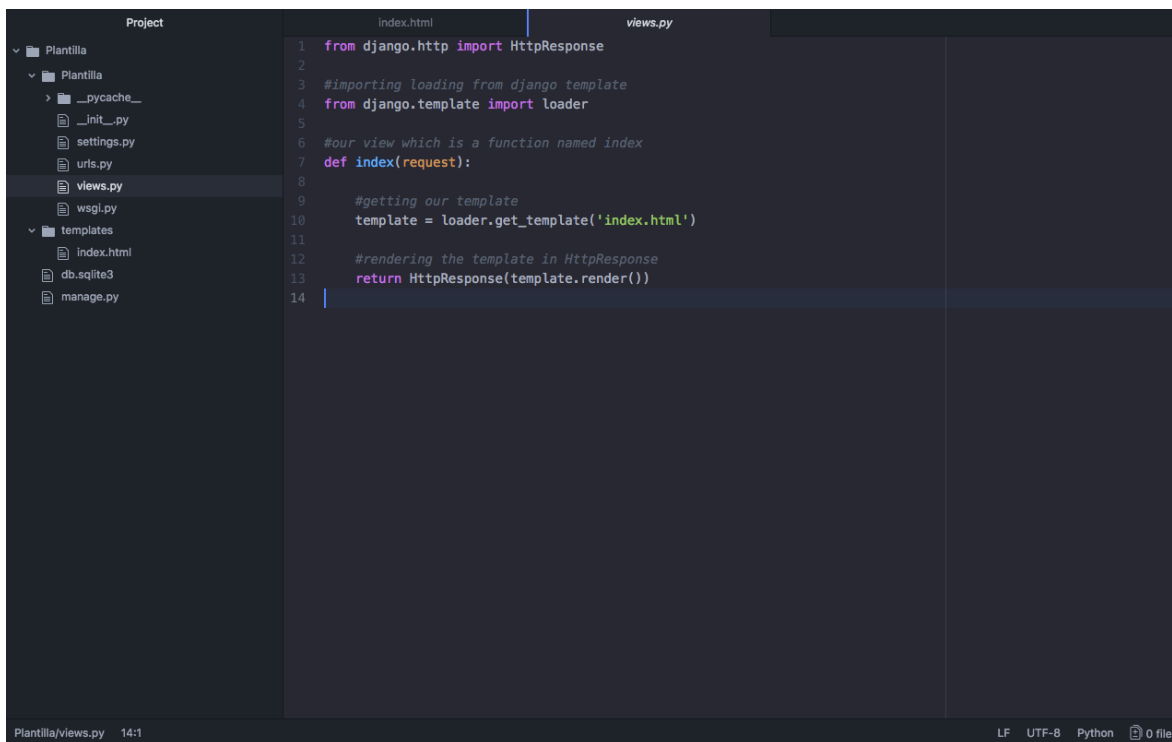
```
41
42 MIDDLEWARE = [
43     'django.middleware.security.SecurityMiddleware',
44     'django.contrib.sessions.middleware.SessionMiddleware',
45     'django.middleware.common.CommonMiddleware',
46     'django.middleware.csrf.CsrfViewMiddleware',
47     'django.contrib.auth.middleware.AuthenticationMiddleware',
48     'django.contrib.messages.middleware.MessageMiddleware',
49     'django.middleware.clickjacking.XFrameOptionsMiddleware',
50 ]
51
52 ROOT_URLCONF = 'Plantilla.urls'
53
54 TEMPLATES = [
55     {
56         'BACKEND': 'django.template.backends.django.DjangoTemplates',
57         'DIRS': [os.path.join(BASE_DIR, 'templates')],
58         'APP_DIRS': True,
59         'OPTIONS': {
60             'context_processors': [
61                 'django.template.context_processors.debug',
62                 'django.template.context_processors.request',
63                 'django.contrib.auth.context_processors.auth',
64                 'django.contrib.messages.context_processors.messages',
65             ],
66         },
67     },
68 ]
69
70 WSGI_APPLICATION = 'Plantilla.wsgi.application'
71
72
73 # Database
74 # https://docs.djangoproject.com/en/dev/ref/settings/#databases
75
76 DATABASES = {
```

En urls.py agregamos la ruta que debería renderizar el template html que acabamos de crear



```
1 """Plantilla URL Configuration
2
3 The 'urlpatterns' list routes URLs to views. For more information please see:
4     https://docs.djangoproject.com/en/dev/topics/http/urls/
5 Examples:
6 Function views
7     1. Add an import: from my_app import views
8     2. Add a URL to urlpatterns: path('', views.home, name='home')
9 Class-based views
10    1. Add an import: from other_app.views import Home
11    2. Add a URL to urlpatterns: path('', Home.as_view(), name='home')
12 Including another URLconf
13    1. Import the include() function: from django.urls import include, path
14    2. Add a URL to urlpatterns: path('blog/', include('blog.urls'))
15 """
16 from django.conf.urls import url
17 from django.contrib import admin
18
19 #importing the views we created
20 from . import views
21
22 urlpatterns = [
23     url(r'^admin/', admin.site.urls),
24
25     #defining a new URL pattern
26     #when we will go here our template will render
27     url(r'^primeraplantilla/', views.index),
28 ]
29
```

En views.py lo siguiente



```
Project
├── Plantilla
│   ├── __pycache__
│   ├── __init__.py
│   ├── settings.py
│   ├── urls.py
│   └── views.py
├── wsgi.py
└── templates
    ├── index.html
    ├── db.sqlite3
    └── manage.py
```

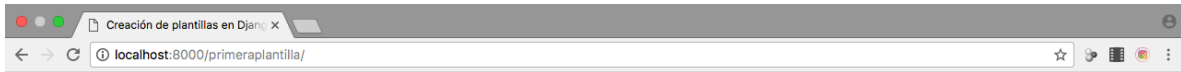
```
1 from django.http import HttpResponse
2
3 #importing loading from django template
4 from django.template import loader
5
6 #our view which is a function named index
7 def index(request):
8
9     #getting our template
10    template = loader.get_template('index.html')
11
12    #rendering the template in HttpResponse
13    return HttpResponse(template.render())
14
```

Plantilla/views.py 14:1 LF UTF-8 Python 0 files

Ya podemos correr nuestro proyecto

```
python manage.py runserver
```

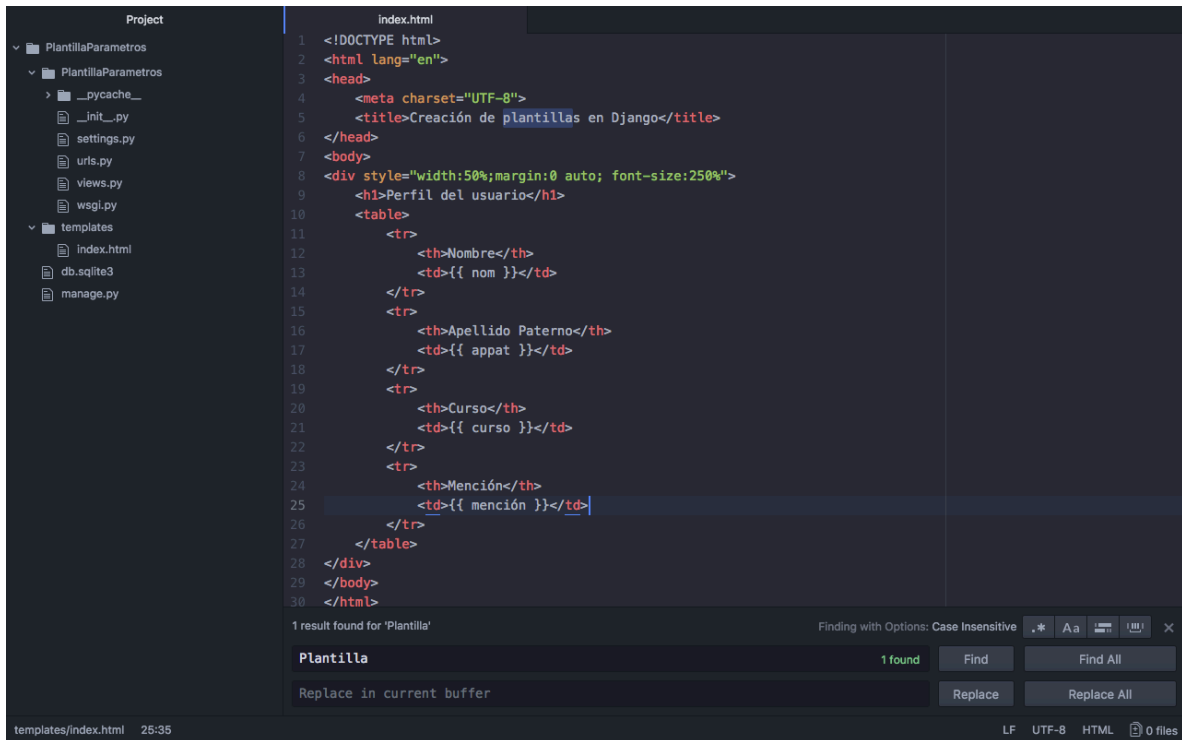
<http://localhost:8000/primeraplantilla/>



Nuestra primera plantilla

Pasaremos ahora parámetros, te recomiendo crear otro proyecto y configurarlo como lo hicimos en los pasos anteriores

Modificamos el archivo index.html



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Creación de plantillas en Django</title>
6 </head>
7 <body>
8   <div style="width:50%;margin:0 auto; font-size:250%">
9     <h1>Perfil del usuario</h1>
10    <table>
11      <tr>
12        <th>Nombre</th>
13        <td>{{ nom }}</td>
14      </tr>
15      <tr>
16        <th>Apellido Paterno</th>
17        <td>{{ appat }}</td>
18      </tr>
19      <tr>
20        <th>Curso</th>
21        <td>{{ curso }}</td>
22      </tr>
23      <tr>
24        <th>Mención</th>
25        <td>{{ mención }}</td>
26      </tr>
27    </table>
28  </div>
29 </body>
30 </html>
```

1 result found for 'Plantilla'

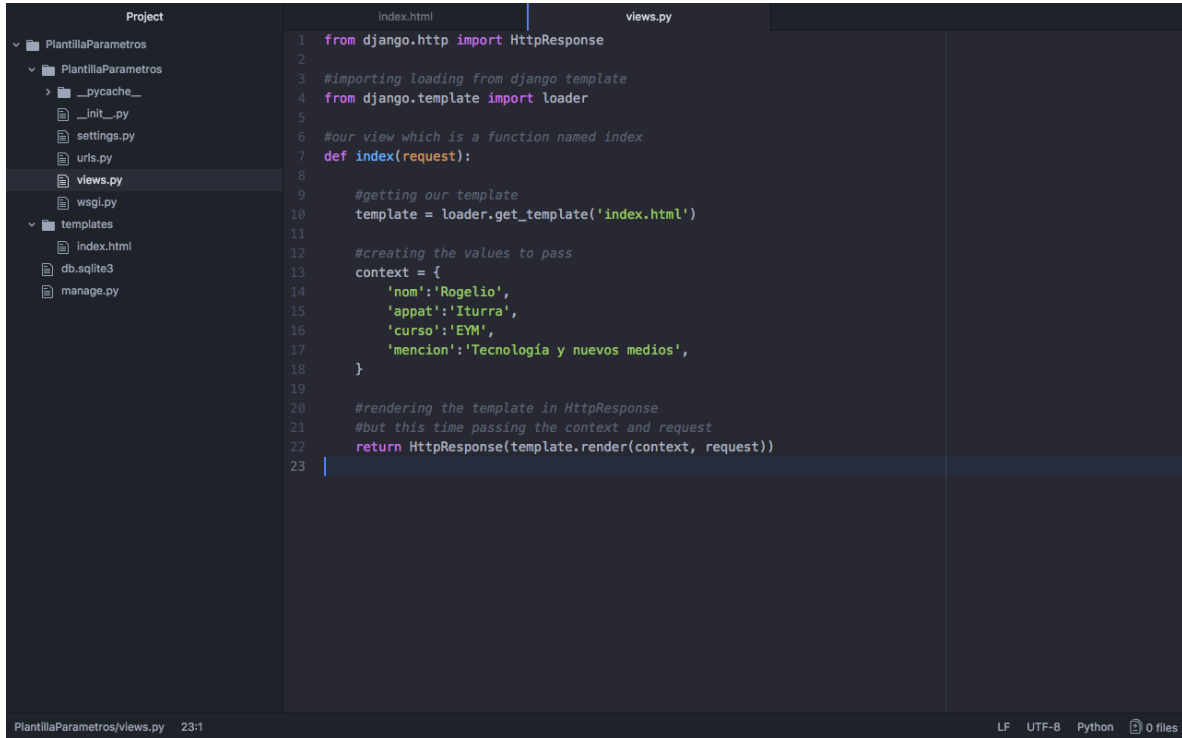
Finding with Options: Case Insensitive

Plantilla 1 found Find Find All

Replace in current buffer Replace Replace All

templates/index.html 26:35 LF UTF-8 HTML 0 files

Editamos views.py

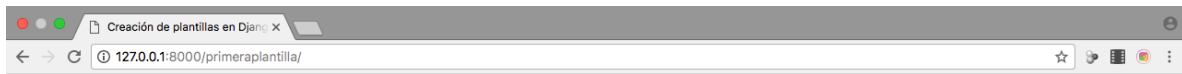


```
1 from django.http import HttpResponse
2
3 #importing loading from django template
4 from django.template import loader
5
6 #our view which is a function named index
7 def index(request):
8
9     #getting our template
10    template = loader.get_template('index.html')
11
12    #creating the values to pass
13    context = {
14        'nom': 'Rogelio',
15        'appat': 'Iturra',
16        'curso': 'EYM',
17        'mencion': 'Tecnología y nuevos medios',
18    }
19
20    #rendering the template in HttpResponse
21    #but this time passing the context and request
22    return HttpResponse(template.render(context, request))
23
```

PlantillaParametros/views.py 23:1 LF UTF-8 Python 0 files


```
python manage.py runserver
```

<http://localhost:8000/primeraplantilla/>



Perfil del usuario

Nombre	Rogelio
Apellido Paterno	Iturra
Curso	EYM
Mención	Tecnología y nuevos medios