



Universidad Autónoma de Baja California
Campus Tijuana
Facultad de contaduría y administración



Alumno:

García Bretado David Antonio

Grupo:

372

“Modelos”

Programación en Python

Tijuana B.C. a 23 de octubre de 2018

Modelos

Un modelo es la fuente única y definitiva de información sobre sus datos. Contiene los campos y comportamientos esenciales de los datos que está almacenando. En general, cada modelo se asigna a una sola tabla de base de datos.

Los basicos:

- Cada modelo es una clase de Python que subclases **django.db.models.Model**.
- Cada atributo del modelo representa un campo de base de datos.
- Con todo esto, Django le ofrece una API de acceso a la base de datos generada automáticamente; Consulte Hacer consultas .

Ejemplo rápido

Este modelo de ejemplo define a **Person**, que tiene una **first_name** y **last_name**:

```
from django.db import models

class Person(models.Model):

    first_name = models.CharField(max_length=30)

    last_name = models.CharField(max_length=30)
```

first_name y **last_name** son campos del modelo. Cada campo se especifica como un atributo de clase, y cada atributo se asigna a una columna de base de datos.

El **Person** modelo anterior crearía una tabla de base de datos como esta:

```
CREATE TABLE myapp_person (
    "id" serial NOT NULL PRIMARY KEY,
    "first_name" varchar(30) NOT NULL,
    "last_name" varchar(30) NOT NULL
);
```

Algunas notas técnicas:

- El nombre de la tabla **myapp_person**, se deriva automáticamente de algunos metadatos del modelo, pero se puede anular. Ver nombres de tablas para más detalles.
- Un **id** campo se agrega automáticamente, pero este comportamiento puede ser anulado. Ver campos de clave primaria automáticos .
- El SQL en este ejemplo está formateado usando la sintaxis de PostgreSQL, pero vale la pena notar que Django usa SQL adaptado al backend de base de datos especificado en su archivo de configuración **.CREATE TABLE**

Usando modelos

Una vez que haya definido sus modelos, debe decirle a Django que va a *usar* esos modelos. Para ello, edite el archivo de configuración y cambie la **INSTALLED_APPS** configuración para agregar el nombre del módulo que contiene su **models.py**.

Por ejemplo, si los modelos para su aplicación viven en el módulo **myapp.models**(la estructura del paquete que se crea para una aplicación mediante el script), debe leer, en parte:**manage.py startappINSTALLED_APPS**

```
INSTALLED_APPS = [  
    #...  
    'myapp',  
    #...  
]
```

Cuando agregue nuevas aplicaciones a **INSTALLED_APPS**, asegúrese de ejecutar , opcionalmente, hacer migraciones para ellas primero **.manage.py migrate****manage.py makemigrations**.

Fuentes

<https://docs.djangoproject.com/es/2.1/topics/db/models/>