

15. April 2010

Semantic Web Services

Exercise sheet 4
Web services**Exercise 1 (WSDL) (4 points)**

Complete the following WSDL file in a way such that the operations it provides correspond to the following procedure calls

- float add(float x, int y)
- boolean lessThan(int a, int b)
- String trim(String s)

The service shall be available at the following endpoint-address:
<http://www.swsbook.com/theService>

```
<?xml version="1.0" encoding="UTF-8"?>
<definitions name="CompService" targetNamespace="urn:Foo"
xmlns:tns="urn:Foo"
xmlns="http://schemas.xmlsoap.org/wsdl/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/">
  <types/>
  <message name="_____">
    <part name="parameter_1" type="xsd:integer"/>
    <part name="parameter_2" type="_____" />
  </message>
  <message name="_____">
    <part name="par_1" type="_____" />
    <part name="par_2" type="_____" />
  </message>
  <message name="_____">
    <part name="echoRequestIn" type="_____" />
  </message>
  <message name="_____">
    <part name="resultdiv" type="_____" />
  </message>
  <message name="_____">
    <part name="resultmul" type="_____" />
  </message>
</definitions>
```

```
<message name="_____">
  <part name="resultecho" type="_____" />
</message>
<portType name="CompIF">
  <operation name="div"
    parameterOrder="_____">
    <input message="_____" />
    <output message="_____" />
  </operation>
  <operation name="mul" parameterOrder="_____">
    <input message="_____" />
    <output message="_____" />
  </operation>
  <operation name="echo">
    <input message="_____" />
    <output message="_____" />
  </operation>
</portType>
<binding name="CompIFBinding" type="tns:CompIF">
  <operation name="mul">
    <input>
      <soap:body
        encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
        use="encoded" namespace="urn:Foo" />
    </input>
    <output>
      <soap:body
        encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
        use="encoded" namespace="urn:Foo" />
    </output>
    <soap:operation soapAction="" />
  </operation>
  <!-- other bindings skipped for space reasons -->
</binding>
<soap:binding transport="http://schemas.xmlsoap.org/soap/http"
  style="rpc" />
</binding>
<service name="CompService">
  <port name="CompIFPort" binding="tns:CompIFBinding">
    <soap:address location="_____" />
  </port>
</service>
</denitions>
```

Exercise 2 (SOAP) (5 points)

Given the following WSDL file (<http://www.webserviceX.net/stockquote.asmx?WSDL>) for a Stock Quote Web service, design a SOAP request and a SOAP response for the GetQuote operation in the StockQuoteSoap port type.

Listing 1: Stock Quote Web service – WSDL

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:denitions xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:s="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:tns="http://www.webserviceX.NET/"
  xmlns:tm="http://microsoft.com/wsdl/mime/textMatching/"
  xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
  targetNamespace="http://www.webserviceX.NET/"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
```

```

<wsdl:types>
  <s:schema elementFormDefault="qualified" targetNamespace="http://www.webserviceX.NET/">
    <s:element name="GetQuote">
      <s:complexType>
        <s:sequence>
          <s:element minOccurs="0" maxOccurs="1" name="symbol" type="s:string" />
        </s:sequence>
      </s:complexType>
    </s:element>
    <s:element name="GetQuoteResponse">
      <s:complexType>
        <s:sequence>
          <s:element minOccurs="0" maxOccurs="1" name="GetQuoteResult" type="s:string" />
        </s:sequence>
      </s:complexType>
    </s:element>
    <s:element name="string" nillable="true" type="s:string" />
  </s:schema>
</wsdl:types>
<wsdl:message name="GetQuoteSoapIn">
  <wsdl:part name="parameters" element="tns:GetQuote" />
</wsdl:message>
<wsdl:message name="GetQuoteSoapOut">
  <wsdl:part name="parameters" element="tns:GetQuoteResponse" />
</wsdl:message>
<wsdl:message name="GetQuoteHttpGetIn">
  <wsdl:part name="symbol" type="s:string" />
</wsdl:message>
<wsdl:message name="GetQuoteHttpGetOut">
  <wsdl:part name="Body" element="tns:string" />
</wsdl:message>
<wsdl:message name="GetQuoteHttpPostIn">
  <wsdl:part name="symbol" type="s:string" />
</wsdl:message>
<wsdl:message name="GetQuoteHttpPostOut">
  <wsdl:part name="Body" element="tns:string" />
</wsdl:message>
<wsdl:portType name="StockQuoteSoap">
  <wsdl:operation name="GetQuote">
    <documentation xmlns="http://schemas.xmlsoap.org/wsdl/">Get Stock quote for a company Symbol</documentation>
    <wsdl:input message="tns:GetQuoteSoapIn" />
    <wsdl:output message="tns:GetQuoteSoapOut" />
  </wsdl:operation>
</wsdl:portType>
<wsdl:portType name="StockQuoteHttpGet">
  <wsdl:operation name="GetQuote">
    <documentation xmlns="http://schemas.xmlsoap.org/wsdl/">Get Stock quote for a company Symbol</documentation>
    <wsdl:input message="tns:GetQuoteHttpGetIn" />
    <wsdl:output message="tns:GetQuoteHttpGetOut" />
  </wsdl:operation>
</wsdl:portType>
<wsdl:portType name="StockQuoteHttpPost">
  <wsdl:operation name="GetQuote">
    <documentation xmlns="http://schemas.xmlsoap.org/wsdl/">Get Stock quote for a company Symbol</documentation>
    <wsdl:input message="tns:GetQuoteHttpPostIn" />
    <wsdl:output message="tns:GetQuoteHttpPostOut" />
  </wsdl:operation>
</wsdl:portType>
<wsdl:binding name="StockQuoteSoap" type="tns:StockQuoteSoap">
  <soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="document" />
  <wsdl:operation name="GetQuote">
    <soap:operation soapAction="http://www.webserviceX.NET/GetQuote" style="document" />
    <wsdl:input>
      <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>

```

```
<soap:body use="literal" />
</wsdl:output>
</wsdl:operation>
</wsdl:binding>
<wsdl:binding name="StockQuoteHttpGet" type="tns:StockQuoteHttpGet">
  <http:binding verb="GET" />
  <wsdl:operation name="GetQuote">
    <http:operation location="/GetQuote" />
    <wsdl:input>
      <http:urlEncoded />
    </wsdl:input>
    <wsdl:output>
      <mime:mimeXml part="Body" />
    </wsdl:output>
  </wsdl:operation>
</wsdl:binding>
<wsdl:binding name="StockQuoteHttpPost" type="tns:StockQuoteHttpPost">
  <http:binding verb="POST" />
  <wsdl:operation name="GetQuote">
    <http:operation location="/GetQuote" />
    <wsdl:input>
      <mime:contentType="application/x-www-form-urlencoded" />
    </wsdl:input>
    <wsdl:output>
      <mime:mimeXml part="Body" />
    </wsdl:output>
  </wsdl:operation>
</wsdl:binding>
<wsdl:service name="StockQuote">
  <wsdl:port name="StockQuoteSoap" binding="tns:StockQuoteSoap">
    <soap:address location="http://www.webservicex.net/stockquote.asmx" />
  </wsdl:port>
  <wsdl:port name="StockQuoteHttpGet" binding="tns:StockQuoteHttpGet">
    <http:address location="http://www.webservicex.net/stockquote.asmx" />
  </wsdl:port>
  <wsdl:port name="StockQuoteHttpPost" binding="tns:StockQuoteHttpPost">
    <http:address location="http://www.webservicex.net/stockquote.asmx" />
  </wsdl:port>
</wsdl:service>
</wsdl:definitions>
```

Exercise 3 (WSDL Design) (6 points)

Consider a bus transportation company as one company that provides tourism services as part of the VTA scenario introduced in [1].

The VTA scenario is build around a Virtual Traveling Agency, called VTA for short, which is an end user service providing eTourism services to customers. These services can cover all kinds of information services concerned with tourism information – from information about events and sights in an area to services that support booking of flights, hotels, rental cars, etc. online. The partners of the VTA are integrated via conventional B2B integration. A customer submits her/his request to the VTA service which in turn provides a response to the customer by aggregating and invoking various Web services offered by tourism service providers.

Design a WSDL Web service for one of the services this company provides and explain all its elements. The service must contain at least 2 operations with di_ferent types. The binding to SOAP should be used. All the messages must have complex types.

Exercise 4 (UDDI) (5 points)

Create the XML representations of the UDDI elements `businessEntity`, `businessService`, `bindingTemplate` and `tModel` for the bus transportation company and its bus service. Additionally explain why UDDI can be described as being both a name service and a directory service, mentioning the types of enquiries that can be made.

References

- [1] Stollberg, M., Lausen, H., Keller, U., Zarembo, M., Haller, A., Fensel, D., Kifer, M.: D3.3 v0.1 wsmo use case virtual travel agency. Working Draft D3.3v0.1, WSMO (2004). Available from <http://www.wsmo.org/2004/d3/d3.3/v0.1/>