Q1) Name all XML elements and child elements.

Orders

Order

Customerid

Status

item

Name

Price

Qty

Q-2 ) Is this a well form document? Is it a valid document?

No,this is a well form document

And it is a valid document

Q3) Design the DTD for this document.

<!ELEMENT orders (order+)>

<!ELEMENT order (customerid,status,item+)>

<!ELEMENT customerid (#PCDATA)>

<!ELEMENT status (#PCDATA)>

<!ELEMENT item (name,price,qty)>

<!ELEMENT name (#PCDATA)>

<!ELEMENT price (#PCDATA)>

<!ELEMENT qty (#PCDATA)>

<!ATTLIST item instock (N|Y) #REQUIRED>

<!ATTLIST item itemid ID #REQUIRED>

Q4) Design XML Schema for this document? Add a proper XML Schema to this document.

<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified" xmlns:xs="http://www.w3.org/2001/XMLSchema">

<xs:element name="orders">

<xs:complexType>

<xs:sequence>

<xs:element name="order" maxOccurs="unbounded" minOccurs="0">

<xs:complexType>

<xs:sequence>

<xs:element type="xs:short" name="customerid"/>

<xs:element type="xs:string" name="status"/>

<xs:element name="item" maxOccurs="unbounded" minOccurs="0">

<xs:complexType>

<xs:sequence>

<xs:element type="xs:string" name="name"/>

<xs:element type="xs:float" name="price"/>

<xs:element type="xs:short" name="qty"/>

</xs:sequence>

<xs:attribute type="xs:string" name="instock" use="optional"/>

<xs:attribute type="xs:string" name="itemid" use="optional"/>

</xs:complexType>

</xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:schema>

Q5) Write an XSLT document to transform order.xml into an HTML document that displays the orders in a table.

<?xml version="1.0"?>

<!DOCTYPE orders SYSTEM "order.dtd">

<?xml-stylesheet type="text/xsl" href="order5.xsl" ?>

<orders>

<order>

<customerid>2384</customerid>

<status>pending</status>

<item instock="Y" itemid="SD93">

<name>Flying By Roller Skates</name>

<price>25.00</price>

<qty>25</qty>

</item>

<item instock="N" itemid="B12">

<name>Bounce-o Ball</name>

<price>.35</price>

<qty>150</qty>

</item>

</order>

<order>

<customerid>5268</customerid>

<status>complete</status>

<item instock="Y" itemid="Q52">

<name>Crash N Burn Skis</name>

<price>20</price>

<qty>10</qty>

</item>

</order>

<order>

<customerid>3384</customerid>

<status>pending</status>

<item instock="Y" itemid="PS93">

<name>All Star Shoe</name>

<price>55.00</price>

<qty>12</qty>

</item>

<item instock="Y" itemid="M12">

<name>All Star Hat</name>

<price>44.35</price>

<qty>15</qty>

</item>

</order>

<order>

<customerid>9008</customerid>

<status>pending</status>

<item instock="N" itemid="F32">

<name>Fancy Shirt</name>

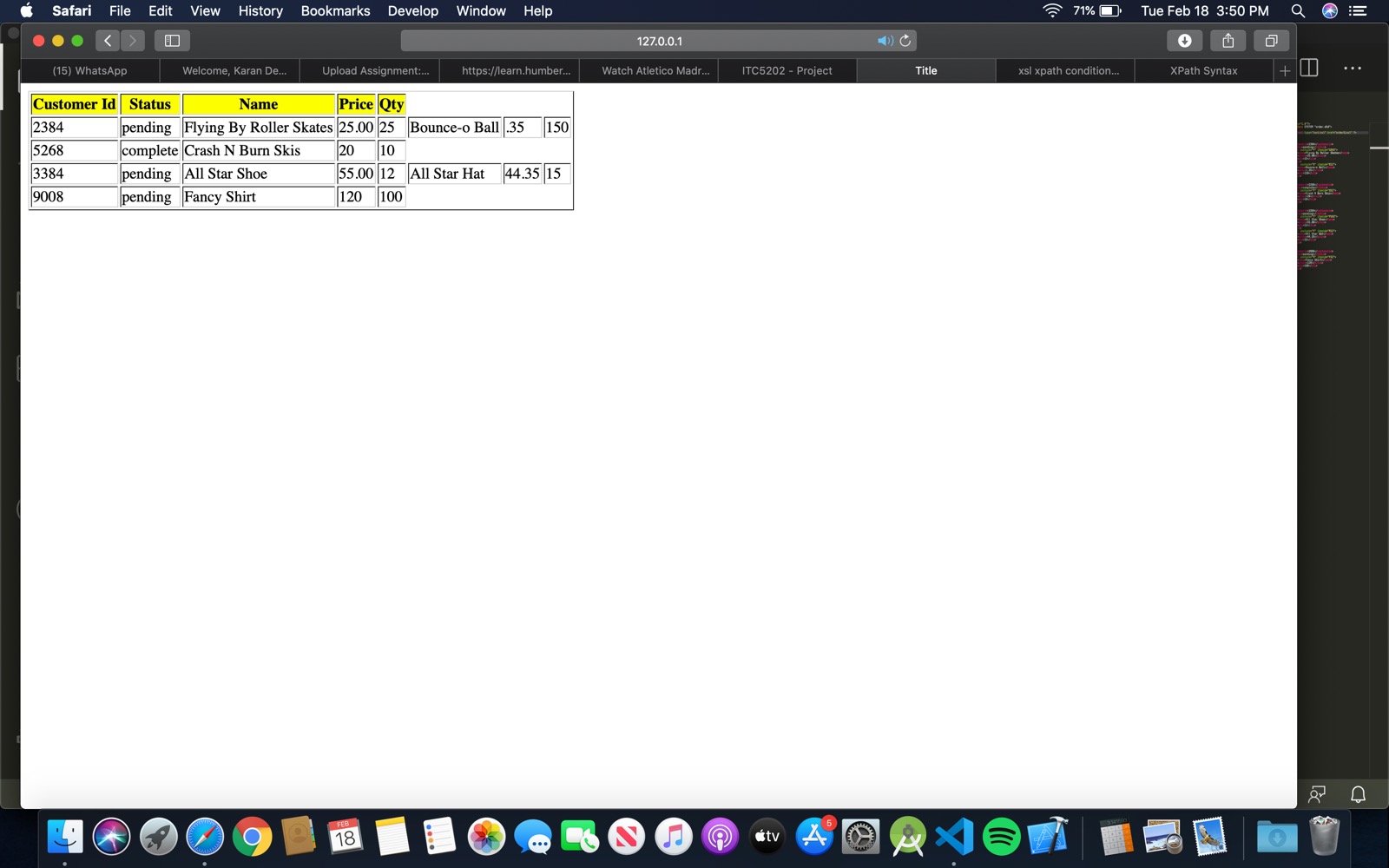
<price>120</price>

<qty>100</qty>

</item>

</order>

</orders>



Q-6)

<?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>Title</title>

</head>

<body>

<table border="1" >

<tr style="background-color:red">

</tr>

<xsl:for-each select="orders/order">

<tr>

<td> <xsl:value-of select="item[@instock='N']"/> </td>

</tr>

</xsl:for-each>

</table>

</body>

</html>

</xsl:template>

</xsl:stylesheet>

Q-7)Using Q7.html, you need to develop a proper JavaScript function which will be invoked when user clicks on the show pending orders. The JavaScript function processes the order.xml data and displays all orders that their status are “pending”

<!doctype html>

<html>

<head>

<meta charset="utf-8">

<title>ITC5202 - Project</title>

</head>

<body>

<h1> Order Report</h1>

<button onclick="loadData()"> Show pending orders </button>

<p id="demo"></p>

</body>

<script>

//Create the XHR object

function loadData() {

var xhttp = new XMLHttpRequest();

//Open the file

xhttp.open("GET", "order.xml", false);

//Send the request

xhttp.send();

// Read data

xmlDoc = xhttp.responseXML;

var order = xmlDoc.getElementsByTagName("order");

var customerid = xmlDoc.getElementsByTagName("customerid");

var status = xmlDoc.getElementsByTagName("status");

var item = xmlDoc.getElementsByTagName("item");

var name = xmlDoc.getElementsByTagName("name");

var price = xmlDoc.getElementsByTagName("price");

var qty = xmlDoc.getElementsByTagName("qty");

let txt = "<table border='1'>";

txt += "<th>ID</th><th>Customer Id</th>";

for (let i = 0; i < order.length; i++) {

if (status[i].innerHTML == "pending") {

txt += "<tr>"

txt += "<td>" + (i + 1) + "</td>"

txt += "<td>" + customerid[i].innerHTML + "</td>";

//txt += "<td>" + status[i].innerHTML + "</td>";

//txt += "<td>" + name[i].innerHTML + "</td>";

//txt += "<td>" + item[i].innerHTML.length + "</td>";

//console.log("hi 1: " + item[i].length);

/\*for (let j = 0; j < item[i].length; j++) {

txt += "<td>" + name[j].innerHTML + "</td>";

txt += "<td>" + price[j].innerHTML + "</td>";

txt += "<td>" + qty[j].innerHTML + "</td>";

}\*/

txt += "</tr>";

}

}

console.log("hi : " + item.length);

txt += "</table>"

document.getElementById("demo").innerHTML = txt;

}

</script>

</html>

