

# WT – Assignment-3

## Code

### employees.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="employees.xsl"?>
<employees>
  <employee>
    <id>1</id>
    <name>Kshitij K</name>
    <position>DevOps Engineer</position>
    <department>IT</department>
    <salary>6000000</salary>
  </employee>
  <employee>
    <id>2</id>
    <name>Ayush Kalas</name>
    <position>Project Manager</position>
    <department>IT</department>
    <salary>8000</salary>
  </employee>
  <employee>
    <id>3</id>
    <name>Ombase</name>
    <position>Frontend Developer</position>
    <department>IT</department>
    <salary>500</salary>
  </employee>
</employees>
```

### employees.xsl

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:template match="/">
    <html>
      <head>
        <title>Employee List</title>
        <style>
          table {
            width: 100%;
            border-collapse: collapse;
          }
        </style>
      </head>
    </html>
  </template>
</xsl:stylesheet>
```

```

        th, td {
            border: 1px solid black;
            padding: 8px;
            text-align: left;
        }
        th {
            background-color: #f2f2f2;
        }
    </style>
</head>
<body>
    <h2>Employee List</h2>
    <table>
        <tr>
            <th>ID</th>
            <th>Name</th>
            <th>Position</th>
            <th>Department</th>
            <th>Salary</th>
        </tr>
        <xsl:for-each select="employees/employee">
            <tr>
                <td><xsl:value-of select="id"/></td>
                <td><xsl:value-of select="name"/></td>
                <td><xsl:value-of select="position"/></td>
                <td><xsl:value-of select="department"/></td>
                <td><xsl:value-of select="salary"/></td>
            </tr>
        </xsl:for-each>
    </table>
</body>
</html>
</xsl:template>
</xsl:stylesheet>

```

## employees.dtd

```

<!DOCTYPE employees [
    <!ELEMENT employees (employee+)>
    <!ELEMENT employee (id, name, position, department, salary)>
    <!ELEMENT id (#PCDATA)>
    <!ELEMENT name (#PCDATA)>
    <!ELEMENT position (#PCDATA)>
    <!ELEMENT department (#PCDATA)>
    <!ELEMENT salary (#PCDATA)>
]>

```

## employees.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="employees">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="employee" maxOccurs="unbounded">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="id" type="xs:integer"/>
              <xs:element name="name" type="xs:string"/>
              <xs:element name="position" type="xs:string"/>
              <xs:element name="department" type="xs:string"/>
              <xs:element name="salary" type="xs:decimal"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

## Output

Employee List				
ID	Name	Position	Department	Salary
1	Kshitij K	DevOps Engineer	IT	6000000
2	Ayush Kalas	Project Manager	IT	8000
3	Ombase	Frontend Developer	IT	500