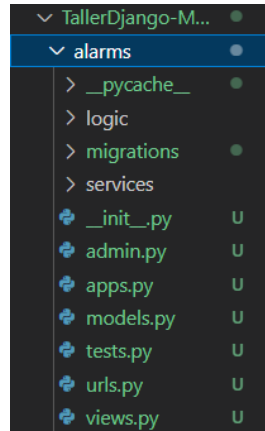


Taller 2 - Modelos en Django

Procedimiento 1

- Creación app Alarm



Relación ManyToMany

```
1 from django.db import models
2 from measurements.models import Measurement
3
4 class Alarm(models.Model):
5     name = models.CharField(max_length=50)
6     measurement = models.ManyToManyField(Measurement)
7     def __str__(self) -> str:
8         return '{}'.format(self.name)
```

Add alarm

Name:

Alarm02

Measurement:

1.0 Degrees Celsius
1.0 Degrees Celsius
1.0 mg/L
12.0 Degrees Celsius
12.0 Degrees Celsius
43.0 mg/L
54.0 mg/L
32.0 Degrees Celsius



Hold down "Control", or "Command" on a Mac, to select more than one.

- **¿Qué es un ORM?**

R/. ORM es un acrónimo de Object-Relational Mapping. Esta es una técnica que permite **mapear** objetos de una aplicación a registros en una base de datos relacional. Esto significa que puedes trabajar los datos en la base de datos más fácil porque se utilizan objetos en lugar de tener que escribir consultas SQL manualmente.

- **¿Cuáles son las ventajas y desventajas de usar un ORM de Django?**

Ventajas:

1. permite trabajar con la base de datos a nivel de objetos y no con consultas SQL, lo que resulta en un código más claro y legible.
2. Los ORM permiten cambiar de base de datos sin necesidad de modificar el código de la aplicación.
3. Trabajar con objetos en lugar de SQL manualmente reduce la posibilidad de cometer errores en la lógica de la aplicación.

Desventajas

1. Las consultas generadas por un ORM pueden ser más lentas que las consultas SQL manuales debido a la abstracción adicional que proporciona.
2. Los ORM a veces no soportan todas las características de una base de datos, lo que puede limitar la funcionalidad de la aplicación.
3. Trabajar con un ORM puede ser más complejo que trabajar directamente con SQL, especialmente para desarrolladores no familiarizados con el concepto de mapeo objeto-relacional.

- **Consigne el significado de DTO y DAO y explique cual pertenece al ORM de Django.**

DTO es un acrónimo de Data Transfer Object, que es un patrón de diseño que se utiliza para transferir datos entre capas de una aplicación. Un DTO es un objeto simple que contiene los datos que se van a transferir, y se utiliza para evitar la exposición de entidades de la base de datos directamente a la capa de presentación. Por otro lado, DAO es un acrónimo de Data Access Object, que es un patrón de diseño que se utiliza para abstraer el acceso a datos en una base de datos. Un DAO es responsable de realizar operaciones de base de datos como consultas, inserciones y actualizaciones.

En Django, el ORM se considera una implementación de los patrones DAO y DTO. La capa de modelos de Django es responsable de proporcionar una abstracción de los datos de la base de datos y de realizar operaciones de base de datos en nombre de la aplicación. Los modelos de Django se utilizan para definir los objetos de la aplicación y se mapean a tablas en la base de datos. De esta manera, Django actúa como un DAO que permite a la aplicación acceder a los datos de la base de datos de manera eficiente y segura, y también proporciona una forma de transferir datos a través de la aplicación como DTOs.

- ¿Qué alternativas hay al ORM integrado en Django para bases de datos relacionales?
 1. SQLAlchemy: es un ORM popular y altamente configurable para Python que admite una amplia variedad de bases de datos relacionales. A diferencia de algunos ORMs, SQLAlchemy no genera automáticamente el código SQL para usted. Esto significa que se tiene un mayor control sobre la forma en que se accede a la base de datos y puede ejecutar consultas más complejas. También está optimizado para un rendimiento óptimo y se ejecuta de manera eficiente en diferentes entornos.
 2. Peewee: es un ORM minimalista para Python que se enfoca en la simplicidad y la rapidez. Peewee tiene una sintaxis clara y fácil de entender, lo que lo hace ideal para principiantes y para proyectos pequeños. También es compatible con una amplia variedad de bases de datos relacionales, incluyendo PostgreSQL, MySQL, SQLite y Oracle.