

# JSON

JavaScript Object Notation

## การเปิดเผยข้อมูลด้วย Web API

- Web API (Application Programming Interface) คือ การเผยแพร่ข้อมูลที่อยู่ในรูปแบบมาตรฐาน ให้บุคคลภายนอกสามารถเข้าถึงได้
- Web Site มนุษย์เป็นผู้อ่าน แต่ Web API แอปพลิเคชันเป็นผู้อ่าน
- การเผยแพร่ข้อมูลในแบบ Web API เป็นวิธีที่ช่วยให้นักพัฒนาโปรแกรมสามารถนำข้อมูลไปใช้ประมวลผล หรือแสดงผล ในวิธีการของตนเองได้
- รูปแบบมาตรฐานในปัจจุบัน ได้แก่ XML และ JSON

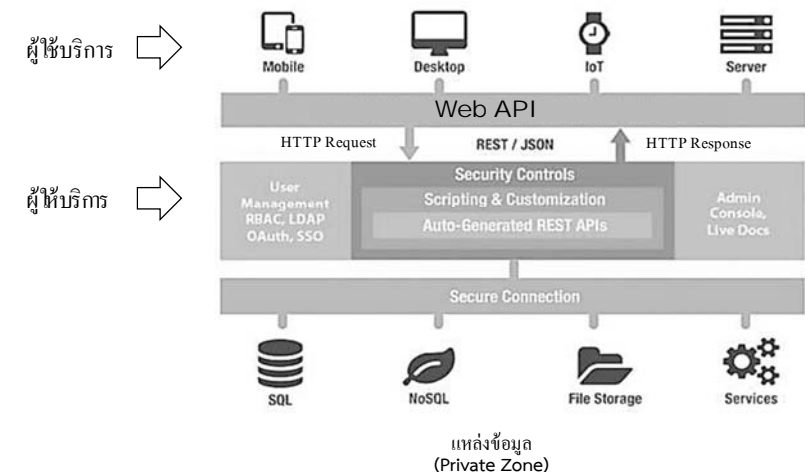
2

## ผู้ให้บริการและ ผู้ใช้บริการ

- ผู้ให้บริการ API
  - Twitter API
  - Facebook API ให้บริการข้อมูลเพื่อให้แอปพลิเคชันอื่นๆ ไปวิเคราะห์ เช่น การตลาด กระแสความนิยม
  - Google Maps Maps ให้บริการข้อมูลเกี่ยวกับแผนที่
  - รัฐบาลไทย ให้ข้อมูลเกี่ยวกับประเทศไทย
- ผู้ใช้ API
  - Mobile Application
  - Web Application
  - IoT Application

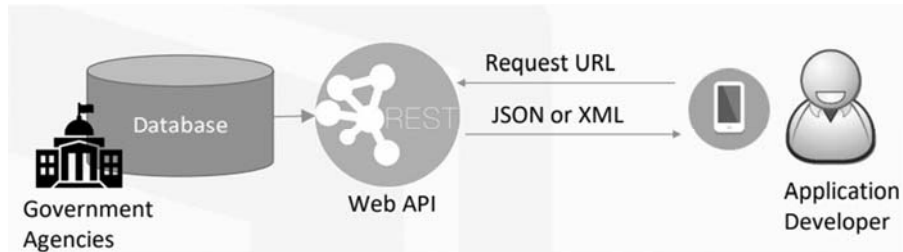
3

## สถาปัตยกรรมระบบที่มี Web API



4

## Web API ชุดข้อมูลเปิดภาครัฐ



ที่มา: [https://www.nstda.or.th/nac/2018/images/13\\_OpenData\\_Marut\\_I.pdf](https://www.nstda.or.th/nac/2018/images/13_OpenData_Marut_I.pdf)

5

## JSON

- รูปแบบข้อความมาตรฐานสำหรับการแทนโครงสร้างข้อมูล
- ใช้ในการส่งข้อมูลระหว่างไคลเอนต์และเซิร์ฟเวอร์
  - Web Browser <-> Server
  - Mobile Application <-> Server
  - IoT Application <-> Server

6

## JSON กับภาษา JavaScript

- JSON มีรูปแบบไวยากรณ์คล้ายกับ Object และ array ของ JavaScript

```
<html>
<body>
<script>
  let person = {
    id: 69,
    fullname: 'John Smith',
    weight: 72.5,
    option: ['move', 'stop', 'slow']
  }
  console.log(person.weight * 0.5)
</script>
</body>
</html>
```

Object ของ JavaScript

7

## JSON Syntax

- ข้อมูลที่อยู่ในรูปแบบ JSON อยู่ในรูปแบบลำดับของคู่ key และ value

**"key1":"value1", "key2":"value2",...**

- key คือ คำอธิบายข้อมูลจะต้องขึ้นต้นและปิดท้ายด้วยเครื่องหมาย “
- หากมีหลายคู่ key และ value จะคั่นด้วยเครื่องหมาย comma
- รายการ key และ value เป็นรายการแบบไม่สนใจลำดับ

8

## ชนิดข้อมูลของ JSON

ชนิด	คำอธิบาย
Number	เป็นชนิด integer, double หรือ floating-point โดยไม่ต้องใส่เครื่องหมาย " " ครอบ
String	จะต้องขึ้นต้นและปิดท้ายด้วยเครื่องหมาย " "
Boolean	ค่าเป็น true หรือ false
null	ค่าว่าง

ตัวอย่าง เช่น

```
{  
  "name": "John",  
  "age": 35,  
  "distinction": true  
}
```

9

## JSON Object

- JSON Object คือ โครงสร้างข้อมูลที่ประกอบด้วยรายการ ของ คู่ key และ value ในเครื่องหมาย { และ } เช่น

```
{  
  "id": "01",  
  "bookName": "Java Programming",  
  "edition": "third",  
  "author": "Robert"  
}
```

Diagram labels: key points to the opening curly brace, value points to the string "01", and another value points to the string "Java Programming".

10

## JSON Array

- เป็นรายการแบบมีลำดับ ภายได้เครื่องหมาย [ .....]
- สิ่งที่แตกต่างจาก JSON Object คือ สมาชิกของอาร์เรย์จะไม่มีชื่อ key ขณะที่ JSON Object สมาชิกจะมีชื่อ key เสมอ
- แต่ละสมาชิกจะคั่นด้วย ,

```
"destination": [  
  "Udonthani",  
  "Bangkok",  
  "Khon Kaen"  
]
```

Diagram labels: key points to the opening curly brace, and a bracket on the right indicates the elements are members of the array.

11

## Object ที่มี Array

```
{  
  "President": "Alan Isaac",  
  "CEO": "David Richardson",  
  "manager": [  
    "Sachin Tendulkar",  
    "Virender Sehwag",  
    "Gautam Gambhir"  
  ],  
  "staff": [  
    "Alastair Cook",  
    "Jonathan Trott",  
    "Kevin Pietersen"  
  ]  
}
```

Diagram labels: manager และ staff มีหลายคนจึงกำหนดให้ value มีค่าเป็นอาร์เรย์

12

## Array ของ Object

```
{
  "book": [
    {
      "id": "01",
      "language": "Java",
      "edition": "third",
      "author": "Herbert Schildt"
    },
    {
      "id": "07",
      "language": "C++",
      "edition": "second",
      "author": "E.Balagurusamy"
    }
  ]
}
```

13

## Object ซ่อน Object

object ของ address ที่  
ซ่อนอยู่ และภายใน  
object ยังมี object  
geolocation อีกด้วย

```
{
  "fullName": "John",
  "lastName": "Smith",
  "address": {
    "zipCode": "40000",
    "country": "Thailand",
    "geolocation": {
      "lat": 16.4331702,
      "lng": 102.8407473
    }
  }
}
```

14

## การแปลง JavaScript Object เป็น String JSON

- Object ในภาษา JavaScript สามารถแปลงเป็น String ได้ด้วยคำสั่ง

```
JSON.stringify( ตัวแปรที่เป็นobject หรือ array )
```

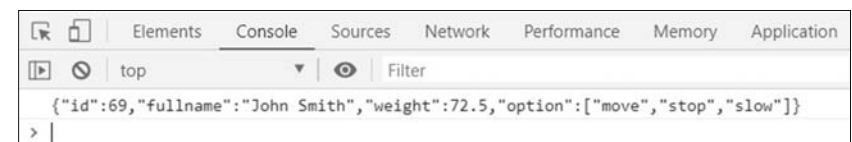
- จะใช้เมื่อต้องการส่งข้อมูลไปยัง server

15

## ตัวอย่าง

```
<html><body>
<script>
  let person = {
    id: 69,
    fullname: 'John Smith', weight: 72.5,
    option: ['move', 'stop', 'slow']
  }

  // แปลงจาก object เป็น string รูปแบบ JSON
  let jsonString = JSON.stringify(person)
  console.log(jsonString)
</script>
</body></html>
```



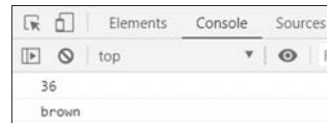
## การแปลง String เป็น Object ของ JavaScript

- String ที่อยู่ในรูปแบบ JSON สามารถแปลงเป็น object เพื่อให้นำไปใช้งานต่อด้วยคำสั่ง JSON.parse()

```
<html>
<body>
<script>
  // string ในรูปแบบ JSON ก่อนถูกแปลง
  let jsonString = '{"height":1.9, "age":36, "eyeColor":"brown"}'

  // ใช้ JSON.parse() เพื่อแปลงเป็น Object
  let person = JSON.parse(jsonString)

  // นำ object ไปใช้งาน
  console.log(person.age)
  console.log(person.eyeColor)
</script>
</body></html>
```



17

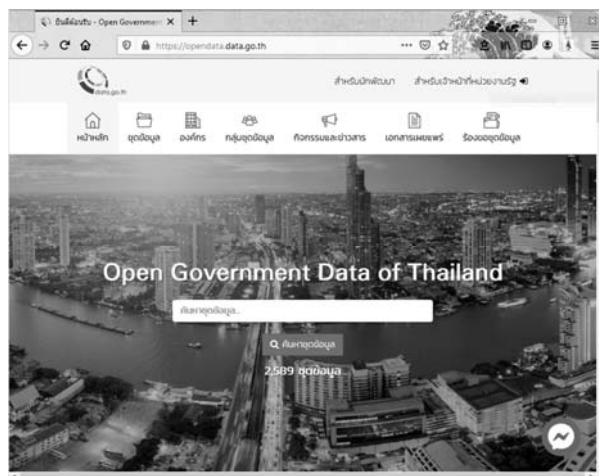
## การดึงข้อมูล JSON จาก Web API

- ศึกษารูปแบบของ URL สำหรับขอข้อมูล JSON จากคู่มือของ Web API
  - บาง Web API จะต้องสมัครสมาชิก เพื่อนำ token แแนบไปกับ URL
- ใช้โปรแกรมทดสอบดึงข้อมูล เช่น Postman
- เขียนคำสั่งดึงข้อมูลผ่าน HTTP โดยภาษาโปรแกรมใดๆ เช่น ภาษา JavaScript ใช้ Fetch API หรือ XMLHttpRequest
- ใช้ Library หรือคำสั่งช่วยแปลง JSON ให้เป็น object แล้วนำไปประมวลผล หรือแสดงผล

18

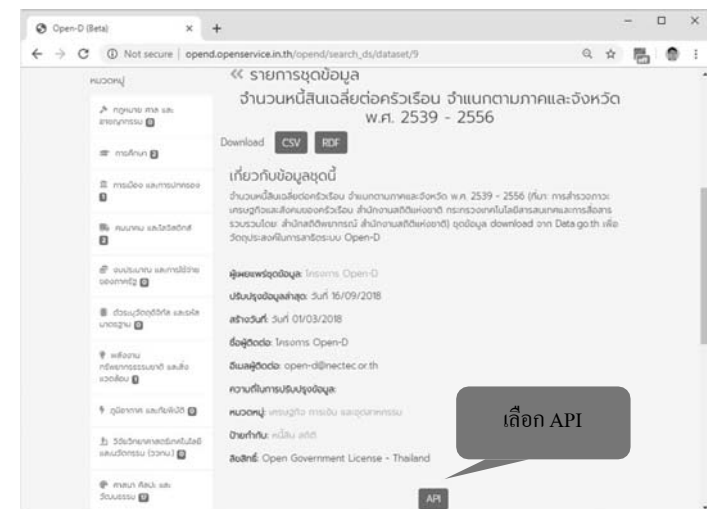
## Web API ข้อมูลเปิดภาครัฐ

<https://opendata.data.go.th/>



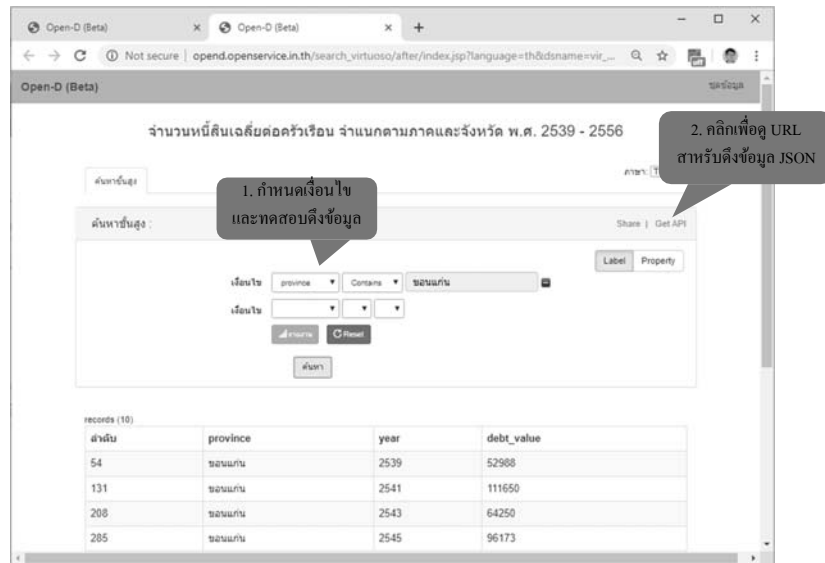
19

## เลือกข้อมูลที่ต้องการ

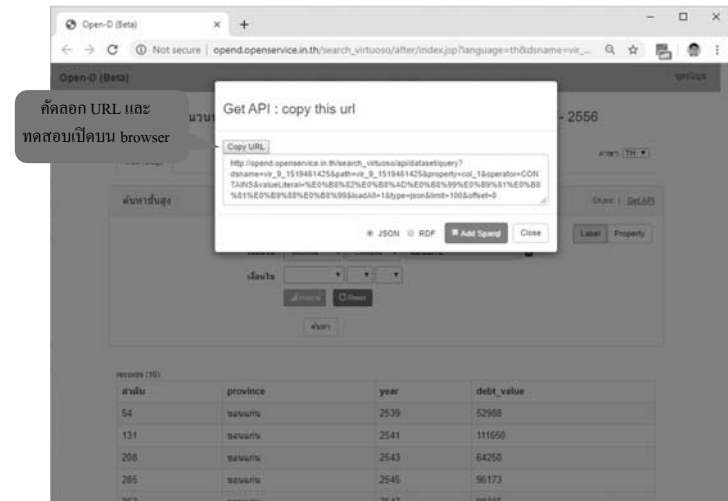


20

## ระบุเงื่อนไขข้อมูลที่ต้องการ

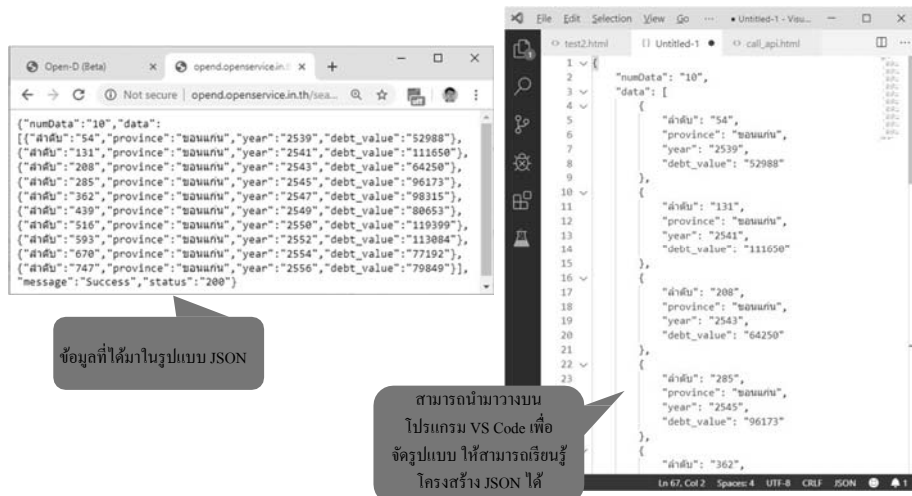


ขอ URL สำหรับดึงข้อมูล



22

## ทดสอบเปิดดูข้อมูล



23

## Extension ช่วยจัดมุมมอง JSON

- JSONView: <https://jsonview.com/>



24

# การเขียนคำสั่งอ่านข้อมูล JSON

```
<html>
<head>
  <script>
    async function getDataFromAPI() {
      let response = await fetch('http://opend.openservice.in.th/sea..')
      let rawData = await response.text() // อ่านผลลัพธ์
      let objectData = JSON.parse(rawData) // แปลงผลลัพธ์เป็น object
      let result = document.getElementById('result') // สิ่ง <ul> เพื่อใช้ในการเพิ่มแท็ก <li>

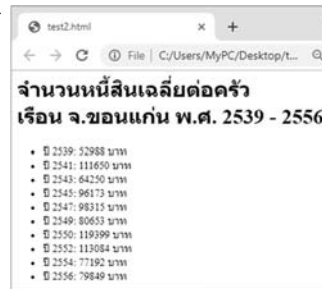
      for (let i = 0; i < objectData.data.length; i++) {
        let content = 'ปี' + objectData.data[i].year + ': ' // ดึงข้อมูลจาก object มาจัดรูปแบบ
        content += objectData.data[i].debt_value + ' บาท'

        let li = document.createElement('li') // สร้างแท็ก <li>
        li.innerHTML = content // นำข้อมูลที่จัดแล้วมาไว้ในแท็ก <li>
        result.appendChild(li) // เพิ่มแท็ก <li> ใหม่
      }

      getDataFromAPI() // เรียกฟังก์ชัน
    }
  </script>
</head>
<body>
  <h1>จำนวนหนี้สินเฉลี่ยต่อครัวเรือน จ.ขอนแก่น พ.ศ. 2539 - 2556</h1>
  <ul id="result"></ul>
</body></html>
```

async ใช้กับ await เพื่อบอกว่าการทำงานแบบ asynchronous  
await คือ การรอให้ประมวลผลเสร็จก่อน

รอให้การดึงข้อมูลจาก  
Web API เสร็จก่อน



## Web API สำหรับทดสอบดึงข้อมูล

- โปรแกรมที่แบ่งส่วน front-end และ back-end ออกจากกัน ฝั่ง front-end จำเป็นต้องออกแบบหน้าจอแสดงผล
- หาก back-end ยังไม่สามารถเตรียม Web API ให้ได้ทันที front-end อาจต้องจำลอง Web API ขึ้นมาก่อน
- front-end อาจใช้ Fake Online Web API ทดสอบดึงข้อมูลได้

<https://jsonplaceholder.typicode.com/>