

Guía para iniciar con el proyecto final para los programadores en Python.

Ejercicio para la Parcial.

El trabajo debe realizarse de forma individual.

No se permite dos trabajos en una misma computadora.

Práctica

Instalar las librerías de pillow y reportlab

Crear el proyecto con tu nombre, por ejemplo proyecto_moises y la app para gestionar clientes:

El sistema debe tener CRUD de clientes y generar un PDF

Iniciar Sesión

Username:

Password:

Nuevo Cliente

Cédula:

Nombre:

Apellido:

Dirección:

564 General C. Vega

Teléfono:

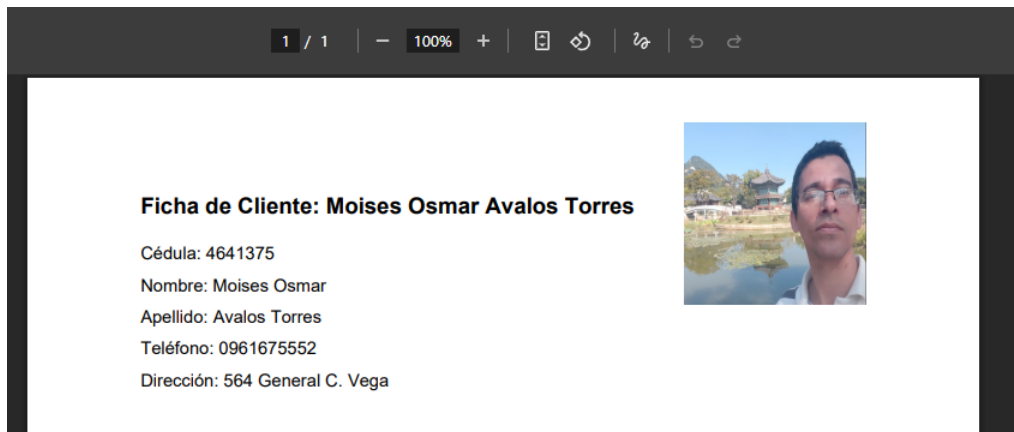
Foto: foto.png

Proyecto-TS

Lista de Clientes

[+ Nuevo Cliente](#)

Cédula	Nombre	Teléfono	Acciones
4641375	Moises Osmar Avalos Torres	0961675552	Ver Editar Eliminar

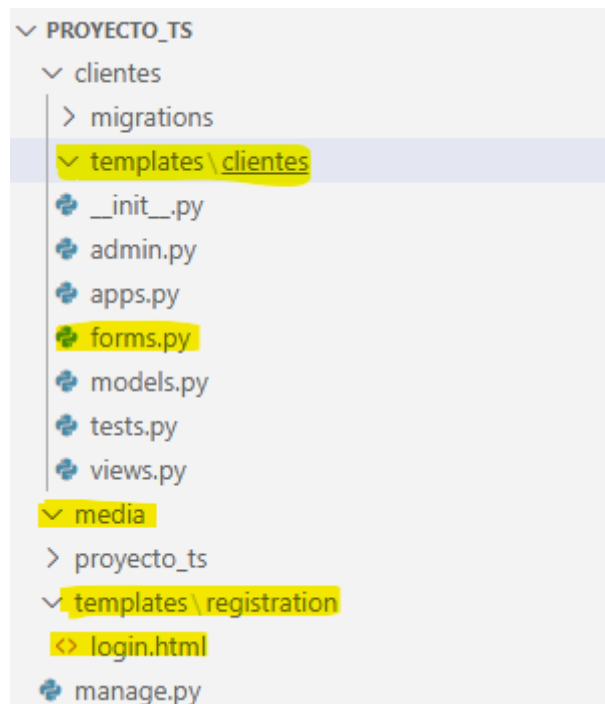


```
python> python -m django startproject proyecto_ts
```

```
ion> cd .\proyecto_ts\
```

```
ion\proyecto_ts> python manage.py startapp clientes
```

Crea los directorios y archivos



Agrega las configuraciones para la aplicación, directorio static y templates en setting.py

Ajusta la base de datos según el motor seleccionado: Mysql, Sqlserver, Oracle, PostgreSQL.

models.py para clientes

models.py 3 ×

clientes > models.py > Cliente

```
1 from django.db import models
2 from django.core.validators import RegexValidator
3
4 numeric_validator = RegexValidator(r'^\d+$', 'Solo dígitos permitidos')
5
6 class Cliente(models.Model):
7     cedula = models.CharField(
8         max_length=20,
9         primary_key=True,
10        validators=[numeric_validator],
11        verbose_name='Cédula'
12    )
13     nombre = models.CharField(max_length=100)
14     apellido = models.CharField(max_length=100)
15     direccion = models.TextField(blank=True)
16     telefono = models.CharField(max_length=20, unique=True, validators=[numeric_validator])
17     foto = models.ImageField(upload_to='clientes_fotos/', null=True, blank=True)
18
19
20     created_at = models.DateTimeField(auto_now_add=True)
21     updated_at = models.DateTimeField(auto_now=True)
22
23
24     def __str__(self):
25         return f"{self.cedula} - {self.nombre} {self.apellido}"
```

clientes > admin.py > ClienteAdmin

```
1 from django.contrib import admin
2 from .models import Cliente
3
4 @admin.register(Cliente)
5 class ClienteAdmin(admin.ModelAdmin):
6     list_display = ('cedula', 'nombre', 'apellido', 'telefono')
7     search_fields = ('cedula', 'nombre', 'apellido', 'telefono')
```

```
forms.py 2 x
clientes > forms.py > ClienteForm > Meta
1  # clientes/forms.py
2  from django import forms
3  from .models import Cliente
4
5
6  class ClienteForm(forms.ModelForm):
7
8      class Meta:
9          model = Cliente
10         fields = ['cedula', 'nombre', 'apellido', 'direccion', 'telefono', 'foto']
11
12
13     def clean_telefono(self):
14         telefono = self.cleaned_data.get('telefono')
15         # ejemplo: quitar espacios
16         telefono = telefono.replace(' ', '')
17         return telefono
```

```
urls.py 5 x
proyecto_ts > urls.py > ...
1  from django.contrib import admin
2  from django.urls import path, include
3  from django.conf import settings
4  from django.conf.urls.static import static
5
6
7  urlpatterns = [
8      path('admin/', admin.site.urls),
9      path('accounts/', include('django.contrib.auth.urls')), # login/logout/password reset
10     path('', include(('clientes.urls', 'clientes'), namespace='clientes')),
11 ]
12
13
14 if settings.DEBUG:
15     urlpatterns += static(settings.MEDIA_URL, document_root=settings.MEDIA_ROOT)
```

```
clientes > urls.py > ...
1  # clientes/urls.py
2  from django.urls import path
3  from . import views
4
5  app_name = 'clientes'
6
7  urlpatterns = [
8      path('', views.ClienteListView.as_view(), name='dashboard'),
9      path('nuevo/', views.ClienteCreateView.as_view(), name='cliente_nuevo'),
10     path('editar/<str:pk>/', views.ClienteUpdateView.as_view(), name='cliente_editar'),
11     path('borrar/<str:pk>/', views.ClienteDeleteView.as_view(), name='cliente_borrar'),
12     path('ver/<str:pk>/', views.ClienteDetailView.as_view(), name='cliente_ver'),
13     path('ver/<str:pk>/pdf/', views.generar_pdf_cliente, name='cliente_pdf'),
14 ]
```

views.py 9+ X

clientes > views.py > generar_pdf_cliente

```
20
21 class ClienteListView(LoginRequiredMixin, ListView):
22     model = Cliente
23     template_name = 'clientes/cliente_list.html'
24     context_object_name = 'clientes'
25     paginate_by = 20
26
27
28 class ClienteCreateView(LoginRequiredMixin, CreateView):
29     model = Cliente
30     form_class = ClienteForm
31     template_name = 'clientes/cliente_form.html'
32     success_url = reverse_lazy('clientes:dashboard')
33
34
35 class ClienteUpdateView(LoginRequiredMixin, UpdateView):
36     model = Cliente
37     form_class = ClienteForm
38     template_name = 'clientes/cliente_form.html'
39     success_url = reverse_lazy('clientes:dashboard')
40
41
42 class ClienteDeleteView(LoginRequiredMixin, DeleteView):
43     model = Cliente
44     template_name = 'clientes/cliente_confirm_delete.html'
45     success_url = reverse_lazy('clientes:dashboard')
46
47
48 class ClienteDetailView(LoginRequiredMixin, DetailView):
49     model = Cliente
50     template_name = 'clientes/cliente_detail.html'
51     context_object_name = 'cliente'
52
53
```

Siguiendo en views.py

```
54 @login_required
55 def generar_pdf_cliente(request, pk):
56     cliente = get_object_or_404(Cliente, pk=pk)
57
58
59     buffer = BytesIO()
60     p = canvas.Canvas(buffer, pagesize=A4)
61     width, height = A4
62
63
64     x_margin = 25 * mm
65     y = height - 30 * mm
66
67
68     p.setFont('Helvetica-Bold', 14)
69     p.drawString(x_margin, y, f"Ficha de Cliente: {cliente.nombre} {cliente.apellido}")
70     y -= 10 * mm
71
72
73     p.setFont('Helvetica', 11)
74     p.drawString(x_margin, y, f"Cédula: {cliente.cedula}")
75     y -= 7 * mm
76     p.drawString(x_margin, y, f"Nombre: {cliente.nombre}")
77     y -= 7 * mm
78     p.drawString(x_margin, y, f"Apellido: {cliente.apellido}")
79     y -= 7 * mm
80     p.drawString(x_margin, y, f"Teléfono: {cliente.telefono}")
81     y -= 7 * mm
82     p.drawString(x_margin, y, f"Dirección: {cliente.direccion}")
83     y -= 12 * mm
84
85
86     # Si tiene foto, intentar ponerla a la derecha
87     if cliente.foto:
88         try:
89             foto_path = os.path.join(settings.MEDIA_ROOT, cliente.foto.name)
90             # ajustar tamaño de imagen
91             img = RLImage(foto_path)
92             img.drawHeight = 40 * mm
93             img.drawWidth = 40 * mm
94             img.hAlign = 'RIGHT'
95             img.wrapOn(p, width, height)
96             img.drawOn(p, width - x_margin - (40 * mm), height - 50 * mm)
97         except Exception as e:
98             # si hay problema con la imagen, pasamos
99             pass
100
101     p.showPage()
102     p.save()
103
104     buffer.seek(0)
105     return HttpResponse(buffer, content_type='application/pdf')
```

templates\clientes

- cliente_confirm_delete.html
- cliente_detail.html
- cliente_form.html
- cliente_list.html

cliente_list.html

clientes > templates > clientes > cliente_list.html > table.table.table-striped > thead > tr > th

```

1 {% extends "base.html" %}
2 {% block title %}Clientes{% endblock %}
3 {% block content %}
4 <h2>Lista de Clientes</h2>
5 <a href="{% url 'clientes:cliente_nuevo' %}" class="btn btn-success mb-3">+ Nuevo Cliente</a>
6 <table class="table table-striped">
7     <thead>
8         <tr>
9             <th>Cédula</th> <th>Nombre</th> <th>Teléfono</th> <th>Acciones</th>
10        </tr>
11    </thead>
12    <tbody>
13        {% for cliente in object_list %}
14            <tr>
15                <td>{{ cliente.cedula }}</td>
16                <td>{{ cliente.nombre }} {{ cliente.apellido }}</td>
17                <td>{{ cliente.telefono }}</td>
18                <td>
19                    <a href="{% url 'clientes:cliente_ver' cliente.pk %}" class="btn btn-info btn-sm">Ver</a>
20                    <a href="{% url 'clientes:cliente_editar' cliente.pk %}" class="btn btn-warning btn-sm">Editar</a>
21                    <a href="{% url 'clientes:cliente_borrar' cliente.pk %}" class="btn btn-danger btn-sm">Eliminar</a>
22                </td>
23            </tr>
24        {% empty %}
25            <tr>
26                <td colspan="4" class="text-center">No hay clientes registrados.</td>
27            </tr>
28        {% endfor %}
29    </tbody>
30 </table>
31 {% endblock %}

```

cliente_form.html

clientes > templates > clientes > cliente_form.html > form > a.btn.btn-secondary

```

1 {% extends "base.html" %}
2 {% block title %}Formulario Cliente{% endblock %}
3 {% block content %}
4 <h2>{% if object %}Editar Cliente{% else %}Nuevo Cliente{% endif %}</h2>
5 <form method="post" enctype="multipart/form-data">
6     {% csrf_token %}
7     {{ form.as_p }}
8     <button type="submit" class="btn btn-primary">Guardar</button>
9     <a href="{% url 'clientes:dashboard' %}" class="btn btn-secondary">Cancelar</a>
10 </form>
11 {% endblock %}
12

```

<> cliente_detail.html ●

clientes > templates > clientes > <> cliente_detail.html > div.card > div.card-body > a.btn.btn-outline-primary

```

1  {% extends "base.html" %}
2  {% block title %}Detalle Cliente{% endblock %}
3  {% block content %}
4  <h2>Detalle de Cliente</h2>
5  <div class="card" style="max-width: 400px;">
6      {% if cliente.foto %}
7          
8      {% endif %}
9      <div class="card-body">
10         <h5 class="card-title">{{ cliente.nombre }} {{ cliente.apellido }}</h5>
11         <p class="card-text"><strong>Cédula:</strong> {{ cliente.cedula }}</p>
12         <p class="card-text"><strong>Dirección:</strong> {{ cliente.direccion }}</p>
13         <p class="card-text"><strong>Teléfono:</strong> {{ cliente.telefono }}</p>
14         <a href="{% url 'clientes:cliente_pdf' cliente.pk %}" class="btn btn-outline-primary">
15             Generar PDF
16         </a>
17     </div>
18 </div>
19 <a href="{% url 'clientes:dashboard' %}" class="btn btn-secondary mt-3">Volver</a>
20 {% endblock %}
21

```

<> cliente_confirm_delete.html ×

clientes > templates > clientes > <> cliente_confirm_delete.html > ...

```

1  {% extends "base.html" %}
2  {% block title %}Eliminar Cliente{% endblock %}
3  {% block content %}
4  <h2>¿Seguro que deseas eliminar este cliente?</h2>
5  <p><strong>{{ object.nombre }} {{ object.apellido }}</strong> ({{ object.cedula }})</p>
6  <form method="post">
7      {% csrf_token %}
8      <button type="submit" class="btn btn-danger">Sí, eliminar</button>
9      <a href="{% url 'clientes:cliente_list' %}" class="btn btn-secondary">Cancelar</a>
10 </form>
11 {% endblock %}
12

```


v templates
 v registration
 <> login.html
 <> base.html

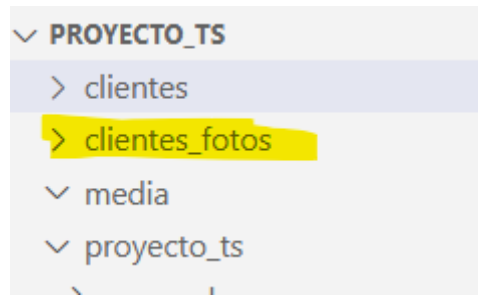
```

templates > <> base.html > ...
1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4      <meta charset="UTF-8">
5      <title>{% block title %}Proyecto-TS{% endblock %}</title>
6      <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet">
7  </head>
8  <body>
9      <nav class="navbar navbar-expand-lg navbar-dark bg-dark mb-4">
10         <div class="container-fluid">
11             <a class="navbar-brand" href="{% url 'clientes:dashboard' %}">Proyecto-TS</a>
12             <div class="collapse navbar-collapse">
13                 <ul class="navbar-nav ms-auto">
14                     {% if user.is_authenticated %}
15                     <li class="nav-item">
16                         <a class="nav-link" href="#">{{ user.username }}</a>
17                     </li>
18                     <li class="nav-item">
19                         <a class="nav-link" href="{% url 'logout' %}">Salir</a>
20                     </li>
21                     {% else %}
22                     <li class="nav-item">
23                         <a class="nav-link" href="{% url 'login' %}">Ingresar</a>
24                     </li>
25                     {% endif %}
26                 </ul>
27             </div>
28         </div>
29     </nav>
30     <div class="container">
31         {% block content %}{% endblock %}
32     </div>
33 </body>
34 </html>
  
```

<> login.html X

```

templates > registration > <> login.html > ...
1  {% extends "base.html" %}
2  {% block title %}Login{% endblock %}
3  {% block content %}
4      <div class="row justify-content-center">
5          <div class="col-md-4">
6              <h2 class="mb-3 text-center">Iniciar Sesión</h2>
7              <form method="post">
8                  {% csrf_token %}
9                  {{ form.as_p }}
10                 <button type="submit" class="btn btn-primary w-100">Entrar</button>
11             </form>
12         </div>
13     </div>
14 {% endblock %}
  
```



Probar el sistema