Assignment 1

- You should work on the assignments in teams of three students (if possible, keep the same groups for further assignments).
- You have to submit your solution in Ilias until May 30, 2025, 10 pm.
- Only **one** file per group should be submitted
- Include the team number and your last names in the filename

Task 1 - XML Schema

14 points

Given are the following XML Schema, describing the export format of an online fashion shop, and an associated XML document.

Schema:

```
<?xml version="1.0" encoding="UTF-8"?>
1
     <xs:schema</pre>
       xmlns:tns="http:///IPVS/AS/Onlineshop"
3
 4
       xmlns:xs="http://www.w3.org/2001/XMLSchema"
       targetNamespace="http:///IPVS/AS/Onlineshop">
 5
 6
 7
       <xs:element name="Orders">
8
         <xs:complexType>
9
           <xs:sequence>
              <xs:element name="Order" type="tns:tOrder"</pre>
10
                          minOccurs="0" maxOccurs="unbounded" />
11
12
           </xs:sequence>
13
         </xs:complexType>
14
         </xs:element>
15
       <xs:complexType name="tOrder">
16
         <xs:sequence>
17
           <xs:element name="Customer" type="tns:tCustomer" />
18
           <xs:element name="BillingAddress" type="tns:tAddress"/>
19
           <xs:element name="DeliveryAddress" type="tns:tAddress"</pre>
20
                        minOccurs="0" maxOccurs="1"/>
21
           <xs:element name="Items">
22
             <xs:complexType>
23
                <xs:sequence>
24
                  <xs:element name="Item" type="tns:tOrderItem"</pre>
25
                               minOccurs="1" maxOccurs="unbounded"/>
26
                </xs:sequence>
27
             </xs:complexType>
28
           </xs:element>
29
         </xs:sequence>
30
         <xs:attribute name="orderID" type="xs:ID" use="required" />
31
32
       </xs:complexType>
33
```

```
<xs:complexType name="tOrderItem">
34
35
         <xs:sequence>
           <xs:element name="Name" type="xs:string" />
36
           <xs:element name="Size" type="xs:string" />
37
           <xs:element name="Quantity">
38
             <xs:simpleType>
39
               <xs:restriction base="xs:integer">
40
                 <xs:minInclusive value="1"/>
41
                 <xs:maxExclusive value="10"/>
42
43
               </xs:restriction>
             </xs:simpleType>
44
           </xs:element>
45
46
         </xs:sequence>
         <xs:attribute name="itemID" type="xs:ID" use="required" />
47
       </xs:complexType>
48
49
       <xs:complexType name="tCustomer">
50
51
         <xs:sequence>
           <xs:element name="Name" type="xs:string" />
52
53
           <xs:element name="Birthdate" type="xs:date" />
54
         </xs:sequence>
55
         <xs:attribute name="customerID" type="xs:ID" use="required" />
56
       </xs:complexType>
57
       <xs:complexType name="tAddress">
58
         <xs:all>
59
           <xs:element name="StreetName" type="xs:string" />
60
           <xs:element name="StreetNumber" type="xs:int" />
61
           <xs:element name="ZipCode">
62
             <xs:simpleType>
63
               <xs:restriction base="xs:integer">
64
                  <xs:pattern value="[0-9][0-9][0-9][0-9][0-9]" />
65
               </xs:restriction>
66
             </xs:simpleType>
67
68
           </xs:element>
           <xs:element name="City" type="xs:string" />
69
70
         </xs:all>
         <xs:attribute name="addressID" type="xs:ID" use="required" />
71
72
       </xs:complexType>
73
74
     </xs:schema>
75
```

Document:

```
<?xml version="1.0" encoding="UTF-8"?>
 1
 2
     <tns:Orders</pre>
 3
       xmlns:tns="http:///IPVS/AS/Onlineshop"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 4
 5
       xsi:schemaLocation="http:///IPVS/AS/Onlineshop Onlineshop.xsd">
       <Order orderID="0000458">
 6
 7
         <Customer customerID="C123583">
           <Name>Carly Customer</Name>
 8
           <Birthdate>1992-07-12</Birthdate>
 9
10
         </Customer>
         <BillingAddress addressID="A5483">
11
12
           <StreetName>Milchstr.</StreetName>
           <StreetNumber>17</StreetNumber>
13
           <ZipCode>73728</ZipCode>
14
15
           <City>Stuttgart</City>
16
         </BillingAddress>
17
         <DeliveryAddress addressID="A1235">
           <StreetName>Universitätsstr. 38</StreetName>
18
           <ZipCode>70569</ZipCode>
19
20
           <City>Stuttgart</City>
21
         </DeliveryAddress>
         <Items>
22
           <Item itemID="I000489">
23
              <Name>Cable RJ45 blue</Name>
24
25
              <Size>0.5m</Size>
26
              <Quantity>10</Quantity>
27
           </Item>
           <Item itemID="I000472">
28
              <Name>Mouse Logitech M510 Wireless</Name>
29
              <Size>0</Size>
30
31
              <Quantity>1</Quantity>
           </Item>
32
33
         </Items>
       </Order>
34
35
       <Order orderID="0001258">
36
37
         <Customer customerID="C623583">
           <Name>Sven Shopper</Name>
38
           <Birthdate>1987-01-25</Birthdate>
39
         </Customer>
40
         <BillingAddress>
41
42
           <StreetName>Kurze Straße</StreetName>
           <StreetNumber>7</StreetNumber>
43
           <ZipCode>1080</ZipCode>
44
45
           <City>Wien</City>
         </BillingAddress>
46
47
         <Items>
           <Item itemID="I000328">
48
              <Quantity>2</Quantity>
49
              <Name>Keyboard Cherry G230</Name>
50
```

```
<Size>0</Size>
51
52
           </Item>
53
           <Item itemID="I000328">
             <Name>Cable USB 3.0 A/B</Name>
54
55
              <Size>2m</Size>
56
             <Quantity>1</Quantity>
57
           </Item>
           <Item itemID="I000452">
58
             <Name>Headset Over-Ear Stereo</Name>
59
              <Size>5</Size>
60
             <Quantity>1</Quantity>
61
62
           </Item>
         </Items>
63
       </Order>
64
     </tns:Orders>
65
66
```

1.1 (3 points)

What are the two most important rules for well-formed XML documents? **Explain these rules** briefly, then check whether the given XML document is well-formed.

Is the document well-formed according to those rules? Explain your answer.

1.2 (10 points)

Presume the given XML document is well-formed. Is it also valid? List and explain at least four invalidities. Please use the line numbers for easier reference.

1.3 (1 point)

Can a valid XML document not be well-formed?

Task 2 - XQuery

17 points

The following XML file "hotels.xml" provides **an extract** of an XML document to manage hotels. To solve the following tasks, please use XQUERY expressions as introduced in the lecture and in the exercises, respectively. Unnecessary statements may lead to deduction of points. Note that the structure of the output does not have to match the given examples exactly.

```
1
      <?xml version="1.0"?>
 2
      <Hotels>
 3
          <Hotel hotelNumber="137865">
 4
               <Name>Seaside Hotel</Name>
 5
               <Address>
 6
                   <TownName>Stralsund</TownName>
 7
                   <StreetAndHouseNumber>Friedrich-Naumann-Street 25</StreetAndHouseNumber>
 8
                   <ZipCode>18435</ZipCode>
 9
               </Address>
10
               <Stars>5</Stars>
11
               <Guests>
12
                     <Guest guestNumber="458">
13
                        <Name>Gary Guest</Name>
14
                        <Address>
15
                             <TownName>Stuttgart</TownName>
16
                             <StreetAndHouseNumber>Universitaetsstr. 38</StreetAndHouseNumber>
17
                             <ZipCode>70569</ZipCode>
18
                       </Address>
19
                  </Guest>
                  <Guest guestNumber="672">
20
21
                      <Name>Vicky Visitor</Name>
22
23
                           <TownName>Hamburg</TownName>
24
                           <StreetAndHouseNumber>Bahnhofstr. 3</StreetAndHouseNumber>
25
                           <ZipCode>22767</ZipCode>
26
27
                  </Guest>
28
              </Guests>
29
              <Reservations>
30
                  <Reservation roomNumber="5" guestNumber="458">
31
                     <RoomType>
32
                          <BasicRoom>
                               <AdditionalBed>false</AdditionalBed>
33
34
                               <MiniBar>false</MiniBar>
35
                               <Price>70</Price>
36
                           </BasicRoom>
37
                       </RoomType>
38
                       <Arrival>2015-10-24</Arrival>
39
                       <Departure>2015-10-30</Departure>
                       <Smoker>false</Smoker>
40
41
                  </Reservation>
42
                  <Reservation roomNumber="6" guestNumber="672">
43
                       <RoomType>
44
                           <LuxuryRoom>
45
                               <OceanView>true</OceanView>
46
                               <RoomService>true</RoomService>
47
                               <Price>140</Price>
48
                           </LuxuryRoom>
49
                       </RoomType>
50
                       <Arrival>2015-21-09</Arrival>
51
                       <Departure>2015-10-24</Departure>
52
                       <Smoker>true</Smoker>
53
                   </Reservation>
```

```
54
55
56
57
58
```

2.1 (4 points)

Create a list containing the **name**, the **town name** and the total **number of reservations** for each hotel in hotels.xml.

2.2 (4 points)

Provide a **list** of **reservations** for each guest **(guest name)**. The list should contain only a single reservation per entry, which e.g., results in 2 entries for 2 reservations of the same guest.

2.3 (4 points)

Provide a list of guests for each hotel. The list has to contain the **hotel number** as well as the **names** of the visiting guests. Consider only guests with names that have less than 11 characters.

```
Example: <output> 137865 Gary Guest <output>...
```

2.4 (5 points)

List the **street and house numbers** of all guests (**guestNumber**) that have **reservations** for a **basic room without** a **minibar**. **Sort** the result list according to the **street and house numbers** alphabetically **ascending**.

Example: <output> 458 Universitaetsstr. 38 </output>...

Task 3 - DTD and XML Schema

24 points

3.1 (10 points)

Write a DTD to describe a document storing machine and machine incident information. The DTD must satisfy the following aspects. Try to cover all of them. If this is not possible for some of the mentioned aspects, provide a detailed explanation on the limitations of DTDs. Write the complete DTD, do not abbreviate!

- Multiple machines can be stored.
- Each machine is covered by an element machine containing exactly once the sub-elements machine-name and maintenance-engineer. It also contains multiple sub-elements incident, grouped by an element incidents.
- Each machine has a unique **machine-number** as a mandatory attribute.
- Each incident has a sub-element **incident-message**, **incident-code**, **incident-date** and an optional sub-element **machine-temperature**.
- A machine-number must have exactly 5 characters.
- There has to be at least one incident per machine.
- A machine-temperature must contain a floating point number.

3.2 (12 points)

Provide an XML schema for the document storing machine and machine incident information as described in task 3.1.

3.3 (2 points)

Name two advantages XML Schema has over DTD.