**1)DIFFERENCE BETWEEN DTD AND XSD**

**🡪**

| S.NO. | DTD | XSD |
| --- | --- | --- |
| 1. | DTD are the declarations that define a document type for SGML. | XSD describes the elements in a XML document. |
| 2. | It doesn’t support namespace. | It supports namespace. |
| 3. | It is comparatively harder than XSD. | It is relatively more simpler than DTD. |
| 4. | It doesn’t support datatypes. | It supports datatypes. |
| 5. | SGML syntax is used for DTD. | XML is used for writing XSD. |
| 6. | It is not extensible in nature. | It is extensible in nature. |
| 7. | It doesn’t give us much control on structure of XML document. | It gives us more control on structure of XML document. |

2) PCDATA and CDATA in DTD

🡪

**CDATA:** (Unparsed Character data): CDATA contains the text which is not parsed further in an XML document. Tags inside the CDATA text are not treated as markup and entities will not be expanded.

Let's take an example for CDATA:

1. **<?xml** version="1.0"**?>**
2. <!DOCTYPE employee SYSTEM "employee.dtd"**>**
3. **<employee>**
4. <![CDATA[
5. <firstname>sid</firstname>
6. <lastname>ghosalkar</lastname>
7. <email>sid@javatpoint.com</email>
8. ]]>
9. **</employee>**

**PCDATA**: (Parsed Character Data): XML parsers are used to parse all the text in an XML document. PCDATA stands for Parsed Character data. PCDATA is the text that will be parsed by a parser. Tags inside the PCDATA will be treated as markup and entities will be expanded.

In other words you can say that a parsed character data means the XML parser examine the data and ensure that it doesn't content entity if it contains that will be replaced.

Let's take an example:

1. **<?xml** version="1.0"**?>**
2. <!DOCTYPE employee SYSTEM "employee.dtd"**>**
3. **<employee>**
4. **<firstname>**SID**</firstname>**
5. **<lastname>**GHOSALKAR**</lastname>**
6. **<email>**sid@javatpoint.com**</email>**
7. **</employee>**

3) Difference between Throw and Throws with implementation in program.

🡪

|  |  |
| --- | --- |
| throw | throws |
| throw keyword is used to throw an exception explicitly. | throws keyword is used to declare one or more exceptions, separated by commas. |
| Only single exception is thrown by using throw. | Multiple exceptions can be thrown by using throws. |
| throw keyword is used within the method. | throws keyword is used with the method signature. |
| Syntax wise throw keyword is followed by the instance variable. | Syntax wise throws keyword is followed by exception class names. |
| Checked exception cannot be propagated using throw only.Unchecked exception can be propagated using throw. | For the propagation checked exception must use throws keyword followed by specific exception class name. |

4)One line definition for different types of exceptions

|  |  |
| --- | --- |
| **Name** | **Description** |
| IOException | While using file input/output stream related exception |
| SQLException. | While executing queries on database related to SQL syntax |
| DataAccessException | Exception related to accessing data/database |
| ClassNotFoundException | Thrown when the JVM can’t find a class it needs, because of a command-line error, a classpath issue, or a missing .class file |
| InstantiationException | Attempt to create an object of an abstract class or interface. |