

JSON

(JavaScript Object Notation)

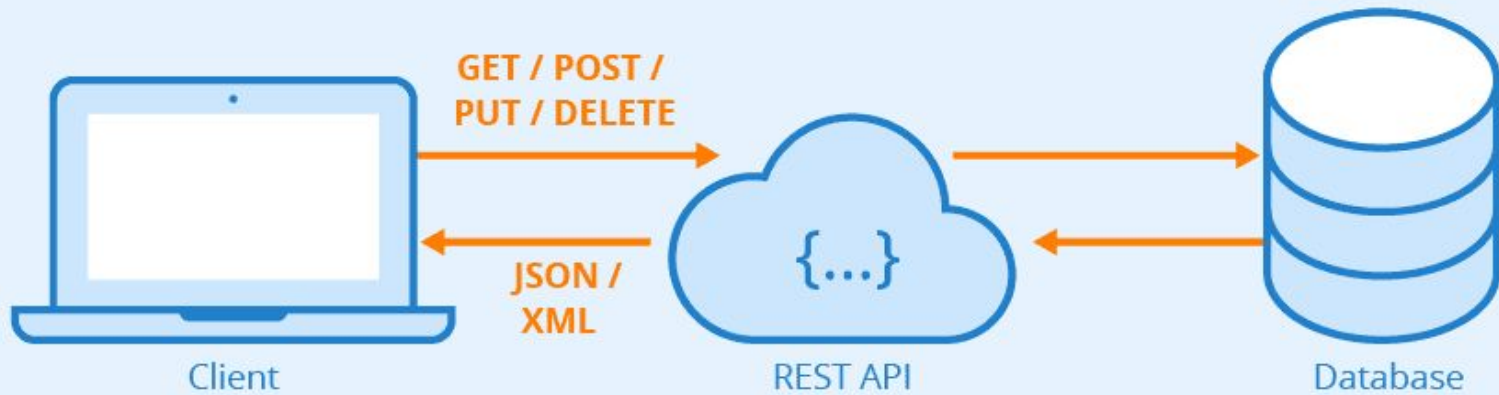
รู้จักกับ JSON

JSON (JavaScript Object Notation) เป็นรูปแบบการแลกเปลี่ยนหรือรับส่งข้อมูลในระบบคอมพิวเตอร์หรือแอปพลิเคชัน

ในอดีตการแลกเปลี่ยนหรือรับส่งข้อมูลนั้นจะใช้รูปแบบ XML แต่เนื่องจาก XML มีโครงสร้างที่ซับซ้อนและมีขนาดใหญ่จึงมีการเปลี่ยนมาใช้ JSON แทน

รู้จักกับ JSON

คุณสมบัติของ JSON เป็นไฟล์ประเภทข้อความ (Text)
มีโครงสร้างคำสั่งที่มนุษย์สามารถอ่าน-เขียนแล้วเข้าใจได้เลย
อีกทั้งยังมีขนาดเล็ก น้ำหนักเบา เป็นมาตรฐานกลางของทุก
ภาษา สำหรับการแลกเปลี่ยนข้อมูลข้ามแพลตฟอร์มบนระบบ
ปฏิบัติการที่แตกต่างกัน



<https://www.datamounts.com/difference-rest-api-restful-api/>

JSON (JavaScript Object Notation) ในตอนเริ่มต้นถูกใช้ในการเขียนโปรแกรมด้วย JavaScript ในการแลกเปลี่ยนข้อมูลกับ Server ปัจจุบันนิยมนำ JSON มาใช้เป็นรูปแบบในการแลกเปลี่ยนข้อมูลในเว็บหรือแอปพลิเคชัน มีหลายภาษาที่รองรับการแลกเปลี่ยนข้อมูล เช่น Java , Python , C# เป็นต้น

ชนิดข้อมูล (Data Types)

รูปแบบ	คำอธิบาย
Number	ตัวเลข (ตัวเลขจำนวนเต็ม , ตัวเลขทศนิยม)
String	ชุดข้อความใช้เครื่องหมาย double-quote (“)
Boolean	ค่าทางตรรกศาสตร์ (True / False)
Null	ค่าว่าง

ชนิดข้อมูล (Data Types)

รูปแบบ	คำอธิบาย
Array	อาร์เรย์หรือชุดข้อมูลซึ่งจะเป็นชนิดข้อมูลใดก็ได้เขียนในพื้นที่สัญลักษณ์ [] และค้นสมาชิกในอาร์เรย์แต่ละค่าด้วยคอมม่า (,) เช่น [var1,var2]
Object	ชุดข้อมูลที่เป็นคู่ Key-Value แบบสตริงใช้สัญลักษณ์ปีกกา และคอมม่า (,) เป็นตัวแบ่งแต่ละคู่ และใช้โคลอน (:) เป็นตัวแบ่งฝั่งระหว่าง key และ value เช่น {"key1":value1,"key2":value2}

JavaScript Object & JSON

การสร้าง JSON นั้นหลักการจะคล้ายกับ Object ใน JavaScript แต่จะมีข้อกำหนดบางอย่างที่แตกต่างกัน

```
let objectName = {  
    property1 : value1,  
    property2 : value2  
}
```

JavaScript Object

```
let objectName = {  
    property1 : value1,  
    property2 : value2  
}
```


JavaScript Object

```
let objectName = {  
    property1 : value1,  
    property2 : value2  
}
```

```
let user = {  
    name : "kongruksiam",  
    age : 27  
}
```

การเข้าถึงข้อมูลใน JavaScript Object

```
let user = {  
  name : "kongruksiam",  
  age : 27  
}
```

- user.name
- user.age

หรือ

- user["name"]
- user["age"]

การเข้าถึงข้อมูลใน JavaScript Object

```
let user = {  
  name : "kongruksiam",  
  age : 27  
}
```

- user.name

- user.age

หรือ

- user["name"]

- user["age"]

การเข้าถึงข้อมูลใน JavaScript Object

- `user.name="ก้องรักสยาม"`
- `user.age = 30`

หรือ

- `user["name"] = "ก้องรักสยาม"`
- `user["age"] = 30`

การเข้าถึงข้อมูลใน JavaScript Object

```
let user = {  
  name : "kongruksiam",  
  age : 27 ,  
  getData(){  
    //คำสั่งต่างๆ  
  }  
}
```

- user.getData()

JavaScript Object

```
let objectName = {  
    property1 : value1,  
    property2 : value2  
}
```

JavaScript Object

```
let objectName = {  
  property1 : value1,  
  property2 : value2  
}
```

JSON

```
let objectName = {  
  key1 : value1,  
  key2 : value2  
}
```

JavaScript Object

```
let objectName = {  
    property1 : value1,  
    property2 : value2  
}
```

JSON

```
let objectName = {  
    "key1" : value1,  
    "key2" : value2  
}
```


JavaScript Object

```
let objectName = {  
  property1 : value1,  
  property2 : value2  
}
```

Object

JSON

```
let objectName = {  
  "key1" : value1,  
  "key2" : value2  
}
```

Text

JavaScript Object

```
let objectName = {  
  property1 : value1,  
  property2 : value2  
}
```

Object

JSON

```
let objectName = {  
  "key1" : value1,  
  "key2" : value2  
}
```

String

JSON Syntax Rule

```
let objectName = {  
    "key1" : value1,  
    "key2" : value2  
}
```

- ชื่อคีย์จะเขียนในเครื่องหมาย Double Quote (“”) เท่านั้น
- ข้อมูล (value) กำหนดตามชนิดข้อมูลที่ JSON รองรับ
- JSON ไม่สามารถสร้างฟังก์ชันหรือเมธอดได้

JSON Syntax Rule

```
let objectName = {  
    "key1" : value1,  
    "key2" : value2  
}
```

- การเข้าถึงข้อมูลจะใช้วิธีการแบบเดียวกันกับ JavaScript Object
- นามสกุลไฟล์ **.json**
- MIME Type :
"Application/json"

JavaScript Object

```
let user = {  
  name : "kongruksiam",  
  age : 27  
}
```

JSON

```
let user = {  
  "name" : "kongruksiam",  
  "age" : 27  
}
```

JavaScript Object

```
let user = {  
  name : "kongruksiam",  
  age : 27  
}
```

Type: Object

JSON

```
let user = {  
  "name" : "kongruksiam",  
  "age" : 27  
}
```

Type: String

JavaScript Object

```
let user = {  
  name : "kongruksiam",  
  age : 27  
}
```

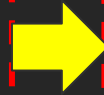
JSON String

```
let user = {  
  "name" : "kongruksiam",  
  "age" : 27  
}
```

JSON Parse

- แปลง JSON String ให้อยู่ในรูปแบบ JavaScript Object

```
let user = {  
  "name" : "kongruksiam",  
  "age" : 27  
}
```

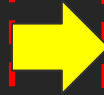


```
let user = {  
  name : "kongruksiam",  
  age : 27  
}
```


JSON Parse

- แปลง JSON String ให้อยู่ในรูปแบบ JavaScript Object

```
let user = `{  
  "name" : "kongruksiam",  
  "age" : 27  
}`
```

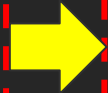


```
let user = {  
  name : "kongruksiam",  
  age : 27  
}
```

JSON Stringify

- แปลง JavaScript Object ให้อยู่ในรูปแบบเป็น JSON String

```
let user = {  
  name : "kongruksiam",  
  age : 27  
}
```



```
let user = {  
  "name" : "kongruksiam",  
  "age" : 27  
}
```

JSON Array & Objects

รูปแบบ	คำอธิบาย
Array	อาร์เรย์หรือชุดข้อมูลซึ่งจะเป็นชนิดข้อมูลใดก็ได้เขียนในพื้นที่สัญลักษณ์ [] และคั่นสมาชิกในอาร์เรย์แต่ละค่าด้วยคอมม่า (,) เช่น [var1,var2]
Object	ชุดข้อมูลที่เป็นคู่ Key-Value แบบสตริงใช้สัญลักษณ์ปีกกา และคอมม่า (,) เป็นตัวแบ่งแต่ละคู่ และใช้โคลอน (:) เป็นตัวแบ่งฝั่งระหว่าง key และ value เช่น {"key1":value1,"key2":value2}

JSON Array

```
let user = {  
  "name" : "kongruksiam",  
  "age" : 27,  
  "languages" : ["ไทย", "จีน", "ญี่ปุ่น"],  
}
```

```
user.languages[0] //ไทย
```

JSON Objects

```
let user = {  
  "name" : "kongruksiam",  
  "general" : {  
    "weight":70,  
    "height":175,  
  }  
}
```

- user.general.weight
หรือ
- user.general["weight"]