

## Manual

Nombre: Tipanluisa Walter

Creamos la clase Tipo con sus respectivos atributos

```
class Tipo(models.Model):
    id=models.AutoField(primary_key=True)
    nombre=models.CharField(max_length=150)
    descripcion=models.TextField(max_length=200)
    def __str__(self):
        fila="{0} => {1} => {2}"
        return fila.format(self.id,self.descripcion, self.nombre)
```

En el modelo Cliente actualizamos y agregamos el tipo

// Importante designar como null y blank en modo true para poder realizar las migraciones

```
class Cliente(models.Model):
    id=models.AutoField(primary_key=True)
    cedula=models.CharField(max_length=10)
    apellido=models.CharField(max_length=100)
    nombre=models.CharField(max_length=100)
    direccion=models.TextField(max_length=100)
    fechas_nacimiento=models.DateField()
    correo=models.EmailField()
    tipo=models.ForeignKey(Tipo, null=True, blank=True, on_delete=models.PROTECT)
    def __str__(self):
        fila="{0} => {1} => {2} => {3} => {4} => {5} => {6} => {7}"
        return fila.format(self.id, self.tipo,self.cedula,self.apellido,self.nombre,self.direccion,self.fechas_nacimiento,self.correo)
```

Realizamos las migraciones con makemigrations

```
python manage.py makemigrations
Migrations for 'Facturacion':
  Aplicaciones/Facturacion/migrations/0002_tipocliente_cliente_tipo_cliente.py
  - Create model TipoCliente
  - Add field tipo_cliente to cliente
python manage.py migrate
Operations to perform:
  Apply all migrations: Facturacion, admin, auth, contenttypes, sessions
Running migrations:
  Applying Facturacion.0002_tipocliente_cliente_tipo_cliente... OK
python manage.py runserver
```

Registramos en el archivo admin.py el modelo Tipo que creamos en models

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

Corremos el servidor y evidenciamos que ya está nuestra nueva tabla creada como modelo y podemos agregar la información a nuestra disposición

FACTURACION		
Cientes	+ Añadir	Modificar
Típos	+ Añadir	Modificar

### Añadir tipo

Nombre:

Descripcion:

### Seleccione tipo a modificar

Acción:  Ir seleccionados 0 de 3

- ☐ TIPO
- ☐ 3 => una vez al anio => Ocacional
- ☐ 2 => una vez al mes => Regular
- ☐ 1 => Clientes que compran a diario => Frecuente

3 tipos

Definimos en nuestra lista de clientes para la extraccion de datos en este caso se llamada tipoBdd

```

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

```

Actualizamos nuestra funcion para guardar clientes, agregamos el tipo y el tipoSeleccionado que capturara en la base de datos

```
def guardarCliente (request):
    tipo=request.POST['id_tipo']
    tipoSeleccionado=Tipo.objects.get(id=tipo)
    cedula=request.POST['cedula']
    nombre=request.POST['nombre']
    apellido=request.POST['apellido']
    fechas_nacimiento=request.POST['fechas_nacimiento']
    correo=request.POST['correo']
    direccion=request.POST['direccion']
    nuevoCliente=Cliente.objects.create(cedula=cedula, nombre=nombre, apellido=apellido, fechas_nacimiento=fechas_nacimiento, correo=correo,
    direccion=direccion,tipoSeleccionado=tipoSeleccionado,)
    messages.success(request, 'Cliente guardado correctamente')
    return redirect ('/')
```

Realizamos lo mismo para el proceso de actualizacion de cliente y editar cliente

```
def editarCliente (request, id):
    clienteEditar=Cliente.objects.get(id=id)
    tipoBdd=Tipo.objects.all()
    return render (request, 'editarCliente.html', {'cliente': clienteEditar, 'tipos': tipoBdd})
```

```
def procesarActualizacionCliente (request):
    id=request.POST['id']
    id_tipo=request.POST['id_tipo']
    tipoSeleccionado=Tipo.objects.get(id=id_tipo)
    cedula=request.POST['cedula']
    nombre=request.POST['nombre']
    apellido=request.POST['apellido']
    fechas_nacimiento=request.POST['fechas_nacimiento']
    correo=request.POST['correo']
    direccion=request.POST['direccion']
    # Actualizar el cliente
    cliente=Cliente.objects.get(id=id)
    cliente.cedula=cedula
    cliente.tipo=tipoSeleccionado
    cliente.nombre=nombre
    cliente.apellido=apellido
    cliente.fechas_nacimiento=fechas_nacimiento
    cliente.correo=correo
    cliente.direccion=direccion
    cliente.save()
    messages.success(request, 'Cliente actualizado correctamente')
    return redirect ('/')
```

En nuestro formulario aumentamos un select con jinja2 y llamamos a nuestros datos que insertamos en el admin de django

```
<select name="id_tipo" required class="form-control m-3" id="id_tipo">
    <option value="" disabled>— Seleccione una opcion --</option>
    {% for tipo in tipos %}
    <option value="{{tipo.id}}">{{tipo.nombre}}</option>
    {% endfor %}
</select>
```

Realizamos lo mismo en el formulario de editar debemos ingresar obligatoriamente el id y el name en este caso se llama id\_tipo

```
ou, hace 1 segundo • Uncommitted changes
<label for="" class="form-label">Tipo de cliente</label>
<select name="id_tipo" required class="form-control m-1" id="id_tipo">
  <option value="" disabled>— Seleccione una opcion --</option>
  {% for tipo in tipos %}
    <option value="{{tipo.id}}">{{tipo.nombre}}</option>
  {% endfor %}
</select>
<script>
  document.getElementById('id_tipo').value = '{{cliente.tipo.id}}'
</script>
```

Ahora podemos observar que ya nos sale la informacion tanto en el formulario como en la tabla.

Ademas de ello podemos agregar eliminar editar la informacion que nosotros requerimos

Tipo de cliente

Frecuente

Cedula

Ingrese la cedula

Nombres

Ingrese los nombres

Apellido

Ingrese los apellidos

Fecha de nacimiento

dd/mm/aaaa

Correo Electronico

Ingrese su Correo Elet

Direccion

Ingrese su direccion

Guardar

Cancelar

Listado Clientes



Id	Cedula	Nombres	Apellidos	Fecha de Nacimiento	Correo Electronico	Direccion	Tipo	Acciones
26	0504235714	Walter Alexander	Tipanluisa Tipanluisa	19 de enero de 2024	alexg@gmail.com	Regular	Sangolqo	<div>Editar</div> <div>Eliminar</div>
27	0504235714	Alex Dario	Tipanluisa Tapia	14 de enero de 2024	alexgdyt22@gmail.com	Regular	Latacunga	<div>Editar</div> <div>Eliminar</div>
28	0504235714	Alex	Tipanluisa	8 de enero de 2024	alexgdyt22@gmail.com	Ocacional	Latacunga	<div>Editar</div> <div>Eliminar</div>
29	0504235714	Alex	Tipanluisa	6 de enero de 2024	maria@hotmail.com	Frecuente	dwedw	<div>Editar</div> <div>Eliminar</div>

Editar Cliente

Tipo de cliente

Regular

Cedula

0504235714

Nombres

Walter Alexander

Apellido

Tipanluisa Tipanluisa

Fecha de nacimiento

19/01/2024

Correo Electronico

alexg@gmail.com

Direccion

Sangolqo

Actualizar

Cancelar