**XML**

1. **Classes and Functions Used:**

Different programming languages provide various libraries and tools to handle XML files. Below are commonly used libraries in different programming languages:

* **Python**: xml.etree.ElementTree, minidom, lxml
* **Java**: javax.xml.parsers, org.w3c.dom, SAXParser
* **C#**: System.Xml, XDocument, XmlDocument
* **JavaScript**: DOMParser, XMLSerializer

**Key Functions:**

* Parsing XML: Functions like parse(), fromstring(), load()
* Creating Elements: Element(), createElement()
* Modifying Data: setAttribute(), appendChild(), removeChild()
* Saving XML: write(), toxml(), tostring()

1. **Saving XML Files:**

To save XML files, the data needs to be serialized into a structured format and stored as a .xml file. Below is an example in Python using xml.etree.ElementTree:

import xml.etree.ElementTree as ET

data = ET.Element("root")

child = ET.SubElement(data, "child")

child.text = "Sample Data"

tree = ET.ElementTree(data)

tree.write("output.xml")

1. **How to Read XML Files:**

Reading XML files involves parsing the file and extracting necessary data. Below is an example in Python:

import xml.etree.ElementTree as ET

tree = ET.parse("input.xml")

root = tree.getroot()

for child in root:

print(child.tag, child.text)

**How to Analyze Data in Elements and Attributes**

To analyze XML data, we extract elements and their attributes. Example in Python:

root = tree.getroot()

for element in root.findall("child"):

print(element.get("attribute"), element.text)