# Lab #8

**Objective:** XML schema and parsing an XML document.

1. Create an XML document structuring information for a store. (shopping store of your choice)

<?xml version="1.0"?>

<store xmlns="http://www.w3schools.com"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://www.w3schools.com store.xsd">

<product> Laptop</product>

<model>Hp EliteBook 8470p </model>

<price>18000 Rs</price>

<color>Silver</color>

<product> Mobile</product>

<model>Q-mobile i6 Metal one</model>

<price> 6000 Rs</price>

<color>Golden</color>

<product>Tablet</product>

<model>s5E</model>

<price>20000 Rs</price>

<color>Black</color>

<product> Gaming Console</product>

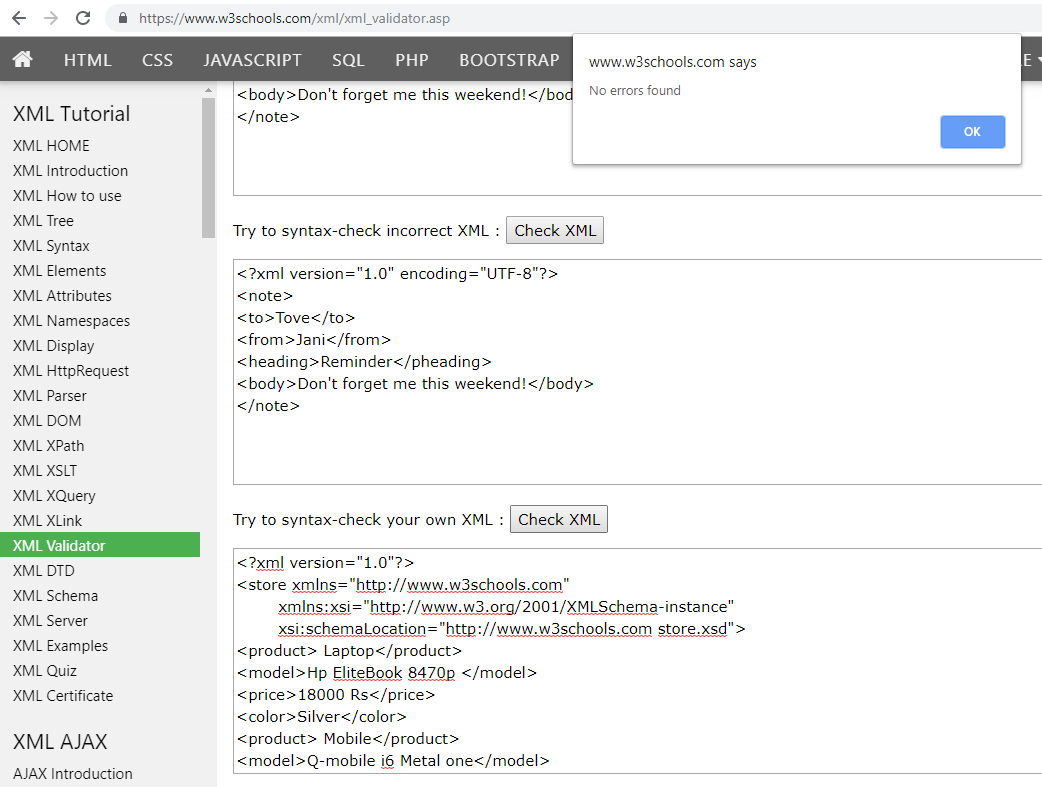
<model>PlayStation4</model>

<price>50000 Rs</price>

<color>white</color>

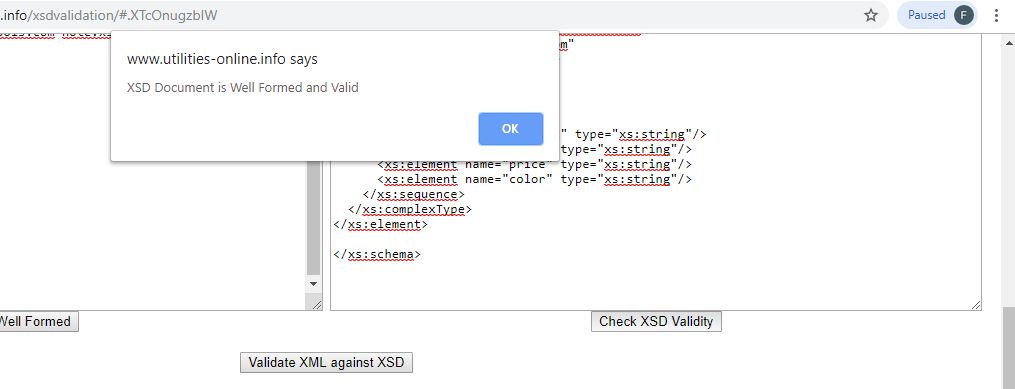
</store>

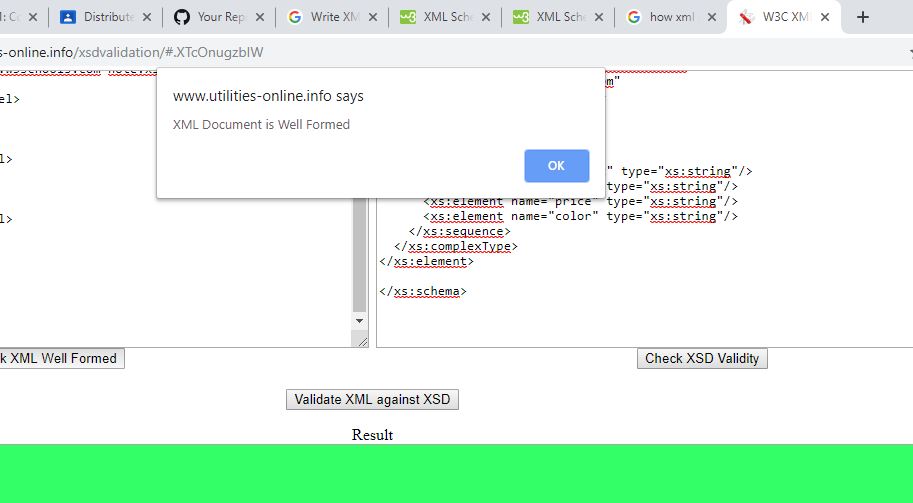
1. **Create DTD for store XML Document. And validate on** [**http://www.w3schools.com/xml/xml\_validator.asp**](http://www.w3schools.com/xml/xml_validator.asp)



1. **Write XML Schema for the following XML document and then validate the XML document along with its schema on the following website:**

[**http://www.utilities-online.info/xsdvalidation/#.UlFx2tKmhIg**](http://www.utilities-online.info/xsdvalidation/)







**4. Write an XML instance for following schema and then parse that document in Java using either DOM or SAX API.**

<?xml version=“1.0” ?>

<xs:schema xmlns:xs=“http://www.w3.org/2001/XMLSchema”>

<xs:element name="person“ type="persontype"/>

<xs:complexType name="persontype">  
<xs:sequence>  
 <xs:element name="firstname“ type="xs:string"/>  
<xs:element name="lastname" type="xs:string"/>  
<xs:element name=“date\_of\_birth” type=“date”/>

</xs:sequence>  
</xs:complexType>

<xs:complexType name=“date”>

<xs:sequence>

<xs:element name=“day” type=“xs:integer”/>

<xs:element name=“month” type=“xs:integer”/>

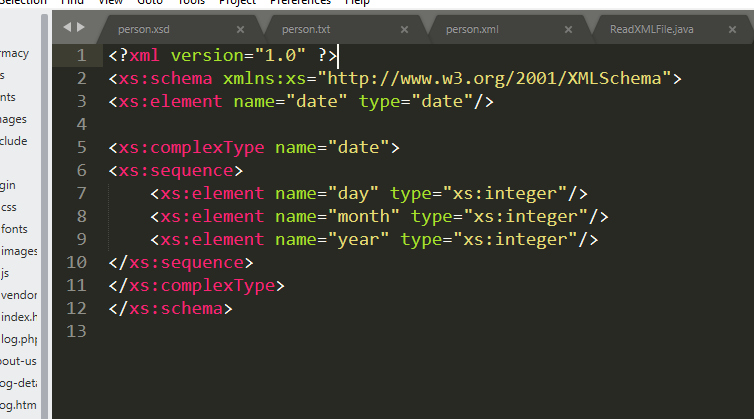
<xs:element name=“year” type=“xs:integer”/>

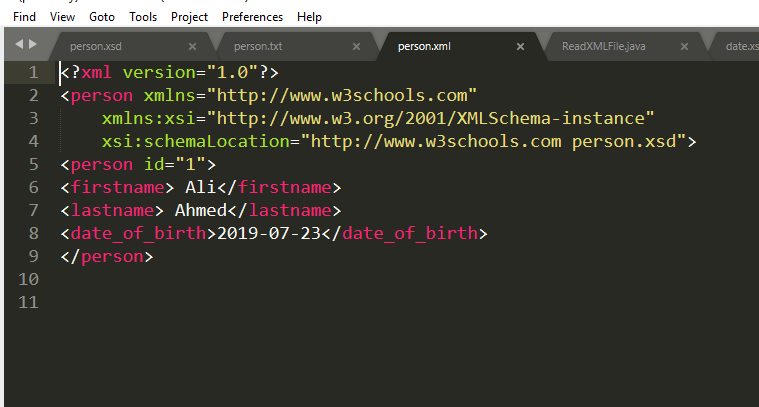
</xs:sequence>

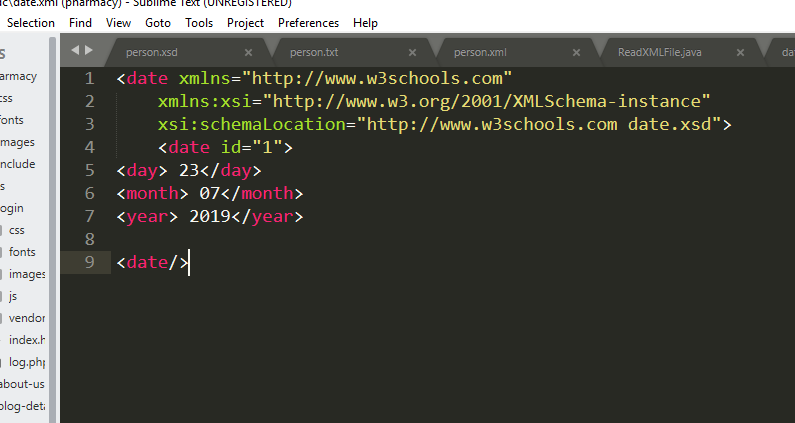
</xs:complexType>

</xs:schema>

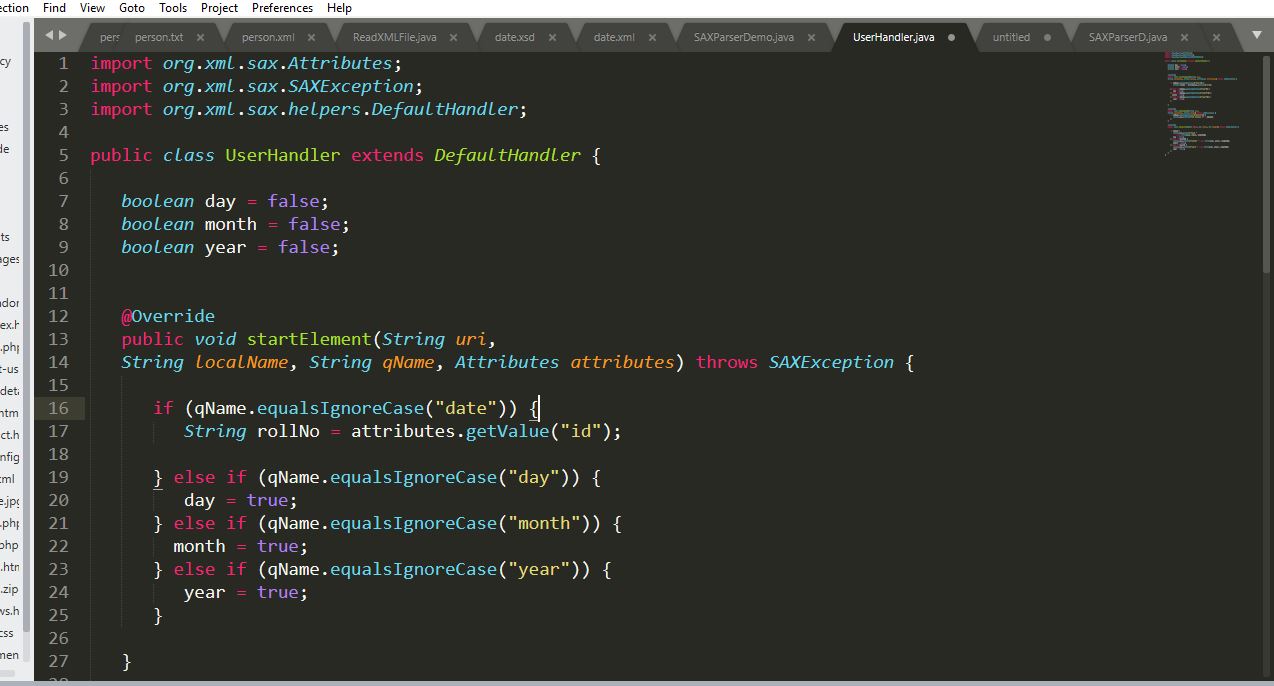


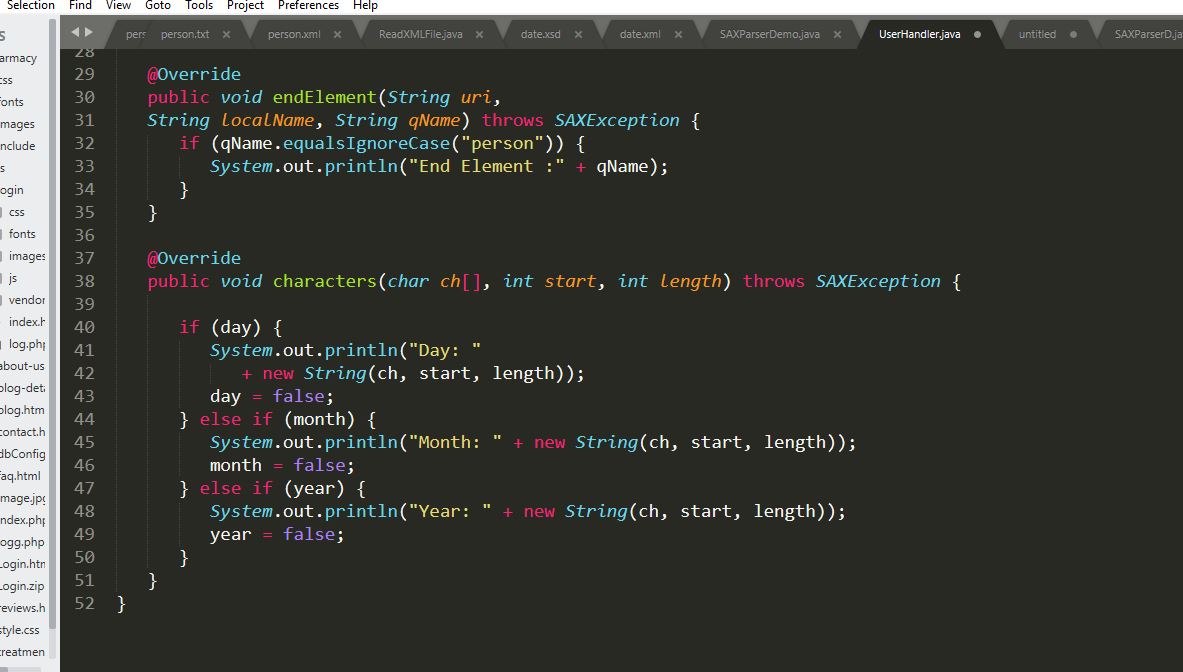


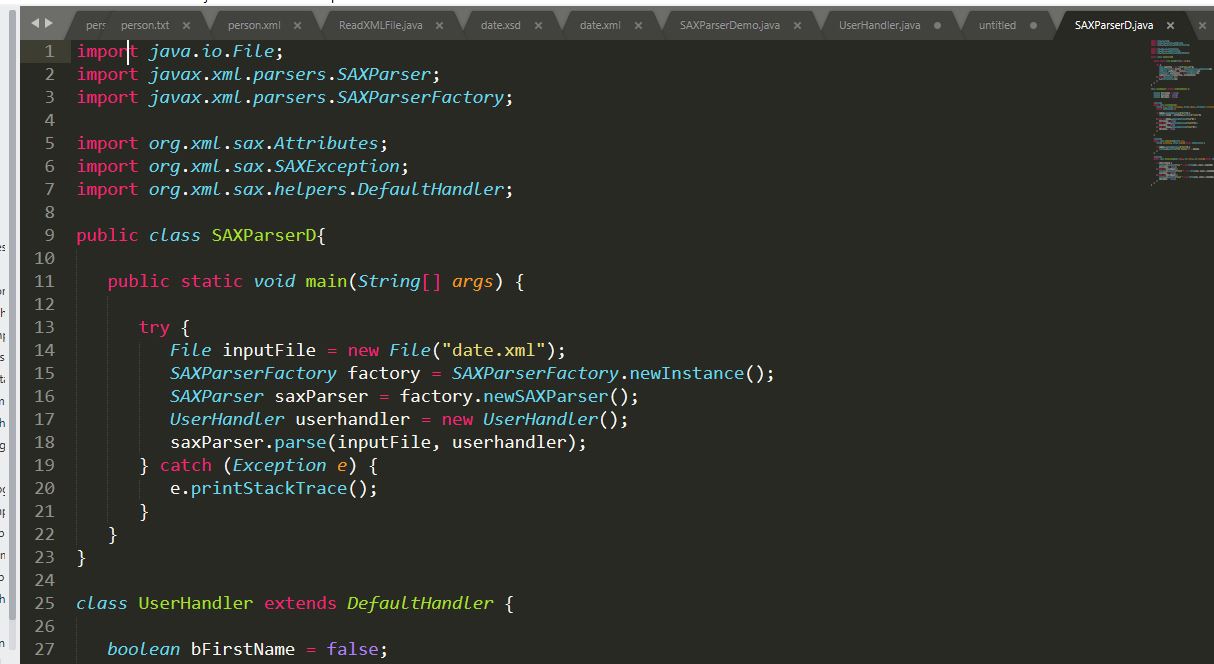


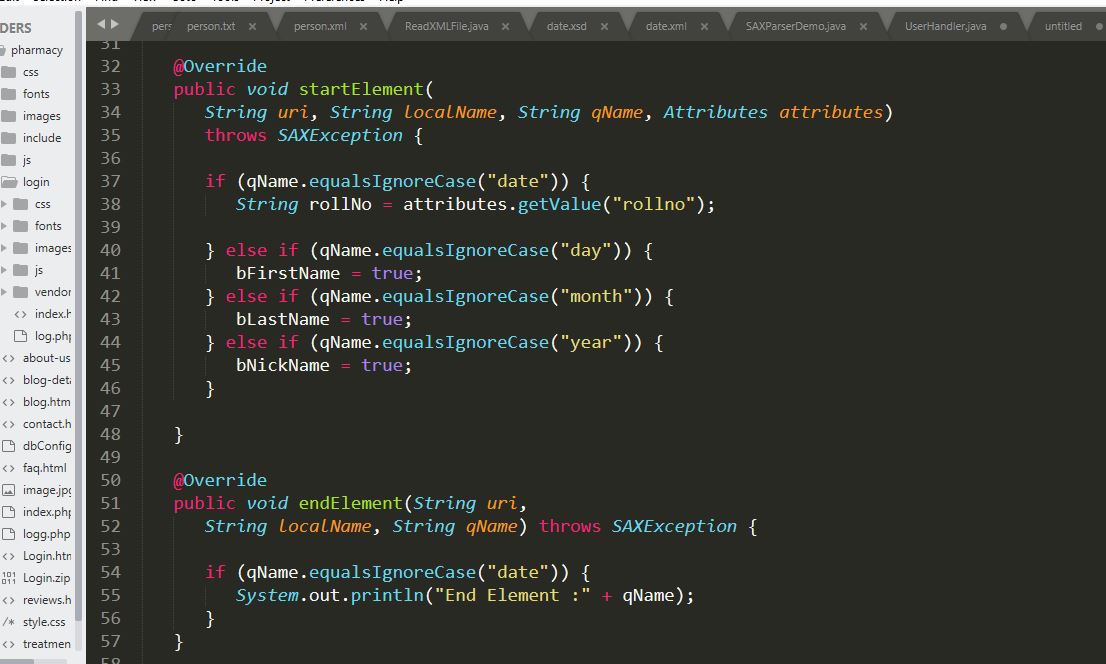


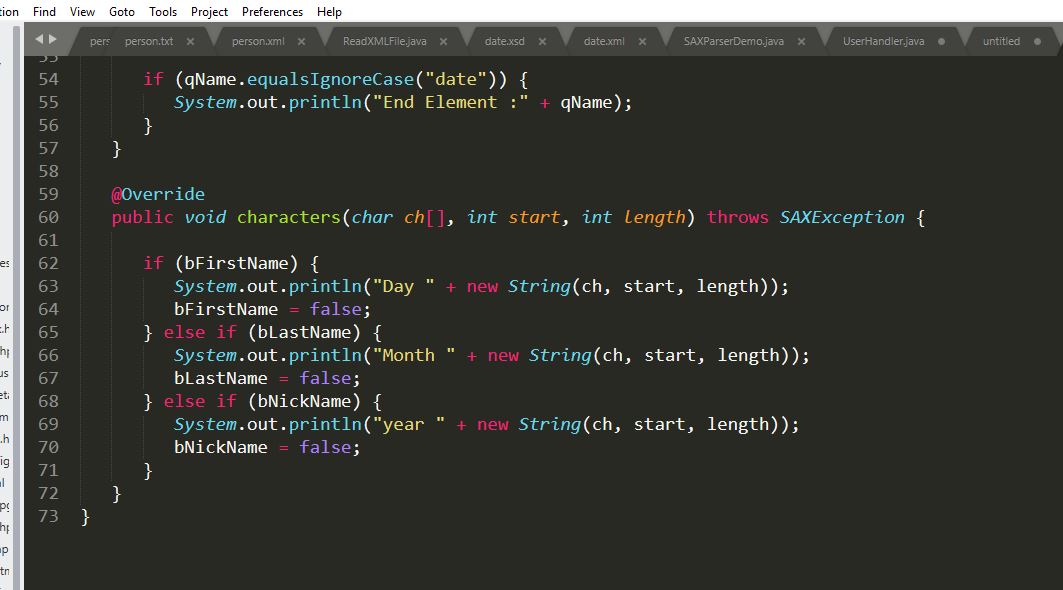
**Parse Java Code Using SAX Api:**



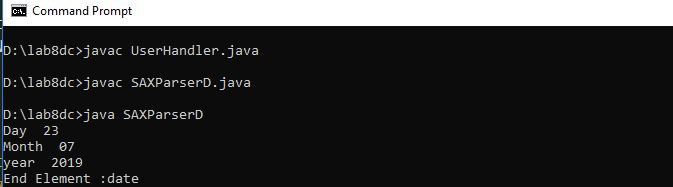
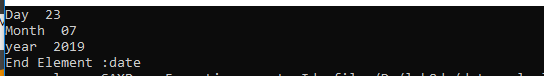




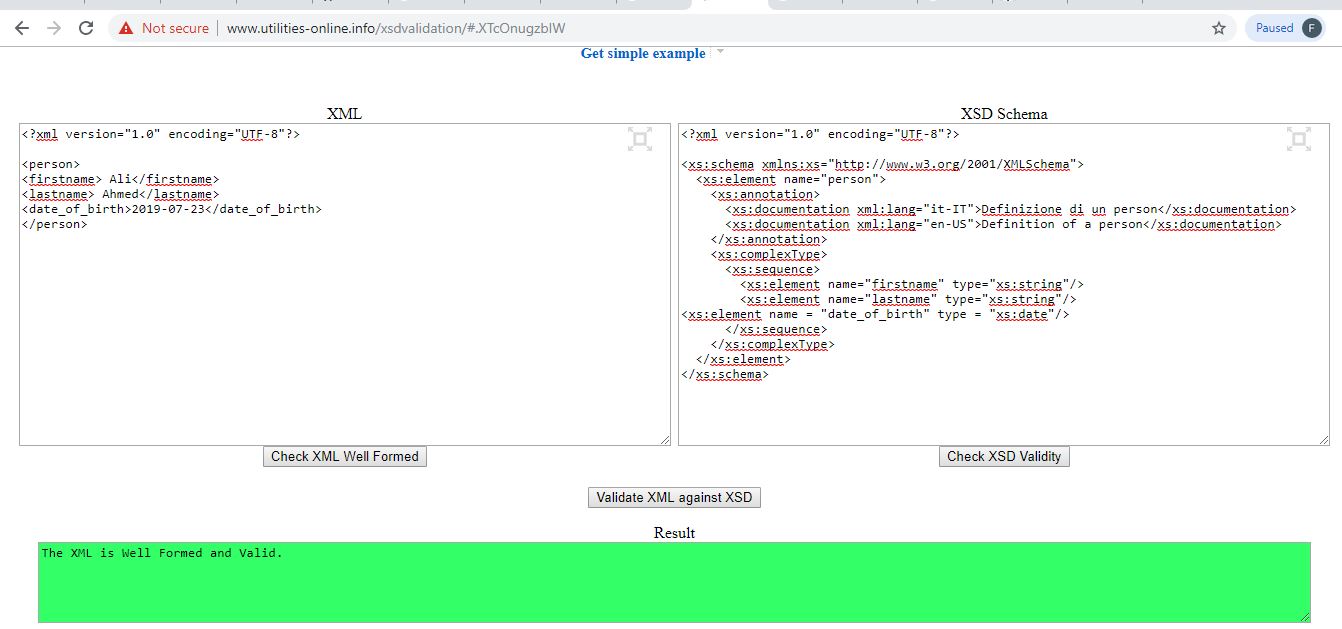




**Output:**



Person:



Date:

