Part 1

打sudo apt-get update

打sudo apt-get install python3-pip

python --version

python3 --version

pip3 install --upgrade pip

pip3 install --upgrade pip

pip3 --version

mkdir myproject

cd myproject

pip3 install virtualenv

sudo pip3 install virtualenv

virtualenv VENV

source VENV/bin/activate

pip --version

pip install django

python

import django

按(Ctrl+Z)

django-admin startproject mysite

python manage.py runserver

(error是因為沒進mysite)

cd mysite

python manage.py runserver

火箭出來後按Ctrl+C

python manage.py startapp polls

開始貼資料到資料夾

python manage.py runserver

連到127.0.0.1:8000/polls

Part 2

先按ctrl+C

然後把Part2一開始的資料貼上

python manage.py makemigrations

**出現結果：**Migrations for 'polls':

polls/migrations/0001\_initial.py

- Create model Choice

- Create model Question

- Add field question to choice

python manage.py migrate

python manage.py shell

打from polls.models import Question, Choice

打Question.objects.all()

**出現結果：**<QuerySet []>

打from django.utils import timezone

打q = Question(question\_text="What's new?", pub\_date=timezone.now())

打q.save()

打q.id

**出現結果：**1

打q.question\_text

**出現結果：**"What's new?"

打q.pub\_date

**出現結果：**datetime.datetime(2018, 3, 15, 12, 28, 18, 222270, tzinfo=<UTC>)

(出現的是日期 數字不一樣不用荒)

打q.question\_text = "What's up?"

打q.save()

打Question.objects.all()

**出現結果：**<QuerySet [<Question: Question object (1)>]>

打Question.objects.all()[0]

**出現結果：**<Question: Question object (1)>

打Question.objects.all()[0].value

**出現結果：**Traceback (most recent call last):

File "<console>", line 1, in <module>

AttributeError: 'Question' object has no attribute 'value'

打Question.objects.all()[0].question\_text

**出現結果：**"What's up?"

對資料庫polls/models.py 資料庫做更改把

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

**return** self.question\_text

**並貼在 (‘data published’)的下一行**

再把

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

return self.choice\_text

貼在default = 0的下面

在對polls/models.py做更改複製

import datetime

from django.db import models

from django.utils import timezone

複製完後把一開頭的from django.db import models刪掉並把剛剛複製的貼上

在複製**def** was\_published\_recently(self):

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

貼在polls/models.py中的return self.question\_text的下一行

貼完後按Ctrl+Z

打python manage.py makemigrations

**出現結果：**No changes detected

打python manage.py migrate

**出現結果：**Operations to perform:

Apply all migrations: admin, auth, contenttypes, polls, sessions

Running migrations:

No migrations to apply.

打python manage.py shell

**出現結果：**Python 3.5.2 (default, Nov 23 2017, 16:37:01)

[GCC 5.4.0 20160609] on linux

Type "help", "copyright", "credits" or "license" for more information.

(InteractiveConsole)

打from polls.models import Question, Choice

打Question.objects.all()

**出現結果：**<QuerySet [<Question: What's up?>]>

打Question.objects.filter(id=1)

**出現結果：**<QuerySet [<Question: What's up?>]>

打Question.objects.filter(question\_text\_\_startswith='What')

出現結果：Question.objects.filter(question\_text\_\_startswith='What')

打from django.utils import timezone

打Question.objects.get(pub\_date\_\_year=current\_year)

打current\_year

出現結果：2018

打 Question.objects.get(pub\_date\_\_year=current\_year)

出現結果: <Question: What's up?>

打Question.objects.get(id=2)

出現結果:錯是正常的

打Question.objects.get(pk=1)

出現結果：<Question: What's up?>

打q = Question.objects.get(pk=1)

打q.was\_published\_recently()

出現結果：Ture

打q = Question.objects.get(pk=1)

打q.choice\_set.all()

出現結果: <QuerySet []>

打q.choice\_set.create(choice\_text='Not much', votes=0)

出現結果：<Choice: Not much>

打q.choice\_set.create(choice\_text='The sky', votes=0)

出現結果: <Choice: The sky>

打c = q.choice\_set.create(choice\_text='Just hacking again', votes=0)

打c.question

出現結果: <Question: What's up?>

打q.choice\_set.all()

出現結果：<QuerySet [<Choice: Not much>, <Choice: The sky>, <Choice: Just hacking again>]>

打q.choice\_set.count()

出現結果：3

打Choice.objects.filter(question\_\_pub\_date\_\_year=current\_year)

出現結果：<QuerySet [<Choice: Not much>, <Choice: The sky>, <Choice: Just hacking again>]>

打c = q.choice\_set.filter(choice\_text\_\_startswith='Just hacking')

打c.delete()

打q.choice\_set.count()

出現結果：2

打q.choice\_set.all()

出現結果：<QuerySet [<Choice: Not much>, <Choice: The sky>

先按Crtl+Z

打python manage.py createsuperuser

按enter直到要輸入密碼

在打python manage.py runserver

再輸入網址127.0.0.1:8000/admin

在polls/admin.py中貼上資料

Part3

在polls/views.py貼上資料不可以蓋到上面的

在polls/urls.py資料全選然後把原本的刪掉再貼上新的

在polls/views.py 把原本資料中的第一段(從from django到結尾polls index)刪掉貼上剛剛在View中複製的(註解不要複製)

創建templates資料夾在裡面再創polls資料夾開新的檔案叫index.html把資料全部貼上

去polls/views把

from django.http import HttpResponse

from django.template import loader

from .models import Question複製並刪掉原本前兩行(Http結尾那個跟Question結尾)然後貼新的

在複製

def index(request):

latest\_question\_list = Question.objects.order\_by('-pub\_date')[:5]

template = loader.get\_template('polls/index.html')

context = {

'latest\_question\_list': latest\_question\_list,

}

return HttpResponse(template.render(context, request))

把第二段的(結尾是[:5]和Output那段)刪掉貼上新的

連到127.0.0.1:8000/polls

polls/views.py

**from** **django.shortcuts** **import** render

**from** **.models** **import** Question

將from開頭的換貼成上面兩行

Raising a 404 error

polls/views.py

**from** **django.http** **import** Http404

**from** **django.shortcuts** **import** render

**from** **.models** **import** Question

將from開頭的換貼成上面三行

**def** detail(request, question\_id):

**try**:

question = Question.objects.get(pk=question\_id)

**except** Question.DoesNotExist:

**raise** Http404("Question does not exist")

**return** render(request, 'polls/detail.html', {'question': question})

將def detail換貼成此段

polls/templates/polls/detail.html

在此路徑建detail.html檔案，貼上{{ question }}

polls/views.py

**from** **django.shortcuts** **import** get\_object\_or\_404, render

**from** **.models** **import** Question

將from開頭的換貼成上面兩行

**def** detail(request, question\_id):

question = get\_object\_or\_404(Question, pk=question\_id)

**return** render(request, 'polls/detail.html', {'question': question})

將def detail換貼成此段

polls/templates/polls/detail.html

<**h1**>{{ question.question\_text }}</**h1**>

<**ul**>

{% **for** choice **in** question.choice\_set.all %}

<**li**>{{ choice.choice\_text }}</**li**>

{% **endfor** %}

</**ul**>

將此段全部貼至上述路徑

瀏覽127.0.0.1:8000/polls/1

polls/urls.py

**from** **django.urls** **import** path

**from** **.** **import** views

app\_name = 'polls'

urlpatterns = [

path('', views.index, name='index'),

path('<int:question\_id>/', views.detail, name='detail'),

path('<int:question\_id>/results/', views.results, name='results'),

path('<int:question\_id>/vote/', views.vote, name='vote'),

]

將此段全部貼至上述路徑

polls/templates/polls/index.html

<**li**><**a** href="{% **url** 'detail' question.id %}">{{ question.question\_text }}</**a**></**li**>

將此段貼製至{%...................%}</li>(覆蓋上)

<**li**><**a** href="{% **url** 'polls:detail' question.id %}">{{ question.question\_text }}</**a**></**li**>

將此段貼製至同樣的位置(覆蓋上)

Part4

polls/templates/polls/detail.html

<**h1**>{{ question.question\_text }}</**h1**>

{% **if** error\_message %}<**p**><**strong**>{{ error\_message }}</**strong**></**p**>{% **endif** %}

<**form** action="{% **url** 'polls:vote' question.id %}" method="post">

{% **csrf\_token** %}

{% **for** choice **in** question.choice\_set.all %}

<**input** type="radio" name="choice" id="choice{{ forloop.counter }}" value="{{ choice.id }}" />

<**label** for="choice{{ forloop.counter }}">{{ choice.choice\_text }}</**label**><**br** />

{% **endfor** %}

<**input** type="submit" value="Vote" />

</**form**>

全部貼至上述路徑

瀏覽127.0.0.1:8000/polls/1

polls/views.py

**from** **django.shortcuts** **import** get\_object\_or\_404, render

**from** **django.http** **import** HttpResponseRedirect, HttpResponse

**from** **django.urls** **import** reverse

**from** **.models** **import** Choice, Question

將from開頭的換貼成上面四行

再將**def** vote(request, question\_id):

question = get\_object\_or\_404(Question, pk=question\_id)

**try**:

selected\_choice = question.choice\_set.get(pk=request.POST['choice'])

**except** (**KeyError**, Choice.DoesNotExist):

*# Redisplay the question voting form.*

**return** render(request, 'polls/detail.html', {

'question': question,

'error\_message': "You didn't select a choice.",

})

**else**:

selected\_choice.votes += 1

selected\_choice.save()

*# Always return an HttpResponseRedirect after successfully dealing*

*# with POST data. This prevents data from being posted twice if a*

*# user hits the Back button.*

**return** HttpResponseRedirect(reverse('polls:results', args=(question.id,)))

貼至def vote覆蓋住

瀏覽127.0.0.1:8000/polls/1/results

polls/views.py

**def** results(request, question\_id):

question = get\_object\_or\_404(Question, pk=question\_id)

**return** render(request, 'polls/results.html', {'question': question})

將此段貼至貼至def results覆蓋住

polls/templates/polls/results.html

<**h1**>{{ question.question\_text }}</**h1**>

<**ul**>

{% **for** choice **in** question.choice\_set.all %}

<**li**>{{ choice.choice\_text }} -- {{ choice.votes }} vote{{ choice.votes|pluralize }}</**li**>

{% **endfor** %}

</**ul**>

<**a** href="{% **url** 'polls:detail' question.id %}">Vote again?</**a**>

在此路徑建results.html檔案，將此段貼上

瀏覽127.0.0.1:8000/polls/1/results

polls/views.py

**from** **django.shortcuts** **import** get\_object\_or\_404, render

**from** **django.http** **import** HttpResponseRedirect

**from** **django.urls** **import** reverse

**from** **django.views** **import** generic

**from** **.models** **import** Choice, Question

將from開頭的換貼成上面五行

**class** **IndexView**(generic.ListView):

template\_name = 'polls/index.html'

context\_object\_name = 'latest\_question\_list'

**def** get\_queryset(self):

*"""Return the last five published questions."""*

**return** Question.objects.order\_by('-pub\_date')[:5]

**class** **DetailView**(generic.DetailView):

model = Question

template\_name = 'polls/detail.html'

**class** **ResultsView**(generic.DetailView):

model = Question

template\_name = 'polls/results.html'

貼在def index至def results上

!!!!!可能出現bug

**from** **django.urls** **import** path

**from** **.** **import** views

app\_name = 'polls'

urlpatterns = [

path('', views.IndexView.as\_view(), name='index'),

path('<int:pk>/', views.DetailView.as\_view(), name='detail'),

path('<int:pk>/results/', views.ResultsView.as\_view(), name='results'),

path('<int:question\_id>/vote/', views.vote, name='vote'),

]

將此段貼全貼至polls/urls.py

再將**def** vote(request, question\_id):

question = get\_object\_or\_404(Question, pk=question\_id)

**try**:

selected\_choice = question.choice\_set.get(pk=request.POST['choice'])

**except** (**KeyError**, Choice.DoesNotExist):

*# Redisplay the question voting form.*

**return** render(request, 'polls/detail.html', {

'question': question,

'error\_message': "You didn't select a choice.",

})

**else**:

selected\_choice.votes += 1

selected\_choice.save()

*# Always return an HttpResponseRedirect after successfully dealing*

*# with POST data. This prevents data from being posted twice if a*

*# user hits the Back button.*

**return** HttpResponseRedirect(reverse('polls:results', args=(question.id,)))

貼至polls/views.py

瀏覽127.0.0.1:8000/polls/1

Part6

polls/static/polls/style.css

在polls建立static資料夾再建立polls資料夾再建立style.css檔案

**li** **a** {

**color**: **green**;

}

將此段貼上

polls/templates/polls/index.html

{% **load** static %}

<**link** rel="stylesheet" type="text/css" href="{% **static** 'polls/style.css' %}" />

將此段貼至index.html的最上方(不要覆蓋任何東西)

polls/static/polls/style.css

**body** {

**background**: **white** url("images/background.gif") **no-repeat**;

}

增加至上述路徑

找一張圖儲存至**polls/static/polls/images/background.jpg**

將polls/static/polls/style.css

**body** {

**background**: **white** url("images/background.gif") **no-repeat**;

}

Gif改成jpg

!!!!!回到part3

polls/templates/polls/index.html

{% **if** latest\_question\_list %}

<**ul**>

{% **for** question **in** latest\_question\_list %}

<**li**><**a** href="/polls/{{ question.id }}/">{{ question.question\_text }}</**a**></**li**>

{% **endfor** %}

</**ul**>

{% **else** %}

<**p**>No polls are available.</**p**>

{% **endif** %}

將此段覆蓋{%if%}至{%end if%}

瀏覽127.0.0.1:8000/polls

Part7

polls/admin.py **from** **django.contrib** **import** admin

**from** **.models** **import** Question

**class** **QuestionAdmin**(admin.ModelAdmin):

fields = ['pub\_date', 'question\_text']

admin.site.register(Question, QuestionAdmin)

此段全部貼上

瀏覽127.0.0.1:8000/admin

polls/admin.py

**from** **django.contrib** **import** admin

**from** **.models** **import** Question

**class** **QuestionAdmin**(admin.ModelAdmin):

fieldsets = [

(**None**, {'fields': ['question\_text']}),

('Date information', {'fields': ['pub\_date']}),

]

admin.site.register(Question, QuestionAdmin)

此段全部貼上

polls/admin.py

**from** **django.contrib** **import** admin

**from** **.models** **import** Choice, Question

將from開頭的換貼成上面兩行

再將admin.site.register(Choice)貼至admin.site.register(Question,Questionadmin)上方

polls/admin.py

將**class** **ChoiceInline**(admin.StackedInline):

model = Choice

extra = 3

貼至class questionadmin上方

再將**class** **QuestionAdmin**(admin.ModelAdmin):

fieldsets = [

(**None**, {'fields': ['question\_text']}),

('Date information', {'fields': ['pub\_date'], 'classes': ['collapse']}),

]

inlines = [ChoiceInline]

覆蓋原本的class questionadmin

polls/admin.py

**class** **ChoiceInline**(admin.TabularInline):

覆蓋原本的class ChoiceInline

polls/admin.py

再inlines下貼上list\_display = ('question\_text', 'pub\_date')

貼上list\_display = ('question\_text', 'pub\_date', 'was\_published\_recently')

(覆蓋上剛剛貼的)

polls/models.py

**def** was\_published\_recently(self):

now = timezone.now()

**return** now - datetime.timedelta(days=1) <= self.pub\_date <= now

was\_published\_recently.admin\_order\_field = 'pub\_date'

was\_published\_recently.boolean = **True**

was\_published\_recently.short\_description = 'Published recently?'

(覆蓋原本的def was\_published\_recently(self))

將list\_filter = ['pub\_date']加至 list\_display = ('question\_text', 'pub\_date', 'was\_published\_recently')下，將search\_fields = ['question\_text']繼續往下加

mysite/settings.py

'DIRS': [os.path.join(BASE\_DIR, 'templates')],

(覆蓋'DIRS')

在polls的templates裡面建admin資料夾裡面再建**base\_site.html**檔案，裡面貼上

{% **block** branding %}

<**h1** id="site-name"><**a** href="{% **url** 'admin:index' %}">Polls Administration</**a**></**h1**>

{% **endblock** %}

瀏覽127.0.0.1:8000/admin

在manage.py同位置開templates資料夾裡面再開admin資料夾，從VENV資料夾中搜尋django找到contrib中的admin中的templates中的admin裡面的**base\_site.html**檔案，然後複製貼至templates/admin中

編輯**base\_site.html**

將{%block%}至{%endblock%}覆蓋上

{% **block** branding %}

<**h1** id="site-name"><**a** href="{% **url** 'admin:index' %}">Polls Administration</**a**></**h1**>

{% **endblock** %}

瀏覽127.0.0.1:8000/admin