

Name	Arnab Sen
Roll	510519006
Task	Subtask-2

Parsing XML to a Data Structure for further processing

The input can be of two types:

1. XML Doc
2. XML in string format

XML Doc

If we have a XML document object then we can iterate through the nodes of the document get the node value and if it has a children, iterate again. Basically we will recursively call the function for its node. The code for this is

```
function xmlToJson(xml) {
    var obj = {};
    if (xml.nodeType == 3) {
        obj = xml.nodeValue;
    }
    if (xml.hasChildNodes()) {
        for(var i = 0; i < xml.childNodes.length; i++) {
            var item = xml.childNodes.item(i);
            var nodeName = item.nodeName;
            if (typeof obj[nodeName] == "undefined") {
                obj[nodeName] = xmlToJson(item);
            } else {
                if (typeof obj[nodeName].push == "undefined") {
                    var old = obj[nodeName];
                    obj[nodeName] = [];
                    obj[nodeName].push(old["#text"]);
                }
                obj[nodeName].push(xmlToJson(item)["#text"]);
            }
        }
    }
}
```

```
    return obj;  
};
```

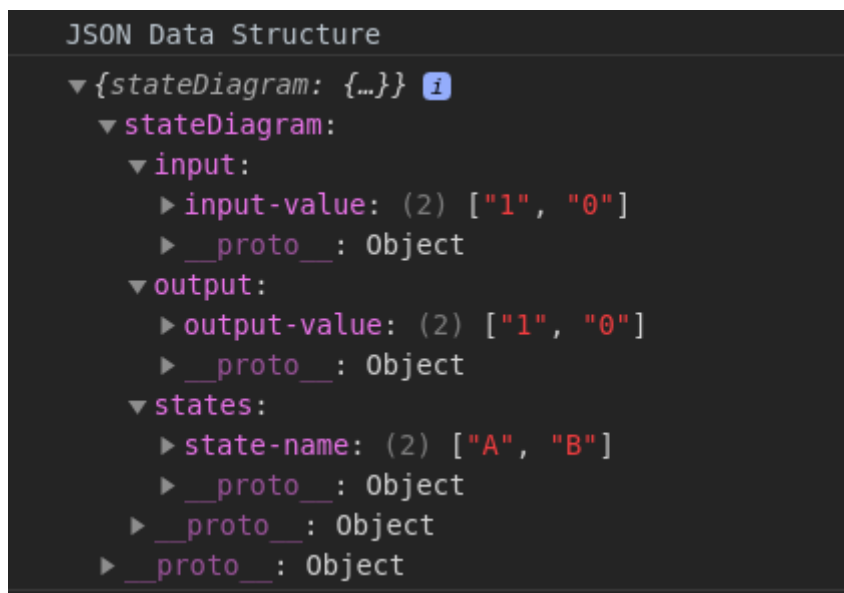
Let's consider a simple XML as:

```
<stateDiagram>  
  <input>  
    <input-value>1</input-value>  
    <input-value>0</input-value>  
  </input>  
  <output>  
    <output-value>1</output-value>  
    <output-value>0</output-value>  
  </output>  
  <states>  
    <state-name>A</state-name>  
    <state-name>B</state-name>  
  </states>  
</stateDiagram>
```

In browser it get's rendered as this:

```
▼<stateDiagram>  
  ▼<input>  
    <input-value>1</input-value>  
    <input-value>0</input-value>  
  </input>  
  ▼<output>  
    <output-value>1</output-value>  
    <output-value>0</output-value>  
  </output>  
  ▼<states>  
    <state-name>A</state-name>  
    <state-name>B</state-name>  
  </states>  
</stateDiagram>
```

When the XML doc is passed through the function `xmlToJson` it displays a JSON object as:



String

If the same XML is passed as a string we first have to convert that to XML doc then use the function `xmlToJson` to get the Data Structure. Here we are using the `DOMParser` method of window to parse the XML string. Also the `replace` function removes unwanted line breaks and whitespaces.

For that :

```
function parseXml(xml) {
    xml = xml.replace("/\r?\n|\r/g", "");
    var dom = null;
    if (window.DOMParser) {
        try {
            dom = (new DOMParser()).parseFromString(xml, "text/xml");
        }
        catch (e) { dom = null; }
    }
    else
        alert("cannot parse xml string!");
    return dom;
}
```

So if we rather had the same XML as

```
sampleXML = `<stateDiagram><input><input-value>1...`
```

Then we can run `parseXML` and it will return:

```
XML DOC
▼ #document
  <stateDiagram>
    ▼ <input>
      <input-value>1</input-value>
      <input-value>0</input-value>
    </input>
    ▼ <output>
      <output-value>1</output-value>
      <output-value>0</output-value>
    </output>
    ▼ <states>
      <state-name>A</state-name>
      <state-name>B</state-name>
    </states>
  </stateDiagram>
```

We can further use this XML doc.