

---

# **Clinical Quality Language (CQL) Style Guide**

---

[Version 4.0 – Last Revised: May 2020](#)

The Centers for Medicare & Medicaid Services (CMS) has contracted with Mathematica and its partners, including The Joint Commission, to develop this style guide to support the electronic specification and maintenance of electronic clinical quality measures (eCQMs) that eligible professionals, eligible clinicians, eligible hospitals, and critical access hospitals can potentially use for reporting using certified electronic health records (EHRs) for CMS quality programs. Mathematica and its partners developed this document under two Measure and Instrument Development and Support (MIDS) indefinite delivery, indefinite quantity master contract vehicle task order contracts, Electronic Clinical Quality Measures Development and Maintenance for Eligible Clinicians contract (CMS contract #75FCMC18D0032, Task Order #75FCMC19F0004) and the Inpatient and Outpatient Measure Maintenance contract (CMS contract #75FCMC18D0032, Task Order #75FCMC19F0003).

# CONTENTS

---

1.	PREREQUISITE .....	1
2.	INTRODUCTION.....	2
3.	STANDARDS – LIBRARIES .....	2
	A. Best practices for using CQL libraries .....	3
	B. Standards for naming a local CQL library .....	3
4.	STANDARDS – DEFINITIONS .....	4
	A. Best practices for naming CQL definitions .....	4
	B. Standards for naming definitions across measures .....	6
5.	STANDARDS – ALIASES OR ARGUMENT NAMES .....	6
	A. Best practices for using CQL aliases and argument names .....	6
	B. Standards for naming aliases across measures .....	10
6.	FUNCTIONS .....	11
	A. Best practices and standards for naming new CQL functions .....	11
	B. Selecting functions .....	11
7.	OTHER CQL BEST PRACTICES .....	13
	A. Population criteria .....	13
	B. Additional timing phrases .....	14
	C. Operator precedence.....	14
	D. Direct Reference Codes .....	15
	VERSION HISTORY .....	16
	APPENDIX A: THE ORDER OF OPERATOR PRECEDENCE IN CQL FROM HIGHEST TO LOWEST .....	A.1
	APPENDIX B: CASE TYPE DEFINITIONS .....	B.1

## 1. PREREQUISITE

---

Clinical Quality Language (CQL)<sup>1</sup> is an HL7 standard developed as part of the Clinical Quality Framework (CQF) initiative. CQL is intended to promote standardization and harmonization; it is also intended to be clinically focused, author friendly, and human readable.<sup>2</sup> Users of this style guide are expected to have a baseline knowledge of CQL. Please refer to the following links for more information on CQL:

- [eCQI Resource Center](#)
- [CQL Formatting and Usage Wiki](#)

Measure and Instrument Development and Support (MIDS) contractors, in coordination with the Centers for Medicare & Medicaid Services (CMS), created this document using guidelines from the [CQL Formatting and Usage Wiki](#) to promote consistency and reusability in measure specification within measures included in federal reporting programs. Measure developers and stakeholders contributed to the development of the standards included in this guide.

---

<sup>1</sup> The guidance in this document is based on CQL STU 1.4, located at: <https://cql.hl7.org/01-introduction.html>.

<sup>2</sup> Raw CQL files are human readable, but there is also an HTML version in the eCQM package exported from MAT. HTML human readable provides a view consistent with the style provided with earlier eCQMs based on Quality Data Model (QDM) logic.

---

## 2. INTRODUCTION

---

The purpose of the style guide is to standardize expression of measure concepts across electronic clinical quality measures (eCQMs) and define a uniform “look and feel” for eCQM logic using CQL. The style guide focuses on a set of common best practices that have been implemented across CQL-based eCQMs in CMS reporting programs. It also promotes the use of consistent language within the framework of CQL, including libraries, aliases, definitions, and functions, and provides guidance on other conventions, such as operator precedence. Measure stewards or developers who are developing or specifying eCQMs for potential inclusion in CMS reporting programs should align with these best practices.

## 3. STANDARDS – LIBRARIES

---

Libraries, the basic units of sharing CQL, consist of a foundation of CQL statements used within an eCQM. Every eCQM has at least one CQL library.

The measure name and measure library name do *not* have to be the same. When naming a new measure CQL library, adhere to the following standards:

- **DO** use PascalCase<sup>3</sup> (capitalizing the first letter of every word, without spaces between words).

Example CQL library name: **AdultOutpatientEncounters**

- **DO** use names that are short, descriptive, and easy to read, and accurately inform the reader of the contents of the library.

Example CQL library name: **Hospice**

- **DO NOT** start the library name with a special character,<sup>4</sup> number, or underscore.  
*Example:* \*AdultEncounters

- **DO NOT** use spaces or special characters<sup>4</sup> within the library name.

- Example: Adult+PediatricEncounter

eCQMs can also use CQL expressions stored in shared libraries. Shared libraries enable similar logic to be used repeatedly across multiple measures, thus improving standardization and harmonization across eCQMs. Shared libraries can exist at the *local* or *global* level.

- **Local library:** CQL libraries can be shared among several eCQMs or a specific group of eCQMs.

---

<sup>3</sup> See Appendix B for a complete list of case type definitions.

<sup>4</sup> Special characters are symbols such as +, -, \*, and /.

The local CQL library ‘**Hospice**’ is used in a CQL definition statement below. The ‘**Hospice**’ library contains one CQL definition named “Has Hospice”.

Example CQL library: **Hospice**

Denominator Exclusions

```
Hospice."Has Hospice"  
  or exists "Malignant Neoplasm"  
  or exists "Total Colectomy Performed"
```

- **Global library**: A global library is a shared CQL library created in the Measure Authoring Tool (MAT) that contains CQL expressions for all measure developers to use in the specification of an eCQM. The common library, **MATGlobalCommonFunctions**, is accessible to all MAT users and is maintained and updated by Mathematica. The common library is updated with each published update to the Quality Data Model (QDM) and CQL standards incorporated into the MAT. Using this common library helps reduce duplication and maintains consistency across measure specifications. The version of the common library to be used with each measure update is specified on the eCQI Resource Center.

The CQL library ‘**Global**’ is used in the definition below. Note: The common library definition “**Inpatient Encounter**” is recommended for use in hospital measures.

Example CQL Library Name: **Global**

```
Global."Inpatient Encounter"  
  
  ["Encounter, Performed": "Encounter  
  Inpatient"]EncounterInpatient  
  where "LengthInDays"(EncounterInpatient.relevantPeriod)<= 120  
    and EncounterInpatient.relevantPeriod ends during  
    "Measurement Period"
```

## **A. Best practices for using CQL libraries**

A local library should be used when similar functions or definitions are used across measures; it is required if *five or more* measures are impacted.

## **B. Standards for naming a local CQL library**<sup>5</sup>

When naming a new local CQL library, adhere to established best practices for naming libraries referenced above.

---

<sup>5</sup> A library can also have an alias. For standards on naming aliases, refer to Section 5.

## 4. STANDARDS – DEFINITIONS

---

Definitions are concise logical CQL expressions that provide detail about the meaning of measure concepts. Definitions are also referenced in the measure population logic. They should be reused and referenced within other CQL expressions, whenever appropriate.

### A. Best practices for naming CQL definitions

When naming definitions, use the following best practices:

- **DO** use Title Case<sup>3</sup> (capitalizing principal words only; not articles, conjunctions, or prepositions, and allow for spaces between words).

Example CQL definition name:

- **Baseline DEXA Scan Two Years Prior to the Start of or Less than Three Months After the Start of ADT**

- **DO** create names that are easy to read, describe the contents of the logical expression, and provide context as to what makes the statement unique and clinically meaningful.

Example CQL definition names:

- **End Stage Renal Disease Encounter**
- **Estimated Gestational Age Captured on the Day of Delivery**
- **Initial Congestive Heart Failure Functional Assessment**

- **DO** use only abbreviations or acronyms that are consistent with terminology used in the measure’s narrative.

Example CQL definition names:

- **Encounter with Prior or Present Diagnosis of Atrial Fibrillation or VTE**
- **ASCVD Procedure before End of Measurement Period**

- **DO** create definition names that are clear and indicate the return<sup>6</sup> (a yes/no should be named like a question [for example, “Is” or “Has”], a list of encounters should be named “Encounters...”, a list of procedures should be named “Procedures.....”, and so on).

Example CQL definition names:

- **Has** Initial Major Depression Diagnosis
- **Has** Complete Hearing Screening
- **Has** Continuation of Treatments
- **Has** Liver Disease
- Qualifying **Encounters**
- **Procedures** Identifying Sexual Activity

---

<sup>6</sup> The MAT will display the “Return Type” in the definition form (when there are no CQL errors).

o Ischemic Stroke **Encounters** with Discharge Status

- **DO NOT** name a definition the same as a value set name.<sup>7</sup>

*Example:*

#### **Cognitive Assessment**

```
["Intervention, Performed": "Cognitive Assessment"]
```

- **DO NOT** name a definition the same as a CQL operator.<sup>8</sup>
- **DO NOT** use special characters<sup>4</sup> in definition names.

*Example:* %Patients with Test Result

Below are additional examples of CQL definitions (used in logic) that use descriptive and clinically meaningful text, title case, and appropriate spacing.

Example CQL definition:

#### **Newborn Fed Breast Milk Only Since Birth**

```
"Single Live Birth Encounter With Gestational Age 37 Weeks or More"  
QualifyingEncounter  
  with ["Substance, Administered": "Breast Milk"]  
  BreastMilkFeeding  
    such that BreastMilkFeeding.relevantPeriod starts during  
    QualifyingEncounter.relevantPeriod  
  without ["Substance, Administered": "Dietary Intake Other than  
  Breast Milk"] OtherFeeding  
    such that OtherFeeding.relevantPeriod starts during  
    QualifyingEncounter.relevantPeriod
```

Example CQL definition:

#### **Has Most Recent HbA1c Without Result**

```
Last(["Laboratory Test, Performed": "HbA1c Laboratory Test"]  
HbA1cNoResult  
  where HbA1cNoResult.relevantDatetime during "Measurement  
  Period"  
  sort by resultDatetime  
) .result is null
```

---

<sup>7</sup> In CQL, a definition name and value set name cannot share the same identifier.

<sup>8</sup> Please refer to the CQL specification for a full list of operators: <https://cql.hl7.org/09-b-cqlreference.html>.

Use Table 1 below as a guide for naming definitions. The left column lists examples of definition names. The right column provides example alternatives that offer improved description and readability.

**Table 1. Making good definition names better (more clear and concise)**

Good definition name	Better definition name
Anticoagulant Not Given at Discharge	Reason for Not Giving Anticoagulant at Discharge
In Demographic	Single Live Birth Encounter with Gestational Age 37 Weeks or More
Lab Test with Result	Most Recent Elevated HbA1c with Result

## B. Standards for naming definitions across measures

Use Table 2 as a guide for naming definitions that use common concepts across measures. The left column presents the concept and the right column presents the recommended standard naming convention.

**Table 2. Standard definition names for use across measures**

Concept	Recommended definition name
Hospice Exclusions Exclusions for Hospice	Has Hospice
Encounters <i>or</i> Eligible Encounters <i>or</i> Valid Encounters	Qualifying Encounters

## 5. STANDARDS – ALIASES OR ARGUMENT NAMES

Aliases, or argument names, are identifiers that refer to individual CQL expressions or libraries. Aliases should correlate clearly to their source and can be reused to avoid restating key expressions. This allows for a more fluid, concise, and standardized CQL expression. Alias names should maintain meaning and uniformity within and across measures. Authors can develop aliases for libraries, functions, and definitions.

### A. Best practices for using CQL aliases and argument names

When naming aliases and argument names, use the following standards:

- **DO** use PascalCase<sup>9</sup> (capitalizing the first letter of every word, without spaces between words).

Example CQL aliases:

- ["Medication, Order": "High Risk Medications for the Elderly"]  
**HighRiskMedicationOrder**

<sup>9</sup> See Appendix B for a complete list of case type definitions.

- (["Intervention, Order": "Referral to Alternative Provider / Primary Care Provider"] **Referral**)
- **DO** use names that are short, descriptive, easy to read, and provide an accurate representation of the identified concept.

Example CQL aliases:

- ["Diagnosis": "Allergy to Eggs"] **EggAllergy**
- ["Diagnosis": "Malignant Neoplasm of Colon"] **ColorectalCancer**
- ["Procedure, Performed": "General or Neuraxial Anesthesia"] **AnesthesiaProcedure**
- ["Procedure, Performed": "Influenza Vaccination"] **FluVaccination**
- **DO** create alias names that are clinically focused.

Example CQL aliases:

- ["Physical Exam, Performed": "Diastolic blood pressure"] **DiastolicBP**
- ["Medication, Order": "Beta Blocker Therapy for LVSD"] **BetaBlockerOrdered**
- (["Encounter, Performed": "Alcohol and Drug Dependence Treatment"]  
union ["Encounter, Performed": "Psych Visit - Psychotherapy"]) **InitiatedTreatment**
- **DO** use only abbreviations or acronyms that are consistent with terminology used in the measure's narrative sections.

Example CQL aliases (note: in each of these examples, the acronym is defined in the narrative):

- ["Assessment, Performed": "VR12 Physical Component T Score"] **VR12PhysicalScore**
- "No VTE Prophylaxis Medication Administered or Ordered" **NoVTEMedication**
- **DO NOT** reuse aliases, unless a single alias carries the same content and scope as its original definition when reused within a measure.

See below for examples to avoid. The alias **HeartRate** is reused in two different definitions within the same measure, each with a different scope.

*Example:*

```
First(["Encounter, Performed": "Heart Rate Visit"] HeartRate
with ["Diagnosis": "Essential Hypertension"] Hypertension
such that HeartRate.relevantPeriod overlaps
Hypertension.prevalencePeriod)
```

*Example:*

```
Last(["Physical Exam, Performed": "Heart Rate Exam"] HeartRate
with "Initial Blood Pressure Visit" InitialEncounter
such that HeartRate.relevantDatetime during
InitialEncounter.relevantPeriod
where HeartRate.result is not null
sort by start of relevantDatetime
)
```

- **DO NOT** give an alias the same name as the definition name.

*Example:*

- Lower Back Procedure  
["Procedure, Performed": "Lumbar Surgical Procedures"]  
**LowerBackProcedure**  
where LowerBackProcedure.relevantDateTime overlaps  
"Measurement Period"

- **DO NOT** use an alias if the definition statement does not require additional logic.

*Example:*

- Blood Transfusion  
["Substance, Order": "Blood Administration"] **BloodTransfusion**

Use Table 3 as a guide for naming aliases. The left column lists examples of alias names that measure developers should avoid. The example alternatives in the right column offer improved description and readability.

**Table 3. Making aliases better (easy to read and clinically focused)**

Alias names to avoid	Better alias names
D or Dx or Diagnosis	HeartFailure Pregnancy Asthma Bradycardia
Med Medication	BetaBlocker Antidepressant
P or Proc or Procedure	CardiacSurgery Dialysis
Lab or LabTest	HepBAntigenTest MumpsTiter PregnancyTest
E or Enc	Encounter* <i>(use with caution if referring to several types of encounters in measure)</i> InpatientEncounter HeartFailureEncounter Psychotherapy
["Physical Exam, Performed": "Heart Rate"] <b>Exam</b>	HeartRate
["Diagnostic Study, Performed": "Ejection Fraction"] <b>Study</b>	EjectionFraction

Use Table 4 as a guide for improving alias names even further. The left column lists examples of aliases. The right column includes example alternatives that offer improved description and clarity.

**Table 4. Making good alias names better (add description)**

Good alias name	Better alias name
HeartRate	FirstHeartRate
AntithromboticNotGiven	NoAntithrombotic
VisualExam	VisualFootExam
Fracture	LowerBodyFracture
THAProcedure	TotalHip
HeightExam	Height

Use Table 5 as a guide for creating distinction between two aliases with similar characteristics within a measure (examples in left column) by adding specificity (examples in right column).

**Table 5. Creating distinction between aliases with similar concepts (add specificity)**

Similar concepts	Similar alias names with specificity
Heart failure encounter <i>and</i> Heart failure diagnosis	HeartFailureEncounter <i>and</i> HeartFailureDiagnosis

## B. Standards for naming aliases across measures

Use Table 6 as a guide for naming aliases that use common concepts across measures. The left column presents the alias concept, and the right column presents the recommended standard alias naming convention.

**Table 6. Standard alias names for use across measures**

Concept	Standardized alias
Patient date of birth	BirthDate
Hospice discharge	DischargeHospice
Hospice care order	HospiceOrder
Hospice intervention performed	HospicePerformed

Refer to the following examples of standard alias names.

- The alias **BirthDate** is used consistently in different CQL definition statements, and in multiple places within and across measures.

- Example CQL alias: BirthDate

```
exists ( ["Patient Characteristic Birthdate": "Birth date"]
BirthDate
  where
    Global."CalendarAgeInYearsAt"(BirthDate.birthDatetime, start
of "Measurement Period")>= 18
)
```

- Example CQL alias: DischargeHospice, HospiceOrder, HospicePerformed

```
exists ( ["Encounter, Performed": "Encounter Inpatient"]
DischargeHospice
  where ( DischargeHospice.dischargeDisposition ~ "Discharge to
home for hospice care (procedure)"
or DischargeHospice.dischargeDisposition ~ "Discharge to
healthcare facility for hospice care (procedure)"
)
  and DischargeHospice.relevantPeriod ends during
"Measurement Period"
)
or exists ( ["Intervention, Order": "Hospice care ambulatory"]
HospiceOrder
  where HospiceOrder.authorDatetime during "Measurement Period"
)
or exists ( ["Intervention, Performed": "Hospice care
ambulatory"] HospicePerformed
  where HospicePerformed.relevantPeriod overlaps "Measurement
Period"
)
```

## 6. FUNCTIONS

---

A function is a named CQL expression that can perform any variety of calculations. Before creating new functions, measure developers should review and—to the extent possible and applicable—use the predefined functions available in the MAT or in the shared “Global” common library.

### A. Best practices and standards for naming new CQL functions

New function names should be short, descriptive, and easy to read, and should provide an accurate representation of the identified concept. When naming functions, use the following standards:

- **DO** use PascalCase.<sup>10, 11</sup>

Example CQL function name:

- **EmergencyDepartmentArrivalTime()**

- **DO** use spaces after commas to separate arguments, always.

Example CQL function:

- **Global.CalendarAgeInYearsAt**(BirthDateTime DateTime, AsOf DateTime)

- **DO NOT** name a function the same as a MAT pre-defined function.
  - Process step: Please review the predefined operators available in the MAT<sup>12</sup> to ensure the name of your function is not the same.

### B. Selecting functions

To differentiate among similar functions, choose from the pre-defined list in the MAT or from the “Global” common library. Select the one that’s most appropriate to meet the measure’s intent. See the following examples of preferred functions from the “Global” common library.

- **Age functions:**
  - Example CQL function: **Global.CalendarAgeInYearsAt()**
  - Calculates age using **only birth date**, not time.
  - Use this function for measures pertaining to *adult* patients, or as applicable.
  - For example, to determine all adult patients aged 65 or older at the beginning of the Measurement Period.

---

<sup>10</sup> See Appendix B for complete definitions of case types.

<sup>11</sup> Use of double quotes in invoking is not required and should be discouraged.

<sup>12</sup> See the MAT user guide for list of operators:  
<https://www.emeasuretool.cms.gov/training-resources>.

Example CQL function (used in logic):

```
exists ( ["Patient Characteristic Birthdate": "Birth date"]
BirthDate
  where Global.CalendarAgeInYearsAt(Birthdate.birthDatetime,
  start of "Measurement Period")>= 65
```

- **Length-of-stay functions** (generally used for hospital measures):

- Example CQL function: **Global.LengthInDays()**

- **LengthInDays()** calculates the difference in calendar days between the start and end of the given interval.

```
Global.LengthInDays(Value Interval<DateTime>)
```

```
difference in days between start of Value and end of
Value
```

- This function can be used to calculate length of hospital stay for an inpatient encounter from admission to discharge. See example below.

- Example CQL function (used in logic): **Global.LengthInDays()**

```
["Encounter, Performed": "Encounter Inpatient"]
EncounterInpatient
  where
    LengthInDays(EncounterInpatient.relevantPeriod) <=
    120
    and EncounterInpatient.relevantPeriod ends
    during "Measurement Period"
```

- Example CQL function: **Global.HospitalizationLengthOfStay()**

- Returns the length of stay in days (i.e., the number of days between admission and discharge) for the given encounter, or from the admission of any immediately prior emergency department visit to the discharge of the encounter.

```
Global.HospitalizationLengthOfStay(Encounter "Encounter,
Performed")
```

```
LengthInDays("Hospitalization"(Encounter))
```

- Example CQL function (used in logic):

```
Definition name: Encounter Less Than Two Days
```

```
TJC."Ischemic Stroke Encounter" IschemicStrokeEncounter
```

```

where
Global.HospitalizationLengthOfStay (IschemicStrokeEncounter) <
2

```

- Example CQL function: **Global.Hospitalization()**
  - This function returns the total interval for admission to discharge for the given encounter, or for the admission of any immediately prior emergency department visit to the discharge of the given encounter.

**Global.Hospitalization**(Encounter "Encounter, Performed")

```

Definition name: Encounter Visit
let EDVisit: Last(["Encounter, Performed": "Emergency
Department Visit"]) LastED
    where LastED.relevantPeriod ends 1 hour or less on or
    before start of Visit.relevantPeriod
    sort by
    end of relevantPeriod
)
return Interval[Coalesce(start of EDVisit.relevantPeriod,
start of Visit.relevantPeriod),
end of Visit.relevantPeriod]

```

- Example CQL function (used in logic): **Global.Hospitalization()**

```

"Ischemic Stroke Encounter" IschemicStrokeEncounter
with "Intervention Comfort Measures" ComfortMeasure
such that Coalesce(start of ComfortMeasure.relevantPeriod,
ComfortMeasure.authorDatetime) during
Global."Hospitalization" (IschemicStrokeEncounter)

```

## 7. OTHER CQL BEST PRACTICES

---

### A. Population criteria

When using population criteria, be descriptive and specific, making sure names are descriptive and easy to read. Below is an example of how to improve naming of population criteria.

Current population criteria	Improved population criteria
<b>Initial Population</b> 'In Demographic'	<b>Initial Population</b> 'Single Live Birth Encounter with Gestational Age 37 Weeks or More'

When the denominator population criteria are equivalent to the initial population criteria, state '**Initial Population**' for the Denominator, as illustrated below:

Example CQL: Initial Population = Denominator

**Initial Population**

---

```

exists ( ["Patient Characteristic Birthdate":"Birth date"]
BirthDate
      where
Global."CalendarAgeInYearsAt"(BirthDate.birthDatetime,
start of "Measurement Period")>= 65
)
and exists "Qualifying Encounters"

```

## Denominator

"Initial Population"

## B. Additional timing phrases

CQL supports precision-based date/time comparisons. Be sure to consider whether day or time should be considered in timing phrases. Additional timing phrases, besides **day of** expressions, may be needed when making a statement such as: **A starts/ends before/after or concurrent with start of B**. However, when anchoring a timing to the parameter “Measurement Period,” it is not necessary to use **day of**.

Example CQL timing phrase: ends 1 day after day of start of

A supplementary timing constraint is added to ensure the timing of the relevant period for the AnesthesiaProcedure ends 1 day after the start of relevant period of the QualifyingEncounter.

Definition name: No VTE Prophylaxis Medication Due to Medical Reason on Day of or Day after Procedure

```

from
  "Encounter with Age Range and without VTE Diagnosis or
  Obstetrical Conditions" QualifyingEncounter,
  ["Procedure, Performed": "General or Neuraxial Anesthesia"]
  AnesthesiaProcedure,
  "No VTE Prophylaxis Medication Administered or Ordered"
  NoVTEMedication
  where NoVTEMedication.negationRationale in "Medical Reason"
        and AnesthesiaProcedure.relevantPeriod ends 1 day after
        day of start of QualifyingEncounter.relevantPeriod
        and NoVTEMedication.authorDatetime during
        VTEICU."CalendarDayOfOrDayAfter"(end of
        AnesthesiaProcedure.relevantPeriod)
  return QualifyingEncounter

```

## C. Operator precedence

Precedence in CQL expressions is determined by the order of appearance in the expression, left to right. To ensure consistent and predictable behavior in the order of operations within CQL expressions, use parentheses around a grouping to enforce higher precedence. Refer to the table in Appendix A for more detail on operator precedence.

Refer to the example below, which uses parentheses to promote operator precedence around **exists 'Right Mastectomy'** and **exists 'Left Mastectomy'** to make the grouping clear.

Example CQL operator precedence:

Denominator Exclusions

```
Hospice."Has Hospice"  
  or ( Count("Unilateral Mastectomy  
  Procedure")= 2 )  
  or ( exists "Right Mastectomy"  
        and exists "Left Mastectomy"  
  )  
  or exists "History Bilateral Mastectomy"  
  or exists "Bilateral Mastectomy Procedure"
```

## D. Direct Reference Codes

Direct Reference Codes (DRCs) are single terminology codes that can be referenced directly within CQL logic, instead of creating a value set for a single code. DRCs are recommended for all single-use LOINC codes, and may be used for other single-use terminology codes. When included in a definition, the DRC is incorporated in the CQL syntax through use of the code descriptor. The DRC's specific code and corresponding descriptor will always be called out in the Terminology section of the human readable and may be referenced in the Data Criteria section, if used as part of a QDM element and not just as an attribute of a previously defined QDM element.

Example of Direct Reference Code (used in CQL logic):

### Denominator Exclusions

```
"Initial Population" LiveBirthEncounter  
  where LiveBirthEncounter.dischargeDisposition ~ "Patient  
  deceased during stay (discharge status = dead) (finding)"  
  and not exists "Has Complete Hearing Screening"
```

Example of Direct Reference Code (used in CQL logic):

### Rotavirus Numerator Inclusion Conditions

```
( ( ["Diagnosis": "Anaphylaxis due to rotavirus vaccine (disorder)"]  
  union ["Diagnosis": "Severe Combined Immunodeficiency"]  
  union ["Diagnosis": "Intussusception"] ) RotavirusConditions  
  with ["Patient Characteristic Birthdate": "Birth date"]  
  BirthDate  
  such that  
  Global."CalendarAgeInDaysAt"(BirthDate.birthDatetime,  
  start of RotavirusConditions.prevalencePeriod)<= 730)
```

## VERSION HISTORY

---

Version	Date	Description of Change
2.0	August 21, 2018	Initial publication
3.0	May 2019	Removed references related to retired QDM logic
		Updated examples of logic as needed to reflect most recent use
		Added additional examples of logic in each section to provide context
		Added clarifying language to content in each section
		Updated language/content to align with standards changes QDM 5.4 and CQL 1.3
		Removed reference to Keyword-Distinct
4.0	May 2020	Updated examples using birthdate to reflect the addition of birth date, a direct reference code
		Updated language/content to align with standards changes QDM 5.5 and CQL 1.4
		Updated examples of logic as needed to reflect most recent use
		Removed references related to retired QDM logic

**APPENDIX A:**

**THE ORDER OF OPERATOR PRECEDENCE IN CQL  
FROM HIGHEST TO LOWEST**

**Table A.1. The order of operator precedence in CQL from highest to lowest**

Category	Operators
Primary	. [] ()
Conversion Phrase	convert..to
Unary Arithmetic	unary +/-
Extractor	start end difference duration width successor predecessor of <i>component</i> singleton from
Exponentiation	^
Multiplicative	* / div mod
Additive	+ - &
Conditional	if..then..else case..else..end
Unary List	distinct collapse flatten expand
Unary Test	is null true false
Type Operators	is as cast..as
Unary Logical	not exists
Between	between <i>precision</i> between duration in <i>precision</i> between difference in <i>precision</i> between
Comparison	<= < > >=
Timing Phrase	same as includes during before/after within

Category	Operators
Interval Operators	meets overlaps starts ends
Equality	= != ~ !~
Membership	in contains
Conjunction	and
Disjunction	or xor
Implication	implies
Binary List	union intersect except

Source: <https://cql.hl7.org/03-developersguide.html#operator-precedence>

**APPENDIX B:**  
**CASE TYPE DEFINITIONS**

## Case Type Definitions:

*(Note: CQL is a case-sensitive language)*

- **lowercase** – All letters are lowercase
- **camelCase** – First letters of words are capitalized, except the first word, with no whitespace characters allowed (used for QDM attributes)
- **PascalCase** – First letters of words are capitalized, including words not capitalized in Title Case like "and" and "of," with no whitespace characters allowed
- **Title Case** – Standard title casing including spaces and tabs, but no other whitespace characters allowed