

Django 存取資料庫

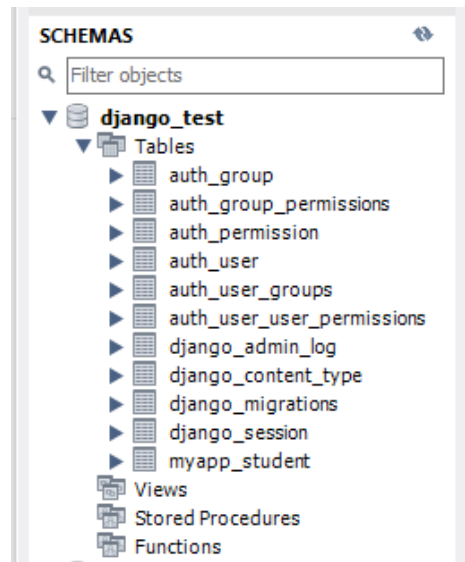
py manage.py migrate (執行資料庫遷移命令)

建立內建 APP 所需之資料表

```
(django) C:\Users\226083\django_test\myweb>py manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
  Applying auth.0010_alter_group_name_max_length... OK
  Applying auth.0011_update_proxy_permissions... OK
  Applying auth.0012_alter_user_first_name_max_length... OK
  Applying sessions.0001_initial... OK
```

=====

進入MYSQL，進去後就會看見多了一些東西



改動資料模型 `models.py`

資料模型預設是以應用程式作為對應的單位，開啟 `myapp\models.py`

並且在裡面建立一個資料表，如下

```
from django.db import models
from django.utils import timezone

# Create your models here.
class student(models.Model):
    SEX_CHOICES = [
        ('M', '男'),
        ('F', '女'),
    ]
    cName = models.CharField('姓名', max_length=20, null=False)
    cSex = models.CharField('性別', max_length=1, choices=SEX_CHOICES, default='', null=False)
    cBirthday = models.DateField('生日', null=False)
    cEmail = models.EmailField('Email', max_length=100, blank=True, default='')
    cPhone = models.CharField('手機', max_length=50, blank=True, default='')
    cAddr = models.CharField('地址', max_length=255, blank=True, default='')
    last_modified = models.DateTimeField('最後修改日期', auto_now = True)
    created = models.DateTimeField('保存日期', default = timezone.now)

    def __str__(self):
        return self.cName
```

儲存後，此時已完成異動，也就是發生了更改動作，此時資料庫尚未發生改變

補充：

Django Models :

定義裡面的屬性， Django 會依據這個建立資料表，以及資料表裡的欄位設定

資料型態	說明	參數
CharField	標題	<code>max_length=100</code> -- 標題不可以超過 100 個字元
TextField	內文	<code>blank=True</code> -- 非必填欄位（表單驗證時使用），預設所有欄位都是 <code>blank=False</code>
URLField	照片網址	同 content，非必填欄位
CharField	地點	同 title
DateTimeField	建立時間	<code>auto_now_add=True</code> -- 物件新增的時間。若想設成物件修改時間，則用 <code>auto_now=True</code>

∴ 補充：

+ ∴ Django Models :

定義裡面的屬性， Django 會依據這個建立資料表，以及資料表裡的欄位設定

資料型態	說明	參數
CharField	標題	<code>max_length=100</code> -- 標題不可以超過 100 個字元
TextField	內文	<code>blank=True</code> -- 非必填欄位（表單驗證時使用），預設所有欄位都是 <code>blank=False</code>
URLField	照片網址	同 content，非必填欄位
CharField	地點	同 title
DateTimeField	建立時間	<code>auto_now_add=True</code> -- 物件新增的時間。若想設成物件修改時間，則用 <code>auto_now=True</code>

=====

將模型的異動產生 migrations 遷移檔

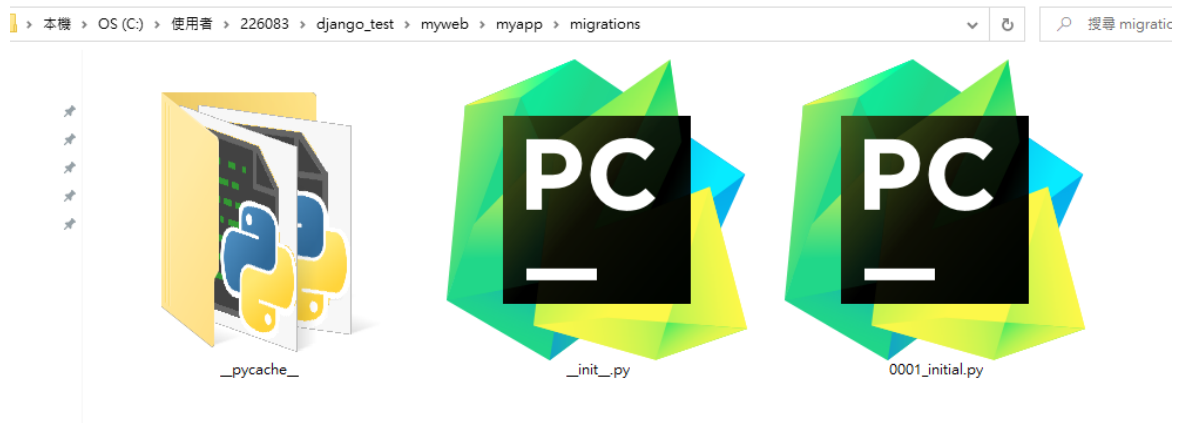
改動後，執行“makemigrations”命令(“python `manage.py` makemigrations”)

Django 會重新生成一個新的資料庫遷移檔案用來記錄表結構之間的差異，命名規則是對上一個 遷移檔案的序列號加1，如 0002_xxx、0003_xxx

```
(django) C:\Users\226083\django_test\myweb>py manage.py migrate myapp
Operations to perform:
  Apply all migrations: myapp
Running migrations:
  Applying myapp.0001_initial... OK
```

執行上述指令後，會在應用程式 myapp 下增加 migrations 目錄如下圖

遷移檔案為"0001_initial.py"



檢視指定的 migration 會產生的 SQL 敘述

上述完成改動資料models.py、改動後產生遷移檔案"0001_initial.py"

在正式執行遷移之前，可選擇先檢視執行遷移動作之後，會發生什麼事(預覽結果，不代表正式執行)

py manage.py sqlmigrate myapp 0001

```
(django) C:\Users\226083\django_test\myweb>py manage.py sqlmigrate myapp 0001
--
-- Create model student
--
CREATE TABLE `myapp_student` (`id` bigint AUTO_INCREMENT NOT NULL PRIMARY KEY, `cName` varchar(20) NOT NULL, `cSex` varchar(1) NOT NULL, `cBirthday` date NOT NULL, `cEmail` varchar(100) NOT NULL, `cPhone` varchar(50) NOT NULL, `cAddr` varchar(255) NOT NULL);
```

可以看到底下跳出一些code，稍加整理之後可得：

```
BEGIN;
--
-- Create model student
--
CREATE TABLE "myapp_student" (
  "id" integer NOT NULL PRIMARY KEY AUTOINCREMENT,
```

```
"cName" varchar(20) NOT NULL,  
"cSex" varchar(1) NOT NULL,  
"cBirthday" date NOT NULL,  
"cEmail" varchar(100) NOT NULL,  
"cPhone" varchar(50) NOT NULL,  
"cAddr" varchar(255) NOT NULL,  
"last_modified" datetime NOT NULL,  
"created" datetime NOT NULL  
);  
  
COMMIT;
```

這其實就是前面改動資料模型 `models.py` 做的事

可以一項項核對，看看是否有遺漏或出問題的部分

補充：

發現多了一行前面沒寫到的指令 `"id" integer NOT NULL PRIMARY KEY
AUTOINCREMENT, "`

那是因為以 `model.py` 新建資料庫欄位時，系統預設會在欄位最前面自動新增一個 `"id"`，所以可以不必自己寫

該欄位的功能就像之前學的資料庫 `"SQL primary key auto create"`，寫入資料換下一列時，數值會自動增加

=====

需要再次執行 `migrate` 命令 (`"py manage.py migrate "`) 讓新的遷移檔案生效並同步回資料庫，從而完成結構定義的修改

`py manage.py migrate myapp`

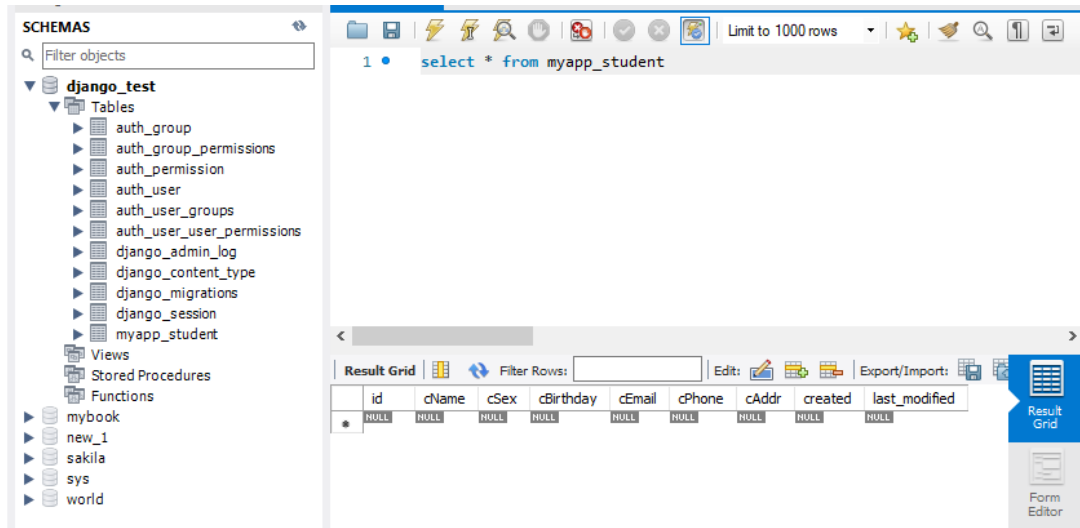
```
(django) C:\Users\226083\django_test\myweb>py manage.py migrate myapp  
Operations to perform:  
  Apply all migrations: myapp  
Running migrations:  
  Applying myapp.0001_initial... OK
```

=====

打開MYSQL

可以看到內部多了一個“myapp_student 的資料表”，選擇該資料表看看內容

```
select * from myapp_student
```



出現之前建立的各項欄位 ↑