

Chapter 7

จำลองการทดสอบส่งข้อมูลระหว่างแอปพลิเคชัน และ Server ด้วย
Django rest framework

ผลลัพธ์วันนี้

Django REST framework pom

All Todolist

All Todolist

OPTIONS GET

GET /api/all-todolist/

```
HTTP 200 OK
Allow: GET, OPTIONS
Content-Type: application/json
Vary: Accept

[
  {
    "id": 1,
    "title": "พักผ่อน",
    "detail": "กินอาหารอร่อยๆ"
  },
  {
    "id": 2,
    "title": "ไปตลาด",
    "detail": "ซื้อยาหมอยแชนบ ะ"
  },
  {
    "id": 3,
    "title": "กินกาแฟ",
    "detail": "กินกาแฟลาเต้ อร่อยๆ"
  }
]
```



Django follow MTV Architectural Pattern

MTV หรือ Model Template View เป็นคอนเซปต์การเขียนเว็บไซต์ที่มีส่วนประกอบการทำงานที่เชื่อมโยงกันระหว่าง

M - Model คือ ระบบจัดการฐานข้อมูล

T - Template คือ หน้าตาของเว็บที่ใช้แสดงผลว่าแต่ละหน้าจะออกมาเป็นอย่างไร

V- View คือ ส่วนสำหรับจัดการข้อมูลเพื่อนำไปแสดงผลใน **template**

1.ทดสอบระบบเดิมที่ทำไว้

1.1 cd เข้า folder project เราก่อน

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.3930]
(c) Microsoft Corporation. All rights reserved.

C:\napaphatFlutter\webapi>
```

1.2 พิมพ์ `.\venv\scripts\activate` เพื่อเข้าสู่โหมด venv

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.3930]
(c) Microsoft Corporation. All rights reserved.

C:\napaphatFlutter\webapi>.\venv\scripts\activate

(venv) C:\napaphatFlutter\webapi>
```

1.3 cd เข้า project ที่สร้าง

Git C:\Windows\System32\cmd.exe

```
Microsoft Windows [Version 10.0.19045.3930]  
(c) Microsoft Corporation. All rights reserved.
```

```
C:\napaphatFlutter\webapi>. \venv\scripts\activate
```

```
(venv) C:\napaphatFlutter\webapi>cd flutterapi
```

```
(venv) C:\napaphatFlutter\webapi\flutterapi>
```

1.4 พิมพ์คำสั่ง `python manage.py runserver` เพื่อ `runserver`

```
C:\Windows\System32\cmd.exe - python manage.py runserver
```

```
Microsoft Windows [Version 10.0.19045.3930]
(c) Microsoft Corporation. All rights reserved.

C:\napaphatFlutter\webapi>. \venv\scripts\activate

(venv) C:\napaphatFlutter\webapi>cd flutterapi

(venv) C:\napaphatFlutter\webapi\flutterapi>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
February 15, 2024 - 20:47:29
Django version 4.2, using settings 'flutterapi.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```


1.5 ใช้ browser > เข้า localhost:8000 ได้ผลดังรูป

```
[{"title": "Laptop คืออะไร?", "subtitle": "คอมพิวเตอร์ คือ อุปกรณ์ที่ใช้สำหรับการคำนวณและทำงานอื่นๆ?", "detail": "คอมพิวเตอร์ (อังกฤษ: computer) หรือศัพท์บัญญัติราชบัณฑิตยสภาว่า คณิตกรณ์[2][3] เป็นเครื่องจักรแบบสั่งการได้ที่ออกแบบมาเพื่อดำเนินการกับลำดับตัวดำเนินการทางตรรกศาสตร์หรือคณิตศาสตร์ โดยอนุกรมนี้อาจเปลี่ยนแปลงได้เมื่อพร้อม ส่งผลให้คอมพิวเตอร์สามารถแก้ปัญหาได้มากมาย\n\nคอมพิวเตอร์ถูกประดิษฐ์ออกมาให้ประกอบไปด้วยความจำรูปแบบต่าง ๆ เพื่อเก็บข้อมูล อย่างน้อยหนึ่งส่วนที่มีหน้าที่ดำเนินการคำนวณเกี่ยวกับตัวดำเนินการทางตรรกศาสตร์ และตัวดำเนินการทางคณิตศาสตร์ และส่วนควบคุมที่ใช้เปลี่ยนแปลงลำดับของตัวดำเนินการโดยยึดสารสนเทศที่ถูกเก็บไว้เป็นหลัก อุปกรณ์เหล่านี้จะยอมให้นำเข้าข้อมูลจากแหล่งภายนอก และส่งผลการคำนวณตัวดำเนินการออกไป"}, {"title": "Flutter 2 คือ?", "subtitle": "Tools สำหรับออกแบบ UI ของ Google ", "detail": "Flutter 2 สร้างโดย Google บริษัทเทคโนโลยีอันดับต้นๆของโลก ที่มาพร้อมกับความสวยงาม อลังการ พาไปวัดไปวาแบบไม่ต้องอายุใคร 55 อีกทั้งสามารถเขียนโปรแกรมครั้งเดียวด้วยภาษา Dart ที่ถูกออกแบบมาให้เขียนง่าย เข้าใจได้ไม่ยาก เขียนแค่ครั้งเดียว ส่งไปใช้งานได้ทุกแพลตฟอร์ม ไม่ว่าจะเป็น Android, iOS, Web Application, Windows, Linux, MacOS เรียกได้ว่าเขียนครั้งเดียวจบ ส่งไปใช้งานได้เลย"}, {"title": "Python คือ?", "subtitle": "ภาษาเขียนโปรแกรมชนิดหนึ่ง สร้างขึ้นเมื่อ 1991", "detail": "ภาษาไพทอน (Python programming language) หรือที่มักเรียกกันว่าไพทอน เป็นภาษาระดับสูงซึ่งสร้างโดยคีโด ฟัน โรสซิม โดยเริ่มในปีพ.ศ. 2533 การออกแบบของภาษาไพทอนมุ่งเน้นให้ผู้โปรแกรมสามารถอ่านชุดคำสั่งได้โดยง่ายผ่านการใช้งานอักขระเว้นว่าง (whitespaces) จำนวนมาก นอกจากนั้นการออกแบบภาษาไพทอนและการประยุกต์ใช้แนวคิดการเขียนโปรแกรมเชิงวัตถุในตัวภาษายังช่วยให้นักเขียนโปรแกรมสามารถเขียนโปรแกรมที่เป็นระเบียบ อ่านง่าย มีขนาดเล็ก และง่ายต่อการบำรุง[3]"}]
```

2.สร้างข้อมูลในฐานข้อมูล

2.1 เปิด project ใน vscode > ไปที่ไฟล์ models.py ของ myapp และพิมพ์โค้ดเพิ่มตามรูป

models.py ×

flutterapi > myapp > models.py > ...

```
1 from django.db import models
2
3 class Todolist(models.Model):
4     title = models.CharField(max_length=100)
5     detail = models.TextField(null=True, blank=True)
```

2.2 ไปที่ไฟล์ admin.py ของ myapp > พิมพ์โค้ดตามรูป

```
models.py  admin.py ●
flutterapi > myapp > admin.py
1  from django.contrib import admin
2  from .models import Todolist
3
4  admin.site.register(Todolist)
5
```

2.3 ออกจากการรัน server ด้วยการกด `ctrl+c` เพื่อให้เข้าโหมดปกติ

คั่งรูป

```
C:\Windows\System32\cmd.exe
(venv) C:\napaphatFlutter\webapi>cd flutterapi

(venv) C:\napaphatFlutter\webapi\flutterapi>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
February 08, 2024 - 20:01:30
Django version 5.0.1, using settings 'flutterapi.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.

[08/Feb/2024 20:01:40] "GET / HTTP/1.1" 200 4468
C:\napaphatFlutter\webapi\flutterapi\myapp\models.py changed, reloading.
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
February 08, 2024 - 20:09:45
Django version 5.0.1, using settings 'flutterapi.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.

(venv) C:\napaphatFlutter\webapi\flutterapi>
```

2.4 พิมพ์คำสั่ง `python manage.py makemigrations` เพื่ออัปเดตการเปลี่ยนแปลง ว่ามีการเปลี่ยนแปลงอะไรบ้าง

C:\Windows\System32\cmd.exe

```
Watching for file changes with StatReloader
Performing system checks...
```

```
System check identified no issues (0 silenced).
February 08, 2024 - 20:01:30
Django version 5.0.1, using settings 'flutterapi.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

```
[08/Feb/2024 20:01:40] "GET / HTTP/1.1" 200 4468
C:\napaphatFlutter\webapi\flutterapi\myapp\models.py changed, reloading.
Watching for file changes with StatReloader
Performing system checks...
```

```
System check identified no issues (0 silenced).
February 08, 2024 - 20:09:45
Django version 5.0.1, using settings 'flutterapi.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

```
(venv) C:\napaphatFlutter\webapi\flutterapi>python manage.py makemigrations
Migrations for 'myapp':
  myapp\migrations\0001_initial.py
  - Create model Todolist
```

```
(venv) C:\napaphatFlutter\webapi\flutterapi>
```

2.5 พิมพ์คำสั่ง `python manage.py migrate` เพื่อนำการเปลี่ยนแปลงข้างต้น ลงฐานข้อมูล

```
(venv) C:\napaphatFlutter\webapi\flutterapi>python manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, myapp, sessions
Running migrations:
  Applying myapp.0001_initial... OK

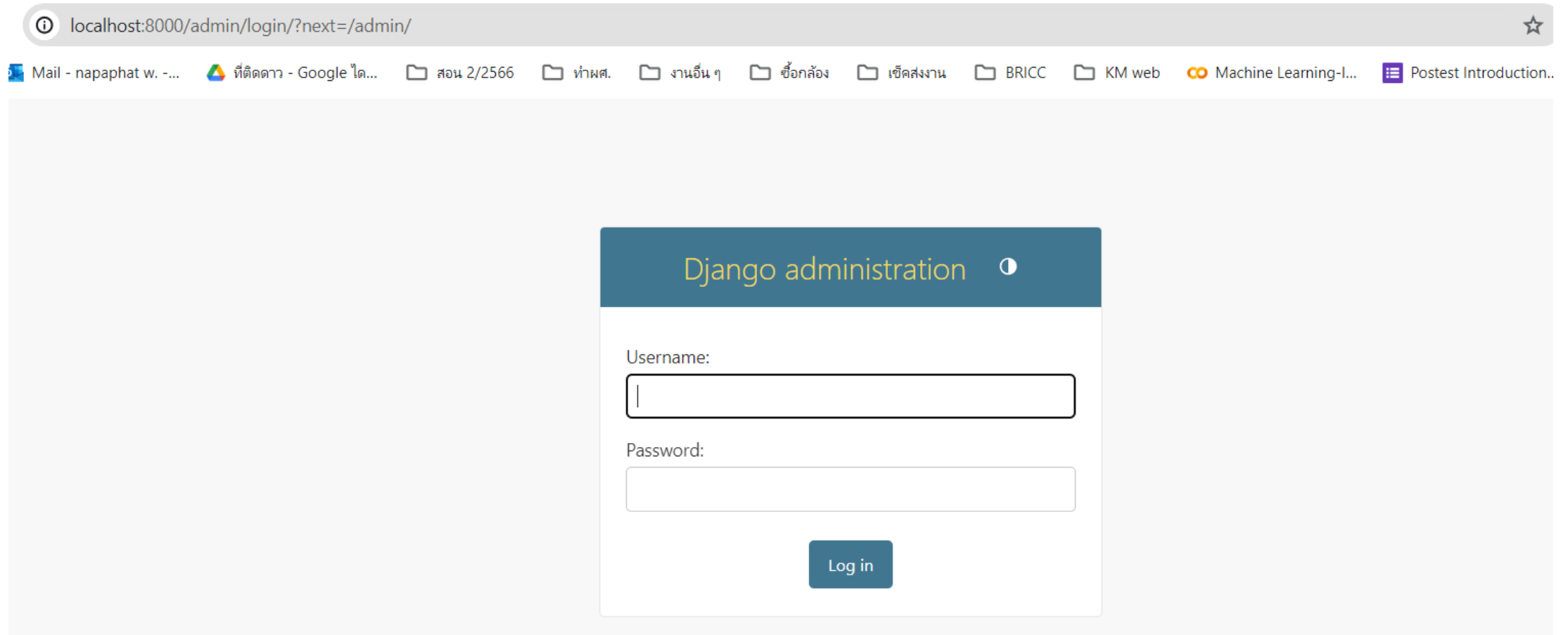
(venv) C:\napaphatFlutter\webapi\flutterapi>
```

2.6 จากนั้นพิมพ์คำสั่ง `python manage.py runserver` เพื่อรัน `runserver`

```
(venv) C:\napaphatFlutter\webapi\flutterapi>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
February 08, 2024 - 20:18:25
Django version 5.0.1, using settings 'flutterapi.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```


2.7 ไปที่ browser > พิมพ์ localhost:8000/admin/ จะเข้าสู่หน้า admin ดังรูป



The screenshot shows a web browser window with the address bar containing the URL `localhost:8000/admin/login/?next=/admin/`. The browser's tab bar shows several open tabs, including "Mail - napaphat w. -...", "ที่ติดดาว - Google ไต...", "สอน 2/2566", "ทำมต.", "งานอื่น ๆ", "ช็อกลิ่ง", "เช็คส่งงาน", "BRICC", "KM web", "Machine Learning-I...", and "Postest Introduction..".

The main content of the browser is the Django administration login page. It features a dark teal header with the text "Django administration" and a small circular icon. Below the header, there are two input fields: "Username:" and "Password:". The "Username:" field contains a vertical cursor. Below the password field is a dark teal "Log in" button.

2.8 login เข้าไปด้วยรหัสผ่านที่เคยตั้ง จะจอหน้าดังรูป

Site administration

AUTHENTICATION AND AUTHORIZATION

Groups [+ Add](#) [Change](#)

Users [+ Add](#) [Change](#)

MYAPP

Todolists [+ Add](#) [Change](#)

Recent actions

My actions

None available

2.9 ไปที่ Todolist ด้านซ้าย > คลิก +Add จะได้ผลลัพธ์ดังรูป

The screenshot displays the Django administration interface for a 'Myapp' project. The top navigation bar shows 'Django administration' on the left and user information 'WELCOME, POM. VIEW SITE / CHANGE PASSWORD / LOG OUT' on the right. The breadcrumb trail indicates the current location: 'Home > Myapp > Todolists > Add todolist'. On the left sidebar, the 'MYAPP' section is expanded, and 'Todolists' is highlighted with a '+ Add' button. The main content area is titled 'Add todolist' and contains a form with a 'Title:' label and an empty text input field. Below the input field is a 'Detail:' label and a large empty text area. At the bottom of the form, there are three buttons: 'SAVE', 'Save and add another', and 'Save and continue editing'.

2.10 กรอกข้อมูล และกดบันทึก

Start typing to filter...

AUTHENTICATION AND AUTHORIZATION

Groups [+ Add](#)

Users [+ Add](#)

MYAPP

Todolists [+ Add](#)

Add todolist

Title:

พักผ่อน

Detail:

กินอาหารอร่อยๆ

SAVE

Save and add another

Save and continue editing

2.11 ได้ผลดังรูป

Start typing to filter...

AUTHENTICATION AND AUTHORIZATION

Groups [+ Add](#)

Users [+ Add](#)

MYAPP

Todolists [+ Add](#)

✔ The todolist "Todolist object (1)" was added successfully.

Select todolist to change

ADD TODOLIST [+](#)

Action: 0 of 1 selected

TODOLIST

Todolist object (1)

1 todolist

2.12 คลิกปุ่มสี่เหลี่ยมด้านขวา เพื่ออีกรายการ > เพิ่มให้ได้จำนวน 2 รายการ ดังรูป

The screenshot shows the Django administration interface for the 'Myapp' application, specifically the 'Todolists' view. The page features a dark blue header with the text 'Django administration' and user information 'WELCOME, POM. VIEW SITE / CHANGE PASSWORD / LOG OUT'. Below the header, a breadcrumb trail reads 'Home > Myapp > Todolists'. A green success message at the top states: 'The todolist "Todolist object (2)" was added successfully.' The left sidebar contains a search bar and two sections: 'AUTHENTICATION AND AUTHORIZATION' with 'Groups' and 'Users' (each with a '+ Add' button), and 'MYAPP' with 'Todolists' (with a '+ Add' button). The main content area is titled 'Select todolist to change' and includes an 'ADD TODOLIST +' button. Below this is an 'Action:' dropdown menu set to '-----' with a 'Go' button and '0 of 2 selected' text. A list of three items is shown, each with a checkbox: 'TODOLIST', 'Todolist object (2)', and 'Todolist object (1)'. At the bottom of the list, it says '2 todolists'.

2.13 แก้ให้นำ title ออกมาแสดงด้านนอกเป็นหัวข้อแทนแบบเดิม ด้วยการไปที่ไฟล์ models.py และเพิ่มโค้ดดังรูป >> กดบันทึก

```
models.py × admin.py urls.py
flutterapi > myapp > models.py > ...
1  from django.db import models
2
3  class Todolist(models.Model):
4      title = models.CharField(max_length=100)
5      detail = models.TextField(null=True, blank=True)
6
7      def __str__(self):
8          return self.title
9
```

หมายเหตุ* โค้ดที่เพิ่มต้องย่อหน้าเข้าไป ดังรูป ให้อยู่ภายใน class

2.14 กด refresh browser จะเห็นผลลัพธ์ดังรูป

The screenshot shows the Django administration interface. At the top, there is a dark blue header with the text "Django administration" on the left and "WELCOME, POM. VIEW SITE / CHANGE PASSWORD / LOG OUT" on the right. Below the header, a breadcrumb trail reads "Home > Myapp > Todolists".

On the left side, there is a sidebar menu. The first section is "AUTHENTICATION AND AUTHORIZATION" with sub-items "Groups" and "Users", each with a "+ Add" link. The second section is "MYAPP" with a sub-item "Todolists" which is highlighted in yellow and also has a "+ Add" link.

The main content area is titled "Select todolist to change". It features an "Action:" dropdown menu with a "Go" button and the text "0 of 2 selected". Below this, there is a list of two items, each with a checkbox and a label: "TODOLIST" and "ปิดลาด" (with a Thai character). The second item is "พักพอน" (with a Thai character). At the bottom of the list, it says "2 todolists".

In the top right corner of the main content area, there is a button labeled "ADD TODOLIST +" with a plus sign icon.

3.ติดตั้ง Django rest framework

Django rest framework คืออะไร

- Django REST Framework คือ toolkit หรือไลบรารีของ Python ในการสร้าง RESTful APIs
- REST : REpresentational State Transfer คือ สถาปัตยกรรมอย่างหนึ่งของซอฟต์แวร์ที่อยู่บนพื้นฐานของ HTTP Protocol ที่เกี่ยวข้องกับการส่งข้อมูลระหว่าง Client และ Server

ที่มา : (Stackpython, 2021).

ข้อดี Django REST Framework

- ง่ายในการเรียนรู้เพราะว่า **base on Python** ซึ่งก็ขึ้นชื่อในเรื่องของไวยากรณ์ที่อ่านเข้าใจง่าย สละสลวยสวยงามเหมือนภาษาอังกฤษ
- สร้าง **RESTful APIs** ได้อย่างรวดเร็ว
- มี **serializers** ที่ช่วยในการแปลง **Django, Python object** ไปเป็น **JSON** ได้อย่างง่ายดาย
- **Browsable API** คือมีหน้า **Interface** ในการเรียกดู ทดสอบ **API** ในตัว ทำให้ง่ายและสะดวกขึ้นมาก
- ได้รับความนิยมนสูงและมีคอมมูนิตี้ขนาดใหญ่ มีแหล่งเรียนรู้ต่าง ๆ ไม่ว่าจะเป็นบทความ คลิป **tutorials** และแหล่งเรียนรู้อื่น ๆ เพิ่มเติมเยอะ
- มี **Documentation** ที่ครบครัน เขียนไว้ได้อย่างครอบคลุมและละเอียด ทำให้เป็น **reference** ได้เป็นอย่างดี

HTTP Methods ที่ควรรู้

- GET --> เมธอดที่ใช้สำหรับดึงข้อมูล
- POST --> เมธอดที่ใช้สำหรับสร้างข้อมูลขึ้นมาใหม่
- PUT --> เมธอดที่ใช้สำหรับอัปเดตข้อมูลที่มีอยู่
- DELETE --> เมธอดที่ใช้สำหรับลบข้อมูล

ที่มา : (Stackpython, 2021).

Status Codes ที่ควรรู้

- ✓ 200 - 299 --> status ที่บ่งบอกว่า response สำเร็จ (Successful Responses)
- ✓ 400 - 499 --> status ที่บ่งบอกว่า response ผิดพลาด (Client Error Responses)
- ✓ 500 - 599 --> status ที่บ่งบอกว่า server ผิดพลาด (Server Error Responses)

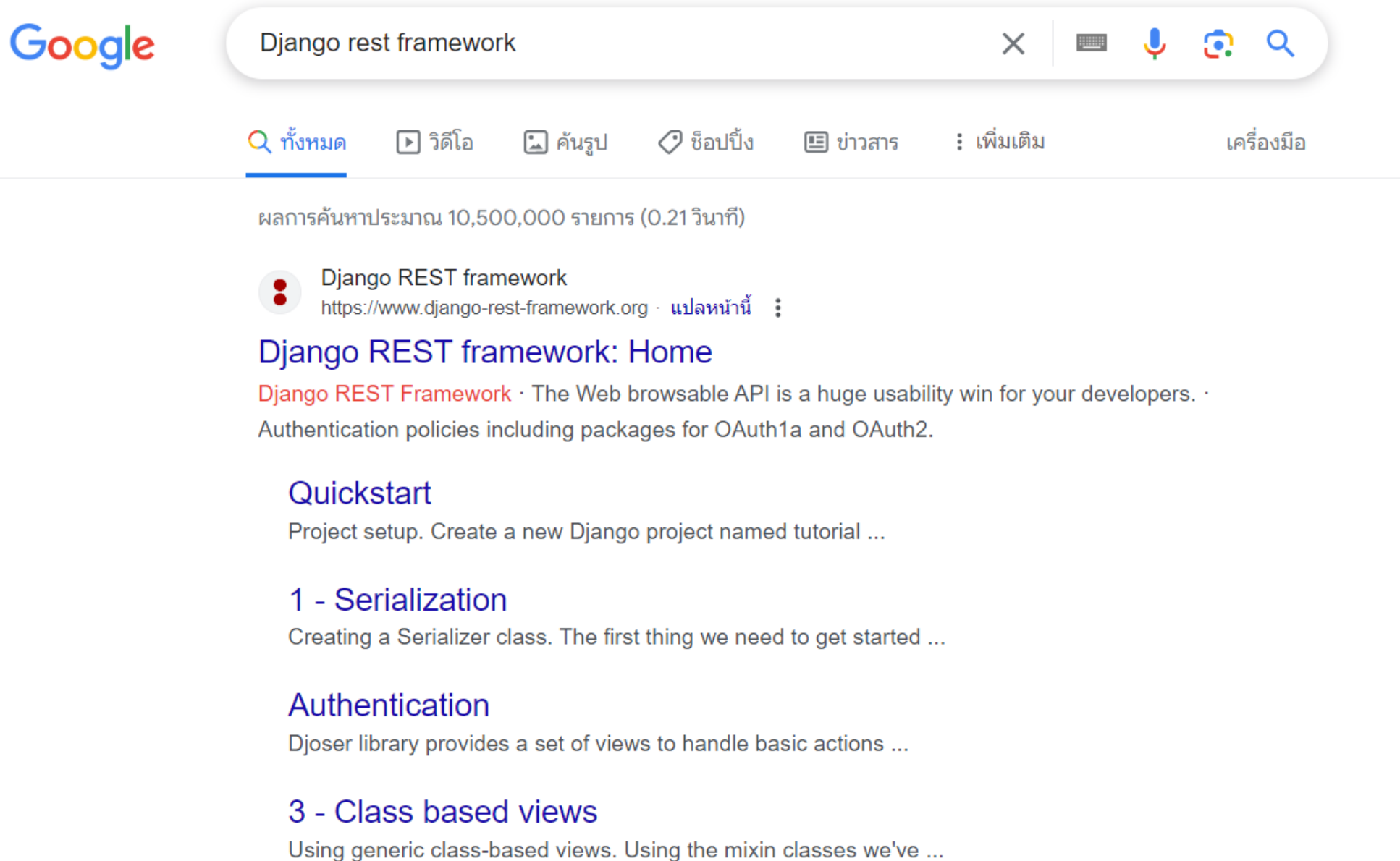
ที่มา : (Stackpython, 2021).

Status Codes ที่ควรรู้

- **200 OK** คือ **response** สำเร็จเป็น **status code** ที่พบเจอบ่อยที่สุด เช่นเมื่อมีการเสิร์ชหาหน้าเว็บต่าง ๆ แล้วเรากดคลิกเข้าดูได้ **server** ก็จะตอบกลับด้วย **status code** นี้
- **201 Created** คือ **response** สำเร็จเมื่อ **resource** ใหม่ถูกสร้างขึ้นมา พบเจอได้บ่อย ๆ เช่นมีการบันทึกข้อมูลผ่านหน้าฟอร์ม และถูก **submit** ไปที่ **server** แล้ว **server** ทำการตอบกลับด้วย **status code** นี้
- **404 Not Found** คือ **Page not found** (ไม่พบหน้าที่ต้องการ/ร้องขอ) หน้านี้ไม่มีอยู่อีกต่อไปหรือถูกลบไปแล้ว

ที่มา : (Stackpython, 2021).

3.1 ค้นคว้า Django rest framework ในเว็บ



The image shows a Google search interface. The search bar contains the text "Django rest framework". Below the search bar, there are navigation icons for "ทั้งหมด" (All), "วิดีโอ" (Videos), "ค้นรูป" (Images), "ข้อปับัง" (Maps), "ข่าวสาร" (News), "เพิ่มเติม" (More), and "เครื่องมือ" (Tools). The search results show approximately 10,500,000 results in 0.21 seconds. The top result is for the Django REST framework website, with the URL "https://www.django-rest-framework.org" and a link to the home page. The snippet for the home page reads: "Django REST Framework · The Web browsable API is a huge usability win for your developers. · Authentication policies including packages for OAuth1a and OAuth2." Below this, there are links to "Quickstart", "1 - Serialization", "Authentication", and "3 - Class based views", each with a brief description of the content.

Google

Django rest framework

ทั้งหมด วิดีโอ ค้นรูป ข้อปับัง ข่าวสาร : เพิ่มเติม เครื่องมือ

ผลการค้นหาประมาณ 10,500,000 รายการ (0.21 วินาที)

Django REST framework
https://www.django-rest-framework.org · แพลหน้านี้

Django REST framework: Home
Django REST Framework · The Web browsable API is a huge usability win for your developers. · Authentication policies including packages for OAuth1a and OAuth2.

Quickstart
Project setup. Create a new Django project named tutorial ...

1 - Serialization
Creating a Serializer class. The first thing we need to get started ...

Authentication
Djoser library provides a set of views to handle basic actions ...

3 - Class based views
Using generic class-based views. Using the mixin classes we've ...

3.2 คลินิกก็ได้หน้าเว็บ ดังรูป


Django REST framework

Home Tutorial API Guide Topics Community

Search Previous Next GitHub

pypi v3.14.0 CI passing Star 27,089

- Django REST framework
- Funding
- Requirements
- Installation
- Example
- Quickstart
- Development
- Support
- Security
- License



Spaciov provides innovative solutions and services in space and on earth, telecommunications and embedded systems.

Fund Django REST framework

django REST framework

Django REST framework is a powerful and flexible toolkit for building Web APIs.

Some reasons you might want to use REST framework:

- The **Web browsable API** is a huge usability win for your developers.
- **Authentication policies** including packages for **OAuth1a** and **OAuth2**.
- **Serialization** that supports both **ORM** and **non-ORM** data sources.
- Customizable all the way down - just use **regular function-based views** if you don't need the **more powerful features**.
- Extensive documentation, and **great community support**.
- Used and trusted by internationally recognised companies including **Mozilla**, **Red Hat**, **Heroku**, and **Eventbrite**.

Funding

3.3 ไปที่ installation > copy ส่วนที่ขีดเส้นใต้สีแดงใส่ notepad ไว้

Installation

Install using `pip`, including any optional packages you want...

```
pip install djangorestframework  
pip install markdown      # Markdown support for the browsable API.  
pip install django-filter # Filtering support
```

...or clone the project from github.

```
git clone https://github.com/encode/django-rest-framework
```

Add `'rest_framework'` to your `INSTALLED_APPS` setting.

```
INSTALLED_APPS = [  
    ...  
    'rest_framework',  
]
```

3.4 อ่าน requirements พบว่า version เดิมที่ติดตั้งไม่รองรับ Django rest framework

Requirements

REST framework requires the following:

- Python (3.6, 3.7, 3.8, 3.9, 3.10, 3.11)
- Django (3.0, 3.1, 3.2, 4.0, 4.1, 4.2)

We **highly recommend** and only officially support the latest patch release of each Python and Django series.

The following packages are optional:

- **PyYAML**, **uritemplate** (5.1+, 3.0.0+) - Schema generation support.
- **Markdown** (3.0.0+) - Markdown support for the browsable API.
- **Pygments** (2.4.0+) - Add syntax highlighting to Markdown processing.
- **django-filter** (1.0.1+) - Filtering support.
- **django-guardian** (1.1.1+) - Object level permissions support.

3.5 ติดตั้ง Django version ใหม่ >> pip install Django==4.2

```
(venv) C:\napaphatFlutter\webapi\flutterapi>pip install Django==4.2
Collecting Django==4.2
  Downloading Django-4.2-py3-none-any.whl (8.0 MB)
  8.0/8.0 MB 3.5 MB/s eta 0:00:00
Requirement already satisfied: asgiref<4,>=3.6.0 in c:\napaphatflutter\webapi\venv\lib\site-packages (from Django==4.2) (3.7.2)
Requirement already satisfied: sqlparse>=0.3.1 in c:\napaphatflutter\webapi\venv\lib\site-packages (from Django==4.2) (0.4.4)
Requirement already satisfied: tzdata in c:\napaphatflutter\webapi\venv\lib\site-packages (from Django==4.2) (2023.4)
Installing collected packages: Django
  Attempting uninstall: Django
    Found existing installation: Django 5.0.1
    Uninstalling Django-5.0.1:
      Successfully uninstalled Django-5.0.1
Successfully installed Django-4.2

[notice] A new release of pip is available: 23.3.2 -> 24.0
[notice] To update, run: python.exe -m pip install --upgrade pip
```

3.6 ctrl+c ออกจากการ runserver >

พิมพ์คำสั่ง `pip install django` (ที่ copy มา)

```
(venv) C:\napaphatFlutter\webapi\flutterapi>pip install django
Collecting django
  Downloading django-3.14.0-py3-none-any.whl (1.1 MB)
  ----- 1.1/1.1 MB 1.5 MB/s eta 0:00:00
Requirement already satisfied: django>=3.0 in c:\napaphatflutter\webapi\venv\lib\site-packages (from django) (4.2)
Collecting pytz (from django)
  Downloading pytz-2024.1-py2.py3-none-any.whl.metadata (22 kB)
Requirement already satisfied: asgiref<4,>=3.6.0 in c:\napaphatflutter\webapi\venv\lib\site-packages (from django>=3.0->django) (3.7.2)
Requirement already satisfied: sqlparse>=0.3.1 in c:\napaphatflutter\webapi\venv\lib\site-packages (from django>=3.0->django) (0.4.4)
Requirement already satisfied: tzdata in c:\napaphatflutter\webapi\venv\lib\site-packages (from django>=3.0->django) (2023.4)
  Downloading pytz-2024.1-py2.py3-none-any.whl (505 kB)
  ----- 505.5/505.5 kB 2.6 MB/s eta 0:00:00
Installing collected packages: pytz, django
Successfully installed django-3.14.0 pytz-2024.1

[notice] A new release of pip is available: 23.3.2 -> 24.0
[notice] To update, run: python.exe -m pip install --upgrade pip

(venv) C:\napaphatFlutter\webapi\flutterapi>
```

3.7 ไปที่ไฟล์ settings.py > วางส่วนที่ 2 ที่ copy มา

```
models.py settings.py × admin.py
flutterapi > flutterapi > settings.py > ...
31 # Application definition
32
33 INSTALLED_APPS = [
34     'django.contrib.admin',
35     'django.contrib.auth',
36     'django.contrib.contenttypes',
37     'django.contrib.sessions',
38     'django.contrib.messages',
39     'django.contrib.staticfiles',
40     'myapp',
41     'rest_framework',
42 ]
43
```

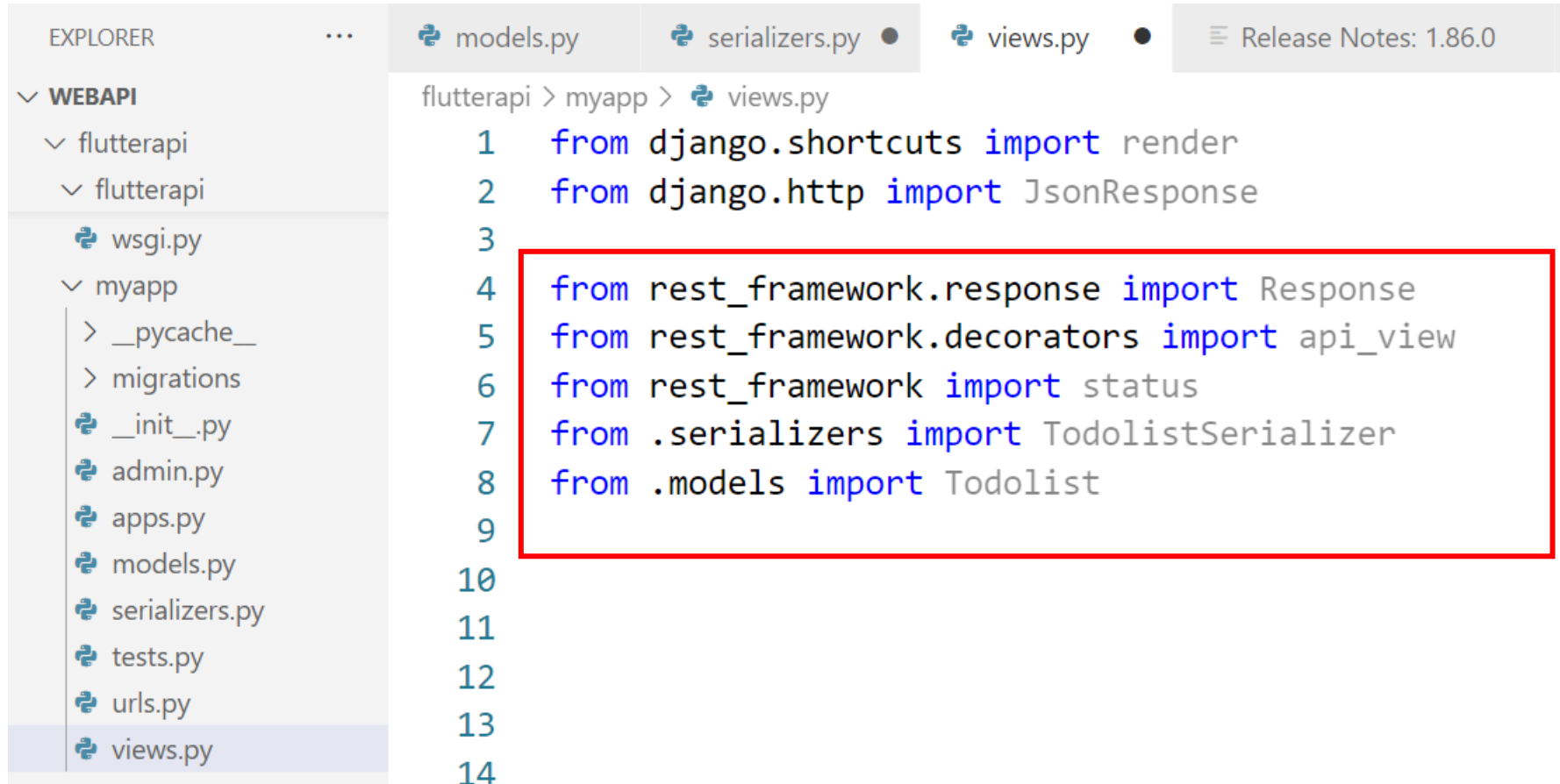
4.สร้างไฟล์ serializers

คล้ายตัว `model` เป็นการระบุว่า จะให้แสดงข้อมูลไหนบ้างใน `api`

4.1 สร้างไฟล์ serializers.py ภายใต้ myapp > พิมพ์โค้ดตามรูป

```
models.py db.sqlite3 serializers.py ● Release Notes: 1.86.0 settings.py
flutterapi > myapp > serializers.py > ...
1  from rest_framework import serializers
2  from .models import Todolist
3
4  class TodolistSerializer(serializers.ModelSerializer):
5      class Meta:
6          model = Todolist
7          fields = ('id', 'title', 'detail')
8
```

4.2 ไปที่ไฟล์ views.py ของ myapp > พิมพ์โค้ดเพิ่ม ตาม กรอบดังรูป

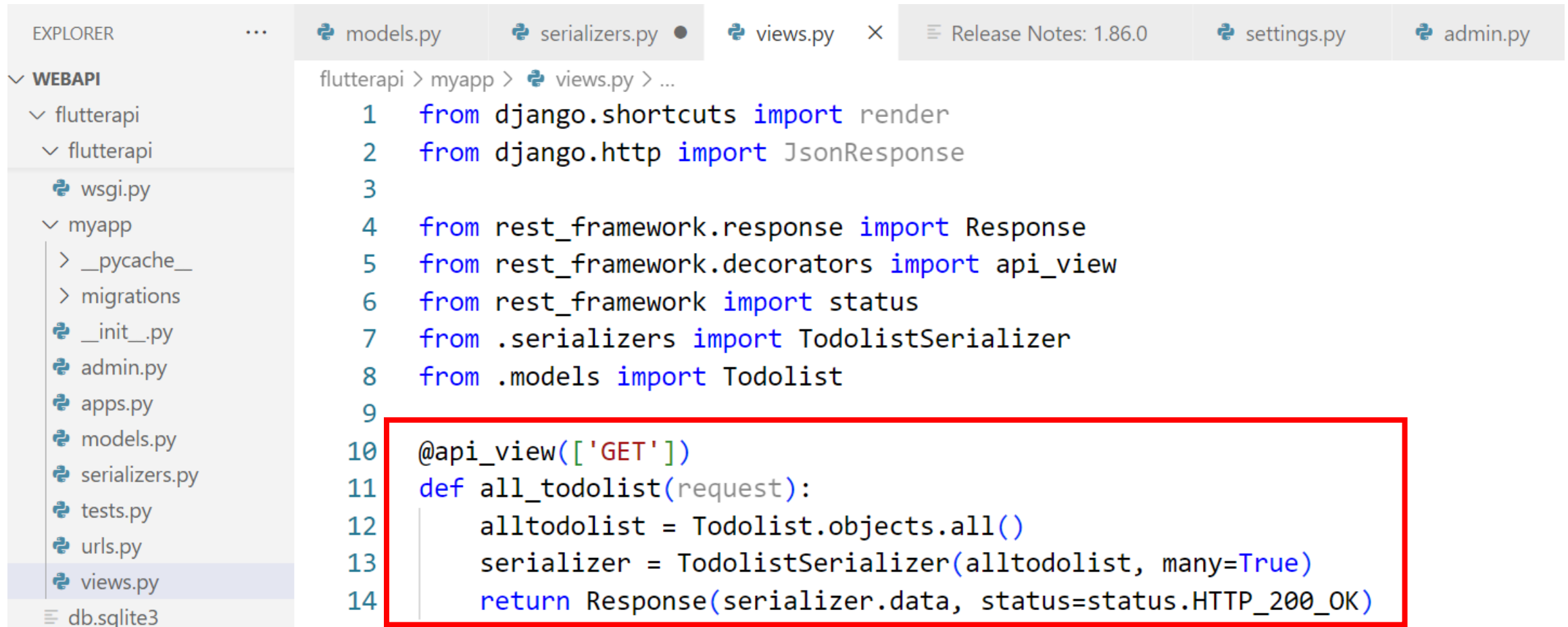


The image shows a screenshot of a code editor (VS Code) with the Explorer sidebar on the left and the Editor window on the right. The Explorer sidebar shows a project structure with a folder named 'myapp' containing several files, including 'views.py' which is selected. The Editor window shows the content of 'views.py' with line numbers 1 through 14. A red rectangular box highlights the code between lines 4 and 8.

```
EXPLORER
...
models.py
serializers.py
views.py
Release Notes: 1.86.0

flutterapi > myapp > views.py
1  from django.shortcuts import render
2  from django.http import JsonResponse
3
4  from rest_framework.response import Response
5  from rest_framework.decorators import api_view
6  from rest_framework import status
7  from .serializers import TodolistSerializer
8  from .models import Todolist
9
10
11
12
13
14
```


4.3 ยังอยู่ที่ไฟล์ views.py > พิมพ์โค้ดเพิ่ม ตามกรอบดังรูป

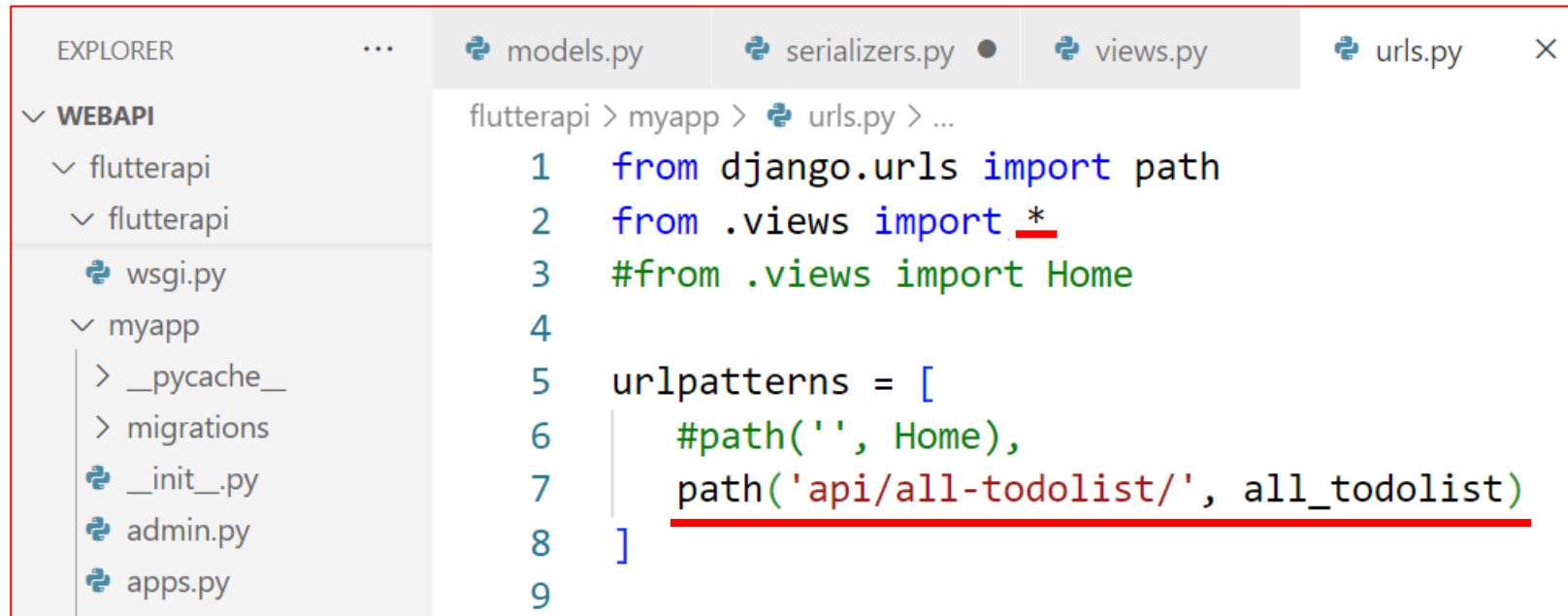


```
EXPLORER  ...  models.py  serializers.py  views.py  ×  Release Notes: 1.86.0  settings.py  admin.py

v WEBAPI
v flutterapi
v flutterapi
  wsgi.py
  myapp
    > __pycache__
    > migrations
    __init__.py
    admin.py
    apps.py
    models.py
    serializers.py
    tests.py
    urls.py
    views.py
  db.sqlite3

flutterapi > myapp > views.py > ...
1  from django.shortcuts import render
2  from django.http import JsonResponse
3
4  from rest_framework.response import Response
5  from rest_framework.decorators import api_view
6  from rest_framework import status
7  from .serializers import TodolistSerializer
8  from .models import Todolist
9
10 @api_view(['GET'])
11 def all_todolist(request):
12     alltodolist = Todolist.objects.all()
13     serializer = TodolistSerializer(alltodolist, many=True)
14     return Response(serializer.data, status=status.HTTP_200_OK)
```

4.4 ไปที่ไฟล์ urls.py ของ myapp > แก้ไขและ พิมพ์โค้ดเพิ่ม ดังรูป



```
EXPLORER
...
models.py  serializers.py  views.py  urls.py  X
v WEBAPI
v flutterapi
v flutterapi
  wsgi.py
v myapp
  > __pycache__
  > migrations
  wsgi.py
  __init__.py
  admin.py
  apps.py

flutterapi > myapp > urls.py > ...
1  from django.urls import path
2  from .views import *
3  #from .views import Home
4
5  urlpatterns = [
6      #path('', Home),
7      path('api/all-todolist/', all_todolist)
8  ]
9
```

4.5 cmd > เข้า folder โปรเจ็ค > พิมพ์ `.\venv\scripts\activate` เพื่อเข้าสู่โหมด venv > cd เข้า folder ที่สร้างขึ้น > รัน server ด้วยการพิมพ์ `python manage.py runserver`

```
C:\Windows\System32\cmd.exe - python manage.py runserver
```

```
Microsoft Windows [Version 10.0.19045.3930]  
(c) Microsoft Corporation. All rights reserved.
```

```
C:\napaphatFlutter\webapi> .\venv\scripts\activate
```

```
(venv) C:\napaphatFlutter\webapi> cd flutterapi
```

```
(venv) C:\napaphatFlutter\webapi\flutterapi> python manage.py runserver  
Watching for file changes with StatReloader  
Performing system checks...
```

```
System check identified no issues (0 silenced).
```

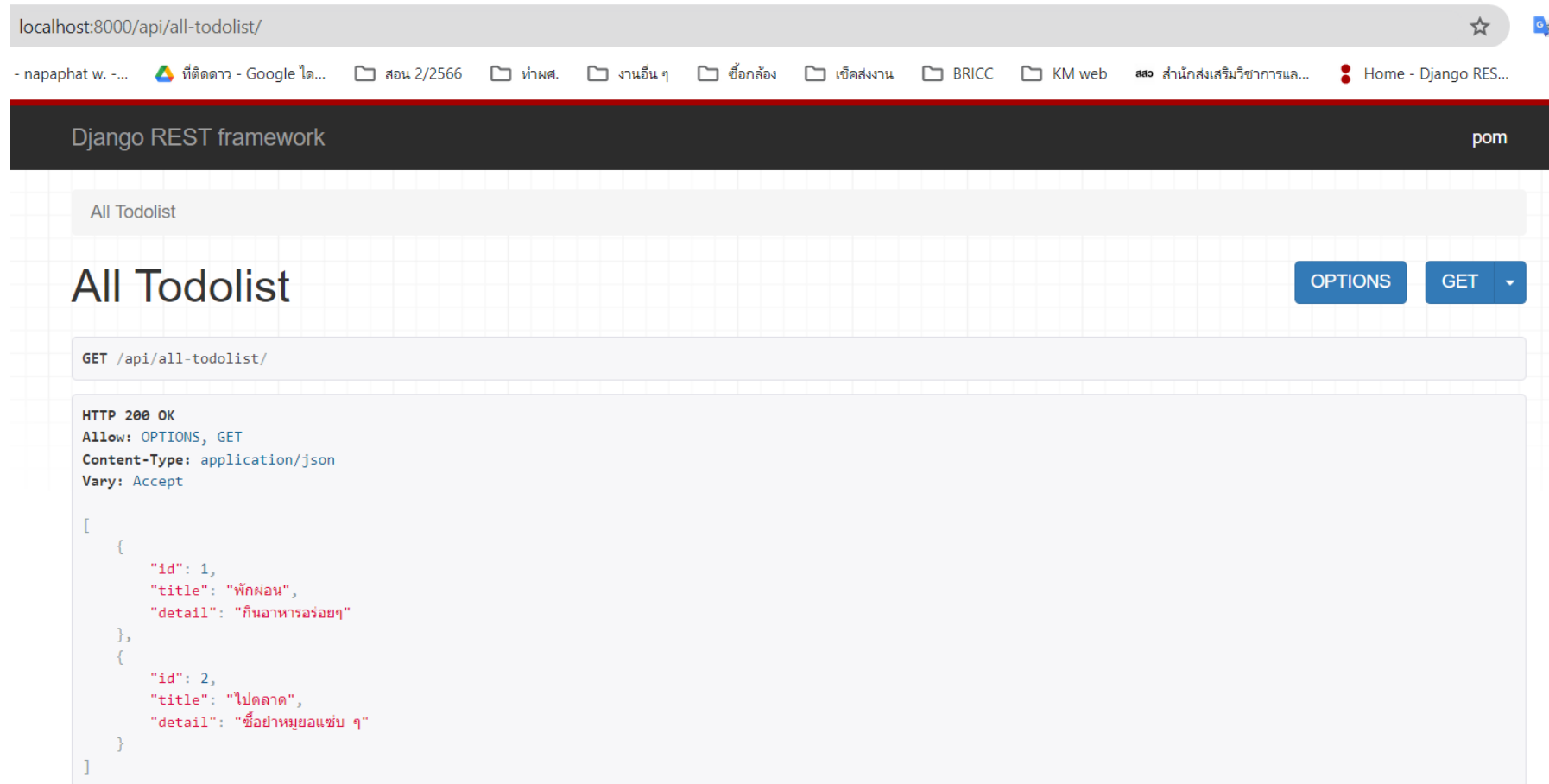
```
February 09, 2024 - 20:36:27
```

```
Django version 4.2, using settings 'flutterapi.settings'
```

```
Starting development server at http://127.0.0.1:8000/
```

```
Quit the server with CTRL-BREAK.
```

4.6 เปิด browser > พิมพ์ localhost:8000/api/all-todolist/ ได้ผลดังรูป



The screenshot shows a web browser window with the address bar containing 'localhost:8000/api/all-todolist/'. The browser's tab bar shows several tabs, including 'napaphat w. -...', 'ที่ติดดาว - Google ได...', 'สอน 2/2566', 'ทำพศ.', 'งานอื่นๆ', 'ช้อกล้าง', 'เช็คสงงาน', 'BRICC', 'KM web', 'สำนักส่งเสริมวิชาการแล...', and 'Home - Django RES...'. The main content area displays the Django REST framework interface for the 'All Todolist' endpoint. The interface includes a header 'Django REST framework' with the user 'pom' on the right. Below the header, there is a section for 'All Todolist' with a title 'All Todolist' and two buttons: 'OPTIONS' and 'GET'. The 'GET' button is selected, and the response is displayed in a code block. The response is a JSON array of two objects, each representing a todo item with 'id', 'title', and 'detail' fields.

```
localhost:8000/api/all-todolist/
- napaphat w. -...  ที่ติดดาว - Google ได...  สอน 2/2566  ทำพศ.  งานอื่นๆ  ช้อกล้าง  เช็คสงงาน  BRICC  KM web  สำนักส่งเสริมวิชาการแล...  Home - Django RES...

Django REST framework pom

All Todolist

All Todolist OPTIONS GET

GET /api/all-todolist/

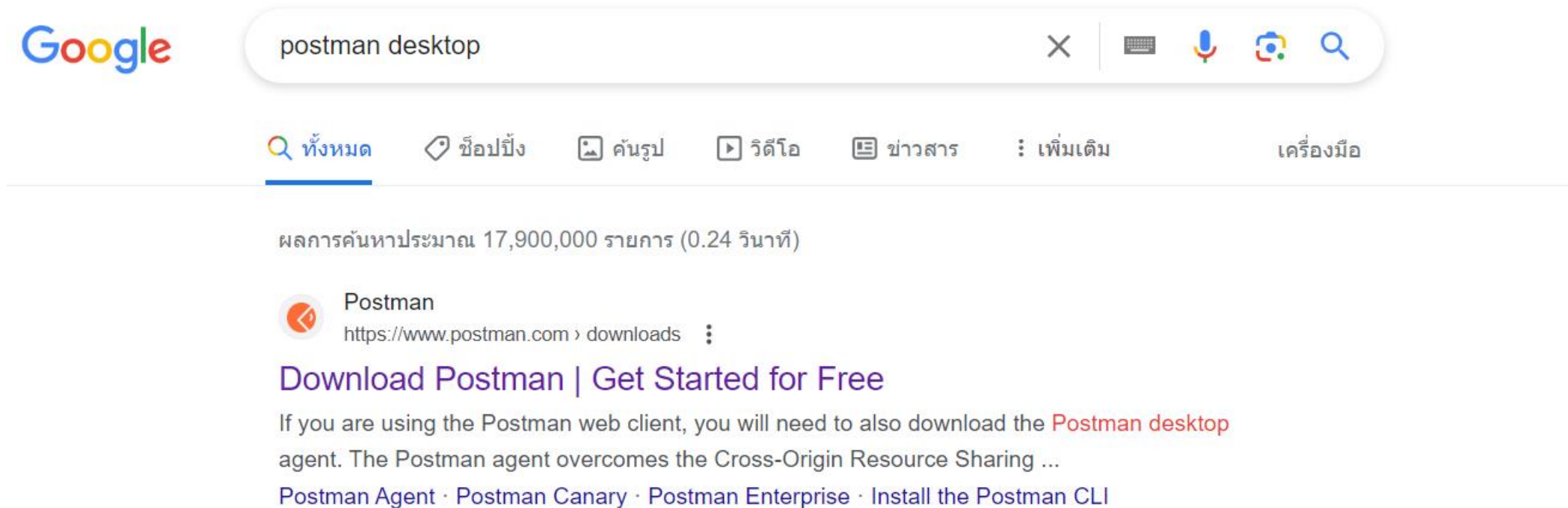
HTTP 200 OK
Allow: OPTIONS, GET
Content-Type: application/json
Vary: Accept

[
  {
    "id": 1,
    "title": "พักผ่อน",
    "detail": "กินอาหารอร่อยๆ"
  },
  {
    "id": 2,
    "title": "ไปตลาด",
    "detail": "ซื้อขนมขบเคี้ยว  ะ"
  }
]
```

5. ติดตั้ง โปรแกรม Postman

เป็นโปรแกรมที่จะใช้ทดสอบจำลอง การส่ง **API** ระหว่าง **server** กับ **flutter**

5.1 คำน่าว่า postman desktop > ไปที่ web postman >




The image shows a Google search interface. The search bar contains the text "postman desktop". Below the search bar, there are navigation options: "ทั้งหมด" (All), "ข้อปั้ง" (Images), "ค้นรูป" (Videos), "วิดีโอ" (Maps), "ข่าวสาร" (More), "เพิ่มเติม" (Tools), and "เครื่องมือ". The search results show approximately 17,900,000 results in 0.24 seconds. The top result is from Postman, with the URL "https://www.postman.com > downloads". The title of the result is "Download Postman | Get Started for Free". The snippet below the title reads: "If you are using the Postman web client, you will need to also download the Postman desktop agent. The Postman agent overcomes the Cross-Origin Resource Sharing ...". At the bottom of the snippet, there are links for "Postman Agent", "Postman Canary", "Postman Enterprise", and "Install the Postman CLI".

Google

postman desktop

ทั้งหมด ข้อปั้ง ค้นรูป วิดีโอ ข่าวสาร เพิ่มเติม เครื่องมือ

ผลการค้นหาประมาณ 17,900,000 รายการ (0.24 วินาที)

 Postman
https://www.postman.com > downloads

[Download Postman | Get Started for Free](#)

If you are using the Postman web client, you will need to also download the **Postman desktop** agent. The Postman agent overcomes the Cross-Origin Resource Sharing ...

[Postman Agent](#) · [Postman Canary](#) · [Postman Enterprise](#) · [Install the Postman CLI](#)

5.2 คลิกลิงก์แรก > download ที่ปุ่มส้ม > ติดตั้ง > สมัครสมาชิก



Product ▾ Pricing Enterprise Resources and Support ▾ Public API Network

Download Postman

Download the app to get started using the Postman API Platform today. Or, if you prefer a browser experience, you can try the web version of Postman.

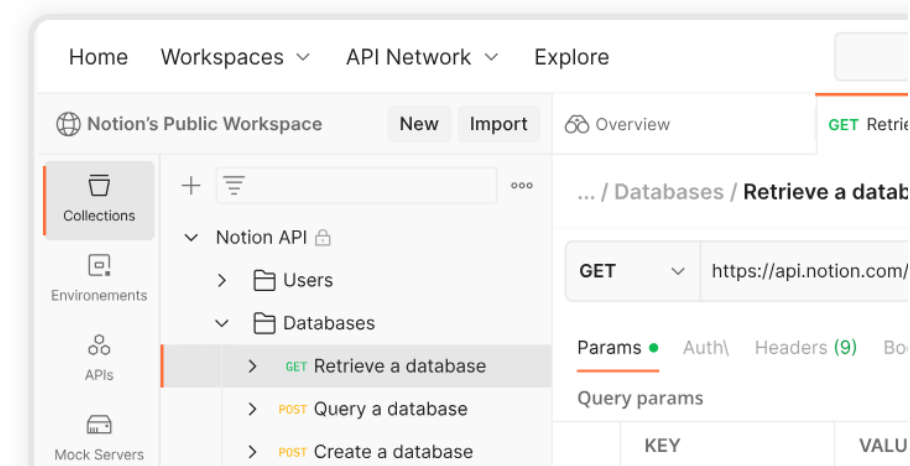
The Postman app

Download the app to get started with the Postman API Platform.

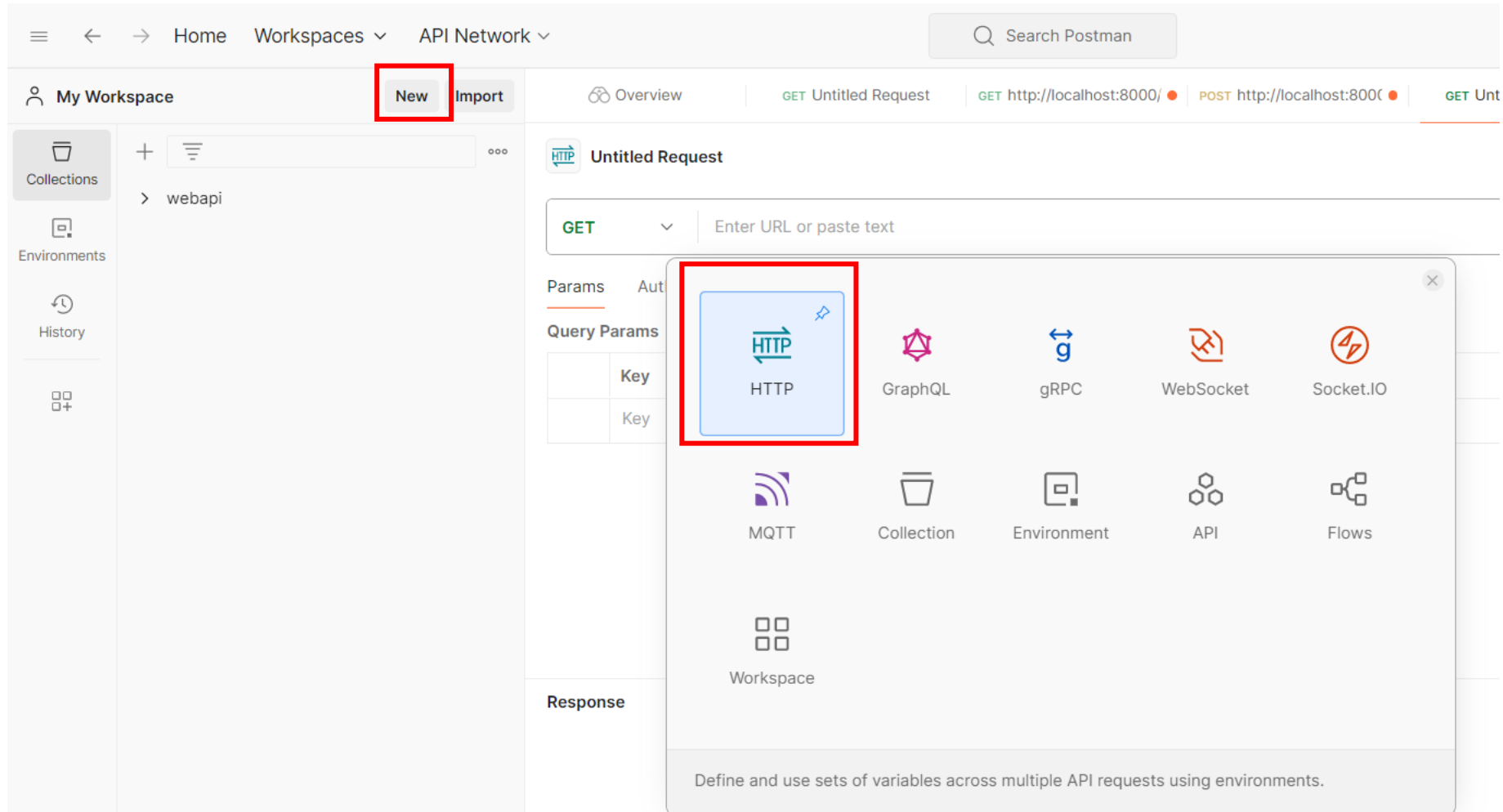
 Windows 64-bit

By downloading and using Postman, I agree to the [Privacy Policy](#) and [Terms](#).

[Release Notes](#)



5.3 คลิก New > HTTP



5.4 เปิดหน้าจอ > ตรวจสอบว่า server รันอยู่ไหม ถ้าไม่รัน ต้อง ตั้งรัน

cmd C:\Windows\System32\cmd.exe - python manage.py runserver

```
Microsoft Windows [Version 10.0.19045.3930]
(c) Microsoft Corporation. All rights reserved.

C:\napaphatFlutter\webapi>. \venv\scripts\activate

(venv) C:\napaphatFlutter\webapi>cd flutterapi

(venv) C:\napaphatFlutter\webapi\flutterapi>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
February 09, 2024 - 20:36:27
Django version 4.2, using settings 'flutterapi.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

5.5 Copy url จากหน้า Django rest framework มาใส่ > กดปุ่ม send ด้านขวา

The screenshot shows the Postman interface. The top bar includes navigation (Home, Workspaces, API Network), a search bar, and utility buttons (Invite, Upgrade). The main workspace shows a collection named 'webapi' with an environment set to 'No Environment'. A request is configured with the method 'GET' and the URL 'http://localhost:8000/api/all-todolist/'. The 'Send' button is highlighted in blue. Below the request configuration, tabs for 'Params', 'Authorization', 'Headers (6)', 'Body', 'Pre-request Script', 'Tests', and 'Settings' are visible. The 'Params' tab is active, showing a table for 'Query Params' with columns for Key, Value, and Description.

Key	Value	Description	...	Bulk Edit
Key	Value	Description		

5.6 จะได้ผลลัพธ์ดังรูป แสดงว่าเมื่อถึงน่าจะดึงข้อมูลจาก server ได้

The screenshot shows the Postman interface with a GET request to `http://localhost:8000/api/all-todolist/` that has been successfully executed. The response is a JSON array of two objects, each representing a todo item. The status is 200 OK, and the response size is 513 B.

Query Params

Key	Value	Description	Bulk Edit
Key	Value	Description	

Body (JSON)

```
1 [
2   {
3     "id": 1,
4     "title": "พักผ่อน",
5     "detail": "กินอาหารอร่อยๆ"
6   },
7   {
8     "id": 2,
9     "title": "ไปตลาด",
10    "detail": "ซื้อขนมยี่ห้อต่างๆ"
11  }
12 ]
```

5.7 ไปที่ไฟล์ views.py ของ myapp > พิมพ์โค้ดดังรูป

```
models.py × serializers.py ● views.py × urls.py Release Notes: 1.86.0 settings.py admin.py
flutterapi > myapp > views.py > ...
8 from .models import Todolist
9
10 @api_view(['GET'])
11 def all_todolist(request):
12     alltodolist = Todolist.objects.all()
13     serializer = TodolistSerializer(alltodolist, many=True)
14     return Response(serializer.data, status=status.HTTP_200_OK)
15
16 @api_view(['POST'])
17 def post_todolist(request):
18     if request.method == 'POST':
19         serializer = TodolistSerializer(data=request.data)
20         if serializer.is_valid():
21             serializer.save()
22             return Response(serializer.data, status=status.HTTP_201_CREATED)
23         return Response(serializer.errors, status=status.HTTP_404_NOT_FOUND)
24
```

5.8 ไปที่ไฟล์ urls.py ของ myapp > พิมพ์โค้ดเพิ่มดังรูป

```
models.py  serializers.py  views.py  urls.py  tests.py
flutterapi > myapp > urls.py > ...
1  from django.urls import path
2  from .views import *
3  #from .views import Home
4
5  urlpatterns = [
6      #path('', Home),
7      path('api/all-todolist/', all_todolist),
8      path('api/post-todolist/', post_todolist)
9  ]
10
```

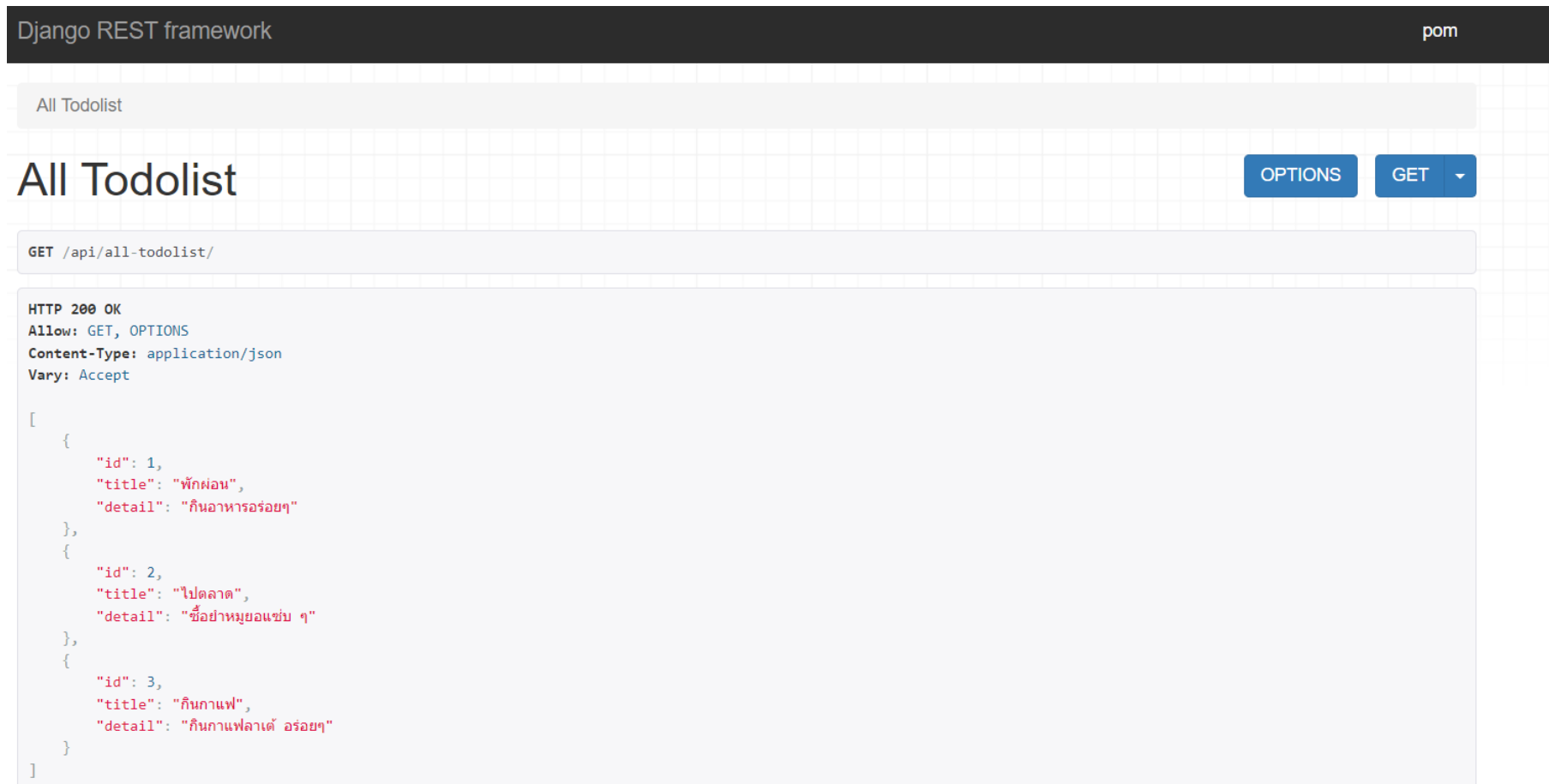
5.9 ไปที่ postman > กด + เพิ่ม tab ใหม่ > เลือก post > ใส่ url ตามรูป > ไปที่ body > เลือก form-data > กรอกข้อมูล > กด send

The screenshot shows the Postman interface. At the top, there's a search bar and navigation icons. Below that, the request configuration is visible: a POST request to `http://localhost:8000/api/post-todolist/`. The 'Body' tab is selected, and 'form-data' is chosen as the body type. A table with two rows is shown, with checkboxes for 'title' and 'detail' selected. The 'title' row has the value 'กินกาแฟ' and the 'detail' row has 'กินกาแฟพลาด์ อร่อยๆ'. Below the table, there's a 'Key' and 'Value' section. At the bottom, the 'Body' tab shows the response in JSON format: `{ "id": 3, "title": "กินกาแฟ", "detail": "กินกาแฟพลาด์ อร่อยๆ" }`.

Key	Value	Description
<input checked="" type="checkbox"/> title	Text <input type="text" value="กินกาแฟ"/>	กินกาแฟพลาด์ อร่อยๆ
<input checked="" type="checkbox"/> detail	Text <input type="text" value="กินกาแฟพลาด์ อร่อยๆ"/>	
Key	Text <input type="text" value="Value"/>	Description

```
1 {
2   "id": 3,
3   "title": "กินกาแฟ",
4   "detail": "กินกาแฟพลาด์ อร่อยๆ"
5 }
```

5.10 กลับไปที่หน้า Django rest framework > กด refresh จะพบว่ามึข้อมูลที่เราเพิ่มใหม่



Django REST framework pom

All Todolist

All Todolist

OPTIONS GET

GET /api/all-todolist/

HTTP 200 OK
Allow: GET, OPTIONS
Content-Type: application/json
Vary: Accept

```
[
  {
    "id": 1,
    "title": "พักผ่อน",
    "detail": "กินอาหารอร่อยๆ"
  },
  {
    "id": 2,
    "title": "ไปตลาด",
    "detail": "ซื้อยาหมอยอแซบ ะ"
  },
  {
    "id": 3,
    "title": "กินกาแฟ",
    "detail": "กินกาแฟลาเต้ อร่อยๆ"
  }
]
```

Reference

ลุงวิศวกร สอนคำนวณ. Flutter 2 Mobile App Bootcamp 2021 - Uncle Engineer Stackpython. (2021). Django REST Framework - สร้าง API ด้วย Python ฉบับเต็ม <https://stackpython.co/tutorial/django-rest-framework-api-python>