

Measure title	Depression Screening and Follow-Up for Adolescents and Adults*	Measure ID	DSF-E
Description	<p>The percentage of members 12 years of age and older who were screened for clinical depression using a standardized instrument and, if screened positive, received follow-up care.</p> <ul style="list-style-type: none"> • <i>Depression Screening.</i> The percentage of members who were screened for clinical depression using a standardized instrument. • <i>Follow-Up on Positive Screen.</i> The percentage of members who received follow-up care within 30 days of a positive depression screen finding. 		
Measurement period	January 1 - December 31.		
Copyright and disclaimer notice	<p><i>*Adapted with financial support from the Centers for Medicare & Medicaid Services (CMS).</i></p> <p>HEDIS® is a registered trademark of the National Committee for Quality Assurance (“NCQA”). The HEDIS measures and specifications were developed by and are owned by NCQA. NCQA holds a copyright in these materials and may rescind or alter these materials at any time. Users of the HEDIS measures and specifications shall not have the right to alter, enhance or otherwise modify the HEDIS measures and specifications, and shall not disassemble, recompile or reverse engineer the HEDIS measures and specifications. Anyone desiring to use or reproduce the materials, subject to licensed user restrictions, without modification for an internal non-commercial purpose may do so without obtaining any approval from NCQA. Use of the Rules for Allowable Adjustments of HEDIS to make permitted adjustments of the materials does not constitute a modification. All other uses, including a commercial use (including but not limited to vendors using the measures and specifications with a product or service to calculate measure results), or any external reproduction, distribution and publication of the HEDIS measures or results (“rates”) therefrom must be approved by NCQA and are subject to a license at the discretion of NCQA. Any use of the materials to identify records or calculate measure results, for example, requires a custom license and may necessitate certification pursuant to NCQA’s Measure Certification Program.</p> <p>HEDIS measures and specifications are not clinical guidelines, do not establish a standard of medical care and have not been tested for all potential applications. The measures and specifications are provided “as is” without warranty of any kind. NCQA makes no representations, warranties or endorsements about the quality of any product, test or protocol identified as numerator compliant or otherwise identified as meeting the requirements of a HEDIS measure or specification. NCQA also makes no representations, warranties or endorsements about the quality of any organization or clinician who uses or reports performance measures. NCQA has no liability to anyone who relies on HEDIS measures and specifications or data reflective of performance under such measures and specifications.</p> <p>Unadjusted Uncertified Measures: A calculated measure result (a “rate”) from a HEDIS measure that has not been certified via NCQA’s Measure Certification Program, and is based on unadjusted HEDIS specifications, may not be called a “Health Plan HEDIS rate” until it is audited and designated reportable by an NCQA-Certified HEDIS Compliance Auditor. Until such time, such measure rates shall be designated or referred to as “Uncertified, Unaudited Health Plan HEDIS Rates.”</p> <p>Adjusted Uncertified Measures: A calculated measure result (a “rate”) from a HEDIS measure that has not been certified via NCQA’s Measure Certification Program, and is</p>		

based on adjusted HEDIS specifications, may not be called an “Adjusted HEDIS rate” until it is audited and designated reportable by an NCQA-Certified HEDIS Compliance Auditor. Until such time, such measure rates shall be designated or referred to as “**Adjusted, Uncertified, Unaudited HEDIS Rates.**”

Limited proprietary coding is contained in the measure specifications for convenience. Users of the proprietary code sets should obtain all necessary licenses from the owners of these code sets. NCQA disclaims all liability for use or accuracy of any coding contained in the specifications.

The American Medical Association holds a copyright to the CPT[®] codes contained in the measure specifications.

The American Hospital Association holds a copyright to the Uniform Billing Codes (“UB”) contained in the measure specifications. The UB Codes in the HEDIS specifications are included with the permission of the AHA. All uses of the UB Codes may require a license from the AHA. Anyone desiring to use the UB Codes in a commercial product to generate HEDIS results, or for any other commercial use, must obtain a commercial use license directly from the AHA. To inquire about licensing, contact ub04@aha.org.

Some measure specifications contain coding from LOINC[®] (<http://loinc.org>). The LOINC table, LOINC codes, LOINC panels and form file, LOINC linguistic variants file, LOINC/RSNA Radiology Playbook, and LOINC/IEEE Medical Device Code Mapping Table are copyright © 1995–2021 Regenstrief Institute, Inc. and the Logical Observation Identifiers Names and Codes (LOINC) Committee and are available at no cost under the license at <http://loinc.org/terms-of-use>.

“SNOMED” and “SNOMED CT” are registered trademarks of the International Health Terminology Standards Development Organisation (IHTSDO).

“HL7” is the registered trademark of Health Level Seven International.

No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopy, recording or any information storage and retrieval system, without the written permission of NCQA.

© 2021 by the National Committee for Quality Assurance
1100 13th Street, NW, Third Floor
Washington, DC 20005

All rights reserved.

NCQA Customer Support: 888-275-7585
NCQA Fax: 202-955-3599
NCQA Website: www.ncqa.org

Submit policy clarification support questions via My NCQA (<http://my.ncqa.org>)

Clinical recommendation statement

The U.S. Preventive Services Task Force (USPSTF) recommends screening for depression among adolescents 12–18 years and the general adult population, including pregnant and postpartum women. (B recommendation)

The USPSTF also recommends that screening be implemented with adequate systems in place to ensure accurate diagnosis, effective treatment and appropriate follow-up. (B recommendation)

Citations

U.S. Preventive Services Task Force. 2016. “Screening for Depression in Children and

Adolescents: U.S. Preventive Services Task Force Recommendation Statement.” *Annals of Internal Medicine* 164:360–6.

U.S. Preventive Services Task Force. 2016. “Screening for Major Depressive Disorder in Adults: US Preventive Services Task Force Recommendation Statement.” *Journal of the American Medical Association* 315(4):380–7.

Characteristics

Scoring

Proportion.

Type

Process.

Stratification

1. Commercial 12–17 years.
2. Commercial 18–64 years.
3. Commercial 65 years and older.
4. Medicaid 12–17 years
5. Medicaid 18–64 years.
6. Medicaid 65 years and older.
7. Medicare 18–64 years.
8. Medicare 65 years and older.

Risk adjustment

None

Improvement notation

A higher rate indicates better performance.

Guidance

Allocation:

The member was enrolled with a medical benefit throughout the Participation Period.

Requirements:

- This measure requires the use of an age-appropriate screening instrument. The member’s age is used to select the appropriate depression screening instrument.
- Depression screening captured in health risk assessments or other types of health assessments are allowed if the questions align with a specific instrument that is validated for depression screening. For example, if a health risk assessment includes questions from the PHQ-2, it counts as screening if the member answered the questions and a total score is calculated.

Reporting:

The total is the sum of the age stratifications.

Age and product line stratifications are not included in the measure calculation logic and need to be programmed manually.

Definitions

Participation

The identifiers and descriptors for each organization’s coverage used to define members’ eligibility for measure reporting. Allocation for HEDIS reporting is based on eligibility during the Participation Period.

Participation Period

The Measurement Period.

Depression Screening Instrument

A standard assessment instrument that has been normalized and validated for the appropriate patient population. Eligible screening instruments with thresholds for positive findings include:

Instruments for Adolescents (≤17 years)	Positive Finding
Patient Health Questionnaire (PHQ-9) [®]	Total Score ≥10
Patient Health Questionnaire Modified for Teens (PHQ-9M) [®]	Total Score ≥10
Patient Health Questionnaire-2 (PHQ-2) ^{®1}	Total Score ≥3
Beck Depression Inventory-Fast Screen (BDI-FS) ^{®1,2}	Total Score ≥8
Center for Epidemiologic Studies Depression Scale-Revised (CESD-R)	Total Score ≥17
Edinburgh Postnatal Depression Scale (EPDS)	Total Score ≥10
PROMIS Depression	Total Score (T Score) ≥60
¹ Brief screening instrument. All other instruments are full-length.	
² Proprietary; may be cost or licensing requirement associated with use.	
Instruments for Adults (18+ years)	Positive Finding
Patient Health Questionnaire (PHQ-9) [®]	Total Score ≥10
Patient Health Questionnaire-2 (PHQ-2) ^{®1}	Total Score ≥3
Beck Depression Inventory-Fast Screen (BDI-FS) ^{®1,2}	Total Score ≥8
Beck Depression Inventory (BDI-II)	Total Score ≥20
Center for Epidemiologic Studies Depression Scale-Revised (CESD-R)	Total Score ≥17
Duke Anxiety-Depression Scale (DUKE-AD) ^{®2}	Total Score ≥30
Geriatric Depression Scale Short Form (GDS) ¹	Total Score ≥5
Geriatric Depression Scale Long Form (GDS)	Total Score ≥10
Edinburgh Postnatal Depression Scale (EPDS)	Total Score ≥10
My Mood Monitor (M-3) [®]	Total Score ≥5
PROMIS Depression	Total Score (T Score) ≥60
Clinically Useful Depression Outcome Scale (CUDOS)	Total Score ≥31
¹ Brief screening instrument. All other instruments are full-length.	
² Proprietary; may be cost or licensing requirement associated with use.	

Initial Population

Members 12 years of age and older at the start of the Measurement Period who also meet criteria for Participation.

Exclusions

- Members with bipolar disorder that starts during the year prior to the Measurement Period.

	<ul style="list-style-type: none"> • Members with depression that starts during the year prior to the Measurement Period. • Members in hospice or using hospice services any time during the Measurement Period.
<p>Denominator</p>	<p>Denominator 1</p> <p>The Initial Population, minus Exclusions.</p> <p>Denominator 2</p> <p>All members from Numerator 1 with a positive depression screen finding between January 1 and December 1 of the Measurement Period.</p>
<p>Numerator</p>	<p>Numerator 1—Depression Screening</p> <p>Members with a documented result for depression screening, using an age-appropriate standardized instrument, performed between January 1 and December 1 of the Measurement Period.</p> <p>Numerator 2—Follow-Up on Positive Screen</p> <p>Members who received follow-up care on or up to 30 days after the date of the first positive screen (31 total days).</p> <p>Any of the following on or up to 30 days after the first positive screen:</p> <ul style="list-style-type: none"> • An outpatient, telephone, e-visit or virtual check-in follow-up visit with a diagnosis of depression or other behavioral health condition. • A depression case management encounter that documents assessment for symptoms of depression or a diagnosis of depression or other behavioral health condition. • A behavioral health encounter, including assessment, therapy, collaborative care or medication management. • A dispensed antidepressant medication. <p>OR</p> <ul style="list-style-type: none"> • Documentation of additional depression screening on a full-length instrument indicating either no depression or no symptoms that require follow-up (i.e., a negative screen) on the same day as a positive screen on a brief screening instrument. <p><i>Note: For example, if there is a positive screen resulting from a PHQ-2 score, documentation of a negative finding from a PHQ-9 performed on the same day qualifies as evidence of follow-up.</i></p>
<p>Data criteria (element level)</p>	
<p>Value Sets:</p> <ul style="list-style-type: none"> • DSFE_HEDIS_MY2022-1.0.0 <ul style="list-style-type: none"> ◦ Antidepressant Medications (https://www.ncqa.org/fhir/valueset/2.16.840.1.113883.3.464.1004.1503) 	

- Behavioral Health Encounter (<https://www.ncqa.org/fhir/valueset/2.16.840.1.113883.3.464.1004.1383>)
- Bipolar Disorder (<https://www.ncqa.org/fhir/valueset/2.16.840.1.113883.3.464.1004.1044>)
- Depression (<https://www.ncqa.org/fhir/valueset/2.16.840.1.113883.3.464.1004.1390>)
- Depression Case Management Encounter (<https://www.ncqa.org/fhir/valueset/2.16.840.1.113883.3.464.1004.1389>)
- Depression or Other Behavioral Health Condition (<https://www.ncqa.org/fhir/valueset/2.16.840.1.113883.3.464.1004.1501>)
- Follow Up Visit (<https://www.ncqa.org/fhir/valueset/2.16.840.1.113883.3.464.1004.1385>)
- Other Bipolar Disorder (<https://www.ncqa.org/fhir/valueset/2.16.840.1.113883.3.464.1004.1399>)
- NCQA_Hospice-1.0.0
 - Hospice Encounter (<https://www.ncqa.org/fhir/valueset/2.16.840.1.113883.3.464.1004.1761>)
 - Hospice Intervention (<https://www.ncqa.org/fhir/valueset/2.16.840.1.113883.3.464.1004.1762>)

Direct Reference Codes and Codesystems:

- DSFE_HEDIS_MY2022-1.0.0
 - codesystem "ICD-10": 'http://hl7.org/fhir/sid/icd-10-cm'
 - codesystem "LOINC": 'http://loinc.org'
 - codesystem "SNOMEDCT": 'http://snomed.info/sct'
 - code "Beck Depression Inventory Fast Screen total score [BDI]": '89208-3' from "LOINC" display 'Beck Depression Inventory Fast Screen total score [BDI]'
 - code "Beck Depression Inventory II total score [BDI]": '89209-1' from "LOINC" display 'Beck Depression Inventory II total score [BDI]'
 - code "Center for Epidemiologic Studies Depression Scale-Revised total score [CESD-R]": '89205-9' from "LOINC" display 'Center for Epidemiologic Studies Depression Scale-Revised total score [CESD-R]'
 - code "Edinburgh Postnatal Depression Scale [EPDS]": '71354-5' from "LOINC" display 'Edinburgh Postnatal Depression Scale [EPDS]'
 - code "Exercise counseling": 'Z71.82' from "ICD-10" display 'Exercise counseling'
 - code "Final score [DUKE-AD]": '90853-3' from "LOINC" display 'Final score [DUKE-AD]'
 - code "Geriatric depression scale (GDS) short version total": '48545-8' from "LOINC" display 'Geriatric depression scale (GDS) short version total'
 - code "Geriatric depression scale (GDS) total": '48544-1' from "LOINC" display 'Geriatric depression scale (GDS) total'
 - code "Patient Health Questionnaire 2 item (PHQ-2) total score [Reported]": '55758-7' from "LOINC" display 'Patient Health Questionnaire 2 item (PHQ-2) total score [Reported]'
 - code "Patient Health Questionnaire 9 item (PHQ-9) total score [Reported]": '44261-6' from "LOINC" display 'Patient Health Questionnaire 9 item (PHQ-9) total score [Reported]'
 - code "Patient Health Questionnaire-9: Modified for Teens total score [Reported.PHQ.Teen]": '89204-2' from "LOINC" display 'Patient Health Questionnaire-9: Modified for Teens total score [Reported.PHQ.Teen]'
 - code "PROMIS-29 Depression score T-score": '71965-8' from "LOINC" display 'PROMIS-29 Depression score T-score'
 - code "Symptoms of depression (finding)": '394924000' from "SNOMEDCT" display 'Symptoms of depression (finding)'
 - code "Total score [CUDOS]": '90221-3' from "LOINC" display 'Total score [CUDOS]'
 - code "Total score [M3]": '71777-7' from "LOINC" display 'Total score [M3]'
- NCQA_Terminology-1.0.0
 - codesystem "ActEncounterCodes": 'http://terminology.hl7.org/CodeSystem/v3-ActCode'
 - codesystem "ConditionClinicalStatusCodes": 'http://terminology.hl7.org/CodeSystem/condition-clinical'
 - code "active": 'active' from "ConditionClinicalStatusCodes"
 - code "managed care policy": 'MCPOL' from "ActEncounterCodes"
 - code "retiree health program": 'RETIRE' from "ActEncounterCodes"
 - code "subsidized health program": 'SUBSIDIZ' from "ActEncounterCodes"

Table of Contents

1. [Population Criteria 1](#)
2. [Population Criteria 2](#)
3. [Definitions](#)
4. [Functions](#)

Population Criteria 1

- **Initial Population 1**

```
define "Initial Population 1":  
  AgeInYearsAt(date from start of "Measurement Period")>= 12  
  and "Enrolled During Participation Period"
```

- **Denominator 1**

```
define "Denominator 1":  
  "Initial Population 1"
```

- **Exclusions 1**

```
define "Exclusions 1":  
  Hospice."Hospice Intervention or Encounter"  
  or exists "Bipolar Disorder Starting during Year Prior to Measurement Period"  
  or exists "Depression Starting during Year Prior to Measurement Period"
```

- **Numerator 1**

```
define "Numerator 1":  
  "Adolescent Depression Screening with Documented Result between January 1 and December 1"  
  or "Adult Depression Screening with Documented Result between January 1 and December 1"
```

Population Criteria 2

- **Initial Population 2**

```
define "Initial Population 2":  
  "Initial Population 1"
```

- **Denominator 2**

```
define "Denominator 2":  
  "Adolescent Depression Screening with Positive Result between January 1 and December 1"  
  or "Adult Depression Screening with Positive Result between January 1 and December 1"
```

- **Exclusions 2**

```
define "Exclusions 2":  
  "Exclusions 1"
```

- **Numerator 2**

```
define "Numerator 2":  
  exists "Follow Up Care on or 30 Days after First Positive Screen"  
  or "Has Positive Brief Screen Same Day as Negative Full Length Screen"
```

Definitions

- **DSFE_HEDIS_MY2022-1.0.0."Adolescent Brief Screen with Documented Result"**

```
define "Adolescent Brief Screen with Documented Result":
  ( "Depression Screening with Documented Result"("Patient Health Questionnaire 2 item (PHQ-2) total score [Reported]")
    union "Depression Screening with Documented Result"("Beck Depression Inventory Fast Screen total score [BDI]")) AdolescentBrief
    where AgeInYearsAt(date from start of FHIRBase."Normalize Interval"(AdolescentBrief.effective))<= 17
```

- **DSFE_HEDIS_MY2022-1.0.0."Adolescent Brief Screen with Positive Result"**

```
define "Adolescent Brief Screen with Positive Result":
  ( "Depression Screening with Positive Result"("Patient Health Questionnaire 2 item (PHQ-2) total score [Reported]", 3)
    union "Depression Screening with Positive Result"("Beck Depression Inventory Fast Screen total score [BDI]", 8)) AdolescentBriefPositive
    where AgeInYearsAt(date from start of FHIRBase."Normalize Interval"
(AdolescentBriefPositive.effective))<= 17
    and start of FHIRBase."Normalize Interval" ( AdolescentBriefPositive.effective ) during
Interval[start of "Measurement Period", "December 1 of Measurement Period"]
```

- **DSFE_HEDIS_MY2022-1.0.0."Adolescent Depression Screening with Documented Result between January 1 and December 1"**

```
define "Adolescent Depression Screening with Documented Result between January 1 and December 1":
  "Has Depression Screening during Period"("Adolescent Full Length Depression Screen with Documented Result"
    union "Adolescent Brief Screen with Documented Result", Interval[start of "Measurement Period",
"December 1 of Measurement Period"]
  )
```

- **DSFE_HEDIS_MY2022-1.0.0."Adolescent Depression Screening with Positive Result between January 1 and December 1"**

```
define "Adolescent Depression Screening with Positive Result between January 1 and December 1":
  "Has Depression Screening during Period"("Adolescent Full Length Depression Screen with Positive Result"
    union "Adolescent Brief Screen with Positive Result", Interval[start of "Measurement Period",
"December 1 of Measurement Period"]
  )
```

- **DSFE_HEDIS_MY2022-1.0.0."Adolescent Full Length Depression Screen with Documented Result"**

```
define "Adolescent Full Length Depression Screen with Documented Result":
  ( "Depression Screening with Documented Result"("Patient Health Questionnaire 9 item (PHQ-9) total score [Reported]")
    union "Depression Screening with Documented Result"("Patient Health Questionnaire-9: Modified for Teens total score [Reported.PHQ.Teen]")
    union "Depression Screening with Documented Result"("Center for Epidemiologic Studies Depression Scale-Revised total score [CESD-R]")
    union "Depression Screening with Documented Result"("Edinburgh Postnatal Depression Scale [EPDS]")
    union "Depression Screening with Documented Result"("PROMIS-29 Depression score T-score")) AdolescentDepressionScreen
    where AgeInYearsAt(date from start of FHIRBase."Normalize Interval"
(AdolescentDepressionScreen.effective))<= 17
```

- **DSFE_HEDIS_MY2022-1.0.0."Adolescent Full Length Depression Screen with Positive Result"**

```
define "Adolescent Full Length Depression Screen with Positive Result":
  ( "Depression Screening with Positive Result"("Patient Health Questionnaire 9 item (PHQ-9) total score [Reported]", 10)
    union "Depression Screening with Positive Result"("Patient Health Questionnaire-9: Modified for Teens total score [Reported.PHQ.Teen]", 10)
    union "Depression Screening with Positive Result"("Center for Epidemiologic Studies Depression Scale-Revised total score [CESD-R]", 17)
```



```

    union "Depression Screening with Positive Result"("Edinburgh Postnatal Depression Scale [EPDS]", 10)
    union "Depression Screening with Positive Result"("PROMIS-29 Depression score T-score", 60))
AdolescentPositiveDepressionScreen
    where AgeInYearsAt(date from start of FHIRBase."Normalize Interval"
(AdolescentPositiveDepressionScreen.effective))<= 17
    and start of FHIRBase."Normalize Interval" ( AdolescentPositiveDepressionScreen.effective ) during
Interval[start of "Measurement Period", "December 1 of Measurement Period"]

```

- **DSFE_HEDIS_MY2022-1.0.0."Adult Brief Screen with Documented Result"**

```

define "Adult Brief Screen with Documented Result":
    ( "Depression Screening with Documented Result"("Patient Health Questionnaire 2 item (PHQ-2) total
score [Reported]")
    union "Depression Screening with Documented Result"("Geriatric depression scale (GDS) short version
total")
    union "Depression Screening with Documented Result"("Beck Depression Inventory Fast Screen total
score [BDI]")) AdultBrief
    where AgeInYearsAt(date from start of FHIRBase."Normalize Interval"(AdultBrief.effective))>= 18

```

- **DSFE_HEDIS_MY2022-1.0.0."Adult Brief Screen with Positive Result"**

```

define "Adult Brief Screen with Positive Result":
    ( "Depression Screening with Positive Result"("Patient Health Questionnaire 2 item (PHQ-2) total score
[Reported]", 3)
    union "Depression Screening with Positive Result"("Beck Depression Inventory Fast Screen total score
[BDI]", 8)
    union "Depression Screening with Positive Result"("Geriatric depression scale (GDS) short version
total", 5)) AdultBriefPositive
    where AgeInYearsAt(date from start of FHIRBase."Normalize Interval"(AdultBriefPositive.effective))>=
18
    and start of FHIRBase."Normalize Interval" ( AdultBriefPositive.effective ) during Interval[start
of "Measurement Period", "December 1 of Measurement Period"]

```

- **DSFE_HEDIS_MY2022-1.0.0."Adult Depression Screening with Documented Result between January 1 and December 1"**

```

define "Adult Depression Screening with Documented Result between January 1 and December 1":
    "Has Depression Screening during Period"("Adult Full Length Depression Screen with Documented Result"
    union "Adult Brief Screen with Documented Result", Interval[start of "Measurement Period",
"December 1 of Measurement Period"]
    )

```

- **DSFE_HEDIS_MY2022-1.0.0."Adult Depression Screening with Positive Result between January 1 and December 1"**

```

define "Adult Depression Screening with Positive Result between January 1 and December 1":
    "Has Depression Screening during Period"("Adult Full Length Depression Screen with Positive Result"
    union "Adult Brief Screen with Positive Result", Interval[start of "Measurement Period", "December
1 of Measurement Period"]
    )

```

- **DSFE_HEDIS_MY2022-1.0.0."Adult Full Length Depression Screen with Documented Result"**

```

define "Adult Full Length Depression Screen with Documented Result":
    ( "Depression Screening with Documented Result"("Patient Health Questionnaire 9 item (PHQ-9) total
score [Reported]")
    union "Depression Screening with Documented Result"("Beck Depression Inventory II total score [BDI]")
    union "Depression Screening with Documented Result"("Center for Epidemiologic Studies Depression
Scale-Revised total score [CESD-R]")
    union "Depression Screening with Documented Result"("Final score [DUKE-AD]")
    union "Depression Screening with Documented Result"("Geriatric depression scale (GDS) total")
    union "Depression Screening with Documented Result"("Edinburgh Postnatal Depression Scale [EPDS]")
    union "Depression Screening with Documented Result"("Total score [M3]")
    union "Depression Screening with Documented Result"("Total score [CUDOS]")
    union "Depression Screening with Documented Result"("PROMIS-29 Depression score T-score"))

```

```
AdultDepressionScreenResult
  where AgeInYearsAt(date from start of FHIRBase."Normalize Interval"
(AdultDepressionScreenResult.effective))>= 18
```

- **DSFE_HEDIS_MY2022-1.0.0."Adult Full Length Depression Screen with Positive Result"**

```
define "Adult Full Length Depression Screen with Positive Result":
  ( "Depression Screening with Positive Result"("Patient Health Questionnaire 9 item (PHQ-9) total score
[Reported]", 10)
  union "Depression Screening with Positive Result"("Beck Depression Inventory II total score [BDI]",
20)
  union "Depression Screening with Positive Result"("Center for Epidemiologic Studies Depression Scale-
Revised total score [CESD-R]", 17)
  union "Depression Screening with Positive Result"("Final score [DUKE-AD]", 30)
  union "Depression Screening with Positive Result"("Geriatric depression scale (GDS) total", 10)
  union "Depression Screening with Positive Result"("Edinburgh Postnatal Depression Scale [EPDS]", 10)
  union "Depression Screening with Positive Result"("Total score [M3]", 5)
  union "Depression Screening with Positive Result"("Total score [CUDOS]", 31)
  union "Depression Screening with Positive Result"("PROMIS-29 Depression score T-score", 60))
AdultPositiveDepressionScreen
  where AgeInYearsAt(date from start of FHIRBase."Normalize Interval"
(AdultPositiveDepressionScreen.effective))>= 18
  and start of FHIRBase."Normalize Interval" ( AdultPositiveDepressionScreen.effective ) during
Interval[start of "Measurement Period", "December 1 of Measurement Period"]
```

- **DSFE_HEDIS_MY2022-1.0.0."Bipolar Disorder Starting during Year Prior to Measurement Period"**

```
define "Bipolar Disorder Starting during Year Prior to Measurement Period":
  ( Status."Active Condition" ( [Condition: "Bipolar Disorder"] )
  union Status."Active Condition" ( [Condition: "Other Bipolar Disorder"] ) ) BipolarExclusion
  where FHIRBase."Prevalence Period" ( BipolarExclusion ) starts during Interval["January 1 of Year
Prior to Measurement Period", start of "Measurement Period" )
```

- **DSFE_HEDIS_MY2022-1.0.0."December 1 of Measurement Period"**

```
define "December 1 of Measurement Period":
  DateTime((year from start of "Measurement Period"), 12, 1, 23, 59, 59, 0, 0)
```

- **DSFE_HEDIS_MY2022-1.0.0."Denominator 1"**

```
define "Denominator 1":
  "Initial Population 1"
```

- **DSFE_HEDIS_MY2022-1.0.0."Denominator 2"**

```
define "Denominator 2":
  "Adolescent Depression Screening with Positive Result between January 1 and December 1"
  or "Adult Depression Screening with Positive Result between January 1 and December 1"
```

- **DSFE_HEDIS_MY2022-1.0.0."Depression Starting during Year Prior to Measurement Period"**

```
define "Depression Starting during Year Prior to Measurement Period":
  ( Status."Active Condition" ( [Condition: "Depression"] ) ) DepressionDiagnosis
  where FHIRBase."Prevalence Period" ( DepressionDiagnosis ) starts during Interval["January 1 of Year
Prior to Measurement Period", start of "Measurement Period" )
```

- **DSFE_HEDIS_MY2022-1.0.0."Enrolled During Participation Period"**

```
define "Enrolled During Participation Period":
  Enrollment."Health Plan Enrollment Criteria" ( "Member Coverage", date from
end of "Measurement Period", Interval[date from start of "Measurement Period", date from
end of "Measurement Period"], 45 )
```

- **DSFE_HEDIS_MY2022-1.0.0."Exclusions 1"**

```
define "Exclusions 1":  
  Hospice."Hospice Intervention or Encounter"  
  or exists "Bipolar Disorder Starting during Year Prior to Measurement Period"  
  or exists "Depression Starting during Year Prior to Measurement Period"
```

- **DSFE_HEDIS_MY2022-1.0.0."Exclusions 2"**

```
define "Exclusions 2":  
  "Exclusions 1"
```

- **DSFE_HEDIS_MY2022-1.0.0."First Positive Adolescent Depression Screen between January 1 and December 1"**

```
define "First Positive Adolescent Depression Screen between January 1 and December 1":  
  { First(("Adolescent Full Length Depression Screen with Positive Result"  
    union "Adolescent Brief Screen with Positive Result")PositiveScreen  
    sort by start of FHIRBase."Normalize Interval"(effective)asc  
  )}
```

- **DSFE_HEDIS_MY2022-1.0.0."First Positive Adolescent Screen is Brief Screen"**

```
define "First Positive Adolescent Screen is Brief Screen":  
  "First Positive Adolescent Depression Screen between January 1 and December 1" PositiveScreen  
  without "Adolescent Full Length Depression Screen with Positive Result" FLScreen  
  such that start of FHIRBase."Normalize Interval" ( PositiveScreen.effective ) same day as start of  
  FHIRBase."Normalize Interval" ( FLScreen.effective )
```

- **DSFE_HEDIS_MY2022-1.0.0."First Positive Adult Depression Screen between January 1 and December 1"**

```
define "First Positive Adult Depression Screen between January 1 and December 1":  
  { First(("Adult Full Length Depression Screen with Positive Result"  
    union "Adult Brief Screen with Positive Result")PositiveScreen  
    sort by start of FHIRBase."Normalize Interval"(effective)asc  
  )}
```

- **DSFE_HEDIS_MY2022-1.0.0."First Positive Adult Screen is Brief Screen"**

```
define "First Positive Adult Screen is Brief Screen":  
  "First Positive Adult Depression Screen between January 1 and December 1" PositiveScreen  
  without "Adult Full Length Depression Screen with Positive Result" FLScreen  
  such that start of FHIRBase."Normalize Interval" ( PositiveScreen.effective ) same day as start of  
  FHIRBase."Normalize Interval" ( FLScreen.effective )
```

- **DSFE_HEDIS_MY2022-1.0.0."Follow Up Care on or 30 Days after First Positive Screen"**

```
define "Follow Up Care on or 30 Days after First Positive Screen":  
  ( ( "First Positive Adolescent Depression Screen between January 1 and December 1"  
    union "First Positive Adult Depression Screen between January 1 and December 1" ) Screening  
    return ( Tuple {  
      hasFollowUpVisit: exists ( Status."Finished Encounter" ( [Encounter: "Follow Up Visit"] ) )  
      FollowUpVisit  
        where Encounters."Encounter Has Diagnosis" ( FollowUpVisit, [Condition: "Depression or Other  
Behavioral Health Condition"] )  
        and date from start of FHIRBase."Normalize Interval" ( FollowUpVisit.period ) 30 days or less  
on or after date from start of FHIRBase."Normalize Interval" ( Screening.effective ),  
      hasDepressionCaseManagementEncounterWithDx: exists ( ( Status."Finished Encounter" ( [Encounter:  
"Depression Case Management Encounter"] ) ) dcmEnc  
        where date from start of FHIRBase."Normalize Interval" ( dcmEnc.period ) 30 days or less on or  
after date from start of FHIRBase."Normalize Interval" ( Screening.effective ) )  
      CaseManagementEncounterWithDx  
        where Encounters."Encounter Has Diagnosis" ( CaseManagementEncounterWithDx, [Condition:
```

```

"Depression or Other Behavioral Health Condition" ] ),
  hasDepressionCaseManagementEncounterWithSymptom: exists ( ( Status."Finished Encounter" (
[Encounter: "Depression Case Management Encounter" ] ) ) dcmEnc
  where date from start of FHIRBase."Normalize Interval" ( dcmEnc.period ) 30 days or less on or
after date from start of FHIRBase."Normalize Interval" ( Screening.effective ) )
CaseManagementEncounterWithSymptom
  where exists [Observation: "Symptoms of depression (finding)"] DepressionSymptoms
  where date from start of FHIRBase."Normalize Interval" ( DepressionSymptoms.effective ) ~ date
from start of FHIRBase."Normalize Interval" ( CaseManagementEncounterWithSymptom.period ),
  hasBehavioralHealthEncounter: exists ( ( Status."Finished Encounter" ( [Encounter: "Behavioral
Health Encounter" ] ) ) BHEnc
  where date from start of FHIRBase."Normalize Interval" ( BHEnc.period ) 30 days or less on or
after date from start of FHIRBase."Normalize Interval" ( Screening.effective )
)
)
or ( exists ( Status."Active Condition" ( [Condition: "Exercise counseling" ] ) ) )
ExerciseDiagnosis
  where date from start of FHIRBase."Prevalence Period" ( ExerciseDiagnosis ) 30 days or less
on or after date from start of FHIRBase."Normalize Interval" ( Screening.effective )
),
  hasAntidepressantMedication: exists ( Status."Dispensed Medication" ( [MedicationDispense:
"Antidepressant Medications" ] ) ) ADMeds
  where date from ADMeds.whenHandedOver 30 days or less on or after date from start of
FHIRBase."Normalize Interval" ( Screening.effective )
} ) FollowUpCare
  return if AnyTrue({ FollowUpCare.hasFollowUpVisit,
FollowUpCare.hasDepressionCaseManagementEncounterWithDx,
FollowUpCare.hasDepressionCaseManagementEncounterWithSymptom, FollowUpCare.hasBehavioralHealthEncounter,
FollowUpCare.hasAntidepressantMedication })then Screening
  else null ) screeningWithFollowUpCare
  where screeningWithFollowUpCare is not null

```

- **DSFE_HEDIS_MY2022-1.0.0."Has Positive Brief Screen Same Day as Negative Full Length Screen"**

```

define "Has Positive Brief Screen Same Day as Negative Full Length Screen":
  exists ( "First Positive Adult Screen is Brief Screen" PositiveScreen
  with ( "Adult Full Length Depression Screen with Documented Result"
  except "Adult Full Length Depression Screen with Positive Result" ) NegativeScreen
  such that start of FHIRBase."Normalize Interval" ( NegativeScreen.effective ) same day as start
of FHIRBase."Normalize Interval" ( PositiveScreen.effective )
)
)
or exists ( "First Positive Adolescent Screen is Brief Screen" PositiveScreen
  with ( "Adolescent Full Length Depression Screen with Documented Result"
  except "Adolescent Full Length Depression Screen with Positive Result" ) NegativeScreen
  such that start of FHIRBase."Normalize Interval" ( NegativeScreen.effective ) same day as start
of FHIRBase."Normalize Interval" ( PositiveScreen.effective )
)
)

```

- **DSFE_HEDIS_MY2022-1.0.0."Initial Population 1"**

```

define "Initial Population 1":
  AgeInYearsAt(date from start of "Measurement Period")>= 12
  and "Enrolled During Participation Period"

```

- **DSFE_HEDIS_MY2022-1.0.0."Initial Population 2"**

```

define "Initial Population 2":
  "Initial Population 1"

```

- **DSFE_HEDIS_MY2022-1.0.0."January 1 of Year Prior to Measurement Period"**

```

define "January 1 of Year Prior to Measurement Period":
  DateTime(((year from start of "Measurement Period")- 1), 1, 1, 0, 0, 0, 0, 0)

```

- **DSFE_HEDIS_MY2022-1.0.0."Member Coverage"**

```
define "Member Coverage":  
  [Coverage] C  
  where FHIRBase."Normalize Interval" ( C.period ) overlaps "Measurement Period"
```

- **DSFE_HEDIS_MY2022-1.0.0."Numerator 1"**

```
define "Numerator 1":  
  "Adolescent Depression Screening with Documented Result between January 1 and December 1"  
  or "Adult Depression Screening with Documented Result between January 1 and December 1"
```

- **DSFE_HEDIS_MY2022-1.0.0."Numerator 2"**

```
define "Numerator 2":  
  exists "Follow Up Care on or 30 Days after First Positive Screen"  
  or "Has Positive Brief Screen Same Day as Negative Full Length Screen"
```

- **NCQA_Hospice-1.0.0."Hospice Intervention or Encounter"**

```
define "Hospice Intervention or Encounter":  
exists ((Status."Completed or Ongoing Procedure"([Procedure:"Hospice Intervention"]))) HospiceInt  
  where NCQAFHIRBase."Normalize Interval"(HospiceInt.performed) overlaps "Measurement Period"  
)  
or exists ((Status."Finished Encounter"(["Encounter": "Hospice Encounter"])) HospiceEnc  
  where NCQAFHIRBase."Normalize Interval" (HospiceEnc.period) overlaps "Measurement Period"  
)
```

Functions

- **DSFE_HEDIS_MY2022-1.0.0."Depression Screening with Documented Result"**

```
define function "Depression Screening with Documented Result"(assessmentDRC System.Code):  
  [Observation: assessmentDRC] Obs  
  where Obs.value is not null
```

- **DSFE_HEDIS_MY2022-1.0.0."Depression Screening with Positive Result"**

```
define function "Depression Screening with Positive Result"(assessmentDRC System.Code,  
positiveScoreThreshold Integer):  
  [Observation: assessmentDRC] Obs  
  where Obs.value >= positiveScoreThreshold
```

- **DSFE_HEDIS_MY2022-1.0.0."Has Depression Screening during Period"**

```
define function "Has Depression Screening during Period"(Assessment List<Observation>, period  
Interval<DateTime>):  
  exists Assessment A  
  where ( date from start of FHIRBase."Normalize Interval" ( A.effective ) ) during Interval[date from  
start of period, date from  
end of period]
```

- **FHIRHelpers-4.0.1."ToCalendarUnit"**

```
define function ToCalendarUnit(unit System.String):  
  case unit  
  when 'ms' then 'millisecond'  
  when 's' then 'second'  
  when 'min' then 'minute'  
  when 'h' then 'hour'  
  when 'd' then 'day'  
  when 'wk' then 'week'  
  when 'mo' then 'month'  
  when 'a' then 'year'
```

```
    else unit
end
```

- **FHIRHelpers-4.0.1."ToCode"**

```
define function ToCode(coding FHIR.Coding):
  if coding is null then
    null
  else
    System.Code {
      code: coding.code.value,
      system: coding.system.value,
      version: coding.version.value,
      display: coding.display.value
    }
  }
```

- **FHIRHelpers-4.0.1."ToConcept"**

```
define function ToConcept(concept FHIR.CodeableConcept):
  if concept is null then
    null
  else
    System.Concept {
      codes: concept.coding C return ToCode(C),
      display: concept.text.value
    }
  }
```

- **FHIRHelpers-4.0.1."ToDate"**

```
define function ToDate(value date): value.value
```

- **FHIRHelpers-4.0.1."ToDateTime"**

```
define function ToDateTime(value dateTime): value.value
```

- **FHIRHelpers-4.0.1."ToDateTime"**

```
define function ToDateTime(value instant): value.value
```

- **FHIRHelpers-4.0.1."ToInteger"**

```
define function ToInteger(value integer): value.value
```

- **FHIRHelpers-4.0.1."ToQuantity"**

```
define function ToQuantity(quantity FHIR.Quantity):
  case
  when quantity is null then null
  when quantity.value is null then null
  when quantity.comparator is not null then
    Message(null, true, 'FHIRHelpers.ToQuantity.ComparatorQuantityNotSupported', 'Error', 'FHIR
Quantity value has a comparator and cannot be converted to a System.Quantity value.')
  when quantity.system is null or quantity.system.value = 'http://unitsofmeasure.org'
    or quantity.system.value = 'http://hl7.org/fhirpath/CodeSystem/calendar-units' then
    System.Quantity { value: quantity.value.value, unit:
ToCalendarUnit(Coalesce(quantity.code.value, quantity.unit.value, '1')) }
  else
    Message(null, true, 'FHIRHelpers.ToQuantity.InvalidFHIRQuantity', 'Error', 'Invalid FHIR
Quantity code: ' & quantity.unit.value & ' (' & quantity.system.value & '|' & quantity.code.value & ')')
  end
```

- **FHIRHelpers-4.0.1."ToString"**

```
define function ToString(value EncounterStatus): value.value
```

- **FHIRHelpers-4.0.1."ToString"**

```
define function ToString(value MedicationDispenseStatus): value.value
```

- **FHIRHelpers-4.0.1."ToString"**

```
define function ToString(value ProcedureStatus): value.value
```

- **FHIRHelpers-4.0.1."ToString"**

```
define function ToString(value string): value.value
```

- **NCQA_CQLBase-1.0.0."Collapse Date Interval Workaround"**

```
define function "Collapse Date Interval Workaround"(intervals List<Interval<Date>>):  
  if intervals is null then { } as List<Interval<Date>>  
  else if Count(intervals)= 0 then { } as List<Interval<Date>>  
  else collapse ( intervals I where not AnyTrue( intervals J return I properly included in J ) ) per day
```

- **NCQA_CQLBase-1.0.0."Collapsed Date Interval Stats"**

```
define function "Collapsed Date Interval Stats"(collapsedIntervals List<Interval<Date>>):  
  Tuple {  
    "Intervals": collapsedIntervals,  
    "Interval Count": Count(collapsedIntervals),  
    "Total Days In Intervals": if Count(collapsedIntervals)= 0 then 0  
      else Sum(collapsedIntervals I  
        return Max({ duration in days of I + 1, 0 })),  
    "Longest Interval": if Count(collapsedIntervals)= 0 then null as Interval<Date>  
      else First(collapsedIntervals I  
        return Tuple {  
          "interval": I,  
          "days": Max({ duration in days of I + 1, 0 })  
        } sort by "days" desc).interval,  
    "Total Days In Longest Interval": if Count(collapsedIntervals)= 0 then 0  
      else Max({ duration in days of First(collapsedIntervals I  
        return Tuple {  
          "interval": I,  
          "days": Max({ duration in  
days of I + 1, 0 })  
        }  
      } sort by "days"  
desc).interval + 1, 0 })  
  }
```

- **NCQA_CQLBase-1.0.0."Date Interval Covering Relative to Base Interval"**

```
define function "Date Interval Covering Relative to Base Interval"(baseInterval Interval<Date>,  
  coveringIntervals List<Interval<Date>>):  
  "Collapse Date Interval Workaround"(("Sort Date Intervals"(coveringIntervals)) sortedInterval  
    return baseInterval intersect sortedInterval)
```

- **NCQA_CQLBase-1.0.0."Date Interval Gaps Relative to Base Interval"**

```
define function "Date Interval Gaps Relative to Base Interval"(baseInterval Interval<Date>,  
  coveringIntervals List<Interval<Date>>):  
  ( Tuple {  
    sortedCoverings: "Date Interval Covering Relative to Base Interval"(baseInterval, coveringIntervals)  
  } ) variableDeclarations  
  return ( Tuple {  
    frontgaps: if Count(variableDeclarations.sortedCoverings)= 0 then { baseInterval }  
      else ( variableDeclarations.sortedCoverings sortedCovering
```

```

sortedCovering) = 0
return if IndexOf(variableDeclarations.sortedCoverings,
sortedCovering) < 0
then ( Interval[start of baseInterval, start of sortedCovering]
intersect baseInterval) except sortedCovering
else Interval( start of
variableDeclarations.sortedCoverings[IndexOf(variableDeclarations.sortedCoverings, sortedCovering)- 1],
end of sortedCovering )
except
variableDeclarations.sortedCoverings[IndexOf(variableDeclarations.sortedCoverings, sortedCovering)- 1]
except sortedCovering),
endgap: if Count(variableDeclarations.sortedCoverings)= 0 then { } as List<Interval<Date>>
else { ( Interval( start of Last(variableDeclarations.sortedCoverings), end of
baseInterval]
except Last(variableDeclarations.sortedCoverings)) intersect baseInterval
}
} ) calculations
return "Collapse Date Interval Workaround"( calculations.frontgaps union calculations.endgap )

```

- **NCQA_CQLBase-1.0.0."Date Interval Gaps Relative to Base Interval Stats"**

```

define function "Date Interval Gaps Relative to Base Interval Stats"(baseInterval Interval<Date>,
coveringIntervals List<Interval<Date>>):
( Tuple {
"Gap Intervals": "Date Interval Gaps Relative to Base Interval"(baseInterval, coveringIntervals)
} ) variableDeclarations
return ( "Collapsed Date Interval Stats"(variableDeclarations."Gap Intervals"))

```

- **NCQA_CQLBase-1.0.0."Sort Date Intervals"**

```

define function "Sort Date Intervals"(intervals List<Interval<Date>>):
if intervals is null then { } as List<Interval<Date>>
else if Count(intervals)= 0 then { } as List<Interval<Date>>
else ( ( intervals I
return Tuple {
interval: I,
startOfInterval: if start of I is null then minimum Date else start of I
} sort by startOfInterval asc
) sortedIntervals
return sortedIntervals.interval
)

```

- **NCQA_Encounter-1.0.0."Encounter Has Diagnosis"**

```

define function "Encounter Has Diagnosis"(Encounter FHIR.Encounter, Conditions List<FHIR.Condition>):
AnyTrue((Encounter.diagnosis D
return D.condition.reference)CRef
return exists(Conditions C
where C.id = FHIRBase."GetId"(CRef)
)
)

```

- **NCQA_FHIRBase-1.0.0."GetId"**

```

define function "GetId"(uri String):
if ( PositionOf('/', uri) > 0 ) then Last(Split(uri, '/'))
else uri

```

- **NCQA_FHIRBase-1.0.0."Normalize Abatement"**

```

define function "Normalize Abatement"(abatement Choice<FHIR.dateTime, FHIR.Age, FHIR.Period, FHIR.Range,
FHIR.string>):
if abatement is FHIR.dateTime then Interval[FHIRHelpers.ToDateTime ( abatement as FHIR.dateTime ),
FHIRHelpers.ToDateTime ( abatement as FHIR.dateTime )]
else if abatement is FHIR.Period then Interval[abatement.start.value, abatement.
end.value]
else if abatement is FHIR.string then Message(null as Interval<DateTime>, true, '1', 'Error', 'Cannot

```



```

compute an interval from a String value')
  else if abatement is FHIR.Age then Interval[FHIRHelpers.ToDate ( Patient.birthDate ) +
FHIRHelpers.ToQuantity ( abatement as FHIR.Age ), FHIRHelpers.ToDate ( Patient.birthDate ) +
FHIRHelpers.ToQuantity ( abatement as FHIR.Age ) + 1 year )
  else if abatement is FHIR.Range then Interval[FHIRHelpers.ToDate ( Patient.birthDate ) +
FHIRHelpers.ToQuantity ( ( abatement as FHIR.Range ).low ), FHIRHelpers.ToDate ( Patient.birthDate ) +
FHIRHelpers.ToQuantity ( ( abatement as FHIR.Range ).high ) + 1 year )
  else null

```

- **NCQA_FHIRBase-1.0.0."Normalize Interval"**

```

define function "Normalize Interval"(choice Choice<FHIR.dateTime, FHIR.date, FHIR.Period, FHIR.Timing,
FHIR.instant, FHIR.string, FHIR.Age, FHIR.Range>):
  case
    when choice is FHIR.dateTime then Interval[FHIRHelpers.ToDateTime ( choice as FHIR.dateTime ),
FHIRHelpers.ToDateTime ( choice as FHIR.dateTime )]
    when choice is FHIR.date then Interval[ToDateTime(choice as FHIR.date), ToDateTime(choice as
FHIR.date)]
    when choice is FHIR.Period then Interval[( choice as FHIR.Period ).start.value, ( choice as
FHIR.Period ).
end.value]
    when choice is FHIR.instant then Interval[FHIRHelpers.ToDateTime ( choice as FHIR.instant ),
FHIRHelpers.ToDateTime ( choice as FHIR.instant )]
    when choice is FHIR.Age then Interval[FHIRHelpers.ToDate ( Patient.birthDate ) +
FHIRHelpers.ToQuantity ( choice as FHIR.Age ), FHIRHelpers.ToDate ( Patient.birthDate ) +
FHIRHelpers.ToQuantity ( choice as FHIR.Age ) + 1 year )
    when choice is FHIR.Range then Interval[FHIRHelpers.ToDate ( Patient.birthDate ) +
FHIRHelpers.ToQuantity ( ( choice as FHIR.Range ).low ), FHIRHelpers.ToDate ( Patient.birthDate ) +
FHIRHelpers.ToQuantity ( ( choice as FHIR.Range ).high ) + 1 year )
    when choice is FHIR.Timing then Message(null as Interval<DateTime>, true, '1', 'Error', 'Cannot
compute a single interval from a Timing type')
    when choice is FHIR.string then Message(null as Interval<DateTime>, true, '1', 'Error', 'Cannot
compute an interval from a String value')
  else null as Interval<DateTime>
end

```

- **NCQA_FHIRBase-1.0.0."Normalize Onset"**

```

define function "Normalize Onset"(onset Choice<FHIR.dateTime, FHIR.Age, FHIR.Period, FHIR.Range,
FHIR.string>):
  if onset is FHIR.dateTime then Interval[FHIRHelpers.ToDateTime ( onset as FHIR.dateTime ),
FHIRHelpers.ToDateTime ( onset as FHIR.dateTime )]
  else if onset is FHIR.Period then Interval[onset.start.value, onset.
end.value]
  else if onset is FHIR.string then Message(null as Interval<DateTime>, true, '1', 'Error', 'Cannot
compute an interval from a String value')
  else if onset is FHIR.Age then Interval[FHIRHelpers.ToDate ( Patient.birthDate ) +
FHIRHelpers.ToQuantity ( onset as FHIR.Age ), FHIRHelpers.ToDate ( Patient.birthDate ) +
FHIRHelpers.ToQuantity ( onset as FHIR.Age ) + 1 year )
  else if onset is FHIR.Range then Interval[FHIRHelpers.ToDate ( Patient.birthDate ) +
FHIRHelpers.ToQuantity ( ( onset as FHIR.Range ).low ), FHIRHelpers.ToDate ( Patient.birthDate ) +
FHIRHelpers.ToQuantity ( ( onset as FHIR.Range ).high ) + 1 year )
  else null

```

- **NCQA_FHIRBase-1.0.0."Prevalence Period"**

```

define function "Prevalence Period"(condition Condition):
  Interval[start of "Normalize Onset"(condition.onset),
end of "Normalize Abatement"(condition.abatement)]

```

- **NCQA_HealthPlanEnrollment-1.0.0."All Coverage Info"**

```

define function "All Coverage Info"(Coverage List<FHIR.Coverage>, participationPeriod Interval<Date>):
  Coverage C
  let IntervalInfo: "CoverageIntervals"(Coverage, participationPeriod),
  Collapsed: "Collapsed Coverage Intervals"(IntervalInfo),
  Adjacent: "Collapsed Coverage Adjacent Intervals"(Collapsed),

```

```

CollapsedFinal: "Collapsed Final Coverage Intervals"(Collapsed, Adjacent)
return Tuple {
  IntervalInfo: IntervalInfo,
  Collapsed: Collapsed,
  Adjacent: Adjacent,
  CollapsedFinal: CollapsedFinal
}

```

- **NCQA_HealthPlanEnrollment-1.0.0."Collapsed Coverage Adjacent Intervals"**

```

define function "Collapsed Coverage Adjacent Intervals"(Intervals List<Interval<Date>>):
  from
    Intervals Coverage1,
    Intervals Coverage2
  where
    end of Coverage1 within 1 day of start of Coverage2
  return Interval[start of Coverage1,
    end of Coverage2]

```

- **NCQA_HealthPlanEnrollment-1.0.0."Collapsed Coverage Intervals"**

```

define function "Collapsed Coverage Intervals"(Intervals List<Interval<Date>>):
  CQLBase."Collapse Date Interval Workaround" ( Intervals )

```

- **NCQA_HealthPlanEnrollment-1.0.0."Collapsed Final Coverage Intervals"**

```

define function "Collapsed Final Coverage Intervals"(collapsedI List<Interval<Date>>, adjacentI
List<Interval<Date>>):
  CQLBase."Collapse Date Interval Workaround" ( "Collapsed Coverage Intervals"(collapsedI)
  union "Collapsed Coverage Adjacent Intervals"(adjacentI))

```

- **NCQA_HealthPlanEnrollment-1.0.0."CoverageIntervals"**

```

define function "CoverageIntervals"(Coverage List<FHIR.Coverage>, participationPeriod Interval<Date>):
  Coverage C
  return Interval[date from start of FHIRBase."Normalize Interval" ( C.period ), date from
  end of FHIRBase."Normalize Interval" ( C.period )]
  intersect Interval[start of participationPeriod,
  end of participationPeriod]

```

- **NCQA_HealthPlanEnrollment-1.0.0."Health Plan Anchor Date Criteria"**

```

define function "Health Plan Anchor Date Criteria"(Coverage List<FHIR.Coverage>, AnchorDate Date,
participationPeriod Interval<Date>):
  if AnchorDate during participationPeriod then exists ( ( "All Coverage Info"("Health Plan Coverage
Resources"(Coverage), participationPeriod).CollapsedFinal ) FinalInterval
  where AnchorDate in FinalInterval
)
  else if not ( AnchorDate during participationPeriod ) then exists ( ( "Health Plan Coverage
Resources"(Coverage).period ) Cperiod
  where AnchorDate during Interval[date from start of FHIRBase."Normalize Interval" ( Cperiod ), date
from
  end of FHIRBase."Normalize Interval" ( Cperiod )]
)
  else if AnchorDate is null then true
  else false

```

- **NCQA_HealthPlanEnrollment-1.0.0."Health Plan Coverage Resources"**

```

define function "Health Plan Coverage Resources"(Coverage List<FHIR.Coverage>):
  ( Coverage C
  where exists C.type.coding cTypeCoding
  where ( FHIRHelpers.ToCode ( cTypeCoding ) ~ Terminology."managed care policy"
  or FHIRHelpers.ToCode ( cTypeCoding ) ~ Terminology."retiree health program"
  or FHIRHelpers.ToCode ( cTypeCoding ) ~ Terminology."subsidized health program"

```

```
) ) HPCoverageResource
where HPCoverageResource is not null
```

- **NCQA_HealthPlanEnrollment-1.0.0."Health Plan Enrollment Criteria"**

```
define function "Health Plan Enrollment Criteria"(Coverage List<FHIR.Coverage>, AnchorDate Date,
participationPeriod Interval<Date>, AllowedGapDays Integer):
  ( CQLBase."Date Interval Gaps Relative to Base Interval Stats" ( participationPeriod, "All Coverage
Info"("Health Plan Coverage Resources"(Coverage), participationPeriod).CollapsedFinal ) )
GapsInEnrollment
  return ( GapsInEnrollment."Interval Count" <= 1
          and GapsInEnrollment."Total Days In Longest Interval" <= AllowedGapDays
          and "Health Plan Anchor Date Criteria"(Coverage, AnchorDate, participationPeriod)
        )
```

- **NCQA_Status-1.0.0."Active Condition"**

```
define function "Active Condition"(Condition List<FHIR.Condition>):
  Condition C
  where C.clinicalStatus = Terminology."active"
```

- **NCQA_Status-1.0.0."Completed or Ongoing Procedure"**

```
define function "Completed or Ongoing Procedure"(Proc List<FHIR.Procedure>):
  Proc P
  where P.status in { 'completed', 'in-progress' }
```

- **NCQA_Status-1.0.0."Dispensed Medication"**

```
define function "Dispensed Medication"(Med List<FHIR.MedicationDispense>):
  Med M
  where M.status = 'completed'
```

- **NCQA_Status-1.0.0."Finished Encounter"**

```
define function "Finished Encounter"(Enc List<FHIR.Encounter>):
  Enc E
  where E.status = 'finished'
```