Part2.

建立資料庫

終端機裡輸入python manage.py migrate，這個是在mysite/settings.py

創立必要的設定。(先打makemigrations 告訴系統更改那些東西，在打migrate更新資料庫)

django.contrib.admin – The admin site. You’ll use it shortly.

django.contrib.auth – An authentication system.

django.contrib.contenttypes – A framework for content types.

django.contrib.sessions – A session framework.

django.contrib.messages – A messaging framework.

django.contrib.staticfiles – A framework for managing static files.

創立models

poll app裡，我們創造兩個models:Question和Choice，這兩個之間都有相關性(這個關係有一對一、一對多、多對多)，找到共通相關性靠的是ForeignKey。

在polls/models.py輸入

from django.db import models

class Question(models.Model):

question\_text = models.CharField(max\_length=200)

pub\_date = models.DateTimeField('date published')

class Choice(models.Model):

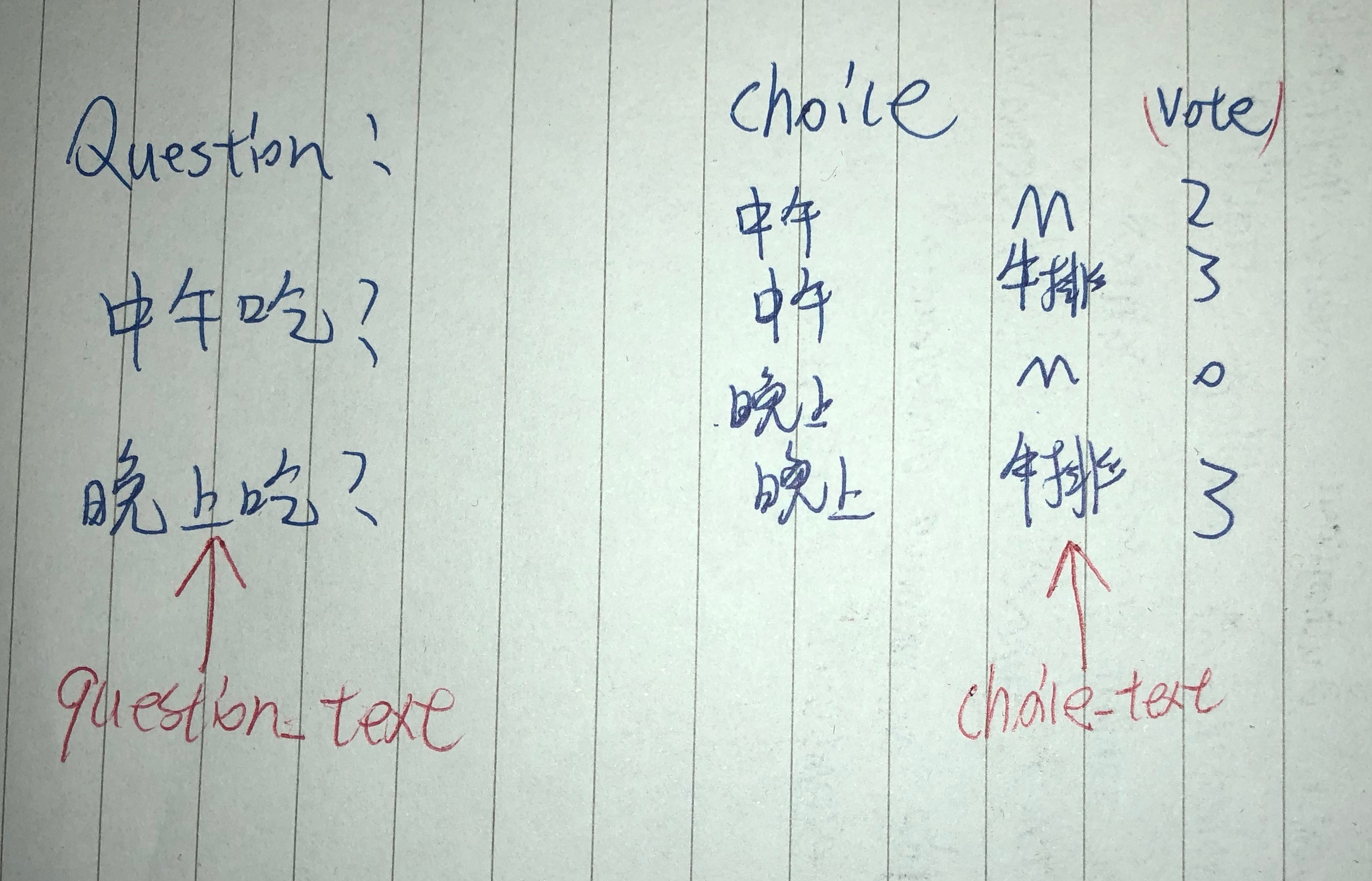
question = models.ForeignKey(Question, on\_delete=models.CASCADE)

choice\_text = models.CharField(max\_length=200)

votes = models.IntegerField(default=0)

Class裡要放字串的話就打CharField，時間類型的話打DateTimeField

每個filed後面都有一個設定值，其中最重要的是CharField，必須要給他一max\_length，範例裡是設定200，vote是數量，設定預設值為0。



啟動models

首先告訴我們的project已經安裝了polls app

在mysite/settings.py裡的installed­\_apps加入'polls.apps.PollsConfig',

會看到以下

INSTALLED\_APPS = [

'polls.apps.PollsConfig',

'django.contrib.admin',

'django.contrib.auth',

'django.contrib.contenttypes',

'django.contrib.sessions',

'django.contrib.messages',

'django.contrib.staticfiles',

]

並在終端機輸入python manage.py makemigrations polls

系統回傳以下代表啟動成功

Migrations for 'polls':

polls/migrations/0001\_initial.py:

- Create model Choice

- Create model Question

- Add field question to choice

最後再輸入python manage.py migrate更新系統

演示程式介面

終端機輸入python manage.py shell (離開按ctrl+z ;runserver按ctrl+c)

>>> from polls.models import Question, Choice #首先輸入剛剛所寫的model classes

>>> Question.objects.all() #創立一個新的Question

>>> from django.utils import timezone

>>> q = Question(question\_text="What's new?", pub\_date=timezone.now()) 設定question\_text為="What's new?" pub\_date為現在時間

>>> q.save() 儲存這個q

>>> q.id 出現這個物件的ID

>>> Question.objects.all() 顯示出所有資料庫裡的Questions

<QuerySet [<Question: Question object (1)>]>

之後再修改polls/models.py

最終變成以下

import datetime

from django.db import models

from django.utils import timezone

class Question(models.Model):

question\_text = models.CharField(max\_length=200)

pub\_date = models.DateTimeField('date published')

def \_\_str\_\_(self):

return self.question\_text

def was\_published\_recently(self):

return self.pub\_date >= timezone.now() - datetime.timedelta(days=1)

class Choice(models.Model):

question = models.ForeignKey(Question, on\_delete=models.CASCADE)

choice\_text = models.CharField(max\_length=200)

votes = models.IntegerField(default=0)

def \_\_str\_\_(self):

return self.choice\_text

\_\_str\_\_(記得為雙底線)，目的為之後檢查資料庫，可以回傳顯示出來

was\_published\_recently 代表最近有沒有發表文章，如果有會顯示true