VISVESVARAYA TECHNOLOGICAL UNIVERSITY

WEB TECHNOLOGY LAB

BCSL504

EXPERIMENT – 07

JSON, CSV AND HASHING USING JAVASCRIPT

5th Semester

JSON → JavaScript Object

JSON Date • JSON → CSV • CSV → JSON

String → Hash using crypto.createHash()

Prepared by Search Creators

Learn • Practice • Execute

1. Experiment 07

Aim

Develop JavaScript programs with HTML and CSS for the following:

a. Convert JSON text into a JavaScript Object.

b. Convert JSON results into a Date.

c. Convert JSON to CSV and CSV to JSON.

d. Create a hash from a string using the `crypto.createHash()` method.

2. Requirements

- **Search Creators Compiler / Code Studio**

  https://searchcreators.org/compiler/

- Modern browser such as Chrome, Edge or Firefox
- HTML, CSS and JavaScript
- Node.js environment for Part (d), because `crypto.createHash()` belongs to the Node.js Crypto API
- Alternatively, VS Code or another suitable code editor

3. Search Creators Compiler

Practice with Search Creators

Use the **Search Creators Compiler / Code Studio** while practising the Web Technology Lab programs.

Open Search Creators Compiler

`https://searchcreators.org/compiler/`

Parts (a), (b) and (c) below are browser-based HTML/JavaScript programs.
Part (d) uses the Node.js `crypto.createHash()` API and therefore requires an environment that supports Node.js.

4. Introduction to JSON

JSON stands for:

JavaScript Object Notation

JSON is a text format used to store and exchange structured data.

Example:

```
{
    "name": "Ravi",
    "usn": "1XX22CS001",
    "semester": 5
}
```

JSON commonly represents data using:

- Objects
- Arrays
- Strings
- Numbers
- Boolean values
- Null values

5. Important JavaScript Methods

Method / Object	Purpose
`JSON.parse()`	Converts JSON text into a JavaScript value/object.
`JSON.stringify()`	Converts a JavaScript value into JSON text.
`new Date()`	Creates a JavaScript Date object.
`Object.keys()`	Returns the property names of an object.
`split()`	Splits a string into an array.
`join()`	Combines array elements into a string.
`crypto.createHash()`	Creates a Node.js Hash object for a specified algorithm.

6. Part (a): JSON Text to JavaScript Object

6.1 Aim

Convert JSON text into a JavaScript object and display its values.

6.2 Program

```
<!DOCTYPE html>
<html lang="en">

<head>

<meta charset="UTF-8">

<title>JSON to JavaScript Object</title>

<style>

body {
    font-family: Arial, sans-serif;
    background-color: #f2f2f2;
    text-align: center;
}

.box {
    width: 500px;
    margin: 50px auto;
    padding: 20px;
    background-color: white;
    border: 2px solid darkgreen;
}

h2 {
    color: darkgreen;
}

#output {
    background-color: lightyellow;
    padding: 15px;
    font-size: 18px;
}

</style>

</head>

<body>

<div class="box">

<h2>JSON Text to JavaScript Object</h2>

<button onclick="convertJSON()">
```

```
    Convert JSON
</button>

<p id="output">
    Result will appear here
</p>

</div>

<script>

function convertJSON() {

    let jsonText =
        '{"name":"Ravi",' +
        '"usn":"1XX22CS001",' +
        '"semester":5}';

    let student = JSON.parse(jsonText);

    document.getElementById("output").innerHTML =
        "Name: " + student.name +
        "<br>USN: " + student.usn +
        "<br>Semester: " + student.semester;
}

</script>

</body>
</html>
```

6.3 Explanation

The JSON text is stored as a string:

```
let jsonText =
    '{"name":"Ravi","usn":"1XX22CS001","semester":5}';
```

It is converted using:

```
let student = JSON.parse(jsonText);
```

Therefore:

$$\text{JSON Text} \xrightarrow{\texttt{JSON.parse()}} \text{JavaScript Object}$$

After conversion, properties can be accessed as:

```
student.name
student.usn
student.semester
```

7. Part (b): JSON Result to Date

7.1 Aim

Read a date value from JSON data and convert it into a JavaScript Date object.

7.2 Program

```
<!DOCTYPE html>
<html lang="en">

<head>

<meta charset="UTF-8">

<title>JSON Date Conversion</title>

<style>

body {
    font-family: Arial, sans-serif;
    text-align: center;
    background-color: #eaf4ef;
}

.box {
    width: 550px;
    margin: 50px auto;
    background-color: white;
    padding: 20px;
    border: 2px solid darkgreen;
}

h2 {
    color: darkgreen;
}

#output {
    font-size: 18px;
    color: darkblue;
```

```
}

</style>

</head>

<body>

<div class="box">

<h2>JSON Result to Date</h2>

<button onclick="convertDate()">
    Convert Date
</button>

<p id="output">
    Result will appear here
</p>

</div>

<script>

function convertDate() {

    let jsonText =
        '{"event":"Web Technology Lab",' +
        '"date":"2026-09-20"}';

    let data = JSON.parse(jsonText);

    let convertedDate =
        new Date(data.date);

    document.getElementById("output").innerHTML =
        "Event: " + data.event +
        "<br>JSON Date: " + data.date +
        "<br>Converted Date: " +
        convertedDate.toDateString();
}

</script>

</body>
</html>
```

7.3 Explanation

First, the JSON text is converted into an object:

```
let data = JSON.parse(jsonText);
```

The date property is then passed to:

```
let convertedDate =
    new Date(data.date);
```

The process is:

JSON Date String → JavaScript Date Object

The date is displayed using:

```
convertedDate.toDateString()
```

8. Part (c): JSON to CSV and CSV to JSON

8.1 Introduction to CSV

CSV stands for:

Comma-Separated Values

Example:

```
name,usn,semester
Ravi,1XX22CS001,5
Asha,1XX22CS002,5
```

8.2 Program

```
<!DOCTYPE html>
<html lang="en">

<head>

<meta charset="UTF-8">

<title>JSON and CSV Conversion</title>

<style>

```

```
body {
    font-family: Arial, sans-serif;
    background-color: #f5f5f5;
    text-align: center;
}

.container {
    width: 700px;
    margin: 30px auto;
    background-color: white;
    padding: 20px;
    border: 2px solid darkgreen;
}

h2 {
    color: darkgreen;
}

button {
    padding: 10px 15px;
    margin: 10px;
    font-size: 16px;
}

pre {
    text-align: left;
    background-color: lightyellow;
    padding: 15px;
    white-space: pre-wrap;
}

</style>

</head>

<body>

<div class="container">

<h2>JSON to CSV and CSV to JSON</h2>

<button onclick="jsonToCSV()">
    JSON to CSV
</button>

<button onclick="csvToJSON()">
    CSV to JSON
</button>
```

```
60
61 <pre id="output">
62 Result will appear here
63 </pre>
64
65 </div>
66
67 <script>
68
69 const jsonData = [
70     {
71         name: "Ravi",
72         usn: "1XX22CS001",
73         semester: 5
74     },
75     {
76         name: "Asha",
77         usn: "1XX22CS002",
78         semester: 5
79     }
80 ];
81
82 function jsonToCSV() {
83
84     let headers =
85         Object.keys(jsonData[0]);
86
87     let csv =
88         headers.join(",") + "\n";
89
90     jsonData.forEach(function(row) {
91
92         let values =
93             headers.map(function(header) {
94                 return row[header];
95             });
96
97         csv += values.join(",") + "\n";
98     });
99
100    document.getElementById("output").textContent =
101        csv;
102 }
103
104 function csvToJSON() {
105
106    let csv =
107        "name,usn,semester\n" +
```

```
        "Ravi,1XX22CS001,5\n" +
        "Asha,1XX22CS002,5";

    let lines =
        csv.trim().split("\n");

    let headers =
        lines[0].split(",");

    let result = [];

    for (let i = 1; i < lines.length; i++) {

        let values =
            lines[i].split(",");

        let obj = {};

        headers.forEach(function(header, index) {

            obj[header] =
                values[index];
        });

        result.push(obj);
    }

    document.getElementById("output").textContent =
        JSON.stringify(result, null, 2);
}

</script>

</body>
</html>
```

8.3 JSON to CSV Explanation

The property names are obtained using:

```
Object.keys(jsonData[0])
```

For example:

name, usn, semester

The headings are joined using commas:

```
headers.join(",")
```

Each object's values are then added as a new CSV row.

The conversion is:

JSON Array → CSV Text

8.4 CSV to JSON Explanation

The CSV text is first separated into lines:

```
let lines =
    csv.trim().split("\n");
```

The first line contains the property names:

```
let headers =
    lines[0].split(",");
```

The remaining rows are converted into JavaScript objects.

Finally:

```
JSON.stringify(result, null, 2)
```

is used to display the result in JSON format.

Important

This is a simple laboratory example. It assumes that the CSV values themselves do not contain embedded commas or quoted multiline values.

9. Part (d): Hash from a String

9.1 Aim

Create a hash from a string using the `crypto.createHash()` method.

Important

`crypto.createHash()` is part of the **Node.js Crypto API**.
Therefore, this part should be executed in a Node.js environment. It should not be treated as ordinary browser JavaScript.

9.2 Node.js Program

Create a file such as:

hash.js

Enter the following program:

```
const crypto = require("crypto");

const text =
    "Search Creators";

const hash =
    crypto
        .createHash("sha256")
        .update(text)
        .digest("hex");

console.log("Original String:");
console.log(text);

console.log("\nSHA-256 Hash:");
console.log(hash);
```

9.3 Explanation

The built-in Node.js Crypto module is loaded using:

```
const crypto = require("crypto");
```

A SHA-256 hash object is created using:

```
crypto.createHash("sha256")
```

The input string is supplied using:

```
.update(text)
```

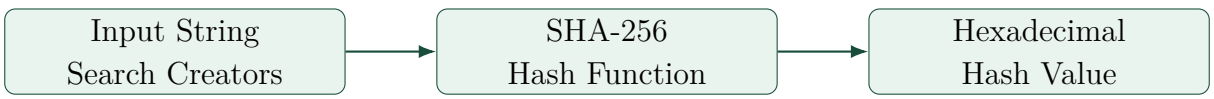
The final hash is returned in hexadecimal format using:

```
.digest("hex")
```

Therefore:

Input String → SHA-256 → Hexadecimal Hash

10. Hashing Process



11. What is Hashing?

Hashing transforms input data into a fixed-size output called a hash or digest.

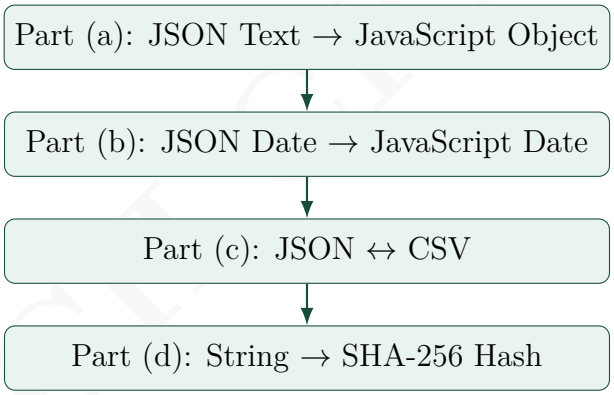
For this experiment:

$$\text{String} \xrightarrow{\text{SHA-256}} \text{Hash}$$

Important characteristics include:

- The same input produces the same hash for the same algorithm.
- A small input change normally produces a very different hash.
- SHA-256 produces a 256-bit digest.
- A hexadecimal representation of a SHA-256 digest contains 64 hexadecimal characters.

12. Overall Experiment Flow



13. How to Execute the Programs

Practice with Search Creators

Search Creators Compiler / Code Studio

Search Creators Compiler

`https://searchcreators.org/compiler/`

Use the coding environment while practising the HTML/CSS/JavaScript portions of the experiment.
For Part (d), use a Node.js execution environment when running the syllabus-required `crypto.createHash()` program.

14. Procedure

1. Open the Search Creators Compiler / Code Studio:

   https://searchcreators.org/compiler/

2. For Part (a), create the HTML program for JSON text to JavaScript object conversion.
3. Define a valid JSON string.
4. Use `JSON.parse()` to convert the JSON string into a JavaScript object.
5. Display the object property values.
6. For Part (b), create JSON data containing a date value.
7. Parse the JSON data and pass its date value to `new Date()`.
8. Display the converted date.
9. For Part (c), create sample JSON records.
10. Obtain the property names using `Object.keys()`.
11. Join headings and values using commas to produce CSV output.
12. For CSV to JSON, split the CSV into lines and values and construct JavaScript objects.
13. Display the JSON result using `JSON.stringify()`.
14. For Part (d), create `hash.js` in a Node.js environment.
15. Import the Crypto module.
16. Create a SHA-256 hash using `crypto.createHash("sha256")`.
17. Pass the input using `update()` and obtain hexadecimal output using `digest("hex")`.
18. Execute all four parts and verify the outputs.
19. VS Code or another suitable editor/environment may also be used where required.

15. Expected Outputs

15.1 Part (a)

For the sample JSON data:

```
Name: Ravi
USN: 1XX22CS001
Semester: 5
```

15.2 Part (b)

The program displays:

```
Event: Web Technology Lab
JSON Date: 2026-09-20
Converted Date: ...
```

The exact human-readable date representation can depend on how the JavaScript runtime interprets and displays the Date object.

15.3 Part (c)

JSON to CSV:

```
name,usn,semester
Ravi,1XX22CS001,5
Asha,1XX22CS002,5
```

CSV to JSON produces an array of objects containing the same fields and values.

15.4 Part (d)

The Node.js program displays:

```
Original String:
Search Creators

SHA-256 Hash:
<64-character hexadecimal hash>
```

16. Important Viva Questions

Important Viva Questions

1. **What is JSON?**

   JSON stands for JavaScript Object Notation. It is a text format used to represent structured data.

2. **What does JSON.parse() do?**

   It parses valid JSON text and returns the corresponding JavaScript value.

3. **What does JSON.stringify() do?**

It converts a JavaScript value into JSON text.

4. **How is a JSON date string converted into a JavaScript Date?**

   By passing the date value to the `Date` constructor.

5. **What is CSV?**

   CSV stands for Comma-Separated Values.

6. **What does Object.keys() return?**

   It returns an array containing an object's own enumerable property names.

7. **What is split() used for?**

   It splits a string into an array based on a separator.

8. **What is join() used for?**

   It joins array elements into a string using a specified separator.

9. **What is hashing?**

   Hashing transforms input data into a fixed-size digest using a hash function.

10. **Which hash algorithm is used in this program?**

    SHA-256.

11. **What is crypto.createHash()?**

    It is a Node.js Crypto API method used to create a Hash object for a specified algorithm.

12. **What does update() do in the hashing program?**

    It supplies data to the hash object.

13. **What does digest("hex") do?**

    It calculates the digest and returns its hexadecimal representation.

14. **Can crypto.createHash() be used as ordinary browser JavaScript?**

    No. The program shown here uses the Node.js Crypto API and requires a Node.js environment.

15. **How many hexadecimal characters are used to represent a SHA-256 digest?**

    64 hexadecimal characters.

16. **Which platform can be used while practising Search Creators lab programs?**

Search Creators Compiler / Code Studio

17. Quick Revision

Concept	Remember
JSON	JavaScript Object Notation
JSON to Object	`JSON.parse()`
Object to JSON Text	`JSON.stringify()`
Date Object	`new Date()`
CSV	Comma-Separated Values
Object Properties	`Object.keys()`
Split String	`split()`
Join Array	`join()`
Hash Module	Node.js `crypto`
Create Hash	`crypto.createHash()`
Hash Algorithm	SHA-256
Add Hash Input	`update()`
Hexadecimal Digest	`digest("hex")`
Practice Platform	Search Creators Compiler
Website	`searchcreators.org/compiler/`

18. Result

Result

JavaScript programs were developed successfully to:

- Convert JSON text into a JavaScript object.
- Convert a JSON date value into a JavaScript Date object.
- Convert JSON data into CSV format.
- Convert CSV data into JSON format.
- Generate a SHA-256 hash from a string using the Node.js `crypto.createHash()` method.

For practice, visit:

Search Creators Compiler / Code Studio

`https://searchcreators.org/compiler/`

END OF EXPERIMENT – 07

BCSL504 – Web Technology Lab

Prepared by Search Creators

Practice Online: searchcreators.org/compiler/