Systems Integration - Practical exercises Module 1

1 Goal

The goal of the module 1 exercises is to define all the external and internal data formats that we need for the messages in our project.
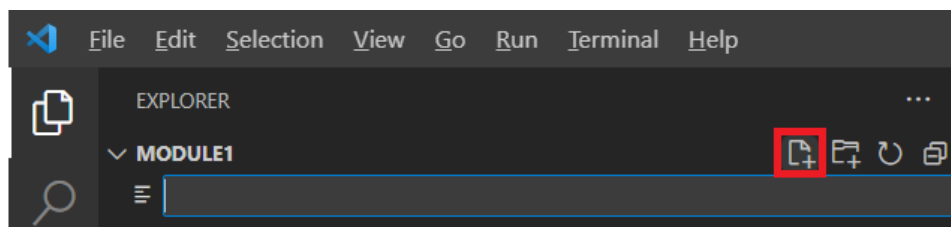
2 Preparation

1. Create a folder somewhere in your filesystem that contains the practical work for this course.
2. Create a subfolder with the name "Module1".
3. Start Visual Studio Code
4. Select File -> Open Folder… and then the created "Module1" folder.

3 General tips

3.1 Creating files in Visual Studio Code

Just click on the File+ icon on folder level and enter the filename:

3.2 JSON schema template

Here you see an example JSON schema representing a simple structure which can be used to solve the JSON schema examples:

Property Name	Type	Required
CommerceRegistrationTemplate	object	X
Name	string	X
DateOfBirth	string, date	

```json
{
    "$schema": "http://json-schema.org/draft-07/schema",
    "$id": "http://technikum-wien.at/mwi/sig/commerce/template",
    "title": "CommerceRegistrationTemplate",
    "type": "object",
    "properties": {
        " CommerceRegistrationTemplate": {

            "type": "object",
            "properties": {
                "Name": {
                    "type": "string"
                },
                "DateOfBirth": {
                    "type": "string",
                    "format": "date"
                }
            },
            "required": [
                "Name"
            ]
        }
    },
    "required": [
        "CommerceRegistrationTemplate"
    ]
}
```

Here you see example JSON code representing a message valid to this schema:

```json
{
    "CommerceRegistrationTemplate": {

        "Name": "Christian Huber",
        "DateOfBirth": "2023-05-01"
    }
}
```

3.3 XSD schema template

Here you see an example XSD schema representing a simple structure which can be used to solve the XSD schema examples:

Property Name	Type	Required
CommerceRegistrationTemplate	complexType	X
Name	string	X
DateOfBirth	date	

```xml
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
    targetNamespace="http://technikum-wien.at/mwi/sig/commerce/template">
    <xs:element name="CommerceRegistrationTemplate">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="Name" type="xs:string"></xs:element>
                <xs:element name="DateOfBirth" type="xs:date" minOccurs="0"></xs:element>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
</xs:schema>
```

Here you see example XML code representing a message valid to this schema.

```xml
<?xml version="1.0" encoding="utf-8"?>
<ns:CommerceRegistrationTemplate xmlns:ns="http://technikum-wien.at/mwi/sig/commerce/template">
    <Name>Christian Huber</Name>
    <DateOfBirth>2023-05-01</DateOfBirth>
</ns:CommerceRegistrationTemplate>
```

3.4 Formatting schemas in Visual Studio Code

Visual Studio Code can format the schema code in the text editor by right clicking into the editor surface and selecting "Format Document".

4 Implementation

4.1 JSON schemas

4.1.1 Commerce registration batch (1 point)

For the processing of the CSV file later in the project we need a JSON schema that represents the data of one line in the CSV format:

Property Name	Type	Required
ApplicantFirstName	string	X
ApplicantLastName	string	X
CommerceRegistrationDescription	string	X
CommerceRegistrationEffectiveDate	string	X
LocationStreetName	string	X
LocationNumber	string	X
LocationCity	string	X
LocationZipCode	string	X

1. Create a file in the solution folder with name "CommerceRegistrationBatch.json".
2. Create the JSON schema in the text editor using the template in the general tips.
3. General settings:
   a. "$id": http://technikum-wien.at/mwi/sig/commerce/commerceregistrationbatch
   b. "title": "CommerceRegistrationBatch"

4.1.2 Commerce registration authority Austria (1 point)

The JSON schema for the commerce registration authority Austria external message format should represent the following data structure:

Property Name	Type	Required
CommerceRegistrationAuthorityAustria	object	X
ApplicantName	string	X
CommerceRegistration	object	X
Profession	string	X
StartDate	string, date	
Location	object	X
StreetName	string	X
Number	string	X
ZipCode	string	X
City	string	X

1. Create a file in the solution folder with name "CommerceRegistrationAuthorityAustria.json".
2. Create the JSON schema in the text editor using the template in the general tips.
3. General settings:
   a. "$id": "http://technikum-wien.at/mwi/sig/commerce/commerceregistrationauthorityaustria"
   b. "title": "CommerceRegistrationAuthorityAustria"

4.1.3   Commerce registration tool (1 point)

The JSON schema for the commerce registration tool external message format should represent the following data structure:

Property Name			Type	Required
CommerceRegistrationTool			object	X
	Applicant		object	X
		FirstName	string	X
		LastName	string	X
	CommerceRegistration		object	X
		Description	string	X
		EffectiveDate	string, date	
	Location		object	X
		StreetName	string	X
		Number	string	X
		ZipCode	string	X
		City	string	X

1. Create a file in the solution folder with name "CommerceRegistrationTool.json".
2. Create the JSON schema in the text editor using the template in the general tips.
3. General settings:
   a. "$id": "http://technikum-wien.at/mwi/sig/commerce/commerceregistrationtool"
   b. "title": "CommerceRegistrationTool"

4.1.4   Commerce registration internal (1 point)

The JSON schema for the commerce registration internal message format should represent the following data structure, which has only some minor differences to the commerce registration tool external format:

Property Name			Type	Required
CommerceRegistrationInternal			object	X
	Applicant		object	X
		FirstName	string	X
		LastName	string	X
	CommerceRegistration		object	X
		Description	string	X
		EffectiveDate	string, date	
	Location		object	X
		StreetName	string	X
		Number	string	X
		ZipCode	string	X
		City	string	X

1. Make a copy of the "CommerceRegistrationTool.json" file and name it "CommerceRegistrationInternal.json".
2. Change the following settings:
   a. "$id": "http://technikum-wien.at/mwi/sig/commerce/commerceregistrationinternal"
   b. "title": "CommerceRegistrationInternal"
   c. Root property name: "CommerceRegistrationInternal"

4.2 XSD schema

4.2.1   Commerce registration authority Vienna (1 point)

The XSD schema for the commerce registration authority Vienna external message format should represent the following data structure:

Property Name	Type	Required
CommerceRegistrationAuthorityVienna	complexType	X
ApplicantName	string	X
CommerceRegistration	complexType	X
Profession	string	X
StartDate	date	
Location	complexType	X
StreetName	string	X
Number	string	X
ZipCode	string	X
City	string	X

1. Create a file in the solution folder with name "CommerceRegistrationAuthorityVienna.xsd".
2. Create the XSD schema in the text editor using the template in the general tips.
3. General settings:
    a. targetNamespace="http://technikum-wien.at/mwi/sig/commerce/commerceregistrationauthorityvienna"