# Exercise Terminology Services: Taking a Closer Look at Applying $closure/Let’s Build!

*Track lead:* Rob Hausam MD

During the Let’s Build! Session, you will learn and gain some hands on familiarity with the ConceptMap $closure operation and explore its use in a simple demo application and consider how you may want to use this potentially quite powerful capability in your own FHIR applications.

After completing this tutorial, you will be able to:

* Understand how to invoke the ConceptMap $closure operation
* Consider how the $closure operation may be used in FHIR applications (including your own)

## Review the Maintaining a Closure Table documentation in the FHIR spec

<http://build.fhir.org/terminology-service.html#closure>

## Also review the Ontoserver $closure Closure Examples documentation, if desired

<https://documenter.getpostman.com/view/784165/ontoserver-51-example-fhir-terminology-requests/RW1hhG1S?version=latest#67ce94c0-ebd8-93fb-6bea-b5b315bc2d98>

## Explore the $closure operation with the Postman REST client

1. Initialize the closure table

POST <https://r4.ontoserver.csiro.au/fhir/$closure>  
(this will be the same base command for all steps)  
  
{

"resourceType" : "Parameters",

"parameter" : [{

"name" : "name",

"valueString" : "test-closure"

}

]

}

1. Add an initial concept  
     
   {

"resourceType" : "Parameters",

"parameter" : [{

"name" : "name",

"valueString" : "test-closure"

},

{

"name" : "concept",

"valueCoding" : {

"system" : "http://snomed.info/sct",

"code" : "22298006"

}

}

]

}

1. Add a second (related) concept  
     
   {

"resourceType" : "Parameters",

"parameter" : [{

"name" : "name",

"valueString" : "test-closure"

},

{

"name" : "concept",

"valueCoding" : {

"system" : "http://snomed.info/sct",

"code" : "56265001"

}

}

]

}

1. Add multiple additional concepts (as desired)  
     
   {

"resourceType" : "Parameters",

"parameter" : [{

"name" : "name",

"valueString" : "test-closure"

},

{

"name" : "concept",

"valueCoding" : {

"system" : "http://snomed.info/sct",

"code" : "128599005"

}

}

]  
}

1. Reset to version 0 (return all rows)  
     
   {

"resourceType" : "Parameters",

"parameter" : [{

"name" : "name",

"valueString" : "test-closure"

},

{

"name" : "version",

"valueId" : "0"

}

]

}

## Requirements

Postman REST client  
<https://www.getpostman.com/>

Access the Ontoserver R4 FHIR Server (and other servers, as available)

<https://r4.ontoserver.csiro.au/fhir>

## Build out a simple starter Java FHIR client to apply the $closure operation output

## Demonstrates how to invoke the $closure operation and retrieve its output with a simple client application (using the HAPI FHIR Java client framework). The starter application prints the returned mapping rows. You can further build out this client to store the operation results in a transitive closure table structure (text file, relational database table, in memory, etc.).

## Starter project source code is available in the GitHub repository below.

## Execute the starter client (from the project directory) with the command: java -jar dist/FHIRJavaClientClosureTest.jar [code1] [code2] [code…]

## Command examples

## Initialize the client closure table:

java -jar dist/FHIRJavaClientClosureTest.jar

## Add a single code:

## (adding multiple codes presently isn’t working)

java -jar dist/FHIRJavaClientClosureTest.jar 70211005

## Requirements

Java >= 1.8

(optional) NetBeans IDE – if you want to launch the project (updated to [Apache NetBeans IDE 11.2](https://github.com/rhausam/closure-test/commit/24684d1a2bb9779fcfc74e350c84b49b2b83076a))

Access the Ontoserver R4 FHIR Server (and other servers, as available)

<https://r4.ontoserver.csiro.au/fhir>

## (Optional) Begin coding your own application using the $closure operation

The tutorial and code for this exercise can be found at:

<https://github.com/rhausam/closure-test.git>

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