Cooking with CQL Q&As

Session 63 - Thursday, April 28, 2022

General CQL

**Q:** In a JAVA implementation, when a *\_retrieve\_* expression is performed on a Fast Healthcare Interoperability Resources® (FHIR®) encounter, does the *\_retrieve\_* expression return a FHIR object, or is the object transformed into a tuple, and is the alias a tuple type?

**A:** In the JAVA implementation, data is physically represented as JAVA objects. The core engine defines interfaces for dealing with data access, terminology, and library loading, and the implementation of the interface is responsible for providing the actual data in response to a \_retrieve\_. In this case, a HAPI FHIR client is used, so the object is an instance of a HAPI FHIR Encounter class. Clinical Quality Language (CQL) supports both class and tuple types (aka anonymous types). The result of a \_retrieve\_ is typed as a class, but class types are generally compatible with tuple types with the same structure, so most operations can be performed on either. One exception is the type testing operations, where \_is\_ and \_as\_ will only work with the actual type. Note also that using the \_return\_ clause to shape the result of a query will typically result in a tuple type as well.

General CQL

**Q:** In the Clinical Quality Language 1.5 library, if you use double quotes for identifiers, can you pull in a value set using double quotes?

**A:** Yes, a value set can be referenced using a double-quoted identifier, for example, the "Office Visit" value set.

|  |  |
| --- | --- |
| define "Qualifying Encounters": | |
|  | ( [Encounter: "Office Visit"] | |
|  | union [Encounter: "Preventive Care Services - Established Office Visit, 18 and Up"] | |
|  | union [Encounter: "Preventive Care Services-Initial Office Visit, 18 and Up"] | |
|  | union [Encounter: "Home Healthcare Services"] | |
|  | union [Encounter: "Telephone Visits"] | |
|  | union [Encounter: "Online Assessments"] | |
|  | ) ValidEncounter | |
|  | where ValidEncounter.status = 'finished' | |
|  | and ValidEncounter.period during "Measurement Period" | |

Functions in CQL

**Q:** In the Clinical Quality Language, library 1.5, is the AgeInYearsAt function in this expression example going to return yes or no without having operator access present?

|  |  |  |
| --- | --- | --- |
| define "Initial Population": | | |
|  |  | AgeInYearsAt(date from start of "Measurement Period") in Interval[23, 64) | |
|  |  | and Patient.gender = 'female' | |
|  |  | and exists "Qualifying Encounters" | |

**A:** The expression returns a Boolean true (yes) or Boolean false (no) because it is part of the “in” operator. Additionally within this code, the bracket - [ - represents a closed boundary [23 indicating that 23 is included; whereas, the parenthesis - ) - indicates an open boundary 64) indicating that up to, but excluding 64. The datatype of the interval (in this case an integer) and the precision of the expression as integer, number, year, day, minute, etc. (in this case the AgeInYearsAt function returns the number of years as an integer) impacts the granularity or how close a value can approach the open boundary.