FHIR DevDays 2017 HL7® FHIR®

EXERCISES*

Track: **Developers**
Tutorial name: **.NET API for Beginners**
Speaker name:: Mirjam Baltus

The speaker will be in the hands-on area to answer your questions!

During the hands-on session of the .NET API for beginners tutorial we will explore working with several of the parts of the FHIR library, to get up and running with FHIR. We will mainly focus on client side aspects.

In the following exercises, each step builds on the previous steps, so you should try them out in sequence:

1. Getting familiar with the model
2. Using the `FhirClient` to communicate with a FHIR server
3. Performing a search, and walking through the results
4. Setting up a transaction

First, open Visual Studio and set up a .Net solution to your liking. Then go to the NuGet Package Manager and browse for the FHIR library. Make sure to install the Hl7.Fhir.STU3 library, which is the most recent version.

1. Getting familiar with the model

- Add a `using` statement for `Hl7.Fhir.Model`
- Create a new instance of `Patient`, and fill it with some details, like a name and an identifier. See http://www.hl7.org/fhir/patient.html for the specification of the Patient resource.
- Can you add multiple names?
- Try to add an extension for place of birth to the Patient. See http://www.hl7.org/fhir/patient-profiles.html for the definition of this, and other extensions defined for Patient.

2. Using the FhirClient to communicate with a FHIR server

- Add a `using` statement for `Hl7.Fhir.Rest`.
- Create a new instance of `FhirClient`, and set the endpoint to `http://vonk.furore.com`. For more test servers, see http://wiki.hl7.org/index.php?title=Publicly_Available_FHIR_Servers_for_testing.

Read

- Read a single resource from the server with id “`example`”, e.g. an Observation, and display the text part.
- Extra: display information from the other resource fields.

Create/Update

- With the `FhirClient.Create` method, send the `Patient` from exercise 1 to the server.
- Display/print the response, specifically the technical id that was generated by the server.
- Extra: set up error handling.
- Extra: change some details of the resource, and use the `Update` method to update the resource on the server

Delete

- Delete the resource you’ve created previously, with the `FhirClient.Delete` method. (please take care not to delete the example resources!)

3. Performing a search, and walking through the results

Search for a resource type

- Search the server for all Resources of a specific type, e.g. Observation, and display the id for each resource.
- Extra: limit the number of results to 5.
- Extra 2: have the UI allow the user to display the next page of search results.

Search for specific resources

- Create `SearchParams` with parameters to limit your search results.
  Tip: Look at the bottom of the page for your chosen resource type in the specification to see which parameters can be used for that type.
- Display the id for each matching resource, plus the value that matched your search parameter, e.g. the code.
- Extra: include another resource type, e.g. if you searched on Observation, include the subject resources. Display information, for example the text parts of the included resources.

4. Setting up a transaction

- Build a transaction Bundle using the `TransactionBuilder`.
- Add at least two transactions to the Bundle, using the `TransactionBuilder` methods. For example, add one create interaction and one read interaction.
- Send the transaction to the server, using the `FhirClient.Transaction` method.
- Bonus: Display the results of the transaction for each interaction you added to it.

Have fun, and remember to ask for help if you get stuck !

*This exercise will not be evaluated and no prizes are attached to any result. The exercise can only be used at HL7 FHIR DevDays 2017.