**Title: XML to JSON Converter**

**Tool Details:**

**Technology Stack:**

* **Frontend:** Web Components (Lit or Vanilla JS)
* **Backend:** Express.js (Node.js)
* **Processing:** Convert XML data into JSON format
* **Data Handling:** API requests & response parsing

**Goal:**  
By completing this assignment, candidates will:

* Learn how to build a **RESTful API** using **Express.js**.
* Implement **XML to JSON conversion** using Node.js.
* Work with **Web Components** to create an interactive frontend.
* Understand **client-server communication** via API calls.
* Gain experience in handling **file processing and structured data transformation**.

**Assignment Description:**

Develop an **XML to JSON Converter** where users can input or upload **XML data** via a **WebComponent-based form**. The backend, built with **Express.js**, will process the data and return a **JSON response** that the user can view or download.

**Tasks & Steps:**

**1. Backend API Development (Express.js):**

* Set up an **Express.js** server to accept **XML input**.
* Implement **XML to JSON conversion** logic.
* Provide an API route that **accepts XML data, converts it to JSON, and returns the structured data**.

**2. Frontend (WebComponent-based UI):**

* Create a **form** using Web Components that allows users to **input or upload XML data**.
* Send the XML data to the backend using the **fetch API**.
* Display the **converted JSON output** after processing.
* Provide a **download button** for the JSON file.

**3. Integration & Testing:**

* Ensure the frontend **properly communicates** with the backend.
* Handle errors gracefully (e.g., **invalid XML format**).
* Test the **XML to JSON conversion process** to ensure accuracy.

**Mathematical Calculation/Steps (if applicable):**

* **Parsing Logic:** Convert XML nodes into **JSON key-value pairs**.
* **Data Structuring:** Maintain XML hierarchy in JSON format.

**Third-Party Packages (if required):**

* express (for backend server)
* xml2js (for XML to JSON conversion)
* lit (for WebComponent development)

**Acceptance Criteria:**

* The **Express.js backend** should successfully **convert XML to JSON**.
* The **WebComponent-based frontend** should have a **form** for XML input or file upload.
* The converted JSON output should be **properly formatted and displayed**.
* Proper **error handling** should be in place for **invalid XML input**.

**Submission Guidelines:**

1. **Fork** the provided GitHub repository.
2. **Create a folder** named xml-to-json-<your-name>.
3. **Implement the backend and frontend** in the respective subfolders.
4. **Push the code** to your forked repository.
5. **Submit a pull request** with a brief description of your implementation.

**Ensure that the backend correctly processes XML data and integrates seamlessly with the WebComponent-based frontend.**