**Practical – 2**

AIM: Create a Jason object and display the same using node.js in the terminal. Create a Node.js script to read and display the content of the external Jason file. Create and display multi-dimensional Jason Arrays, also demonstrate Accessing individual elements of the multi-dimensional Jason Arrays.

Create a web-based application example using JavaScript and JSON to manipulate JSON data. In this example, create a simple to-do list application where tasks are stored in a JSON array. Users can add tasks, mark them as completed, and remove them from the list.

1)  
const ob **=** {

    name **:**"nigam"**,**

    age**:** 21**,**

    roll**:** 7

}**;**

console**.**log(JSON**.**stringify(ob**,** null**,** 1))**;**

2)  
const fs **=** require('fs')**;**

fs**.**readFile('multi\_arr.json'**,** (**err,** **data**) **=>** {

    console**.**log(JSON**.**parse(data))**;**

})

*// have to do .parse because data is read in form of buffers so need to convert it back to JSON object*

*3)*

const multi\_arr **=** [

    [

        {

            name**:** "nigam"**,**

            college**:** "nirma"

        }**,**

        {

            name**:** "xyz"**,**

            college**:** "LD"

        }

    ]**,**

    [

        {

            name**:** "ABC"**,**

            college**:** "DAICT"

        }

    ]

]

console**.**log(multi\_arr[0][0]**.***college*)**;**

4)  
<!DOCTYPE *html*>

<html *lang***=**"en">

<head>

    <meta *charset***=**"UTF-8">

    <meta *name***=**"viewport" *content***=**"width=device-width, initial-scale=1.0">

    <title>Task Manager</title>

    <style>

        body {

            display**:** flex**;**

            justify-content**:** center**;**

            align-items**:** center**;**

            height**:** 100vh**;**

            margin**:** 0**;**

            background-color**:** **#f4f4f9;**

            font-family**:** Arial**,** sans-serif**;**

            text-align**:** center**;**

        }

        button {

            margin-top**:** 10px**;**

            background-color**:** beige**;**

            border-radius**:** 5px**;**

        }

        button*:hover* {

            box-shadow**:** 2px 2px 10px black**;**

        }

*.container* {

            height**:** auto**;**

            width**:** 400px**;**

            background-color**:** rgb(187**,** 164**,** 164)**;**

            border**:** 1px solid **#0a0606;**

            border-radius**:** 10px**;**

            padding**:** 20px**;**

            box-shadow**:** 0 0 10px 10px **#da42421a;**

        }

        form{

            padding**:** 10px**;**

        }

        form input {

            display**:** block**;**

            width**:** 98%**;**

            margin-bottom**:** 10px**;**

            padding**:** 8px**;**

            border**:** 1px solid **#000000;**

            border-radius**:** 5px**;**

        }

    </style>

</head>

<body>

    <div *class***=**"container">

        <h2>Daily Task Sheet</h2>

        <div *class***=**"form-section">

            <form ***id*=**"task-form">

                <input *type***=**"text" ***id*=**"task" *placeholder***=**"Task Name" *required*>

                <input *type***=**"text" ***id*=**"task-desc" *placeholder***=**"Task Description" *required*>

                <input *type***=**"date" ***id*=**"date" *required*>

                <input *type***=**"time" ***id*=**"time" *required*>

            </form>

        </div>

        <div *class***=**"textbox">

            <button *onclick***=**"addTask()">Add</button>

            <button *onclick***=**"removeTask()">Remove Last</button>

            <button *onclick***=**"showTasks()">Show</button>

        </div>

        <div *class***=**"task-list" ***id*=**"task-list">

        </div>

    </div>

    <script>

        let Userlist **=** []**;**

        function addTask() {

            let name **=** document**.**getElementById('task')**.***value***;**

            let desc **=** document**.**getElementById('task-desc')**.***value***;**

            let date **=** document**.**getElementById('date')**.***value***;**

            let time **=** document**.**getElementById('time')**.***value***;**

            const obj **=** {

                name**:** name**,**

                desc**:** desc**,**

                date**:** date**,**

                time**:** time

            }

**if** (**!**name **||** **!**desc **||** **!**date **||** **!**time) {

                alert("First insert all")**;**

**return;**

            }

            Userlist**.**push(obj)**;**

            updateList()**;**

        }

        function updateList() {

            let objlist **=** document**.**getElementById('task-list')**;**

            objlist**.***innerHTML* **=** ""**;**

            Userlist**.**forEach((**value,** **index**) **=>** {

                const taskItem **=** document**.**createElement('div')**;**

                taskItem**.***className* **=** 'taskitem'**;**

                taskItem**.***innerHTML* **=** `

    <span>${value**.***name*} - ${value**.***desc*} (${value**.***date*} at ${value**.***time*})</span>

    <button onclick="deleteTask(${index})">Delete</button>

`**;**

                objlist**.**appendChild(taskItem)**;**

            })**;**

            document**.**getElementById('task-form')**.**reset()**;**

        }

        function removeTask() {

            Userlist**.**pop()**;**

            updateList()**;**

        }

        function showTasks() {

**if** (Userlist**.**length **==** 0) {

                alert("No data are there....")**;**

            }

**else** {

                alert('Tasks:\n' **+** Userlist**.**map(**task** **=>** `${task**.***name*} - ${task**.***desc*}`)**.**join('\n'))**;**

            }

        }

        function deleteTask(**index**) {

            Userlist**.**splice(index**,** 1)**;**

            updateList()**;**

        }

    </script>

</body>

</html>

output:

