

PSG college of Technology
Department of Computer Applications
23MX18 Web Application Development
Worksheet -13
File Handling

Roll no: 25mx102

Date: 03/11/2025

Index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<title>File Handling Output Viewer</title>

<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"
rel="stylesheet">

</head>
<body class="bg-light">

<div class="container py-4">
  <h2 class="text-center mb-4">Node.js File Handling Output Viewer</h2>

  <div class="row g-3">

    <div class="col-md-4">
      <div class="card shadow">
        <div class="card-body">
          <h4>Employee Login</h4>
          <p>View EmpLogin.txt content</p>
          <a href="./data/EmpLogin.txt" class="btn btn-primary"
target="_blank">Open File</a>
        </div>
      </div>
    </div>

    <div class="col-md-4">
      <div class="card shadow">
        <div class="card-body">
          <h4>Product Stock</h4>
          <p>View Prod.csv file content</p>
        </div>
      </div>
    </div>
  </div>
</div>
```

```

        <a href="./data/Prod.csv" class="btn btn-success"
target="_blank">Open File</a>
    </div>
</div>
</div>

<div class="col-md-4">
    <div class="card shadow">
        <div class="card-body">
            <h4>Food Orders</h4>
            <p>View Orders.json details</p>
            <a href="./data/Orders.json" class="btn btn-warning"
target="_blank">Open File</a>
        </div>
    </div>
</div>

</div>
</div>

</body>
</html>

```

Emp.js

```

const fs = require("fs");
const path = "./data/EmpLogin.txt";

/**
 * Add a daily employee login detail
 * @param {*} empID
 * @param {*} date (YYYY-MM-DD)
 * @param {*} timeIn (HH:MM)
 * @param {*} timeOut (HH:MM)
 */
function addLogin(empID, date, timeIn, timeOut) {
    const record = {
        EmpID: empID,
        Date: date,
        TimeIn: timeIn,
        TimeOut: timeOut
    };

    fs.appendFileSync(path, JSON.stringify(record) + "\n", "utf-8");
    console.log("Employee login added.");
}

```

```

/**
 * Counts employees logged after 10 AM on given date
 * @param {*} dateString
 */
function countLateEmployees(dateString) {
  if (!fs.existsSync(path)) {
    console.log("No login file found.");
    return;
  }

  const data = fs.readFileSync(path, "utf-8").trim().split("\n");
  let count = 0;

  data.forEach(line => {
    const obj = JSON.parse(line);
    if (obj.Date === dateString) {
      let hour = parseInt(obj.TimeIn.split(":")[0]);
      if (hour >= 10) count++;
    }
  });

  console.log(`Employees logged after 10AM on ${dateString}: ${count}`);
}

// TEST - uncomment to try
addLogin("EMP01", "2025-11-03", "09:50", "18:00");
addLogin("EMP02", "2025-11-03", "10:15", "17:30");
countLateEmployees("2025-11-03");

module.exports = { addLogin, countLateEmployees };

```

Food.js

```

const fs = require("fs");
const readlineSync = require("readline-sync");
const path = "./data/Orders.json";

/**
 * Initialize file if not present
 */
function initOrders() {
  if (!fs.existsSync(path)) {
    fs.writeFileSync(path, "[]");
  }
}

```

```

/**
 * Place order
 */
function addOrder(id, itemName, qty, price) {
    initOrders();
    let arr = JSON.parse(fs.readFileSync(path, "utf-8"));
    arr.push({ id, itemName, qty, price });
    fs.writeFileSync(path, JSON.stringify(arr, null, 2));
    console.log("Order added.");
}

/**
 * Print total sales
 */
function printTotalSales() {
    initOrders();
    let arr = JSON.parse(fs.readFileSync(path, "utf-8"));
    let total = 0;
    arr.forEach(o => total += o.qty * o.price);
    console.log("Total Sales = ₹" + total);
}

/**
 * Menu driven
 */
function menu() {
    while (true) {
        console.log("\n----- Food Delivery Menu -----");
        console.log("1. Place Order");
        console.log("2. Print Total Sales");
        console.log("3. Exit");

        let ch = readlineSync.questionInt("Enter choice: ");

        if (ch === 1) {
            let id = readlineSync.question("Order ID: ");
            let name = readlineSync.question("Item Name: ");
            let qty = readlineSync.questionInt("Qty: ");
            let price = readlineSync.questionInt("Price: ");
            addOrder(id, name, qty, price);
        } else if (ch === 2) {
            printTotalSales();
        } else {
            console.log("Bye!");
            break;
        }
    }
}

```

```
menu();  
module.exports = { addOrder, printTotalSales, menu };
```

Product.js

```
const fs = require("fs");  
const path = "../data/Prod.csv";  
  
/**  
 * Create new product CSV file (override old)  
 */  
function createCSV() {  
    fs.writeFileSync(path, "ProdID,Description,Price,Stock\n");  
    console.log("Prod.csv created.");  
}  
  
/**  
 * Add new product  
 */  
function addProduct(id, desc, price, stock) {  
    const data = `${id},${desc},${price},${stock}\n`;  
    fs.appendFileSync(path, data);  
    console.log("Product added.");  
}  
  
/**  
 * Search products below stock = 10  
 */  
function getLowStockItems() {  
    const lines = fs.readFileSync(path, "utf-8").trim().split("\n");  
    const low = [];  
  
    for (let i = 1; i < lines.length; i++) {  
        const [id, desc, price, stock] = lines[i].split(",");  
        if (parseInt(stock) < 10)  
            low.push({ id, desc, price, stock });  
    }  
    return low;  
}  
  
/**  
 * Update stock of a given product  
 */  
function updateStock(id, newStock) {  
    let lines = fs.readFileSync(path, "utf-8").trim().split("\n");
```

```

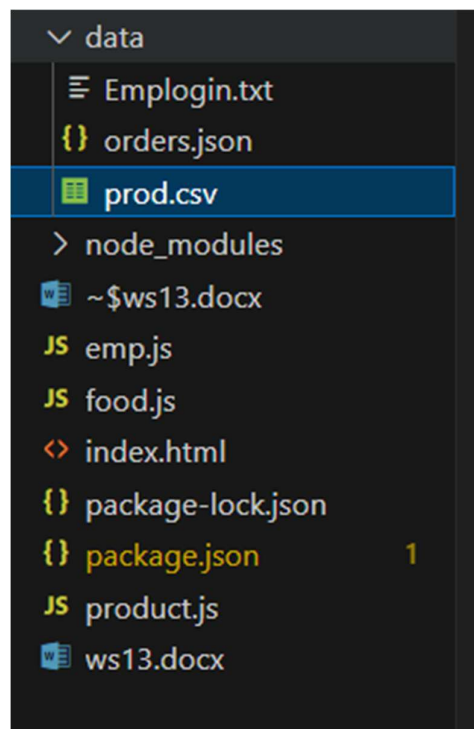
    for (let i = 1; i < lines.length; i++) {
        let cols = lines[i].split(",");
        if (cols[0] === id) {
            cols[3] = newStock;
            lines[i] = cols.join(",");
        }
    }
    fs.writeFileSync(path, lines.join("\n"));
    console.log("Stock updated.");
}

// TEST - uncomment to try
createCSV();
addProduct(101, "Soap", 30, 8);
addProduct(102, "Oil", 150, 12);
addProduct(103, "Paste", 50, 5);
console.log(getLowStockItems());
updateStock(103, 15);

module.exports = { createCSV, addProduct, getLowStockItems, updateStock };

```

o/p:



Node.js File Handling Output Viewer

Employee Login

View EmpLogin.txt content

Open File

Product Stock

View Prod.csv file content

Open File

Food Orders

View Orders.json details

Open File

```
{"EmpID": "EMP01", "Date": "2025-11-03", "TimeIn": "09:50", "TimeOut": "18:00"}  
{"EmpID": "EMP02", "Date": "2025-11-03", "TimeIn": "10:15", "TimeOut": "17:30"}
```

Pretty-print ☐

```
[  
  {  
    "id": "01",  
    "itemName": "pizza",  
    "qty": 5,  
    "price": 200  
  }  
]
```

```
ProdID,Description,Price,Stock  
101,Soap,30,8  
102,Oil,150,12  
103,Paste,50,5
```