1. XML/DB

XML when I need more flexible structure than a database table can offer.
When you have data that you do not know exactly the structure beforehand, it could be profitable to store it as XML. You can store such XML in a relational database field, as many e-learning systems do.
In fact, e-learning course structure is a good example because individual courses could be quite different from each other.
So, XML is somewhere between a document and a relational-table. Imagine a database table where each record would have a very different structure, that would be good use for XML.
Also, XML is very good for data interchange, meanwhile files like .DBF, .XLS etc. are not so good.

References:

https://www.ibm.com/docs/en/db2/11.5?topic=overview-comparison-xml-relational-models

https://dba.stackexchange.com/questions/102233/xml-vs-relational-databases

2. XML/JSON

Both JSON and XML can be used to receive data from a web server.

Using XML
- Fetch an XML document
- Use the XML DOM to loop through the document
- Extract values and store in variables

Using JSON
- Fetch a JSON string
- JSON.Parse the JSON string

Why JSON is Better Than XML
XML is much more difficult to parse than JSON.
JSON is parsed into a ready-to-use JavaScript object.

For AJAX applications, JSON is faster and easier than XML

References:
https://www.w3schools.com/js/js_json_xml.asp