

Python v22.1

(Online Part-time)

<

AJAX

>

AJAX + API's

▲

Overview of API's

✓

Overview of AJAX

✓

Fetching Data

✓

JSON

Knowledge Check: AJAX + APIs

Getting Github

API Key

Dojo Weather

AJAX + Flask

▼

JSON

Learning Objectives:

- Students will explain what JSON data is.
- Students will access the values stored inside a JSON object.

What is JSON?

JSON or JavaScript Object Notation is a lightweight format for storing and transporting data. The data is stored in key:value pairs, just like data is stored in JavaScript objects. JSON is the standard return type of most API's. Below is an example of what JSON could look like.

```
var data = {
  "orders": [
    {
      "orderno": 784692,
      "date": "June 30, 2088 1:54:23 AM",
      "trackingno": "TN000391",
      "customer": {
        "custid": 11045,
        "fname": "Sue",
        "lname": "Hatfield",
        "address": "1409 Silver Street",
        "city": "Ashland",
        "state": "NE",
        "zip": 68003
      }
    },
    {
      "orderno": 784693,
      "date": "March 3, 2088 8:18:14 PM",
      "trackingno": "TN000468",
      "customer": {
        "custid": 11045,
        "fname": "Sue",
        "lname": "Hatfield",
        "address": "1409 Silver Street",
        "city": "Ashland",
        "state": "NE",
        "zip": 68003
      }
    }
  ]
}
```

In order to access the data, we need to know what data type we are working with. Paste the above code in your text editor, along with the following command.

```
console.log(typeof(data));
// 'object'
console.log(Array.isArray(data));
// false
console.log(Array.isArray(data.orders));
// true
```

`typeof()` will return the data type of the argument passed in. We will see, `typeof(data)` returns an object.

Counterintuitively, in JavaScript, the built-in array data structure is also of type `object` if you look 'under the hood'. Thus, we will see "object", if we execute `console.log(typeof(data.orders))`. Luckily, you can use `Array.isArray(someVariable)` to check if a variable is an array-type object.

We can access values in our object with either dot notation, or square bracket notation.



Square Bracket notation:

```
console.log(data['orders'][0].orderno);  
// prints 784692
```

Dot Notation:

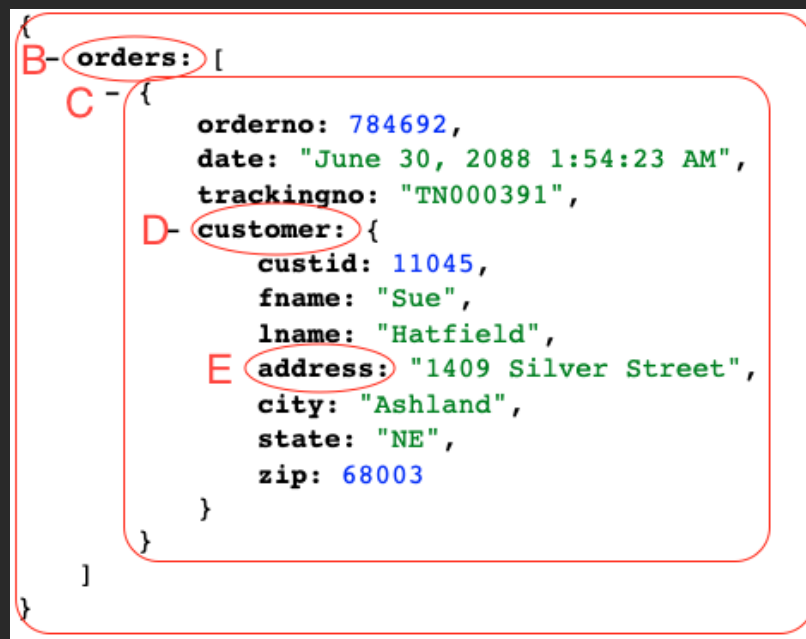
```
console.log(data.orders[0].orderno);  
// prints 784692
```

Accessing JSON

Let's take a look on how we can access a JSON object. In below image we have a JSON object. For this example let's assume this object is assigned to the variable **data**.

```
console.log(data);  
// would print the below image
```

This will be our starting point (A) to get to the customer address (E). Each letter marks a different reference point we need to address in order to get in information we are looking for. Below is a break down of the steps we would need to take.



```
// A - object  
console.log(data);  
// {orders: Array(1)}  
// B - array at 'orders' key in object  
console.log(data.orders);  
// [{...}]  
// C - object at index position 0 at 'orders' key in object  
console.log(data.orders[0]);  
// {"orderno": 784692, "date": "June 30, 2088 1:54:23 AM", "trackingno":  
"TN000391", "customer": {...}}  
// D - object at key 'customer' at index position 0 at 'orders' key in object  
console.log(data.orders[0].customer);  
// {"custid": 11045, "fname": "Sue", "lname": "Hatfield", "address": "1409 Silver Street",  
"city": "Ashland", ...}  
// E - value at key 'address' in object at key 'customer' at index position 0 at 'orders'  
key in object  
console.log(data.orders[0].customer.address);  
// 1409 Silver Street
```

