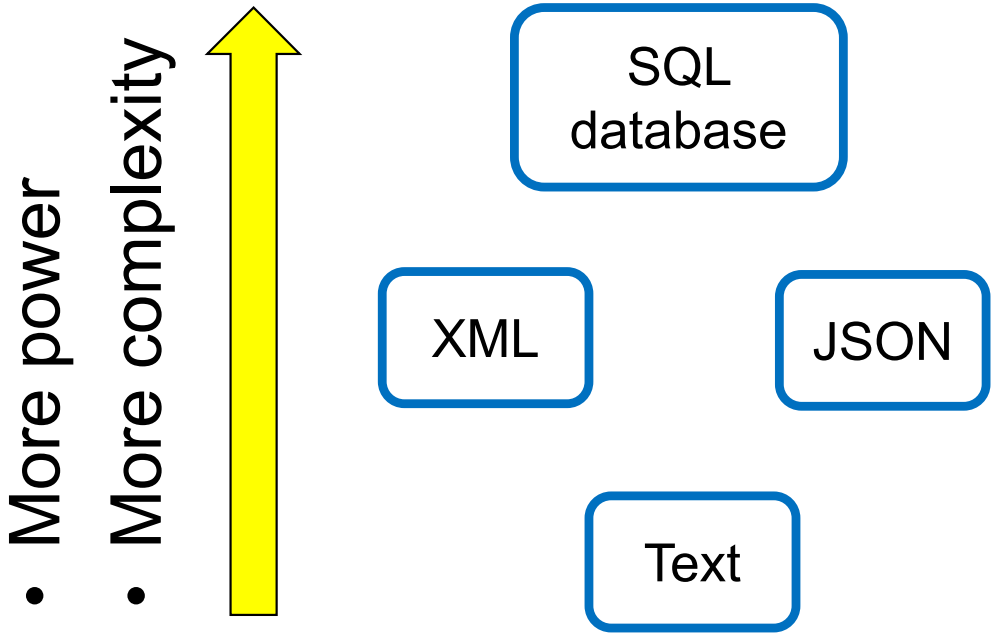
COMP4021
Internet Computing

JSON

David Rossiter

Basic Web Storage Solutions

- These are the basic storage choices for web projects



An Object

- JSON means JavaScript Object Notation
- JSON is almost the same as a JavaScript object
- So first we will look at JavaScript objects
- This is an example of making a JavaScript object
  - The last item doesn't have a comma at the end

```
person = {
      "name":"Dave",
      "midterm":78,
      "lab02":95,
      "lab03":100
}
```

Accessing Object Fields

- For example, to access the name you can use `person.name` or `person["name"]`

```
> person = {
      "name":"Dave",
      "midterm":78,
      "lab02":95,
      "lab03":100
  }
◂ ▶ {name: "Dave", midterm: 78, lab02: 95, lab03: 100}
> console.log( person.name )
  Dave
◂ undefined
> console.log( person["name"] )
  Dave
◂ undefined
```

```
<!DOCTYPE html>
<html>
<head>
<style>
table, td {
border: 1px solid black;
}
</style>
<script>
    person = {
        "name":"Dave",
        "midterm":78,
        "lab02":95,
        "lab03":100
    }
```

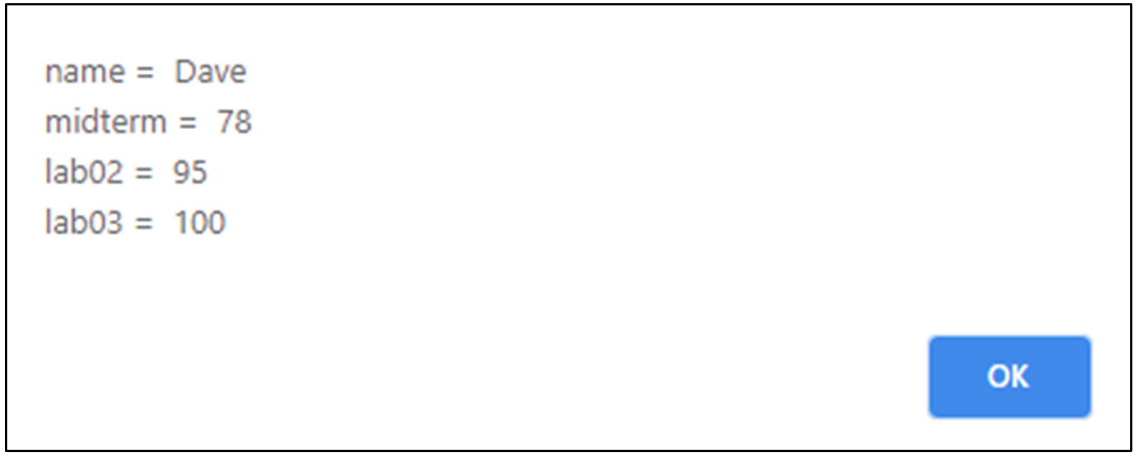
```
function use_js_object() {
        display=document.getElementById("results_go_here");
        display.innerHTML="<tr><td>" + person.name +
                "</td><td>" + person.midterm + "</td></tr>";
}     </script></head>

        <body onload="use_js_object()">
                <table id="results_go_here"></table>
        </body>
        </html>
```

- Result:

Accessing Some Fields in an Object

Accessing all Object Fields

- If you want to, you can access all the fields in an object using a for loop




```
var textToShow="";
    iterate through keys
for (leftSide in person)

    textToShow+=leftSide + " =  " + person[leftSide] + "\n";

alert(textToShow);
```

```
<!DOCTYPE html>
<html>
<head>
<style>
table, td {
border: 1px solid black;
}
</style>
<script>
    person = {
        "name":"Dave",
        "midterm":78,
        "lab02":95,
        "lab03":100
    }
```

```
function use_js_object() {
        display=document.getElementById("results_go_here");
        for (leftSide in person) {
            display.innerHTML=display.innerHTML+
            "<tr><td>" + leftSide + "</td><td>" +
            person[leftSide] + "</td></tr>";
        }
}
    <body onload="use_js_object()">
        <table id="results_go_here">
        </table>
    </body>
    </html>
```

- Result:

name	Dave
midterm	78
lab02	95
lab03	100

Accessing all Fields in an Object

Object Structure

- You can have a structure like this with ‘branches’ of data, as many as you need
- In this example `person["labs"][2]` will give the answer 95

```
person = {
        "name":"Dave",
        "midterm":78,
        "labs": {
              2:95,
              3:100
        }
}
```

注意这里不是index, 而是labs嵌套了一个object，然后这个object有两个key，一个2，一个3

Including Arrays in an Object

- You can include arrays in the object structure
- In this example `person["labs"][0]` will give the answer 95

这次这个是index

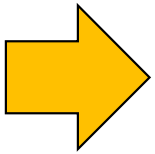
```
person = {
        "name":"Dave",
        "midterm":78,
        "labs": [
                95,
                100
        ]
}
```

A JSON File

- A JSON file is almost the same as a JavaScript object, without the name of the object

```
person = {
      "name":"Dave",
      "midterm":78,
      "lab02":95,
      "lab03":100
}
```

  - A JavaScript object



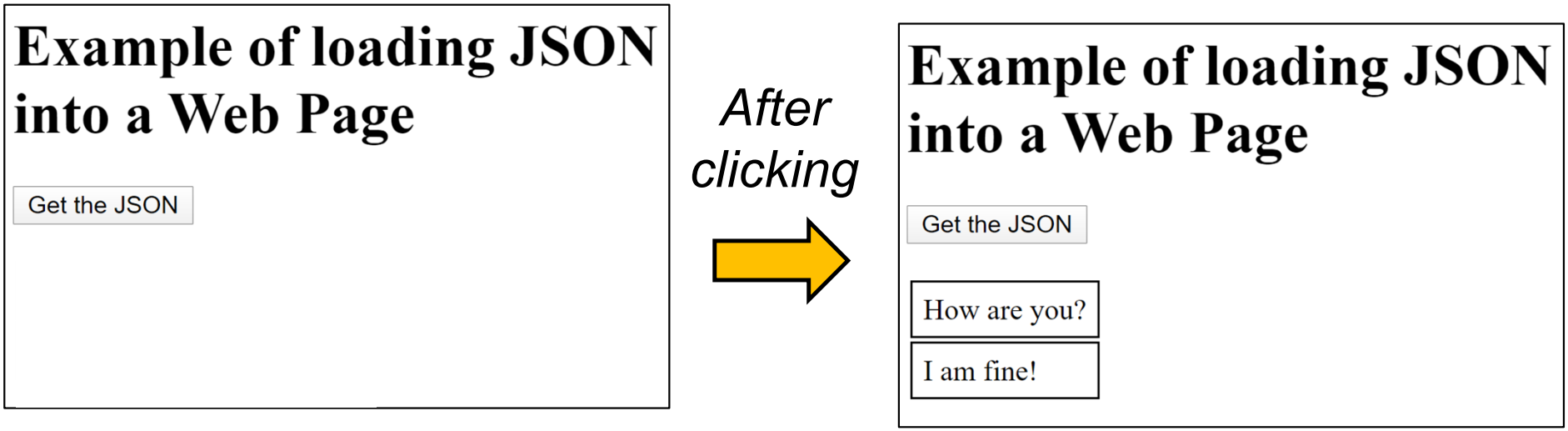
- One difference is that

! ! ! JSON uses double speech marks only, no single speech marks

```
{     "name":"Dave",
      "midterm":78,
      "lab02":95,
      "lab03":100
}
```

  - A JSON file

Loading JSON into a Web Page

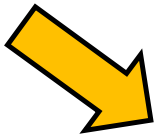


- Commonly, JSON is loaded into the web page memory
- Then the JSON (which you can't see yet) is converted into something which you can see i.e. some HTML
- On the following slides we show a jQuery example

The Basic Idea

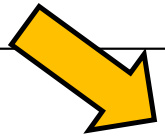
```
{
    "1":"How are you?",
    "2":"I am fine!"
}
```

- JSON is loaded from the server



- JSON is converted into HTML

```
<tr><td>How are you?</td></tr>
<tr><td>I am fine!</td></tr>
```

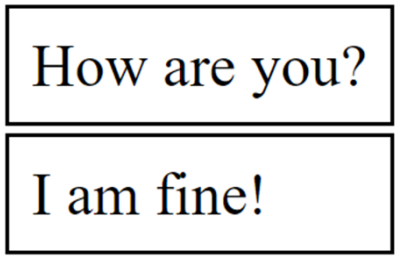


- Browser displays the HTML

How are you?

I am fine!

```
<!DOCTYPE html>
<html> <head>
<script
  src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js">
</script>

<style>
td {
  border:1px solid black;
  padding:6px;
}
</style>
```



- When the JSON content is displayed in the web page, it is displayed in a box (`td`) in a table – this style instruction means it has nice spacing and a border line around each message

Example of Loading JSON into a Web Page 1/3

```
<script>
function processJSON() {
    $.ajax({
      url: '03_chatroom_example.json',
      type: "GET",
      dataType: "json",
      success: function (result) {
            var allContent="";
            for (thisItem in result)
                  allContent+="<tr><td>" + thisItem + "</td><td>"
                        + result[thisItem] + "</td></tr>";
            $("#show_json_content").append(allContent);
      }
      });
} </script>   </head>
```

iterate through keys

- When the request for the JSON file is successful, find the thing called *show_json_content* in the web page and convert each message into a table cell (`td`) which goes inside a table row (`tr`)

Example of Loading JSON into a Web Page 2/3

```
<body>
<h1>Example of loading JSON into a Web Page</h1>

<button type="button" onclick="processJSON()">Get the JSON</button>
<br><br>

<table id="show_JSON_content"></table>

</body>
</html>
```

- Click here to trigger the action
- The JSON content (the messages) are put in here

Example of Loading JSON into a Web Page 3/3

XML and JSON

- XML and JSON have basically the same ‘power’
- JSON has less ‘words’ so less typing is needed, and it usually has a smaller file size
- ! ! ! However, with JSON it’s easy to accidentally add a comma at the end of the last item, which is not OK
- XML has more ‘words’, but that can make the data more readable
- Simple summary - both are good!