|  |
| --- |
| Configuring the RCHC Relevant QIP ECDS Reports  (eCW Health Centers, Version 2) |
| RCHC Logo with words |
| Author: Ben Fouts, Informatics  Redwood Community Health Coalition  1310 Redwood Way, Petaluma, California 94954  [support@rchc.net](mailto:support@rchc.net)  Document Last Updated: 9/19/2022 1:02 PM |

Table of Contents

[Introduction 3](#_Toc114485001)

[Project Overview 3](#_Toc114485002)

[Supporting Documents 3](#_Toc114485003)

[Measurement Period 5](#_Toc114485004)

[Partnership Patients 5](#_Toc114485005)

[Submitting the Files 6](#_Toc114485006)

[Naming the Submission Templates 6](#_Toc114485007)

[Appendices 7](#_Toc114485008)

[General Comments on the SQL Code 8](#_Toc114485009)

[Description of ECDS Measure Report: Breast Cancer Screening (BCS-E) 9](#_Toc114485010)

[Description of ECDS Measure Report: Follow-Up Care for Children Prescribed ADHD Medication (ADD-E) 11](#_Toc114485011)

[Description of ECDS Measure Report: Depression Related Measures (DMS-E, DSF-E, DRR-E, PND-E, PDS-E) 13](#_Toc114485012)

[Description of ECDS Measure Report: Unhealthy Alcohol Use Screening and Follow Up (ASF-E) 17](#_Toc114485013)

[Appendix A: SQL Code for the Report “ECDS: Breast Cancer Screening (BCS-E)” 20](#_Toc114485014)

[Appendix B: SQL Code for the Report “ECDS: Follow-Up Care for Children Prescribed ADHD Medication (ADD-E)” 29](#_Toc114485015)

[Appendix C: SQL Code for the Report “ECDS: Depression Related Measures (DMS-E, DSF-E, DRR-E, PND-E, PDS-E)” and Related Transformers/Data Elements 37](#_Toc114485016)

[Appendix D: SQL Code for the Report “ECDS: Unhealthy Alcohol Use Screening and Follow Up (ASF-E)” and Related Transformers/Data Elements 56](#_Toc114485017)

# Introduction

## Project Overview

Partnership Healthplan is moving towards a model where health centers securely submit detailed patient data, which is then combined with administrative and other data sources Partnership possesses to more accurately evaluate the HEDIS measures reported to the State of California. This is referred to as Electronic Medical Record Data Exchange (ECDS) and Partnership is beginning with a number of measures that are designated by HEDS as electronic measures. There are four groups of measures that correspond to four Relevant reports that will be used to submit four data files to Partnership. These are:

1. Breast Cancer Screening (one measure)
2. Follow-Up Care for Children Prescribed ADHD Medication (one measure, with two aspects)
3. Depression Related Measures (five measures)
4. Unhealthy Alcohol Use Screening and Follow Up (one measure, with two aspects)

RCHC is developing a set of ECDS Quality Measures in Relevant that can be used to track the electronic measures based on data from the health center EHR. In contrast, the data submission reports are designed to suppliment data that Partnership already has from claims, pharmacies, outside screening providers, etc. The Relevant data reports themselves cannot be used to define the denominator or numerator of the Quality Measures.

Partnership will pay a $5,000 incentive to health centers that submit a set of test files and a set of final files for the 2022 calendar year. The deadlines and other details are described in the Supporting Documents below.

## Supporting Documents

This project was developed in collaboration with Partnership Healthplan of California. Therefore, the basic documents come from their website. As of the writing of these instructions, the link to the instroduction presentation slides and recording are on the Partnership Perinatal QIP webpage[[1]](#footnote-1). There are screenshots and additional detail in the 7/2022 RCHC Workgroup presentation described below.

The are important Partnership documents and templates available through the eReports Portal (for PCP QIP). Health centers are strongly encouraged to download these materials to get a better understanding of the project and how to submit the data. These materials include:

1. The measure requirements (i.e., the standards and specifications)
2. The Value Set lists (for your information)
3. Four reporting templates (data will be copied to these and sent securely to Partnership)
4. A document containing SQL code examples for both eCW and NextGen. Even though RCHC health centers can adapt this SQL for their own reports, the SQL code specific to Relevant described below is better suited for this project.

On the RCHC Website (under Initiatives \ Population Health \ Data Analytics and Governance \ Program \ Data Workgroup and Report Documentation \ Additional Resources and Companion Documents \ Partnership ECDS Reporting [[2]](#footnote-2) ), there are two sets of slides and webinars:

1. The first set (both files marked 3/2022) is a project overview and description of the ECDS measures themselves. This presentation is named “Partnership Electronic Clinical Data Systems (ECDS) Measures.”
2. The second set (both files marked 7/2022) describes the ECDS reports and is more relevant to the instructions described in sections below. This presentation is named “New ECDS Measures: Focus on Reports.”

The 2022 RCHC Analytics Academy featured a presentation on how to copy reports from the RCHC Instance of Relevant and adapt them to the health center’s instance of Relevant. This training was called “Can’t Wait for the SQL” and featured slides, handouts and a webinar recording. These materials can also be found on the RCHC website (under Initiatives \ Population Health \ Data Analytics and Governance \ RCHC Analytics Academy \ 2022 Analytics Academy [[3]](#footnote-3) ).

## Measurement Period

To keep things simple and straightforward, all four reports will use the same measurement period for both the test file and the final file: 1/1/2022 to 12/31/2022. The report SQL contains parameters, so the user will be prompted to enter a measurement period date range. The reports automatically make calculations for look-back periods. Note that this is a different approach than described in the general Partnership specifications and SQL templates.

The measurement period is defined on the report by the parameters {{start\_date}} and {{end\_date}}.

## Partnership Patients

Data submission must include Partnership Managed Care patients only. The same approach to identifying these patients should be used in the SQL code of all four reports. The universe of Partnership patients (along with other aspects of the denominator) is identified in the universe TEMPORARY TABLE of the report SQL code.

The SQL default code for the reports identifies Partnership patients based on an insurance group ID for Partnership Managed Care. This is written as **primary\_insurance\_group\_id** = 17 in the WHERE clause of the universe TEMPORARY TABLE. If your health center is going to use this default, identify the unique insurance group ID for Partnership Managed Care patients and replace the default group ID[[4]](#footnote-4).

If a Partnership Managed Care insurance group has not been defined in your EHR (or Relevant), another option is to use individual insurance identification numbers associated with Partnership Managed Care from the field patients.primary\_insurance\_id

If your health center has integrated monthly Partnership patient lists into Relevant, then add SQL code to JOIN these patients (or add a statement to the WHERE clause, etc., depending how they appear in your instance). With this approach, also remove the default SQL code in the WHERE statement for primary\_insurance\_group\_id and the visit date range of the universe TEMPORARY TABLE.

Because the list of patients enrolled in MediCal Managed Care (through Partnership) changes regularly, the term “anchor date” referrs to the list of enrolled patients at a particular time. The test files are due on October 1, 2022, so use an anchor date of September 1, 2022. The final files are due January 28, 2023, so use an anchor date of December 1, 2022 (which is the same as the other QIP reports).

## Submitting the Files

There are two data submissions. The measurement period for both submissions (entered into the parameters) is 1/1/2022 to 12/31/2022.

The first submission is a “test” file that Partnership will use to assess their own data integration systems. They may even provide feedback to health centers that can be used to modify the Relevant report. This file is due by October 1, 2022 and should be prepared some time in September 2022. If the health center has integrated current Partnership patient lists into Relevant, the September 1, 2022 patient list from eReports should be used as the “anchor.”

The second submission is the “final” report and should be run after January 8, 2023 (leaving enough time for visits at the end of December 2022 to be closed and mammogram images reviewed and completed by providers). The final report must be submitted to Partnership before January 28, 2023. If the health center has integrated current Partnership patient lists into Relevant, the December 1, 2022 patient list from eReports should be used as the “anchor.”

Make sure the data copied into the files follow the column formatting that is described in row 2 of the template. It should be possible to do this using the Paste Special function in Excel because the template columns are already formatted by default. NOTE that sometimes the CIN automatically converts to scientific notation when it contains numbers along with the letter E. Scan the list before submitting it to make sure no CIN in the column Member\_Key appear in scientific notation.

## Naming the Submission Templates

Once the data has been validated, the report data should be copied to the appropriate template. There are four templates corresponding to the four Relevant reports.

The blank submission templates have been given general names. When a service provider has entered data and is ready to submit a template, it should be re-named according to the standard on the next page. Utilizing a unique and descriptive report name enables Partnership to ingest the report into their data analysis system.

The following instructions come from the Partnership SQL template document. The introduction presentation (see the section Supporting Documents above) from June 2022 describes how the files should be sent to Partnership.

The file name should follow the convention MEASUREID\_SITENAME\_DATE where:

* **MEASUREID** corresponds to the measure abbreviation and is one of the following:

|  |  |
| --- | --- |
| MEASUREID | Measure name |
| ADD | Follow-Up Care for Children Prescribed ADHD Medication |
| BCS | Breast Cancer Screening |
| ASF | Unhealthy Alcohol Use Screening and Follow-Up |
| DEP | Five depression measures (submitted as one file) |

* **SITENAME** is the name of the health center. The health center should choose a brief and easily recognizable name or abbreviation and use it consistently with all data submissions
* **DATE** is the submission date in the format YYYYMMDD with no dashes, periods or slashes

Examples: ASF\_SHASTA\_20220930

ADD\_PETALUMA\_20230115

## Appendices

The next four sections of this document describe the RCHC ECDS reports and what is needed to adapt the code to the health center’s instance of Relevant. The appendices contain the first draft of the basic SQL code that can be copied to DataGrip or Relevant and customized there. Alternatively, health centers can copy the code directly from the reports section of the RCHC Instance of Relevant.

In the reports section of the RCHC instance of Relevant, there is a Report Set called the PHP ECDS Data Submission that contains the four reports that can be copied to the health center instance. These reports should not be actually run on the RCHC instance of Relevant (they will either show zero rows or an error will appear). Once the report is copied to the health center instance, the “Exposes PHI” checkbox should be checked “on” before it is published.

Health center programmers are encouraged to send feedback on the SQL code (comments on the approach, logic, expressions, etc.) to Ben Fouts ([bfouts@rchc.net](mailto:bfouts@rchc.net)) so that the code can be improved during the validation process. Changes will be discussed on the RCHC Slack Population health channel and the report SQL code on the RCHC Instance updated, if necessary. Therefore, reports on the RCHC Instance reports are the best and most updated source of SQL code.

## General Comments on the SQL Code

The references to SQL code in this document (including the appendices) is for health centers using Relevant and the eCW Electronic Health Record. NextGen SQL code will be developed separately, although the approach and many of the ideas are the same.

The report SQL code will need to be inspected and adapted to each health center’s instance. This is especially true for the two reports that are based on the Staging Database. See the materials from the 2022 RCHC Analytics Academy mentioned above in the section Supporting Documents for an explanation on how this can be done by the SQL programmer. The two reports based on the Production Database will probably require less customization, although one of them requires new Data Elements.

The outline of the SQL code is generally the same for all four reports. Temporary tables are used to define the universe and different aspects of the measure, which are all brought together in the results query. The basic flow of the temporary tables begins with the patient universe and moves on to other aspects of the measure displayed in the columns. Then exclusions are evaluated and displayed in the required columns. The results query that displays the report columns in the same order and format as the reporting template is the final section of the code.

Where the descriptive codes[[5]](#footnote-5) needed by the report follow exactly the HEDIS Value Set codes, a JOIN is made directly to the HEDIS Value Set table in Relevant (the table name is hedis\_value\_set\_codes, which exists on both the Staging and Production Databases). On two of the four reports, the first TEMPORARY TABLE contains additional codes specified by Partnership or codes not included on the HEDIS table.

There are grey comments/notes embedded in the SQL that help explain the objective of the temporary tables or specific aspects of the code. The descriptions of each report below also direct the programmer to locations in the code that must be customized or otherwise scrutinized. It is unlikely that any of the reports besides the Breast Cancer Screening report will run immediately after copying the code to Relevant.

# Description of ECDS Measure Report: Breast Cancer Screening (BCS-E)

This report is based on the Relevant Production Database and so no further mapping is needed. Health centers generally have all Data Elements already mapped for the related UDS and QIP Quality Measures.

Below are some comments about sections of the SQL code. These comments reference the specific temporary tables or the final results table contained in the code of the appendix.

TEMPORARY TABLE universe\_temp\_raw

See notes in the Introduction above on identifying patients with Partnership Insurance

TEMPORARY TABLE mammograms\_temp

Since most health centers do not perform mammograms themselves, the CPT and HCPCS Value Set options are not used. Instead, a SNOMED code is displayed for all mammograms to signify that they come from structured data (i.e., diagnostic images).

Note on Exclusion TEMPORARY TABLES

Partnership wants to know if the patient has any exclusions and provides four columns for codes and dates. The report gathers the Value Set diagnosis codes from two locations in the medical record, prioritizes them (in case there are more than 4 codes at one time, which is possible but likely rare), and displays them in the columns. Both the diagnosis code and the associated date are displayed.

The locations and a description of the data are as follows:

1. Diagnosis codes associated with a visit (from Assessments). These are associated with visits during the 27 months before the end of the measurement period, which is the same period from which mammograms are extracted. In the Results Query, these are displayed by assiciation with a particular date of service (column Date\_of\_Service).
2. Diagnosis codes on the patient’s Problem List. These are not associated with a particular visit and so no date is displayed in the column Date\_of\_Service. These codes can have a start date any time before the end of the measurement period.

It is possible (and even likely) that exclusion codes will be duplicated among rows displaying visits and Problem List results. The intent is for the report to cast a wide net to identify any exclusions. Partnership will determine if the patient should be excluded based on the diagnosis code and associated date.

Results Query

Three groups of records are added together using UNION statements. They are:

1. Mammograms in the 27 months before the and of the measurement period end date. These are associated with a date in the column Date\_of\_Service
2. Exclusions from Assessments associated with a date in the column Date\_of\_Service
3. Exclusions from the Problem List not associated with a date of service (the column Date\_of\_Service will be blank)

The facility where the visit was performed is displayed (in column Site\_Name) when an exclusion is identified from an Assessment. In these rows, the Partnership Site ID number should be added to the report manually (to column Site\_ID\_Number) because it does not exist in Relevant. This can be done in the second subquery (most likely using a CASE WHEN statement depending on site name) or copied into the template directly. The Site\_ID\_Number is the NPI number assigned to the site where care was provided (i.e., the site in the column Site\_Name). Since mammograms and Problem List exclusion codes are not associated with a particular visit to the health center, no Site Name or Site ID Number are needed for these rows.

# Description of ECDS Measure Report: Follow-Up Care for Children Prescribed ADHD Medication (ADD-E)

This report is based on the Relevant Staging Database. Because this Database does not have a high degree of standardization among health centers, some of the SQL code will need to be adapted by the health center programmer using the unique table and field names in the health center instance as well as any “best practices” the health center has already developed to extract particular sets of data.

TEMPORARY TABLE temp\_value\_sets

This table provides codes that are not part of the HEDIS Value Sets but are referenced by Partnership. These consist of some additional visit codes and also the ADHD diagnosis codes.

TEMPORARY TABLE universe\_temp\_raw

See notes in the Introduction above on identifying patients with Partnership Insurance. Visits displayed in the universe must have a combination of a visit code (a code from the six HEDIS Value Sets and the additional Partnership codes) and an ADHD diagnosis code.

The visit measurement period is calculated. To keep things simple for all reports in the ECDS set, the measurement period entered by the user is 1/1/2022 to 12/31/2022. However, the report will make a calculation for the start date from which visits are extracted (the end date is the same as the end date entered by the user).

So, for example, if the measurement period entered by the user is 1/1/2022 to 12/31/2022, the visit span begins on 3/1/2021. The earliest visit extracted by the report corresponds to the start of the measure’s “Intake Period” which is defined as “March 1 of the year prior to the Measurement Period” (the reference is the HEDIS specifications). The report can be run for any year-long measurement period (not just a proper calendar year), but the visit extraction period will be from 10 months before the start of the measurement period to the end of the measurement period.

Results Query

There are two ID numbers needed on this report that likely do not exist in your EHR. However, if they do exist in the EHR, the programmer can reference the field in the SQL code. If they do not exist, they can be added manually to the results in the Excel template or to the SQL code using a CASE WHEN statement.

These ID numbers are:

1. Field “Site\_ID\_Number.” This is the NPI number assigned to the site where care was provided (i.e., the site displayed in the field Site\_Name)
2. Field "Provider\_Key." This is the PCP ID assigned by Partnership to the provider who provided care during the visit (i.e., the provider displayed in Clinician\_First\_Name and Clinician\_Last\_Name)

# Description of ECDS Measure Report: Depression Related Measures (DMS-E, DSF-E, DRR-E, PND-E, PDS-E)

Note that some health centers might already have this report in their instance of Relevant because it was part of a Partnership pilot program last year. If the data on that report has been thoroughly validated, the health center may continue to use it. The SQL code in Appendix C has been improved from the initial version and so programmers may consider using the new code or parts of the new code.

The depression measures report is built on the Relevant Staging Database. Because this Database does not have a high degree of standardization among health centers, some of the SQL code will need to be adapted by the health center programmer using the unique table and field names in the health center instance as well as any “best practices” the health center has already developed to extract particular sets of data.

This report first requires some configuration of Transformers. Although Data Elements are not used directly by the report, the related Quality Measures rely on the Data Elements. Therefore, Transformers and Data Elements will be described here together as pairs (where applicable) and should be configured together.

1. The report displays the actual numerical score of the PHQ-2 and PHQ-9 instruments. Depending how your Transformers are configured, you may already have tables with all PHQ-2 and PHQ-9 dates and scores. The sample report SQL in the appendix (see TEMPORARY TABLE depression\_screenings\_temp) references two tables that were configured at one of the health centers[[6]](#footnote-6). Note that the common Transformer relevant\_depression\_screens may or may not display the raw score and may or may not contain/distinguish between PHQ-2 and PHQ-9 instruments. The numerical screening score is not needed for the Quality Measure (and is therefore not added to any Data Element).
2. Identify other depression screening instruments used by your health center. There is a list of acceptable instruments in the Partnership measure standards document. You may need to communicate with the medical officer or behavioral health department lead at your health center for help. The report RCHC List All Structured Data Items (for eCW users) can be used to find items in structured data (most commonly HPI) where depression screening scores are entered. This report also shows category, symptom and detail ID numbers which are unique to the health center. The presentation with the title “New ECDS Measures: Focus on Reports” mentioned in the section Supporting Documents (above, at the beginning of this document) discusses an approach to finding these screens in structured data.
3. The most common non-PHQ screening instrument is the Edinburgh Depression Screen, which is used with prenatal and postnatal patients (two of the related depression Quality Measures focus on these populations). There is a new Data Element (Edinburgh Depression Screens) that can be configured and enabled if your health center uses this instrument. Example SQL code for the Transformer/Data Element Pair is displayed in Appendix C.
4. Health centers may use additional depression screening instruments. Currently, there are no other standard Transformers/Data Elements for these instruments. However, they can be added to the TEMPORARY TABLE depression\_screenings\_temp in the report using UNION statements and SQL to pull them directly from structured data. There is an example in Appendix C.
5. The report displays the Last Menstrual Period (LMP) and the Estimated Due Date (EDD). These fields should be added to the standard Transformer relevant\_pregnancy and the standard Data Element named Pregnancies (these fields have recently been added to the Relevant Data Model). Example SQL for these fields is displayed in Appendix C.

TEMPORARY TABLE office\_cpt\_codes\_temp

This table provides codes for office visits referenced by Partnership.

TEMPORARY TABLE initial\_denom\_codes\_temp

See notes in the Introduction above on identifying patients with Partnership Insurance

TEMPORARY TABLE preg\_mp\_temp

The report requires that calculations be made for gestational age. Therefore, two new fields need to be added to the Transformer relevant\_pregnancy. These fields are:

* last\_menstrual\_period
* estimated\_due\_date

See Appendix C for suggested SQL that can be added to the Transformer. However, note that since this change is being made on the non-standardized Staging database, the programmer may need to customize the SQL code according to data location or unique approaches.

These two fields should also be added to the standard Data Element named Pregnancies. Appendix C also contains suggested SQL for the Data Element.

TEMPORARY TABLE future\_delivery\_temp

It is possible that there could be more than one pregnancy overlapping the measurement period, as identified in the TEMPORARY TABLE preg\_mp\_temp. In cases where the patient did not deliver before the end of the measurement period and where the Estimated Due Date (EDD) is after the end of the measurement period, the TEMPORARY TABLE future\_delivery\_temp will choose the pregnancy with an EDD closest to the end of the measurement period.

TEMPORARY TABLE past\_delivery\_temp

It is possible that there could be more than one pregnancy overlapping the measurement period, as identified in the TEMPORARY TABLE preg\_mp\_temp. In cases where there is at least one known delivery date during the measurement period, the TEMPORARY TABLE past\_delivery\_temp will choose the earliest delvery date.

Note that it is possible that a single patient can have a delivery in the measurement period (and thus a date in the column "Actual (past) date of delivery within 12 months") and a future due date period (and thus a date in the column "Estimated date of delivery (future)").

TEMPORARY TABLE preg\_columns\_temp

Note that the report will not display a delivery date unless the delivery date was entered into the EHR. Therefore, patients who did not deliver (i.e., pregnancy ended for some reason before delivery) or patients missing this information will have visits where the patient was apparently pregnant (column Pregnant at time of Visit = “Yes”) but with no EDD (column “Estimated date of delivery (future)”) and no delivery date (column “Actual (past) date of delivery within 12 months”).

TEMPORARY TABLE depression\_screenings\_temp

This table gathers the results of all of the screening instruments your health center uses. The PHQ-2 and PHQ-9 are almost universal among health centers, but the programmer should ensure that all other instrument results are also gathered by the code. Each individual instrument is added to the table with a UNION expression.

A list of HEDIS-approved instruments appears in the Partnership specifications document after the section “Additional Background Information.” Also see point #2 above in the section on configurating Transformers for this measure. The list in the Partnership document contains the instrument’s official LOINC code that goes into the column loinc\_code\_screen. The instrument abbreviation from the specifications document goes into the report column tool\_name. The result of the depression screen (either 'Positive' or 'Negative') goes into the report column interp\_result, which depends on the numerical score of the screen (column screen\_score). The calculation used to obtain a result from a score in the report should correspond to the calculation for a positive finding in the table of the specifications document (and should also correspond to how the health center EHR interprets the score).

The SQL code contains SELECT queries for the PHQ-2 and PHQ-9 (see point #1 above in the section on configurating Transformers for this measure). There is also a greyed-out (in other words, “noted”) SELECT query for the Edinburgh Depression Screen (based on the new Transformer, if your health center uses the Edinburgh) and a greyed-out SELECT query for any other screening instrument your health center might have. These can be activated if your health center uses additional depression screens other than the PHQ-2 and PHQ-9. Otherwise, the greyed-out SQL can remain noted or be deleted from the code. Add as many UNION queries as necessary to get all of the depression instruments used. Data for instruments other than the Edinburgh can be extracted directly from Structured Data in the TEMPORARY TABLE or a custom Transformer created (there would be no corresponding Data Element, though).

# Description of ECDS Measure Report: Unhealthy Alcohol Use Screening and Follow Up (ASF-E)

This report is based on the Relevant Production Database, and so a higher degree of standardization here means that the programmer does not need to customize the SQL code as much. However, this report requires two new Transformer-Data Element pairs.

Similar to the other screening/follow-up measures (like depression screening/follow-up or tobacco screening/follow-up), this measure focuses on unhealthy alcohol use screenings for all patients in the denominator and then, for patients with a positive screen, evaluates whether a follow-up activity was performed. The presentation with the title “New ECDS Measures: Focus on Reports” mentioned in the section Supporting Documents (above) contains a flow-chart for this measure.

Since this is a new measure among RCHC health centers, unhealthy alcohol use screens and alcohol follow-up (i.e., counseling in some form) must be identified in the EHR (normally in structured data, if they are already being entered) and extracted by the new Transformer/Data Element pair.

The Partnership ECDS measure specifications document defines three different instruments that can be used for alcohol screening. These are:

* Audit-C
* Audit
* Single Question Screen. Note that this screen must adhere to specific language in the document or be approved by Partnership.

A health center can have one or any combination of these screens. The first step is to work with a clinical or behavioral health contact at your health center to help identify the alcohol use screens being used. Then, identify the screens in your structured data[[7]](#footnote-7). Each screening type should have its own Transformer/Data Element pair.

|  |  |  |
| --- | --- | --- |
| Instrument Name | Suggested Transformer Name | Data Element Name |
| Audit-C | relevant\_audit\_c\_screens | AUDIT C Screens |
| Audit | relevant\_audit\_screens | AUDIT Screens |
| Single Question Screen | relevant\_single\_q\_alc\_screens | Single Question Alcohol use Screens |

Appendix D contains example SQL for the Audit-C Transformer/Data Element pair. If your health center uses the Audit screen and/or single question alcohol use screen, the Transformer/Data Element pair will have similar SQL code. The programmer will need to customize the SQL with the unique internal ID numbers and location of data.

Appendix D contains example SQL for the alcohol counseling Transformer/Data Element pair.

|  |  |
| --- | --- |
| Suggested Transformer Name | Data Element Name |
| relevant\_alcohol\_counseling | Alcohol Counseling Or Other Followups |

TEMPORARY TABLE universe\_temp\_raw

See notes in the Introduction above on identifying patients with Partnership Insurance

The visits are not restricted to medical visits or any other particular kind of visit because we want as many instances of alcohol screenings and counselings as possible.

TEMPORARY TABLE alc\_followup\_struct\_raw\_temp

Partnership will identify alcohol follow-up counseling from structured data through SNOMED codes displayed in the output column Counseling\_SNOMED. The EHR systems used by RCHC health centers do not automatically associate structured data items with SNOMED codes, so the SNOMED code must be added in the SQL. In the section “Additional Background Information” of the Partnership measure specifications document (see section “Supporting Documents” above) is a table with some recommeneded SNOMED codes for counseling. The sample SQL in Appendix D uses the SNOMED code 413473000 (corresponding to “Counseling about alcohol consumption”), but any of the codes in the table can be used if one better describes the exact kind of counseling at your health center.

Results Query

Two groups of records are added together using UNION statements. They are:

1. Individual visits in the measurement period where there was an alcohol screen and/or an alcohol counseling activity. These records feature the visit date in the column Date\_of\_Service and other visit-associated information.
2. Denominator patients with exclusions. These are not associated with a visit date in the column Date\_of\_Service but have dates and codes in the exclusion columns (those with names that begin with “Exclusions\_”).

Note that the data for two output columns associated with individual visits are likely not present in the EHR. These columns are “Site ID number” and “Provider Site Name.” The data in these fields should align with the Partnership ID and recognized name of the site. In the default SQL, a NULL value is present for these fields.

If the site data cannot be extracted from the EHR, there are at least two options. First, a programmer can use a CASE WHEN statement to code the fields based on the Relevant location where the service was given (in the results query, the text location is in the field universe\_temp\_raw.Site\_Name). Alternately, the location can be displayed in the query results and then replaced manually in Excel.

# Appendix A: SQL Code for the Report “ECDS: Breast Cancer Screening (BCS-E)”

**Database:** Production Database

**Description**: This report displays data for the ECDS Breast Cancer Screening (BCS-E) measure that could be submitted to Partnership HealthPlan of California (2022 Edition)

**Parameters**: {{start\_date}} and {{end\_date}}

**Report SQL Code:**

*/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
Report Name: ECDS: Breast Cancer Screening (BCS-E)  
  
Code Edited By: Ben Fouts for RCHC  
  
Description: This report displays data for submission to Partnership Healthplan. It should not be used to  
 directly evaluate a Quality Measure or for case management purposes  
  
RCHC Started: June 2022 (based on code from the Partnership SQL specifications document)  
  
Version Date: July 25, 2022  
  
Revision History: 2022 Version 1  
  
Resources Used: patients, visits, visit\_set\_memberships, locations, mammograms, hedis\_value\_set\_codes,   
 visit\_billing\_codes, billing\_codes, visit\_diagnosis\_codes, patient\_diagnoses, diagnosis\_codes*

*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/*

*-- Universe Definition: visits with a claim that has:  
-- 1) Patient age between age 52 years, 0 days to 74 years, 364 days at the end of the measurement period  
-- 2) Patient sex = female (change this if your health center has a more precise way of defining it)  
-- 3) Patient currently has Partnership HealthPlan insurance -- this is the default, but can be changed  
-- by the programmer to join to internal tables containing current Partnership patients  
-- from the last membership file***DROP TABLE IF EXISTS** universe\_temp\_raw;  
**CREATE TEMPORARY TABLE** universe\_temp\_raw **AS  
SELECT DISTINCT  
 id AS** patient\_id,  
 **mrn**,  
 **subscriber\_number AS** Member\_Key,  
 **first\_name AS** Member\_First\_Name,  
 **NULL AS** Member\_Middle\_Name,  
 **last\_name AS** Member\_Last\_Name,  
 **date\_of\_birth** :: **DATE AS** DOB,  
 *EXTRACT*(**YEAR FROM** *age*({{end\_date}} :: DATE, date\_of\_birth :: DATE)) **AS** Age\_endMP,  
 **'F' AS** Sex  
**FROM** patients  
**WHERE assigned\_female\_at\_birth IS TRUE  
 AND** *EXTRACT*(**YEAR FROM** *age*({{end\_date}} :: DATE, date\_of\_birth :: DATE)) **BETWEEN** 52 **and** 74  
 **AND primary\_insurance\_group\_id** = 17 *-- "PARTNERSHIP MANAGED CARE" -- this will be different at every health center* **AND** *EXISTS*(  
 **SELECT  
 FROM** visits  
 **INNER JOIN** visit\_set\_memberships **ON** visits.**id** = visit\_set\_memberships.**visit\_id  
 WHERE** visits.**visit\_date** :: **DATE BETWEEN** {{start\_date}} **AND** {{end\_date}}  
 **AND** visit\_set\_memberships.standard\_visit\_set\_id = **'uds\_medical'  
 AND** visits.patient\_id = patients.id  
 );  
  
  
*-- Collect mammograms from structured data within the lookback period.  
-- This is the only source of mammograms identified by the report default code  
-- Artificial SNOMED code used so Partnership can identify it***DROP TABLE IF EXISTS** mammograms\_temp;  
**CREATE TEMPORARY TABLE** mammograms\_temp **AS  
SELECT** universe\_temp\_raw.**patient\_id**,  
 **performed\_on AS** Date\_of\_Service,  
 **NULL AS** CPTPX,  
 **NULL AS** CPTMOD,  
 **NULL AS** HCPCSPX,  
 **NULL AS** HCPCSMOD,  
 **NULL AS** LOINC,  
 **'24623002'** :: **VARCHAR AS** SNOMED  
**FROM** universe\_temp\_raw  
 **INNER JOIN** mammograms **ON** mammograms.**patient\_id** = universe\_temp\_raw.**patient\_id  
WHERE performed\_on BETWEEN** ({{end\_date}} :: DATE - **INTERVAL '27 MONTHS'**) **AND** {{end\_date}};  
  
  
*-- Make priority list of exclusion codes that appear as visit-associated diagnosis codes***DROP TABLE IF EXISTS** exclusions\_visit\_list\_temp;  
**CREATE TEMPORARY TABLE** exclusions\_visit\_list\_temp **AS  
SELECT  
 patient\_id**,  
 encounterid,  
 priority,  
 Exclusions\_Code,  
 Exclusions\_DOS,  
 *ROW\_NUMBER* () **OVER** (  
 **PARTITION BY** encounterid  
 **ORDER BY** priority, Exclusions\_Code) **AS** row\_excl  
**FROM**(**SELECT** universe\_temp\_raw.**patient\_id**,  
 visits.**id AS** encounterid,  
 **CASE WHEN value\_set\_name** = **'History of Bilateral Mastectomy' THEN** 1  
 **WHEN value\_set\_name** = **'Absence of Left Breast' THEN** 2  
 **WHEN value\_set\_name** = **'Absence of Right Breast' THEN** 3  
 **WHEN value\_set\_name** = **'Advanced Illness' THEN** 4  
 **WHEN value\_set\_name** = **'Frailty Diagnosis' THEN** 5  
 **ELSE** 9 **END AS** priority,  
 diagnosis\_codes.**code** :: **VARCHAR AS** Exclusions\_Code,  
 visits.**visit\_date** :: **DATE AS** Exclusions\_DOS  
 **FROM** universe\_temp\_raw  
 **INNER JOIN** visits **ON** visits.**patient\_id** = universe\_temp\_raw.**patient\_id  
 INNER JOIN** visit\_diagnosis\_codes **ON** visit\_diagnosis\_codes.**visit\_id** = visits.**id  
 INNER JOIN** diagnosis\_codes **ON** diagnosis\_codes.**id** = visit\_diagnosis\_codes.**diagnosis\_code\_id  
 INNER JOIN** hedis\_value\_set\_codes **ON** hedis\_value\_set\_codes.**code\_value** = diagnosis\_codes.**code**

**AND latest** = **'TRUE'  
 WHERE** visits.**visit\_date** :: **DATE BETWEEN** ({{end\_date}} :: DATE - **INTERVAL '27 MONTHS'**) **AND** {{end\_date}}  
 **AND** hedis\_value\_set\_codes.value\_set\_name **IN**(**'History of Bilateral Mastectomy'**, **'Absence of Left Breast'**,  
 **'Absence of Right Breast'**, **'Advanced Illness'**, **'Frailty Diagnosis'**, **'Frailty Symptom'**)  
 **UNION  
 SELECT** universe\_temp\_raw.patient\_id,  
 visits.id **AS** encounterid,  
 **CASE WHEN** value\_set\_name = **'Frailty Encounter' THEN** 7  
 **ELSE** 8 **END AS** priority,  
 billing\_codes.code:: **VARCHAR AS** Exclusions\_Code,  
 visits.visit\_date :: DATE **AS** Exclusions\_DOS  
 **FROM** universe\_temp\_raw  
 **INNER JOIN** visits **ON** visits.patient\_id = universe\_temp\_raw.patient\_id  
 **INNER JOIN** visit\_billing\_codes **ON** visit\_billing\_codes.visit\_id = visits.id  
 **INNER JOIN** billing\_codes **ON** billing\_codes.id = visit\_billing\_codes.billing\_code\_id  
 **INNER JOIN** hedis\_value\_set\_codes **ON** hedis\_value\_set\_codes.code\_value = billing\_codes.code

**AND latest** = **'TRUE'**  
 **WHERE** visits.visit\_date :: DATE **BETWEEN** ({{end\_date}} :: DATE - **INTERVAL '27 MONTHS'**) **AND** {{end\_date}}  
 **AND** hedis\_value\_set\_codes.value\_set\_name **IN**(**'Frailty Encounter'**, **'Frailty Device'**)) **AS** exclusions\_visit\_list\_raw;  
  
  
*-- Place visit-associated diagnosis codes into columns, by encounter***DROP TABLE IF EXISTS** exclusions\_visit\_list\_final;  
**CREATE TEMPORARY TABLE** exclusions\_visit\_list\_final **AS  
SELECT** enc\_list.**patient\_id**,  
 enc\_list.**encounterid**,  
 enc\_list.**Exclusions\_DOS AS** Date\_of\_Service,  
 excl\_1\_row.Exclusions\_Code\_1,  
 excl\_1\_row.Exclusions\_Code\_1\_DOS,  
 excl\_2\_row.Exclusions\_Code\_2,  
 excl\_2\_row.Exclusions\_Code\_2\_DOS,  
 excl\_3\_row.Exclusions\_Code\_3,  
 excl\_3\_row.Exclusions\_Code\_3\_DOS,  
 excl\_4\_row.Exclusions\_Code\_4,  
 excl\_4\_row.Exclusions\_Code\_4\_DOS  
**FROM**(**SELECT DISTINCT  
 patient\_id**,  
 **encounterid**,  
 **Exclusions\_DOS  
 FROM** exclusions\_visit\_list\_temp) **AS** enc\_list  
**INNER JOIN** (**SELECT DISTINCT  
 patient\_id**,  
 **encounterid**,  
 **Exclusions\_Code AS** Exclusions\_Code\_1,  
 **Exclusions\_DOS AS** Exclusions\_Code\_1\_DOS  
 **FROM** exclusions\_visit\_list\_temp  
 **WHERE row\_excl** = 1) **AS** excl\_1\_row **ON** excl\_1\_row.**encounterid** = enc\_list.**encounterid  
LEFT JOIN** (**SELECT DISTINCT  
 patient\_id**,  
 **encounterid**,  
 **Exclusions\_Code AS** Exclusions\_Code\_2,  
 **Exclusions\_DOS AS** Exclusions\_Code\_2\_DOS  
 **FROM** exclusions\_visit\_list\_temp  
 **WHERE row\_excl** = 2) **AS** excl\_2\_row **ON** excl\_2\_row.**encounterid** = enc\_list.**encounterid  
LEFT JOIN** (**SELECT DISTINCT  
 patient\_id**,  
 **encounterid**,  
 **Exclusions\_Code AS** Exclusions\_Code\_3,  
 **Exclusions\_DOS AS** Exclusions\_Code\_3\_DOS  
 **FROM** exclusions\_visit\_list\_temp  
 **WHERE row\_excl** = 3) **AS** excl\_3\_row **ON** excl\_3\_row.**encounterid** = enc\_list.**encounterid  
LEFT JOIN** (**SELECT DISTINCT  
 patient\_id**,  
 **encounterid**,  
 **Exclusions\_Code AS** Exclusions\_Code\_4,  
 **Exclusions\_DOS AS** Exclusions\_Code\_4\_DOS  
 **FROM** exclusions\_visit\_list\_temp  
 **WHERE row\_excl** = 4) **AS** excl\_4\_row **ON** excl\_4\_row.**encounterid** = enc\_list.**encounterid**;  
  
  
*-- Make priority list of exclusion diagnosis codes that appear on the patient's Problem List***DROP TABLE IF EXISTS** exclusions\_problem\_list\_temp;  
**CREATE TEMPORARY TABLE** exclusions\_problem\_list\_temp **AS  
SELECT  
 patient\_id**,  
 priority,  
 Exclusions\_Code,  
 Exclusions\_DOS,  
 *ROW\_NUMBER* () **OVER** (  
 **PARTITION BY patient\_id  
 ORDER BY** priority, Exclusions\_DOS, Exclusions\_Code) **AS** row\_excl  
**FROM**(**SELECT DISTINCT** patient\_diagnoses.**patient\_id**,  
 **CASE WHEN value\_set\_name** = **'History of Bilateral Mastectomy' THEN** 1  
 **WHEN value\_set\_name** = **'Absence of Left Breast' THEN** 2  
 **WHEN value\_set\_name** = **'Absence of Right Breast' THEN** 3  
 **WHEN value\_set\_name** = **'Advanced Illness' THEN** 4  
 **WHEN value\_set\_name** = **'Frailty Diagnosis' THEN** 5  
 **ELSE** 9 **END AS** priority,  
 diagnosis\_codes.**code AS** Exclusions\_Code,  
 patient\_diagnoses.**start\_date AS** Exclusions\_DOS  
 **FROM** patient\_diagnoses  
 **INNER JOIN** universe\_temp\_raw **ON** universe\_temp\_raw.**patient\_id** = patient\_diagnoses.**patient\_id  
 INNER JOIN** diagnosis\_codes **ON** diagnosis\_codes.**id** = patient\_diagnoses.**diagnosis\_code\_id  
 INNER JOIN** hedis\_value\_set\_codes **ON** hedis\_value\_set\_codes.**code\_value** = diagnosis\_codes.**code**

**AND latest** = **'TRUE'  
 WHERE** hedis\_value\_set\_codes.**value\_set\_name IN**(**'History of Bilateral Mastectomy'**, **'Absence of Left Breast'**,  
 **'Absence of Right Breast'**, **'Advanced Illness'**, **'Frailty Diagnosis'**, **'Frailty Symptom'**)  
 ) **AS** raw\_pl  
**ORDER BY patient\_id**, priority, Exclusions\_DOS, Exclusions\_Code;  
  
  
*-- Place problem list diagnosis codes into columns, by patient***DROP TABLE IF EXISTS** exclusions\_problem\_list\_final;  
**CREATE TEMPORARY TABLE** exclusions\_problem\_list\_final **AS  
SELECT** enc\_list.**patient\_id**,  
 excl\_1\_row.Exclusions\_Code\_1,  
 excl\_1\_row.Exclusions\_Code\_1\_DOS,  
 excl\_2\_row.Exclusions\_Code\_2,  
 excl\_2\_row.Exclusions\_Code\_2\_DOS,  
 excl\_3\_row.Exclusions\_Code\_3,  
 excl\_3\_row.Exclusions\_Code\_3\_DOS,  
 excl\_4\_row.Exclusions\_Code\_4,  
 excl\_4\_row.Exclusions\_Code\_4\_DOS  
**FROM**(**SELECT DISTINCT  
 patient\_id  
 FROM** exclusions\_problem\_list\_temp) **AS** enc\_list  
**INNER JOIN** (**SELECT DISTINCT  
 patient\_id**,  
 **Exclusions\_Code AS** Exclusions\_Code\_1,  
 **Exclusions\_DOS AS** Exclusions\_Code\_1\_DOS  
 **FROM** exclusions\_problem\_list\_temp  
 **WHERE row\_excl** = 1) **AS** excl\_1\_row **ON** excl\_1\_row.**patient\_id** = enc\_list.**patient\_id  
LEFT JOIN** (**SELECT DISTINCT  
 patient\_id**,  
 **Exclusions\_Code AS** Exclusions\_Code\_2,  
 **Exclusions\_DOS AS** Exclusions\_Code\_2\_DOS  
 **FROM** exclusions\_problem\_list\_temp  
 **WHERE row\_excl** = 2) **AS** excl\_2\_row **ON** excl\_2\_row.**patient\_id** = enc\_list.**patient\_id  
LEFT JOIN** (**SELECT DISTINCT  
 patient\_id**,  
 **Exclusions\_Code AS** Exclusions\_Code\_3,  
 **Exclusions\_DOS AS** Exclusions\_Code\_3\_DOS  
 **FROM** exclusions\_problem\_list\_temp  
 **WHERE row\_excl** = 3) **AS** excl\_3\_row **ON** excl\_3\_row.**patient\_id** = enc\_list.**patient\_id  
LEFT JOIN** (**SELECT DISTINCT  
 patient\_id**,  
 **Exclusions\_Code AS** Exclusions\_Code\_4,  
 **Exclusions\_DOS AS** Exclusions\_Code\_4\_DOS  
 **FROM** exclusions\_problem\_list\_temp  
 **WHERE row\_excl** = 4) **AS** excl\_4\_row **ON** excl\_4\_row.**patient\_id** = enc\_list.**patient\_id  
ORDER BY patient\_id**;  
  
  
*-- Results Query  
-- Note: Add "Site\_ID\_Number" from your system or manually to the subquery after the first UNION command.   
-- This subquery displays the site name where the service was provided (other subqueries do not display  
-- visit information). The Site\_ID\_Number is the NPI number assigned to the site  
-- where care was provided (i.e., the Site\_Name)***SELECT DISTINCT** *\** **FROM**(**SELECT** mammograms\_temp.**Date\_of\_Service** :: **DATE**,  
 **null AS** Site\_Name,  
 **null AS** Site\_ID\_Number,  
 universe\_temp\_raw.**Member\_Key**,  
 **null AS** Member\_Name,  
 universe\_temp\_raw.**Member\_First\_Name**,  
 universe\_temp\_raw.**Member\_Middle\_Name**,  
 universe\_temp\_raw.**Member\_Last\_Name**,  
 universe\_temp\_raw.**DOB**,  
 universe\_temp\_raw.**Sex**,  
 mammograms\_temp.**CPTPX** :: **VARCHAR**,  
 mammograms\_temp.**CPTMOD** :: **VARCHAR**,  
 mammograms\_temp.**HCPCSPX** :: **VARCHAR**,  
 mammograms\_temp.**HCPCSMOD** :: **VARCHAR**,  
 mammograms\_temp.**LOINC** :: **VARCHAR**,  
 mammograms\_temp.**SNOMED** :: **VARCHAR**,  
 **NULL AS** Exclusions\_Code\_1,  
 **NULL AS** Exclusions\_Code\_1\_DOS,  
 **NULL AS** Exclusions\_Code\_2,  
 **NULL AS** Exclusions\_Code\_2\_DOS,  
 **NULL AS** Exclusions\_Code\_3,  
 **NULL AS** Exclusions\_Code\_3\_DOS,  
 **NULL AS** Exclusions\_Code\_4,  
 **NULL AS** Exclusions\_Code\_4\_DOS  
**FROM** universe\_temp\_raw  
 **INNER JOIN** mammograms\_temp **ON** mammograms\_temp.**patient\_id** = universe\_temp\_raw.**patient\_id  
UNION  
SELECT** exclusions\_visit\_list\_final.**Date\_of\_Service**,  
 locations.**name AS** Site\_Name,  
 **null AS** Site\_ID\_Number,  
 universe\_temp\_raw.**Member\_Key**,  
 **null AS** Member\_Name,  
 universe\_temp\_raw.**Member\_First\_Name**,  
 universe\_temp\_raw.**Member\_Middle\_Name**,  
 universe\_temp\_raw.**Member\_Last\_Name**,  
 universe\_temp\_raw.**DOB**,  
 universe\_temp\_raw.**Sex**,  
 **NULL AS** CPTPX,  
 **NULL AS** CPTMOD,  
 **NULL AS** HCPCSPX,  
 **NULL AS** HCPCSMOD,  
 **NULL AS** LOINC,  
 **NULL AS** SNOMED,  
 exclusions\_visit\_list\_final.**Exclusions\_Code\_1**,  
 exclusions\_visit\_list\_final.**Exclusions\_Code\_1\_DOS**,  
 exclusions\_visit\_list\_final.**Exclusions\_Code\_2**,  
 exclusions\_visit\_list\_final.**Exclusions\_Code\_2\_DOS**,  
 exclusions\_visit\_list\_final.**Exclusions\_Code\_3**,  
 exclusions\_visit\_list\_final.**Exclusions\_Code\_3\_DOS**,  
 exclusions\_visit\_list\_final.**Exclusions\_Code\_4**,  
 exclusions\_visit\_list\_final.**Exclusions\_Code\_4\_DOS  
FROM** universe\_temp\_raw  
 **INNER JOIN** exclusions\_visit\_list\_final **ON** exclusions\_visit\_list\_final.**patient\_id** = universe\_temp\_raw.**patient\_id  
 INNER JOIN** visits **ON** visits.**id** = exclusions\_visit\_list\_final.**encounterid  
 LEFT JOIN** locations **ON** locations.**id** = visits.**location\_id  
UNION  
SELECT  
 NULL AS** Date\_of\_Service,  
 **null AS** Site\_Name,  
 **null AS** Site\_ID\_Number,  
 universe\_temp\_raw.**Member\_Key**,  
 **null AS** Member\_Name,  
 universe\_temp\_raw.**Member\_First\_Name**,  
 universe\_temp\_raw.**Member\_Middle\_Name**,  
 universe\_temp\_raw.**Member\_Last\_Name**,  
 universe\_temp\_raw.**DOB**,  
 universe\_temp\_raw.**Sex**,  
 **NULL AS** CPTPX,  
 **NULL AS** CPTMOD,  
 **NULL AS** HCPCSPX,  
 **NULL AS** HCPCSMOD,  
 **NULL AS** LOINC,  
 **NULL AS** SNOMED,  
 exclusions\_problem\_list\_final.**Exclusions\_Code\_1**,  
 exclusions\_problem\_list\_final.**Exclusions\_Code\_1\_DOS**,  
 exclusions\_problem\_list\_final.**Exclusions\_Code\_2**,  
 exclusions\_problem\_list\_final.**Exclusions\_Code\_2\_DOS**,  
 exclusions\_problem\_list\_final.**Exclusions\_Code\_3**,  
 exclusions\_problem\_list\_final.**Exclusions\_Code\_3\_DOS**,  
 exclusions\_problem\_list\_final.**Exclusions\_Code\_4**,  
 exclusions\_problem\_list\_final.**Exclusions\_Code\_4\_DOS  
FROM** universe\_temp\_raw  
 **INNER JOIN** exclusions\_problem\_list\_final **ON** exclusions\_problem\_list\_final.**patient\_id** = universe\_temp\_raw.**patient\_id**) **AS** raw\_result  
**ORDER BY Member\_Last\_Name**, **Member\_First\_Name**, **Member\_Key**, **Date\_of\_Service**, Exclusions\_Code\_1\_DOS, Exclusions\_Code\_2\_DOS

# Appendix B: SQL Code for the Report “ECDS: Follow-Up Care for Children Prescribed ADHD Medication (ADD-E)”

**Note:** Text highlighted in yellow indicates a change since version 1 of this document

**Database:** Staging Database

**Description**: This report displays data for the ECDS Follow-Up Care for Children Prescribed ADHD Medication (ADD-E) measure that could be submitted to Partnership HealthPlan of California (2022 Edition)

**Parameters**: {{start\_date}} and {{end\_date}}

**Report SQL Code:**

*/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
Report Name: ECDS: Follow-Up Care for Children Prescribed ADHD Medication (ADD-E)  
  
Code Edited By: Ben Fouts for RCHC  
  
Description: This report displays data for submission to Partnership Healthplan. It should not be used to  
 directly evaluate a Quality Measure or for case management purposes  
  
RCHC Started: June 2022 (based on code from the Partnership SQL specifications document)  
  
Version Date: September 19, 2022  
  
Revision History: 2022 Version 2. Fixed the CASE WHEN statements of the TEMPORARY TABLE universe\_temp\_raw  
 so CPT and HCPCS codes are displayed from either the HEDIS Value set or the "additional codes'  
 from the table temp\_value\_sets  
  
Resources Used: relevant\_patients, relevant\_visits, relevant\_providers, doctors, relevant\_programs,   
 relevant\_genders, hedis\_value\_set\_codes, edi\_invoice, edi\_inv\_cpt, edi\_inv\_diagnosis,   
 relevant\_visit\_diagnosis\_codes, relevant\_patient\_diagnoses, relevant\_diagnosis\_codes  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
  
--Define all codes needed for the measure***DROP TABLE IF EXISTS** temp\_value\_sets;  
**CREATE TEMPORARY TABLE** temp\_value\_sets **AS  
SELECT** *\****FROM** (  
 **VALUES** (**'99492'**,**'CPT'**,**'Additional Visit Codes'**), (**'99493'**,**'CPT'**,**'Additional Visit Codes'**), (**'99494'**,**'CPT'**,**'Additional Visit Codes'**),  
 (**'G0071'**,**'HCPCS'**,**'Additional Visit Codes'**), (**'G0512'**,**'HCPCS'**,**'Additional Visit Codes'**), (**'G2012'**,**'HCPCS'**,**'Additional Visit Codes'**),  
 (**'F90.0'**,**'ICD10CM'**,**'ADHD'**), (**'F90.1'**,**'ICD10CM'**,**'ADHD'**), (**'F90.2'**,**'ICD10CM'**,**'ADHD'**), (**'F90.8'**,**'ICD10CM'**,**'ADHD'**),  
 (**'F90.9'**,**'ICD10CM'**,**'ADHD'**), (**'314'**,**'ICD9CM'**,**'ADHD'**),(**'314.01'**,**'ICD9CM'**,**'ADHD'**)  
 )  
 **AS** value\_set\_values\_values (**code**, **code\_system**, **code\_group**);  
  
  
*-- Universe Definition: visits with a claim that has:  
-- 1) An ADHD diagnosis code (temp\_value\_sets.code\_group = 'ADHD')  
-- 2) An appropriate office or telephone procedure code (temp\_value\_sets.code\_group = 'Visit Codes')  
-- 3) A visit in the calculated measurement period  
-- 4) Patient age between age 6 years, 0 days to 13 years, 363 days at the end of the measurement period  
-- 5) Patient currently has Partnership HealthPlan insurance -- this is the default, but can be changed  
-- by the programmer to join to internal tables containing current Partnership patients  
-- from the last monthly membership file***DROP TABLE IF EXISTS** universe\_temp\_raw;  
**CREATE TEMPORARY TABLE** universe\_temp\_raw **AS  
SELECT DISTINCT** relevant\_patients.**id AS** patient\_id,  
 relevant\_patients.**mrn**,  
 *LEFT*(relevant\_patients.**subscriber\_number**,10) **AS** Member\_Key,  
 relevant\_patients.**first\_name AS** Member\_First\_Name,  
 **NULL AS** Member\_Middle\_Name,  
 relevant\_patients.**last\_name AS** Member\_Last\_Name,  
 relevant\_patients.**date\_of\_birth AS** DOB,  
 *EXTRACT*(**YEAR FROM** *age*(edi\_invoice.**servicedt**, relevant\_patients.**date\_of\_birth**)) **AS** Age,  
 **CASE WHEN** relevant\_genders.**abbreviation IN**(**'F'**,**'M'**,**'U'**) **THEN** relevant\_genders.**abbreviation  
 WHEN** relevant\_genders.**abbreviation** = **'NA' THEN 'U'  
 ELSE 'O' END AS** Sex,  
 edi\_invoice.**encounterid AS** enc\_id,  
 edi\_invoice.**servicedt** :: **DATE AS** Date\_of\_Service,  
 relevant\_programs.**name AS** visit\_facility,  
 relevant\_providers.**first\_name AS** Clinician\_First\_Name,  
 **NULL AS** Clinician\_Middle\_Name,  
 relevant\_providers.**last\_name AS** Clinician\_Last\_Name,  
 relevant\_providers.**npi AS** Clinician\_NPI\_Number,  
 **CASE WHEN** doctors.**deano IS NOT NULL THEN 'Y' ELSE 'N' END AS** Prescriber,  
 **CASE WHEN** temp\_value\_sets\_cpt.code\_system = **'CPT' OR** hedis\_value\_set\_codes.**code\_system\_name** = **'CPT'  
 THEN** edi\_inv\_cpt.**code END AS** visit\_cpt,  
 **CASE WHEN** temp\_value\_sets\_cpt.code\_system = **'CPT' OR** hedis\_value\_set\_codes.**code\_system\_name** = **'CPT'  
 THEN** edi\_inv\_cpt.**mod1 END AS** visit\_cptmod,  
 **CASE WHEN** temp\_value\_sets\_cpt.code\_system = **'CPT' OR** hedis\_value\_set\_codes.**code\_system\_name** = **'CPT'  
 THEN** edi\_inv\_cpt.**displayindex END AS** visit\_cpt\_displayindex,  
 **CASE WHEN** temp\_value\_sets\_cpt.code\_system = **'HCPCS' OR** hedis\_value\_set\_codes.**code\_system\_name** = **'HCPCS'  
 THEN** edi\_inv\_cpt.**code END AS** visit\_hcpcs,  
 **CASE WHEN** temp\_value\_sets\_cpt.code\_system = **'HCPCS' OR** hedis\_value\_set\_codes.**code\_system\_name** = **'HCPCS'  
 THEN** edi\_inv\_cpt.**mod1 END AS** visit\_hcpcsmod,  
 **CASE WHEN** temp\_value\_sets\_cpt.code\_system = **'HCPCS' OR** hedis\_value\_set\_codes.**code\_system\_name** = **'HCPCS'  
 THEN** edi\_inv\_cpt.**displayindex END AS** visit\_hcpcs\_displayindex,  
 edi\_inv\_diagnosis.**code AS** adhd\_dx,  
 edi\_inv\_diagnosis.**primarycode**,  
 edi\_inv\_diagnosis.**icdorder  
FROM** relevant\_patients  
 **INNER JOIN** edi\_invoice **ON** edi\_invoice.**patientid** = relevant\_patients.**id  
 AND** edi\_invoice.**servicedt BETWEEN** ({{start\_date}} :: DATE - **INTERVAL '10 months'** + **INTERVAL '1 DAY'**) **AND** {{end\_date}}  
 **AND** edi\_invoice.deleteflag = 0  
 **AND** edi\_invoice.voidflag = 0  
 **INNER JOIN** edi\_inv\_cpt **ON** edi\_inv\_cpt.invoiceid = edi\_invoice.id  
 **AND** edi\_inv\_cpt.deleteflag = 0  
 **LEFT JOIN** temp\_value\_sets **AS** temp\_value\_sets\_cpt **ON** temp\_value\_sets\_cpt.code = **LEFT**(edi\_inv\_cpt.code, 5)  
 **AND** temp\_value\_sets\_cpt.code\_group = **'Additional Visit Codes'  
 LEFT JOIN** hedis\_value\_set\_codes **ON** hedis\_value\_set\_codes.code\_value = **LEFT**(edi\_inv\_cpt.code, 5)  
 **AND** value\_set\_name **IN**(**'BH Outpatient'**, **'Observation'**, **'Health and Behavior Assessment or Intervention'**,  
 **'Online Assessments'**,**'Telephone Visits'**, **'Visit Setting Unspecified'**)  
 **AND** code\_system\_name **IN**(**'CPT'**,**'HCPCS'**)  
 **AND** latest = **'TRUE'  
 INNER JOIN** edi\_inv\_diagnosis **ON** edi\_inv\_diagnosis.invoiceid = edi\_invoice.id  
 **INNER JOIN** temp\_value\_sets **AS** temp\_value\_sets\_icd **ON** temp\_value\_sets\_icd.code = edi\_inv\_diagnosis.code  
 **AND** temp\_value\_sets\_icd.code\_group = **'ADHD'  
 INNER JOIN** relevant\_providers **ON** relevant\_providers.id = edi\_invoice.dosproviderid  
 **LEFT JOIN** doctors **ON** doctors.doctorid = edi\_invoice.dosproviderid  
 **LEFT JOIN** relevant\_programs **ON** relevant\_programs.id = edi\_invoice.invfacilityid  
 **LEFT JOIN** relevant\_genders **ON** relevant\_genders.id = relevant\_patients.gender\_id  
**WHERE** (temp\_value\_sets\_cpt.code **IS NOT NULL OR** hedis\_value\_set\_codes.code\_value **IS NOT NULL**)  
 **AND EXTRACT**(**YEAR FROM** age({{end\_date}}, relevant\_patients.date\_of\_birth)) **BETWEEN** 6 **and** 13  
 **AND** relevant\_patients.primary\_insurance\_group\_id = 17; *-- "PARTNERSHIP MANAGED CARE"  
  
  
-- This query unduplicates the temporary table "universe\_temp\_raw" (see universe definition above)  
-- so that one row is equal to one visit no matter how many CPT and ICD codes are picked up  
-- If there is more than one CPT code, the display index field is used to prioritize only one  
-- If there is more than one ICD code, the ICD order field is used to prioritize only one***DROP TABLE IF EXISTS** universe\_temp;  
**CREATE TEMPORARY TABLE** universe\_temp **AS  
SELECT DISTINCT** universe\_temp\_raw.**patient\_id**,  
 **mrn**,  
 **Member\_Key**,  
 **Member\_First\_Name**,  
 **Member\_Middle\_Name**,  
 **Member\_Last\_Name**,  
 **DOB**,  
 **Age**,  
 **Sex**,  
 universe\_temp\_raw.**Date\_of\_Service**,  
 **visit\_facility**,  
 **Clinician\_First\_Name**,  
 **Clinician\_Middle\_Name**,  
 **Clinician\_Last\_Name**,  
 **Clinician\_NPI\_Number**,  
 **Prescriber**,  
 cpt\_join.CPTPX,  
 cpt\_join.CPTMOD,  
 hcpcs\_join.HCPCS,  
 hcpcs\_join.HCPCMOD,  
 icd\_join.ICDDX\_10  
**FROM** universe\_temp\_raw  
 **INNER JOIN**(**SELECT DISTINCT ON** (**patient\_id**, **Date\_of\_Service**)  
 **patient\_id**,  
 **Date\_of\_Service**,  
 **visit\_cpt AS** CPTPX,  
 **visit\_cptmod AS** CPTMOD  
 **FROM** universe\_temp\_raw  
 **ORDER BY patient\_id**, **Date\_of\_Service**, **visit\_cpt\_displayindex**, **visit\_cptmod**) **AS** cpt\_join  
 **ON** cpt\_join.**patient\_id** = universe\_temp\_raw.**patient\_id  
 AND** cpt\_join.**Date\_of\_Service** = universe\_temp\_raw.**Date\_of\_Service  
 INNER JOIN**(**SELECT DISTINCT ON** (**patient\_id**, **Date\_of\_Service**)  
 **patient\_id**,  
 **Date\_of\_Service**,  
 **visit\_hcpcs AS** HCPCS,  
 **visit\_hcpcsmod AS** HCPCMOD  
 **FROM** universe\_temp\_raw  
 **ORDER BY patient\_id**, **Date\_of\_Service**, **visit\_hcpcs\_displayindex**, **visit\_hcpcsmod**) **AS** hcpcs\_join  
 **ON** hcpcs\_join.**patient\_id** = universe\_temp\_raw.**patient\_id  
 AND** hcpcs\_join.**Date\_of\_Service** = universe\_temp\_raw.**Date\_of\_Service  
 INNER JOIN**(**SELECT DISTINCT ON** (**patient\_id**, **Date\_of\_Service**)  
 **patient\_id**,  
 **Date\_of\_Service**,  
 **adhd\_dx AS** ICDDX\_10  
 **FROM** universe\_temp\_raw  
 **ORDER BY patient\_id**, **Date\_of\_Service**, **primarycode DESC**, **icdorder**) **AS** icd\_join  
 **ON** icd\_join.**patient\_id** = universe\_temp\_raw.**patient\_id  
 AND** icd\_join.**Date\_of\_Service** = universe\_temp\_raw.**Date\_of\_Service**;  
  
  
*-- Exclusions from the Problem List***DROP TABLE IF EXISTS** exclusions\_problem\_list\_raw;  
**CREATE TEMPORARY TABLE** exclusions\_problem\_list\_raw **AS  
SELECT DISTINCT** relevant\_patient\_diagnoses.**patient\_id**,  
 relevant\_diagnosis\_codes.**code AS** exclude\_code\_problst,  
 relevant\_patient\_diagnoses.**start\_date AS** firstDx,  
 relevant\_patient\_diagnoses.**end\_date AS** ended\_on  
**FROM** relevant\_patient\_diagnoses  
 **INNER JOIN** relevant\_diagnosis\_codes **ON** relevant\_diagnosis\_codes.**id** = relevant\_patient\_diagnoses.**diagnosis\_code\_id  
 INNER JOIN** hedis\_value\_set\_codes **ON** hedis\_value\_set\_codes.**code\_value** = relevant\_diagnosis\_codes.**code  
 AND** hedis\_value\_set\_codes.**value\_set\_name** = **'Narcolepsy'  
 AND latest** = **'TRUE'  
WHERE** relevant\_patient\_diagnoses.**start\_date** <= {{end\_date}};  
  
  
*-- Exclusions from Visit Assessments (with first date the code appeared on an Assessment)***DROP TABLE IF EXISTS** exclusions\_assess\_raw;  
**CREATE TEMPORARY TABLE** exclusions\_assess\_raw **AS  
SELECT** relevant\_visits.**patient\_id**,  
 relevant\_diagnosis\_codes.**code AS** exclude\_code\_assess,  
 *MIN*(relevant\_visits.**visit\_date** :: **DATE**) **AS** firstDx  
**FROM** relevant\_visits  
 **INNER JOIN** relevant\_visit\_diagnosis\_codes **ON** relevant\_visit\_diagnosis\_codes.**visit\_id** = relevant\_visits.**id  
 INNER JOIN** relevant\_diagnosis\_codes **ON** relevant\_diagnosis\_codes.**id** = relevant\_visit\_diagnosis\_codes.**diagnosis\_code\_id  
 INNER JOIN** hedis\_value\_set\_codes **ON** hedis\_value\_set\_codes.**code\_value** = relevant\_diagnosis\_codes.**code  
 AND** hedis\_value\_set\_codes.**value\_set\_name** = **'Narcolepsy'  
 AND latest** = **'TRUE'  
WHERE** relevant\_visits.**visit\_date** :: **DATE** <= {{end\_date}}  
**GROUP BY** relevant\_visits.patient\_id, relevant\_diagnosis\_codes.code;  
  
  
*--Combine and prioritize exclusions. See Standards and Specifications Document for explanation***DROP TABLE IF EXISTS** exclusions\_list\_temp;  
**CREATE TEMPORARY TABLE** exclusions\_list\_temp **AS  
SELECT  
 patient\_id**,  
 code,  
 code\_first\_date,  
 resolve\_date,  
 priority,  
 *ROW\_NUMBER* () **OVER** (  
 **PARTITION BY patient\_id  
 ORDER BY** priority, code\_first\_date) **AS** row\_excl  
**FROM**(  
 **SELECT  
 patient\_id**,  
 **exclude\_code\_problst AS** code,  
 **firstDx** :: **DATE AS** code\_first\_date,  
 **ended\_on** :: **DATE AS** resolve\_date,  
 **CASE WHEN ended\_on IS NOT NULL THEN** 1  
 **ELSE** 2 **END AS** priority  
 **FROM** exclusions\_problem\_list\_raw  
 **UNION  
 SELECT  
 patient\_id**,  
 **exclude\_code\_assess AS** code,  
 **firstDx** :: **DATE AS** code\_first\_date,  
 **NULL AS** resolve\_date,  
 3 **AS** priority  
 **FROM** exclusions\_assess\_raw  
 ) **AS** exclusion\_union;  
  
  
*-- Results query.  
-- Note: Add "Site\_ID\_Number" from your system or manually. This is the NPI number assigned to  
-- the site where care was provided (i.e., the site displayed in the field Site\_Name)  
-- Note: Add "Provider\_Key" from your system or manually. This is the PCP ID assigned by PHC to  
-- the provider who provided care at the visit (i.e., the Clinician\_First\_Name, etc.)  
-- Note: You can choose to combine the clinician names into the field "Clinician\_Name"  
-- and the patient/member names into the field "Member\_Name"***SELECT  
 Date\_of\_Service**,  
 **visit\_facility AS** Site\_Name,  
 **null AS** Site\_ID\_Number,  
 **null AS** Provider\_Key,  
 **Clinician\_NPI\_Number**,  
 **NULL AS** Clinician\_Name,  
 **Clinician\_First\_Name**,  
 **Clinician\_Middle\_Name**,  
 **Clinician\_Last\_Name**,  
 **Prescriber**,  
 **Member\_Key**,  
 **NULL AS** Member\_Name,  
 **Member\_First\_Name**,  
 **Member\_Middle\_Name**,  
 **Member\_Last\_Name**,  
 **DOB**,  
 **Sex**,  
 **CPTPX**,  
 **CPTMOD**,  
 **HCPCS**,  
 **HCPCMOD**,  
 **ICDDX\_10**,  
 exclude\_1.**code AS** Exclusions\_ICDDX10\_1,  
 exclude\_1.**code\_first\_date AS** Exclusions\_ICDDX10\_1\_DOS,  
 **CASE WHEN NOT** exclude\_1.**code** = exclude\_2.**code THEN** exclude\_2.**code  
 END AS** Exclusions\_ICDDX10\_2,  
 **CASE WHEN NOT** exclude\_1.**code** = exclude\_2.**code THEN** exclude\_2.**code\_first\_date  
 END AS** Exclusions\_ICDDX10\_2\_DOS,  
 exclude\_1.**resolve\_date AS** Exclusions\_ICDDX10\_1\_Resolution,  
 **CASE WHEN NOT** exclude\_1.**code** = exclude\_2.**code THEN** exclude\_2.**resolve\_date  
 END AS** Exclusions\_ICDDX10\_2\_Resolution  
**FROM** universe\_temp  
 **LEFT JOIN**(**SELECT  
 patient\_id**,  
 **code**,  
 **code\_first\_date**,  
 **resolve\_date  
 FROM** exclusions\_list\_temp  
 **WHERE row\_excl** = 1) **AS** exclude\_1  
 **ON** exclude\_1.**patient\_id** = universe\_temp.**patient\_id  
 LEFT JOIN**(**SELECT  
 patient\_id**,  
 **code**,  
 **code\_first\_date**,  
 **resolve\_date  
 FROM** exclusions\_list\_temp  
 **WHERE row\_excl** = 2) **AS** exclude\_2  
 **ON** exclude\_2.**patient\_id** = universe\_temp.**patient\_id  
ORDER BY Member\_Last\_Name**, **Member\_First\_Name**, **Date\_of\_Service**

# Appendix C: SQL Code for the Report “ECDS: Depression Related Measures (DMS-E, DSF-E, DRR-E, PND-E, PDS-E)” and Related Transformers/Data Elements

Configure Transformers/Data Elements

Check if your instance of Relevant has the Transformers relevant\_phq2 and relevant\_phq9 and how they are configured. There could be variations in the name and variations in the SQL code between health centers. In the report SQL code below, relevant\_phq2 and relevant\_phq9 each display one total score for one instrument on one date for one patient.

NEW Transformer/Data Element Pair for Edinburgh Depression Screens

Configure only if your health center uses the Edinburgh Depression Screen. Note that the ID numbers for catid, itemid, and detailed will be unique at each health center. One option to find these ID numbers is the report “RCHC List All Structured Data Items.”

**Transformer**: Build relevant\_edinburgh\_screen

**DROP TABLE IF EXISTS** relevant\_edinburgh\_screen;  
**CREATE TABLE** relevant\_edinburgh\_screen **AS  
SELECT DISTINCT** enc.**patientid AS** patient\_id,

enc.**encounterid AS** visit\_id,

enc.**date** :: **DATE AS** performed\_on,  
 structhpi.**value** :: **INT AS** score,  
 **CASE WHEN** structhpi.**value** :: **INT** >= 10 **THEN TRUE** **ELSE FALSE END AS** result  
**FROM** enc  
 **INNER JOIN** structhpi **ON** structhpi.**encounterid** = enc.**encounterid** **AND** (structhpi.**catid** = 170663 **AND** structhpi.**itemid** = 10575 **AND** structhpi.**detailid** = 1549  
 **OR** structhpi.**catid** = 375124 **AND** structhpi.**itemid** = 22423 **AND** structhpi.**detailid** = 2310

**OR** structhpi.**catid** = 250273 **AND** structhpi.**itemid** = 18785 **AND** structhpi.**detailid** = 1516)  
**WHERE** enc.**deleteflag** = 0  
 **AND** enc.**status** = **'CHK'  
 AND** structhpi.**value** ~ **'^\d+$'**

**Data Element**: Edinburgh Depression Screens

**SELECT DISTINCT** **patient\_id**,  
 **performed\_on**,  
 **score**,  
 **result   
FROM** relevant\_edinburgh\_screens

NEW Fields for Existing Pregnancy Transformer/Data Element Pair

**Transformer**: Build relevant\_pregnancy

This is a relatively complicated Transformer that features more than one TEMPORARY TABLE. The final subquery creates the table relevant\_pregnancy. Two fields (last\_menstrual\_period and estimated\_due\_date) should be added to the final subquery. At some health centers, these fields are already available from the existing code. Check the SQL code in the FROM statement for the following references (note there may be slight variations):

**FROM** obf\_pregnancy preg  
 **LEFT JOIN** obf\_edd\_details **AS** lmp  
 **ON** lmp.**pregid** = preg.**pregid AND** lmp.**conf\_lmp\_date** ~ **'^\d+\/\d+\/\d+$'   
 AND** lmp.**delstat** = 0  
 **LEFT JOIN** obf\_edd\_details **AS** final\_edd  
 **ON** final\_edd.**pregid** = preg.**pregid   
 AND** final\_edd.**final\_edd\_date** ~ **'^\d+\/\d+\/\d+$'   
 AND** final\_edd.**delstat** = 0

This code JOINS the alias tables “lmp” and “final\_edd.” If they exist, they can be added to the SELECT statement as the final two columns:

**SELECT DISTINCT . . .** *COALESCE*(*COALESCE*(final\_edd.**final\_edd\_date**, init\_edd.**init\_edd\_date**) :: **DATE** - 280,  
 lmp.**conf\_lmp\_date** :: **DATE**) **AS** last\_menstrual\_period,  
 *COALESCE*(*COALESCE*(final\_edd.**final\_edd\_date**, init\_edd.**init\_edd\_date**) :: **DATE**,  
 lmp.**conf\_lmp\_date** :: **DATE** + 280) **AS** estimated\_due\_date

**Data Element:** Pregnancies

Add the two new fields to the Data Element. An example of the final code is below.

**SELECT DISTINCT  
 id**,  
 **patient\_id**,  
 **started\_on**,  
 **ended\_on**,  
 **prenatal\_treatment\_initial\_trimester**,  
 **prenatal\_treatment\_initiated\_at\_health\_center**,  
 **last\_menstrual\_period**,  
 **estimated\_due\_date  
FROM** relevant\_pregnancy

Report SQL

**Database:** Staging Database

**Description**: This report displays data for the ECDS Depression Related Measures (DMS-E, DSF-E, DRR-E, PND-E, PDS-E) that could be submitted to Partnership HealthPlan of California (2022 Edition)

**Parameters**: {{start\_date}} and {{end\_date}}

**Report SQL Code:**

*/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
Report Name: ECDS: Depression Related Measures (DMS-E, DSF-E, DRR-E, PND-E, PDS-E)  
  
Code Edited By: Ben Fouts for RCHC  
  
Description: This report displays data for submission to Partnership Healthplan. It should not be used to  
 directly evaluate a Quality Measure or for case management purposes  
  
RCHC Started: June 2022 (based on code from the Partnership SQL specifications document)  
  
Version Date: July 25, 2022  
  
Revision History: 2022 Version 2 (previous version was developed in 2021)  
  
Resources Used: relevant\_patients, edi\_invoice, edi\_inv\_cpt, relevant\_payers, relevant\_payer\_groups,  
 relevant\_genders, relevant\_providers, relevant\_pregnancy, relevant\_deliveries, relevant\_phq2,  
 relevant\_phq9, relevant\_visit\_diagnosis\_codes, relevant\_diagnosis\_codes, hedis\_value\_set\_codes,  
 problemlist, itemdetail  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
  
  
--Define Office and Follow-up Visit CPT codes defined by Partnership  
--These are the default codes and your health center may have additional local codes (see PHP instructions)***DROP TABLE IF EXISTS** office\_cpt\_codes\_temp;  
**CREATE TEMPORARY TABLE** office\_cpt\_codes\_temp **AS  
SELECT** *\****FROM** (  
 **VALUES** (**'59400'**,**'CPT'**),(**'59510'**,**'CPT'**),(**'59618'**,**'CPT'**),(**'59619'**,**'CPT'**),(**'90791'**,**'CPT'**),(**'90792'**,**'CPT'**),  
 (**'90832'**,**'CPT'**),(**'90833'**,**'CPT'**),(**'90834'**,**'CPT'**),(**'90836'**,**'CPT'**),(**'90837'**,**'CPT'**),(**'90838'**,**'CPT'**),  
 (**'90839'**,**'CPT'**),(**'90845'**,**'CPT'**),(**'90846'**,**'CPT'**),(**'90847'**,**'CPT'**),(**'90849'**,**'CPT'**),(**'90853'**,**'CPT'**),  
 (**'90865'**,**'CPT'**),(**'90867'**,**'CPT'**),(**'90868'**,**'CPT'**),(**'90869'**,**'CPT'**),(**'90870'**,**'CPT'**),(**'90875'**,**'CPT'**),  
 (**'90876'**,**'CPT'**),(**'90880'**,**'CPT'**),(**'90887'**,**'CPT'**),(**'96116'**,**'CPT'**),(**'96121'**,**'CPT'**),(**'96130'**,**'CPT'**),  
 (**'96131'**,**'CPT'**),(**'96132'**,**'CPT'**),(**'96133'**,**'CPT'**),(**'96136'**,**'CPT'**),(**'96137'**,**'CPT'**),(**'96138'**,**'CPT'**),  
 (**'96139'**,**'CPT'**),(**'96146'**,**'CPT'**),(**'96151'**,**'CPT'**),(**'96156'**,**'CPT'**),(**'96170'**,**'CPT'**),(**'96171'**,**'CPT'**),  
 (**'97165'**,**'CPT'**),(**'97166'**,**'CPT'**),(**'97167'**,**'CPT'**),(**'97799'**,**'CPT'**),(**'98960'**,**'CPT'**),(**'98961'**,**'CPT'**),  
 (**'98962'**,**'CPT'**),(**'98966'**,**'CPT'**),(**'98967'**,**'CPT'**),(**'98968'**,**'CPT'**),(**'99078'**,**'CPT'**),(**'99201'**,**'CPT'**),  
 (**'99202'**,**'CPT'**),(**'99203'**,**'CPT'**),(**'99204'**,**'CPT'**),(**'99205'**,**'CPT'**),(**'99211'**,**'CPT'**),(**'99212'**,**'CPT'**),  
 (**'99213'**,**'CPT'**),(**'99214'**,**'CPT'**),(**'99215'**,**'CPT'**),(**'99217'**,**'CPT'**),(**'99218'**,**'CPT'**),(**'99219'**,**'CPT'**),  
 (**'99220'**,**'CPT'**),(**'99241'**,**'CPT'**),(**'99242'**,**'CPT'**),(**'99243'**,**'CPT'**),(**'99244'**,**'CPT'**),(**'99245'**,**'CPT'**),  
 (**'99304'**,**'CPT'**),(**'99305'**,**'CPT'**),(**'99306'**,**'CPT'**),(**'99307'**,**'CPT'**),(**'99308'**,**'CPT'**),(**'99309'**,**'CPT'**),  
 (**'99310'**,**'CPT'**),(**'99315'**,**'CPT'**),(**'99316'**,**'CPT'**),(**'99318'**,**'CPT'**),(**'99324'**,**'CPT'**),(**'99325'**,**'CPT'**),  
 (**'99326'**,**'CPT'**),(**'99327'**,**'CPT'**),(**'99328'**,**'CPT'**),(**'99334'**,**'CPT'**),(**'99335'**,**'CPT'**),(**'99336'**,**'CPT'**),  
 (**'99337'**,**'CPT'**),(**'99339'**,**'CPT'**),(**'99340'**,**'CPT'**),(**'99341'**,**'CPT'**),(**'99342'**,**'CPT'**),(**'99343'**,**'CPT'**),  
 (**'99344'**,**'CPT'**),(**'99345'**,**'CPT'**),(**'99347'**,**'CPT'**),(**'99348'**,**'CPT'**),(**'99349'**,**'CPT'**),(**'99350'**,**'CPT'**),  
 (**'99366'**,**'CPT'**),(**'99381'**,**'CPT'**),(**'99382'**,**'CPT'**),(**'99383'**,**'CPT'**),(**'99384'**,**'CPT'**),(**'99385'**,**'CPT'**),  
 (**'99386'**,**'CPT'**),(**'99387'**,**'CPT'**),(**'99391'**,**'CPT'**),(**'99392'**,**'CPT'**),(**'99393'**,**'CPT'**),(**'99394'**,**'CPT'**),  
 (**'99394'**,**'CPT'**),(**'99395'**,**'CPT'**),(**'99395'**,**'CPT'**),(**'99396'**,**'CPT'**),(**'99396'**,**'CPT'**),(**'99397'**,**'CPT'**),  
 (**'99397'**,**'CPT'**),(**'99401'**,**'CPT'**),(**'99402'**,**'CPT'**),(**'99403'**,**'CPT'**),(**'99404'**,**'CPT'**),(**'99411'**,**'CPT'**),  
 (**'99412'**,**'CPT'**),(**'99441'**,**'CPT'**),(**'99442'**,**'CPT'**),(**'99443'**,**'CPT'**),(**'99483'**,**'CPT'**),(**'99484'**,**'CPT'**),  
 (**'99492'**,**'CPT'**),(**'99493'**,**'CPT'**),(**'G0071'**,**'HCPCS'**),(**'G0101 '**,**'HCPCS'**),(**'G0155'**,**'HCPCS'**),(**'G0176'**,**'HCPCS'**),  
 (**'G0177'**,**'HCPCS'**),(**'G0402'**,**'HCPCS'**),(**'G0409'**,**'HCPCS'**),(**'G0410'**,**'HCPCS'**),(**'G0411'**,**'HCPCS'**),(**'G0438'**,**'HCPCS'**),  
 (**'G0439'**,**'HCPCS'**),(**'G0444'**,**'HCPCS'**),(**'G0463'**,**'HCPCS'**),(**'G0502'**,**'HCPCS'**),(**'G0503'**,**'HCPCS'**),(**'G0507'**,**'HCPCS'**),  
 (**'G0511'**,**'HCPCS'**),(**'G0512'**,**'HCPCS'**),(**'G2012'**,**'HCPCS'**),(**'G8431'**,**'HCPCS'**),(**'G8510'**,**'HCPCS'**),(**'H0002'**,**'HCPCS'**),  
 (**'H0004'**,**'HCPCS'**),(**'H0031'**,**'HCPCS'**),(**'H0034'**,**'HCPCS'**),(**'H0035'**,**'HCPCS'**),(**'H0036'**,**'HCPCS'**),(**'H0037'**,**'HCPCS'**),  
 (**'H0039'**,**'HCPCS'**),(**'H0040'**,**'HCPCS'**),(**'H1000'**,**'HCPCS'**),(**'H1002'**,**'HCPCS'**),(**'H1003'**,**'HCPCS'**),(**'H2000'**,**'HCPCS'**),  
 (**'H2001'**,**'HCPCS'**),(**'H2010'**,**'HCPCS'**),(**'H2011'**,**'HCPCS'**),(**'H2012'**,**'HCPCS'**),(**'H2013'**,**'HCPCS'**),(**'H2014'**,**'HCPCS'**),  
 (**'H2015'**,**'HCPCS'**),(**'H2016'**,**'HCPCS'**),(**'H2017'**,**'HCPCS'**),(**'H2018'**,**'HCPCS'**),(**'H2019'**,**'HCPCS'**),(**'H2020'**,**'HCPCS'**),  
 (**'S0201'**,**'HCPCS'**),(**'S9480'**,**'HCPCS'**),(**'S9484'**,**'HCPCS'**),(**'S9485'**,**'HCPCS'**),(**'T1015'**,**'HCPCS'**),(**'T1016'**,**'HCPCS'**),  
 (**'T1017'**,**'HCPCS'**),(**'T2022'**,**'HCPCS'**),(**'T2023'**,**'HCPCS'**),(**'Z1032'**,**'HCPCS'**),(**'Z1032'**,**'HCPCS'**),(**'Z1034'**,**'HCPCS'**),  
 (**'Z1036'**,**'HCPCS'**),(**'Z1038'**,**'HCPCS'**),(**'Z6300'**,**'HCPCS'**),(**'Z6302'**,**'HCPCS'**),(**'Z6304'**,**'HCPCS'**),(**'Z6306'**,**'HCPCS'**),  
 (**'Z6300'**,**'HCPCS'**),(**'Z6302'**,**'HCPCS'**),(**'Z6304'**,**'HCPCS'**),(**'Z6306'**,**'HCPCS'**),(**'Z6308'**,**'HCPCS'**),(**'Z6500'**,**'HCPCS'**)  
 )  
 **AS** value\_set\_values\_values (**code**,**code\_system**);  
  
  
*-- Universe Definition: visits with a:  
-- 1) Claim in the measurement period  
-- 2) Claim has a defined office procedure code (office\_cpt\_codes\_temp.code)  
-- 3) Patient who is 12 years or older  
-- 4) Patient sho is enrolled in Partnership Managed Care insurance  
-- Patients in the Universe currently have Partnership HealthPlan insurance -- this is the default,  
-- but can be changed by the programmer to join to internal tables containing current  
-- Partnership patients from the last monthly membership file***DROP TABLE IF EXISTS** initial\_denom\_codes\_temp;  
**CREATE TEMPORARY TABLE** initial\_denom\_codes\_temp **AS  
SELECT DISTINCT** relevant\_patients.**id AS** patient\_id,  
 relevant\_patients.**subscriber\_number**,  
 relevant\_patients.**first\_name AS** memberfirstname,  
 relevant\_patients.**last\_name AS** memberlastname,  
 relevant\_patients.**date\_of\_birth** :: **DATE AS** dob,  
 **CASE WHEN** relevant\_genders.**abbreviation IN**(**'F'**,**'M'**,**'U'**) **THEN** relevant\_genders.**abbreviation  
 WHEN** relevant\_genders.**abbreviation** = **'NA'  
 OR** relevant\_genders.**id IS NULL  
 THEN 'U'  
 ELSE 'O' END AS** Sex,  
 edi\_invoice.**encounterid AS** enc\_id,  
 edi\_invoice.**servicedt** :: **DATE AS** enc\_date,  
 relevant\_providers.**last\_name** || **', '** || relevant\_providers.**first\_name AS** visit\_clinician\_full\_name,  
 relevant\_providers.**npi AS** clinician\_npi\_number,  
 office\_cpt\_codes\_temp.**code AS** visit\_cpt,  
 edi\_inv\_cpt.**mod1 AS** cptmod,  
 office\_cpt\_codes\_temp.**code\_system**,  
 edi\_inv\_cpt.**displayindex  
FROM** relevant\_patients  
 **INNER JOIN** edi\_invoice **ON** edi\_invoice.**patientid** = relevant\_patients.**id  
 AND** edi\_invoice.**servicedt BETWEEN** {{start\_date}} **AND** {{end\_date}}  
 **AND** edi\_invoice.deleteflag = 0  
 **AND** edi\_invoice.voidflag = 0  
 **INNER JOIN** edi\_inv\_cpt **ON** edi\_inv\_cpt.invoiceid = edi\_invoice.id  
 **INNER JOIN** office\_cpt\_codes\_temp **ON** office\_cpt\_codes\_temp.code = **LEFT**(edi\_inv\_cpt.code, 5)  
 **INNER JOIN** relevant\_payers **ON** relevant\_payers.payer\_key = relevant\_patients.primary\_insurance\_id  
 **INNER JOIN** relevant\_payer\_groups **ON** relevant\_payer\_groups.id = relevant\_payers.payer\_group\_key  
 **LEFT JOIN** relevant\_genders **ON** relevant\_genders.id = relevant\_patients.gender\_id  
 **LEFT JOIN** relevant\_providers **ON** relevant\_providers.id = edi\_invoice.dosproviderid  
**WHERE EXTRACT**(**YEAR FROM** age(NOW(), relevant\_patients.date\_of\_birth :: DATE)) >= 12  
 **AND** relevant\_patients.primary\_insurance\_group\_id = 17; *-- "PARTNERSHIP MANAGED CARE"  
  
  
--Unduplicated encounter denominator***DROP TABLE IF EXISTS** total\_denominator\_temp;  
**CREATE TEMPORARY TABLE** total\_denominator\_temp **AS  
SELECT DISTINCT  
 patient\_id**,  
 **subscriber\_number**,  
 **memberfirstname**,  
 **memberlastname**,  
 **dob**,  
 **sex**,  
 **enc\_id**,  
 **enc\_date**,  
 **visit\_clinician\_full\_name**,  
 **clinician\_npi\_number  
FROM** initial\_denom\_codes\_temp;  
**CREATE INDEX** index\_total\_denominator\_temp\_on\_enc\_id **ON** total\_denominator\_temp (**enc\_id**);  
**CREATE INDEX** index\_total\_denominator\_temp\_on\_patient\_id **ON** total\_denominator\_temp (**patient\_id**);  
  
  
*--Produce Outpatient Visit CPT/HCPCS columns for the report output***DROP TABLE IF EXISTS** cpt\_columns\_temp;  
**CREATE TEMPORARY TABLE** cpt\_columns\_temp **AS  
WITH** order\_cpt **AS**(  
 **SELECT  
 enc\_id**,  
 **visit\_cpt**,  
 **code\_system**,  
 *ROW\_NUMBER*() **OVER** (**PARTITION BY enc\_id**, **code\_system  
 ORDER BY displayindex**, **visit\_cpt**) **AS row  
 FROM** (**SELECT DISTINCT  
 enc\_id**,  
 **visit\_cpt**,  
 **code\_system**,  
 **displayindex  
 FROM** initial\_denom\_codes\_temp) **AS** undup  
),cpt1 **AS**(  
 **SELECT enc\_id**, **visit\_cpt  
 FROM** order\_cpt  
 **WHERE code\_system** = **'CPT'  
 AND row** = 1  
),cpt2 **AS**(  
 **SELECT enc\_id**, **visit\_cpt  
 FROM** order\_cpt  
 **WHERE code\_system** = **'CPT'  
 AND row** = 2  
),cpt3 **AS**(  
 **SELECT enc\_id**, **visit\_cpt  
 FROM** order\_cpt  
 **WHERE code\_system** = **'CPT'  
 AND row** = 3  
),cpt4 **AS**(  
 **SELECT enc\_id**, **visit\_cpt  
 FROM** order\_cpt  
 **WHERE code\_system** = **'CPT'  
 AND row** = 4  
),cpt1\_mod **AS**(  
 **SELECT** cpt1.**enc\_id**, initial\_denom\_codes\_temp.**cptmod  
 FROM** cpt1  
 **INNER JOIN** initial\_denom\_codes\_temp **ON** initial\_denom\_codes\_temp.**enc\_id** = cpt1.**enc\_id  
 AND** initial\_denom\_codes\_temp.**visit\_cpt** = cpt1.**visit\_cpt  
 WHERE** initial\_denom\_codes\_temp.**cptmod IS NOT NULL**),hcpcs1 **AS**(  
**SELECT enc\_id**, **visit\_cpt  
 FROM** order\_cpt  
 **WHERE code\_system** = **'HCPCS'  
 AND row** = 1  
),hcpcs1\_mod **AS**(  
 **SELECT** hcpcs1.**enc\_id**, initial\_denom\_codes\_temp.**cptmod  
 FROM** hcpcs1  
 **INNER JOIN** initial\_denom\_codes\_temp **ON** initial\_denom\_codes\_temp.**enc\_id** = hcpcs1.**enc\_id  
 AND** initial\_denom\_codes\_temp.**visit\_cpt** = hcpcs1.**visit\_cpt  
 WHERE** initial\_denom\_codes\_temp.**cptmod IS NOT NULL**),enc\_nbr\_distinct **AS** (**SELECT DISTINCT enc\_id FROM** order\_cpt)  
**SELECT** enc\_nbr\_distinct.**enc\_id**,  
 cpt1.**visit\_cpt AS** cpt1,  
 cpt2.**visit\_cpt AS** cpt2,  
 cpt3.**visit\_cpt AS** cpt3,  
 cpt4.**visit\_cpt AS** cpt4,  
 cpt1\_mod.**cptmod**,  
 hcpcs1.**visit\_cpt AS** hcpcs,  
 hcpcs1\_mod.**cptmod AS** hcpcmod  
**FROM** enc\_nbr\_distinct  
 **LEFT JOIN** cpt1 **ON** cpt1.**enc\_id** = enc\_nbr\_distinct.**enc\_id  
 LEFT JOIN** cpt2 **ON** cpt2.**enc\_id** = enc\_nbr\_distinct.**enc\_id  
 LEFT JOIN** cpt3 **ON** cpt3.**enc\_id** = enc\_nbr\_distinct.**enc\_id  
 LEFT JOIN** cpt4 **ON** cpt4.**enc\_id** = enc\_nbr\_distinct.**enc\_id  
 LEFT JOIN** cpt1\_mod **ON** cpt1\_mod.**enc\_id** = enc\_nbr\_distinct.**enc\_id  
 LEFT JOIN** hcpcs1 **ON** hcpcs1.**enc\_id** = enc\_nbr\_distinct.**enc\_id  
 LEFT JOIN** hcpcs1\_mod **ON** hcpcs1\_mod.**enc\_id** = enc\_nbr\_distinct.**enc\_id  
ORDER BY** enc\_nbr\_distinct.**enc\_id**;  
  
  
*--Pregnancies overlapping measurement period for universe patients***DROP TABLE IF EXISTS** preg\_mp\_temp;  
**CREATE TEMPORARY TABLE** preg\_mp\_temp **AS  
SELECT DISTINCT** total\_denominator\_temp.**patient\_id**,  
 relevant\_pregnancy.**id AS** preg\_id,  
 relevant\_pregnancy.**started\_on**,  
 relevant\_pregnancy.**ended\_on**,  
 relevant\_pregnancy.**last\_menstrual\_period** :: **DATE AS** lmp\_date,  
 relevant\_pregnancy.**estimated\_due\_date** :: **DATE AS** edd\_date,  
 relevant\_deliveries.**delivered\_on** :: **DATE AS** actual\_date\_of\_delivery,  
 *COALESCE*(relevant\_deliveries.**delivered\_on**,relevant\_pregnancy.**estimated\_due\_date**) :: **DATE AS** sort\_date,  
 **CASE WHEN** *COALESCE*(relevant\_deliveries.**delivered\_on**,relevant\_pregnancy.**estimated\_due\_date**) :: **DATE** <= {{end\_date}}  
 **THEN TRUE ELSE FALSE END AS** delivery\_edd\_before\_endmp  
**FROM** total\_denominator\_temp  
 **INNER JOIN** relevant\_pregnancy **ON** relevant\_pregnancy.**patient\_id** = total\_denominator\_temp.**patient\_id  
 LEFT JOIN** relevant\_deliveries **ON** relevant\_deliveries.**pregid** = relevant\_pregnancy.**id  
WHERE** (relevant\_pregnancy.**started\_on**,  
 relevant\_pregnancy.**ended\_on** + 1  
 ) **OVERLAPS** (  
 {{start\_date}},  
 {{end\_date}} :: DATE + 1 );  
  
  
*--Choose one future delivery for the column "Estimated date of delivery (future)"***DROP TABLE IF EXISTS** future\_delivery\_temp;  
**CREATE TEMPORARY TABLE** future\_delivery\_temp **AS  
SELECT DISTINCT ON** (**patient\_id**)  
 **patient\_id**,  
 **preg\_id**,  
 **started\_on**,  
 **ended\_on**,  
 **lmp\_date**,  
 **edd\_date**,  
 **actual\_date\_of\_delivery**,  
 **sort\_date  
FROM** preg\_mp\_temp  
**WHERE delivery\_edd\_before\_endmp IS FALSE  
ORDER BY patient\_id**, **sort\_date ASC**;  
  
  
*--Choose one past pregnancy for the column "Actual (past) date of delivery within 12 months"***DROP TABLE IF EXISTS** past\_delivery\_temp;  
**CREATE TEMPORARY TABLE** past\_delivery\_temp **AS  
SELECT DISTINCT ON** (**patient\_id**)  
 **patient\_id**,  
 **preg\_id**,  
 **started\_on**,  
 **ended\_on**,  
 **lmp\_date**,  
 **edd\_date**,  
 **actual\_date\_of\_delivery**,  
 **sort\_date  
FROM** preg\_mp\_temp  
**WHERE delivery\_edd\_before\_endmp IS TRUE  
ORDER BY patient\_id**, **sort\_date DESC**;  
  
  
*--Make weeks/days gestation calculations and display other relevant pregnancy items for the report output***DROP TABLE IF EXISTS** preg\_columns\_temp;  
**CREATE TEMPORARY TABLE** preg\_columns\_temp **AS  
WITH** preg\_dates **AS** (  
 **SELECT** total\_denominator\_temp.**patient\_id**,  
 total\_denominator\_temp.**enc\_date**,  
 total\_denominator\_temp.**enc\_id**,  
 *COALESCE*(past\_delivery\_temp.**lmp\_date**, future\_delivery\_temp.**lmp\_date**) **AS** calc\_lmp\_date,  
 *COALESCE*(past\_delivery\_temp.**edd\_date**, future\_delivery\_temp.**edd\_date**) **AS** calc\_edd\_date,  
 future\_delivery\_temp.**edd\_date AS** future\_edd\_date,  
 past\_delivery\_temp.**actual\_date\_of\_delivery AS** past\_date\_of\_delivery,  
 past\_delivery\_temp.**lmp\_date AS** past\_delivery\_lmp  
 **FROM** total\_denominator\_temp  
 **LEFT JOIN** past\_delivery\_temp **ON** past\_delivery\_temp.**patient\_id** = total\_denominator\_temp.**patient\_id  
 AND** total\_denominator\_temp.**enc\_date BETWEEN** past\_delivery\_temp.**started\_on AND** past\_delivery\_temp.**ended\_on  
 LEFT JOIN** future\_delivery\_temp **ON** future\_delivery\_temp.**patient\_id** = total\_denominator\_temp.**patient\_id  
 AND** total\_denominator\_temp.**enc\_date BETWEEN** future\_delivery\_temp.**started\_on AND** future\_delivery\_temp.**ended\_on  
 WHERE** past\_delivery\_temp.**patient\_id IS NOT NULL  
 OR** future\_delivery\_temp.**patient\_id IS NOT NULL**)  
**SELECT  
 patient\_id**,  
 **enc\_date**,  
 **enc\_id**,  
 future\_edd\_date,  
 past\_date\_of\_delivery,  
 calc\_lmp\_date,  
 calc\_edd\_date,  
 *TRUNC*(*DATE\_PART*(**'day'**, **enc\_date** ::**timestamp** - calc\_lmp\_date :: **timestamp**)/7) || **'W'** ||  
 *DATE\_PART*(**'day'**, **enc\_date** ::**timestamp** - calc\_lmp\_date :: **timestamp**) -  
 (*TRUNC*(*DATE\_PART*(**'day'**, **enc\_date** ::**timestamp** - calc\_lmp\_date :: **timestamp**)/7) \* 7) || **'D'  
 AS** gestation\_weeks\_and\_days\_at\_time\_of\_service,  
 **CASE WHEN NOT** past\_date\_of\_delivery **IS NULL THEN** *TRUNC*(*DATE\_PART*(**'day'**, past\_date\_of\_delivery ::**timestamp** - past\_delivery\_lmp ::**timestamp**)/7) || **'W'** ||  
 *DATE\_PART*(**'day'**, past\_date\_of\_delivery ::**timestamp** - past\_delivery\_lmp ::**timestamp**) -  
 (*TRUNC*(*DATE\_PART*(**'day'**, past\_date\_of\_delivery ::**timestamp** - past\_delivery\_lmp ::**timestamp**)/7) \* 7) || **'D'  
 END AS** weeks\_and\_days\_gestation\_at\_time\_of\_delivery  
**FROM** preg\_dates;  
  
  
*--Identify depression screenings from structured data. Priority order: PHQ-2, PHQ-9, Edinburgh, GDS (or other screens)***DROP TABLE IF EXISTS** depression\_screenings\_temp;  
**CREATE TEMPORARY TABLE** depression\_screenings\_temp **AS  
SELECT** *-- PHQ-2 add from Transformer (if similar exists) or pull directly from structured data* total\_denominator\_temp.**patient\_id**,  
 total\_denominator\_temp.**enc\_id**,  
 **'55758-7' AS** loinc\_code\_screen,  
 **'PHQ2' AS** tool\_name,  
 **total\_score** :: **VARCHAR AS** screen\_score,  
 **CASE WHEN result IS TRUE THEN 'Positive' ELSE 'Negative' END AS** interp\_result,  
 1 **AS** priority  
**FROM** total\_denominator\_temp  
 **INNER JOIN** relevant\_phq2 **ON** relevant\_phq2.**visit\_id** = total\_denominator\_temp.**enc\_id  
UNION  
SELECT** *-- PHQ-9 add from Transformer (if similar exists) or pull directly from structured data* total\_denominator\_temp.**patient\_id**,  
 total\_denominator\_temp.**enc\_id**,  
 **'44261-6' AS** loinc\_code\_screen,  
 **'PHQ9' AS** tool\_name,  
 **score** :: **VARCHAR AS** screen\_score,  
 **CASE WHEN score** :: **INT** >= 10 **THEN 'Positive' ELSE 'Negative' END AS** interp\_result,  
 2 **AS** priority  
**FROM** total\_denominator\_temp  
 **INNER JOIN** relevant\_phq9 **ON** relevant\_phq9.**visit\_id** = total\_denominator\_temp.**enc\_id**;  
*/\*  
UNION  
 -- Edinburgh Depression Screen add from Transformer (if similar exists) or pull directly from structured data  
SELECT  
 total\_denominator\_temp.patient\_id,  
 total\_denominator\_temp.enc\_id,  
 '71354-5' AS loinc\_code\_screen,  
 'EPDS' AS tool\_name,  
 score :: VARCHAR AS screen\_score,  
 CASE WHEN result IS TRUE THEN 'Positive' ELSE 'Negative' END AS interp\_result,  
 3 AS priority  
FROM total\_denominator\_temp  
 INNER JOIN relevant\_edinburgh\_screens ON relevant\_edinburgh\_screens.visit\_id = total\_denominator\_temp.enc\_id  
UNION  
 -- If your health center uses additional kinds of depression screening tools, add them below using UNION sub-queries  
 -- Otherwise, delete this section  
SELECT -- [Add name of tool here for easy reference]  
 total\_denominator\_temp.patient\_id,  
 total\_denominator\_temp.enc\_id,  
 '48544-1' AS loinc\_code\_screen, -- Add LOINC from Partnership ECDS instructions  
 'GDSL' AS tool\_name, -- Add tool name (or abbreviation)  
 total\_score :: VARCHAR AS screen\_score,  
 CASE WHEN total\_score :: INT >= 10 THEN 'Positive'  
 ELSE 'Negative' END AS interp\_result, -- You may have to use a CASE WHEN statement. See Partnership ECDS instructions  
 4 AS priority -- Priority should start and 4 and increase to 5, 6, etc. depending on number of screens  
FROM total\_denominator\_temp  
 INNER JOIN(  
 SELECT DISTINCT  
 enc.patientid AS patient\_id,  
 enc.encounterid AS enc\_id,  
 structhpi.value :: INT AS total\_score  
 FROM enc  
 INNER JOIN structhpi  
 ON structhpi.encounterid = enc.encounterid AND structhpi.catid = 250273 -- ID numbers are from your system  
 AND structhpi.itemid = 18785 AND structhpi.detailid = 1516  
 AND structhpi.value ~ '\d+'  
 WHERE enc.deleteflag = 0  
 AND enc.status = 'CHK'  
 AND structhpi.encounterid IS NOT NULL) AS gdsl\_given ON gdsl\_given.enc\_id = total\_denominator\_temp.enc\_id  
\*/  
  
--Unduplicate depression screening results and add row number***DROP TABLE IF EXISTS** depression\_screen\_priorities\_temp;  
**CREATE TEMPORARY TABLE** depression\_screen\_priorities\_temp **AS  
SELECT  
 enc\_id**,  
 **patient\_id**,  
 **loinc\_code\_screen**,  
 **tool\_name**,  
 **screen\_score**,  
 **interp\_result**,  
 **priority**,  
 *ROW\_NUMBER*() **OVER**(**PARTITION BY enc\_id ORDER BY priority**) **AS row  
FROM**(**SELECT DISTINCT ON** (**enc\_id**, **priority**)  
 **enc\_id**,  
 **patient\_id**,  
 **loinc\_code\_screen**,  
 **tool\_name**,  
 **screen\_score**,  
 **interp\_result**,  
 **priority  
 FROM** depression\_screenings\_temp  
 **ORDER BY enc\_id**, **priority**, **screen\_score DESC**) **AS** depression\_screenings\_temp\_distinct\_on;  
  
  
*--Prepare depression screening results so they can be entered into columns of the report output***DROP TABLE IF EXISTS** depression\_screen\_columns\_temp;  
**CREATE TEMPORARY TABLE** depression\_screen\_columns\_temp **AS  
SELECT DISTINCT** depression\_screen\_priorities\_temp.**patient\_id**,  
 depression\_screen\_priorities\_temp.**enc\_id**,  
 total\_denominator\_temp.**enc\_date**,  
 first\_columns.**loinc\_code\_screen AS** loinc\_tool\_1,  
 first\_columns.**tool\_name AS** name\_tool\_1,  
 first\_columns.**screen\_score AS** score\_tool\_1,  
 first\_columns.**interp\_result AS** result\_tool\_1,  
 second\_columns.**loinc\_code\_screen AS** loinc\_tool\_2,  
 second\_columns.**tool\_name AS** name\_tool\_2,  
 second\_columns.**screen\_score AS** score\_tool\_2,  
 second\_columns.**interp\_result AS** result\_tool\_2  
**FROM** depression\_screen\_priorities\_temp  
 **INNER JOIN** total\_denominator\_temp **ON** total\_denominator\_temp.**enc\_id** = depression\_screen\_priorities\_temp.**enc\_id  
 INNER JOIN** (**SELECT** *\** **FROM** depression\_screen\_priorities\_temp  
 **WHERE row** = 1) **AS** first\_columns **ON** first\_columns.**enc\_id** = depression\_screen\_priorities\_temp.**enc\_id  
 LEFT JOIN** (**SELECT** *\** **FROM** depression\_screen\_priorities\_temp  
 **WHERE row** = 2) **AS** second\_columns **ON** second\_columns.**enc\_id** = depression\_screen\_priorities\_temp.**enc\_id  
ORDER BY** depression\_screen\_priorities\_temp.**enc\_id**;  
  
  
  
*--Identify a depression or other behavioral health condition code associated with the present visit Assessment***DROP TABLE IF EXISTS** depression\_dx\_column\_temp;  
**CREATE TEMPORARY TABLE** depression\_dx\_column\_temp **AS  
SELECT DISTINCT ON** (total\_denominator\_temp.**enc\_id**)  
 total\_denominator\_temp.**patient\_id**,  
 total\_denominator\_temp.**enc\_id**,  
 relevant\_diagnosis\_codes.**code AS** icd\_code  
**FROM** total\_denominator\_temp  
 **INNER JOIN** relevant\_visit\_diagnosis\_codes **ON** relevant\_visit\_diagnosis\_codes.**visit\_id** = total\_denominator\_temp.**enc\_id  
 INNER JOIN** relevant\_diagnosis\_codes **ON** relevant\_diagnosis\_codes.**id** = relevant\_visit\_diagnosis\_codes.**diagnosis\_code\_id  
 INNER JOIN** hedis\_value\_set\_codes **ON** hedis\_value\_set\_codes.**code\_value** = relevant\_diagnosis\_codes.**code  
 AND** hedis\_value\_set\_codes.**value\_set\_name** = **'Depression or Other Behavioral Health Condition'  
 AND latest** = **'TRUE'  
ORDER BY** total\_denominator\_temp.**enc\_id**, relevant\_diagnosis\_codes.**code**;  
  
  
*--Identify exclusion diagnosis codes (including depression) that occurred in the past two years  
--Take the earliest visit date for each code and order all codes by their dates***DROP TABLE IF EXISTS** exclusion\_dx\_temp;  
**CREATE TEMPORARY TABLE** exclusion\_dx\_temp **AS  
SELECT  
 patient\_id**,  
 icd\_code,  
 first\_visit\_date,  
 *ROW\_NUMBER*() **OVER**(**PARTITION BY patient\_id ORDER BY** first\_visit\_date, icd\_code) **AS row  
FROM**(**SELECT** total\_denominator\_temp.**patient\_id**,  
 relevant\_diagnosis\_codes.**code AS** icd\_code,  
 *MIN*(relevant\_visits.**visit\_date**) :: **DATE AS** first\_visit\_date  
 **FROM** total\_denominator\_temp  
 **INNER JOIN** relevant\_visits **ON** relevant\_visits.**patient\_id** = total\_denominator\_temp.**patient\_id  
 INNER JOIN** relevant\_visit\_diagnosis\_codes **ON** relevant\_visit\_diagnosis\_codes.**visit\_id** = relevant\_visits.**id  
 INNER JOIN** relevant\_diagnosis\_codes **ON** relevant\_diagnosis\_codes.**id** = relevant\_visit\_diagnosis\_codes.**diagnosis\_code\_id  
 INNER JOIN** hedis\_value\_set\_codes **ON** hedis\_value\_set\_codes.**code\_value** = relevant\_diagnosis\_codes.**code  
 AND** hedis\_value\_set\_codes.**value\_set\_name IN**(**'Bipolar Disorder'**,**'Other Bipolar Disorder'**,  
 **'Personality Disorder'**,**'Pervasive Developmental Disorder'**,**'Psychotic Disorders'**,  
 **'Depression or Other Behavioral Health Condition'**)  
 **AND latest** = **'TRUE'  
 WHERE** relevant\_visits.**visit\_date BETWEEN** ({{end\_date}} :: DATE - **INTERVAL '2 YEARS'**) **AND** {{end\_date}}  
 **GROUP BY** total\_denominator\_temp.patient\_id, relevant\_diagnosis\_codes.code) **AS** raw\_codes;  
  
  
*--Prepare the exclusion diagnosis codes by date of service for the report output***DROP TABLE IF EXISTS** exclusion\_columns\_temp;  
**CREATE TEMPORARY TABLE** exclusion\_columns\_temp **AS  
SELECT** denom.**patient\_id**,  
 exclusions\_1.**icd\_code AS** exclusions\_icddx10\_1,  
 exclusions\_1.**first\_visit\_date AS** exclusions\_icddx10\_1\_dos,  
 exclusions\_2.**icd\_code AS** exclusions\_icddx10\_2,  
 exclusions\_2.**first\_visit\_date AS** exclusions\_icddx10\_2\_dos,  
 exclusions\_3.**icd\_code AS** exclusions\_icddx10\_3,  
 exclusions\_3.**first\_visit\_date AS** exclusions\_icddx10\_3\_dos,  
 exclusions\_4.**icd\_code AS** exclusions\_icddx10\_4,  
 exclusions\_4.**first\_visit\_date AS** exclusions\_icddx10\_4\_dos,  
 exclusions\_5.**icd\_code AS** exclusions\_icddx10\_5,  
 exclusions\_5.**first\_visit\_date AS** exclusions\_icddx10\_5\_dos,  
 exclusions\_6.**icd\_code AS** exclusions\_icddx10\_6,  
 exclusions\_6.**first\_visit\_date AS** exclusions\_icddx10\_6\_dos,  
 exclusions\_7.**icd\_code AS** exclusions\_icddx10\_7,  
 exclusions\_7.**first\_visit\_date AS** exclusions\_icddx10\_7\_dos,  
 exclusions\_8.**icd\_code AS** exclusions\_icddx10\_8,  
 exclusions\_8.**first\_visit\_date AS** exclusions\_icddx10\_8\_dos,  
 exclusions\_9.**icd\_code AS** exclusions\_icddx10\_9,  
 exclusions\_9.**first\_visit\_date AS** exclusions\_icddx10\_9\_dos,  
 exclusions\_10.**icd\_code AS** exclusions\_icddx10\_10,  
 exclusions\_10.**first\_visit\_date AS** exclusions\_icddx10\_10\_dos,  
 exclusions\_11.**icd\_code AS** exclusions\_icddx10\_11,  
 exclusions\_11.**first\_visit\_date AS** exclusions\_icddx10\_11\_dos,  
 exclusions\_12.**icd\_code AS** exclusions\_icddx10\_12,  
 exclusions\_12.**first\_visit\_date AS** exclusions\_icddx10\_12\_dos,  
 exclusions\_13.**icd\_code AS** exclusions\_icddx10\_13,  
 exclusions\_13.**first\_visit\_date AS** exclusions\_icddx10\_13\_dos,  
 exclusions\_14.**icd\_code AS** exclusions\_icddx10\_14,  
 exclusions\_14.**first\_visit\_date AS** exclusions\_icddx10\_14\_dos,  
 exclusions\_15.**icd\_code AS** exclusions\_icddx10\_15,  
 exclusions\_15.**first\_visit\_date AS** exclusions\_icddx10\_15\_dos  
**FROM** (**SELECT DISTINCT patient\_id FROM** exclusion\_dx\_temp) **AS** denom  
 **INNER JOIN** (**SELECT DISTINCT patient\_id**, **first\_visit\_date**, **icd\_code  
 FROM** exclusion\_dx\_temp  
 **WHERE row** = 1) **AS** exclusions\_1  
 **ON** exclusions\_1.**patient\_id** = denom.**patient\_id  
 LEFT JOIN** (**SELECT DISTINCT patient\_id**, **first\_visit\_date**, **icd\_code  
 FROM** exclusion\_dx\_temp  
 **WHERE row** = 2) **AS** exclusions\_2  
 **ON** exclusions\_2.**patient\_id** = denom.**patient\_id  
 LEFT JOIN** (**SELECT DISTINCT patient\_id**, **first\_visit\_date**, **icd\_code  
 FROM** exclusion\_dx\_temp  
 **WHERE row** = 3) **AS** exclusions\_3  
 **ON** exclusions\_3.**patient\_id** = denom.**patient\_id  
 LEFT JOIN** (**SELECT DISTINCT patient\_id**, **first\_visit\_date**, **icd\_code  
 FROM** exclusion\_dx\_temp  
 **WHERE row** = 4) **AS** exclusions\_4  
 **ON** exclusions\_4.**patient\_id** = denom.**patient\_id  
 LEFT JOIN** (**SELECT DISTINCT patient\_id**, **first\_visit\_date**, **icd\_code  
 FROM** exclusion\_dx\_temp  
 **WHERE row** = 5) **AS** exclusions\_5  
 **ON** exclusions\_5.**patient\_id** = denom.**patient\_id  
 LEFT JOIN** (**SELECT DISTINCT patient\_id**, **first\_visit\_date**, **icd\_code  
 FROM** exclusion\_dx\_temp  
 **WHERE row** = 6) **AS** exclusions\_6  
 **ON** exclusions\_6.**patient\_id** = denom.**patient\_id  
 LEFT JOIN** (**SELECT DISTINCT patient\_id**, **first\_visit\_date**, **icd\_code  
 FROM** exclusion\_dx\_temp  
 **WHERE row** = 7) **AS** exclusions\_7  
 **ON** exclusions\_7.**patient\_id** = denom.**patient\_id  
 LEFT JOIN** (**SELECT DISTINCT patient\_id**, **first\_visit\_date**, **icd\_code  
 FROM** exclusion\_dx\_temp  
 **WHERE row** = 8) **AS** exclusions\_8  
 **ON** exclusions\_8.**patient\_id** = denom.**patient\_id  
 LEFT JOIN** (**SELECT DISTINCT patient\_id**, **first\_visit\_date**, **icd\_code  
 FROM** exclusion\_dx\_temp  
 **WHERE row** = 9) **AS** exclusions\_9  
 **ON** exclusions\_9.**patient\_id** = denom.**patient\_id  
 LEFT JOIN** (**SELECT DISTINCT patient\_id**, **first\_visit\_date**, **icd\_code  
 FROM** exclusion\_dx\_temp  
 **WHERE row** = 10) **AS** exclusions\_10  
 **ON** exclusions\_10.**patient\_id** = denom.**patient\_id  
 LEFT JOIN** (**SELECT DISTINCT patient\_id**, **first\_visit\_date**, **icd\_code  
 FROM** exclusion\_dx\_temp  
 **WHERE row** = 11) **AS** exclusions\_11  
 **ON** exclusions\_11.**patient\_id** = denom.**patient\_id  
 LEFT JOIN** (**SELECT DISTINCT patient\_id**, **first\_visit\_date**, **icd\_code  
 FROM** exclusion\_dx\_temp  
 **WHERE row** = 12) **AS** exclusions\_12  
 **ON** exclusions\_12.**patient\_id** = denom.**patient\_id  
 LEFT JOIN** (**SELECT DISTINCT patient\_id**, **first\_visit\_date**, **icd\_code  
 FROM** exclusion\_dx\_temp  
 **WHERE row** = 13) **AS** exclusions\_13  
 **ON** exclusions\_13.**patient\_id** = denom.**patient\_id  
 LEFT JOIN** (**SELECT DISTINCT patient\_id**, **first\_visit\_date**, **icd\_code  
 FROM** exclusion\_dx\_temp  
 **WHERE row** = 14) **AS** exclusions\_14  
 **ON** exclusions\_14.**patient\_id** = denom.**patient\_id  
 LEFT JOIN** (**SELECT DISTINCT patient\_id**, **first\_visit\_date**, **icd\_code  
 FROM** exclusion\_dx\_temp  
 **WHERE row** = 15) **AS** exclusions\_15  
 **ON** exclusions\_15.**patient\_id** = denom.**patient\_id  
ORDER BY** denom.**patient\_id**;  
  
  
*--Determine if any of the exclusion diagnosis codes displayed in the exclusion columns have been resolved since  
-- the time of service***DROP TABLE IF EXISTS** exclusion\_dx\_resolved\_temp;  
**CREATE TEMPORARY TABLE** exclusion\_dx\_resolved\_temp **AS  
SELECT** exclusion\_dx\_temp.**patient\_id**,  
 **icd\_code AS** exclusion\_code,  
 problemlist.**resolvedon** :: **DATE AS** date\_resolved,  
 exclusion\_dx\_temp.**row AS** exclusion\_row\_orig,  
 *ROW\_NUMBER*() **OVER**(**PARTITION BY patient\_id ORDER BY row**) **AS** resolution\_row  
**FROM** exclusion\_dx\_temp  
 **INNER JOIN** problemlist **ON** problemlist.**patientid** = exclusion\_dx\_temp.**patient\_id  
 INNER JOIN** itemdetail **ON** itemdetail.**itemID** = problemlist.**asmtid AND propID** = 13  
 **AND** itemdetail.**value** = exclusion\_dx\_temp.**icd\_code  
WHERE row BETWEEN** 1 **and** 15  
 **AND resolvedon IS NOT NULL**;  
  
  
*--Prepare the resolution columns for the report output***DROP TABLE IF EXISTS** exclusion\_resolved\_columns\_temp;  
**CREATE TEMPORARY TABLE** exclusion\_resolved\_columns\_temp **AS  
SELECT** denom.**patient\_id**,  
 excl\_resolve\_1.**exclusion\_code AS** exclusions\_icddx10\_1\_resolution,  
 excl\_resolve\_1.**date\_resolved AS** exclusions\_icddx10\_1\_resolution\_date,  
 excl\_resolve\_2.**exclusion\_code AS** exclusions\_icddx10\_2\_resolution,  
 excl\_resolve\_2.**date\_resolved AS** exclusions\_icddx10\_2\_resolution\_date,  
 excl\_resolve\_3.**exclusion\_code AS** exclusions\_icddx10\_3\_resolution,  
 excl\_resolve\_3.**date\_resolved AS** exclusions\_icddx10\_3\_resolution\_date,  
 excl\_resolve\_4.**exclusion\_code AS** exclusions\_icddx10\_4\_resolution,  
 excl\_resolve\_4.**date\_resolved AS** exclusions\_icddx10\_4\_resolution\_date  
**FROM**(**SELECT DISTINCT patient\_id FROM** exclusion\_dx\_resolved\_temp) **AS** denom  
 **INNER JOIN** (**SELECT patient\_id**, **exclusion\_code**,**date\_resolved  
 FROM** exclusion\_dx\_resolved\_temp  
 **WHERE resolution\_row** = 1) **AS** excl\_resolve\_1  
 **ON** excl\_resolve\_1.**patient\_id** = denom.**patient\_id  
 LEFT JOIN** (**SELECT patient\_id**, **exclusion\_code**,**date\_resolved  
 FROM** exclusion\_dx\_resolved\_temp  
 **WHERE resolution\_row** = 2) **AS** excl\_resolve\_2  
 **ON** excl\_resolve\_2.**patient\_id** = denom.**patient\_id  
 LEFT JOIN** (**SELECT patient\_id**, **exclusion\_code**,**date\_resolved  
 FROM** exclusion\_dx\_resolved\_temp  
 **WHERE resolution\_row** = 3) **AS** excl\_resolve\_3  
 **ON** excl\_resolve\_3.**patient\_id** = denom.**patient\_id  
 LEFT JOIN** (**SELECT patient\_id**, **exclusion\_code**,**date\_resolved  
 FROM** exclusion\_dx\_resolved\_temp  
 **WHERE resolution\_row** = 4) **AS** excl\_resolve\_4  
 **ON** excl\_resolve\_4.**patient\_id** = denom.**patient\_id**;  
  
  
*--Report output***SELECT DISTINCT**  
 total\_denominator\_temp.**enc\_date AS** date\_of\_service,  
 **NULL AS** site\_id\_number,  
 **NULL AS** provider\_site\_name,  
 total\_denominator\_temp.**visit\_clinician\_full\_name**,  
 total\_denominator\_temp.**clinician\_npi\_number**,  
 total\_denominator\_temp.**subscriber\_number AS** member\_cin,  
 **NULL AS** membername,  
 total\_denominator\_temp.**memberfirstname**,  
 **NULL AS** membermiddlename,  
 total\_denominator\_temp.**memberlastname**,  
 total\_denominator\_temp.**dob**,  
 total\_denominator\_temp.**sex**,  
 cpt\_columns\_temp.**cpt1**,  
 cpt\_columns\_temp.**cpt2**,  
 cpt\_columns\_temp.**cpt3**,  
 cpt\_columns\_temp.**cpt4**,  
 cpt\_columns\_temp.**cptmod**,  
 cpt\_columns\_temp.**hcpcs**,  
 cpt\_columns\_temp.**hcpcmod**,  
 **CASE WHEN** preg\_columns\_temp.**enc\_id IS NULL  
 THEN 'No' ELSE 'Yes' END AS** pregnant\_at\_time\_of\_visit,  
 preg\_columns\_temp.**future\_edd\_date AS** estimated\_date\_of\_delivery,  
 preg\_columns\_temp.**past\_date\_of\_delivery AS** actual\_date\_of\_delivery,  
 preg\_columns\_temp.**gestation\_weeks\_and\_days\_at\_time\_of\_service**,  
 **CASE WHEN NOT** del\_date\_postpart.**past\_date\_of\_delivery IS NULL  
 AND** del\_date\_postpart.**past\_date\_of\_delivery** <= total\_denominator\_temp.**enc\_date  
 THEN** *TRUNC*(*DATE\_PART*(**'day'**, total\_denominator\_temp.**enc\_date** ::**timestamp** - del\_date\_postpart.**past\_date\_of\_delivery** :: **timestamp**)/7) || **'W'** ||  
 *DATE\_PART*(**'day'**, total\_denominator\_temp.**enc\_date** ::**timestamp** - del\_date\_postpart.**past\_date\_of\_delivery** :: **timestamp**) -  
 (*TRUNC*(*DATE\_PART*(**'day'**, total\_denominator\_temp.**enc\_date** ::**timestamp** - del\_date\_postpart.**past\_date\_of\_delivery** :: **timestamp**)/7) \* 7) || **'D'  
 END AS** weeks\_and\_days\_postpartum,  
 preg\_columns\_temp.**weeks\_and\_days\_gestation\_at\_time\_of\_delivery**,  
 depression\_screen\_columns\_temp.**loinc\_tool\_1 AS** loinc\_code\_1,  
 depression\_screen\_columns\_temp.**name\_tool\_1 AS** depression\_screening\_tool\_name\_1,  
 depression\_screen\_columns\_temp.**loinc\_tool\_2 AS** loinc\_code\_2,  
 depression\_screen\_columns\_temp.**name\_tool\_2 AS** depression\_screening\_tool\_name\_2,  
 depression\_screen\_columns\_temp.**score\_tool\_1 AS** score\_screening\_tool\_1,  
 depression\_screen\_columns\_temp.**result\_tool\_1 AS** result\_of\_screening\_1,  
 depression\_screen\_columns\_temp.**score\_tool\_2 AS** score\_screening\_tool\_2,  
 depression\_screen\_columns\_temp.**result\_tool\_2 AS** result\_of\_screening\_2,  
 depression\_dx\_column\_temp.**icd\_code AS** icddx10,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_1**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_1\_dos**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_2**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_2\_dos**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_3**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_3\_dos**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_4**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_4\_dos**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_5**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_5\_dos**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_6**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_6\_dos**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_7**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_7\_dos**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_8**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_8\_dos**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_9**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_9\_dos**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_10**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_10\_dos**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_11**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_11\_dos**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_12**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_12\_dos**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_13**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_13\_dos**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_14**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_14\_dos**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_15**,  
 exclusion\_columns\_temp.**exclusions\_icddx10\_15\_dos**,  
 exclusion\_resolved\_columns\_temp.**exclusions\_icddx10\_1\_resolution**,  
 exclusion\_resolved\_columns\_temp.**exclusions\_icddx10\_1\_resolution\_date**,  
 exclusion\_resolved\_columns\_temp.**exclusions\_icddx10\_2\_resolution**,  
 exclusion\_resolved\_columns\_temp.**exclusions\_icddx10\_2\_resolution\_date**,  
 exclusion\_resolved\_columns\_temp.**exclusions\_icddx10\_3\_resolution**,  
 exclusion\_resolved\_columns\_temp.**exclusions\_icddx10\_3\_resolution\_date**,  
 exclusion\_resolved\_columns\_temp.**exclusions\_icddx10\_4\_resolution**,  
 exclusion\_resolved\_columns\_temp.**exclusions\_icddx10\_4\_resolution\_date  
FROM** total\_denominator\_temp  
 **LEFT JOIN** cpt\_columns\_temp **ON** cpt\_columns\_temp.**enc\_id** = total\_denominator\_temp.**enc\_id  
 LEFT JOIN** preg\_columns\_temp **ON** preg\_columns\_temp.**enc\_id** = total\_denominator\_temp.**enc\_id  
 LEFT JOIN** preg\_columns\_temp **AS** del\_date\_postpart **ON** del\_date\_postpart.**patient\_id** = total\_denominator\_temp.**patient\_id  
 AND** del\_date\_postpart.**past\_date\_of\_delivery IS NOT NULL  
 LEFT JOIN** depression\_screen\_columns\_temp **ON** depression\_screen\_columns\_temp.**enc\_id** = total\_denominator\_temp.**enc\_id  
 LEFT JOIN** depression\_dx\_column\_temp **ON** depression\_dx\_column\_temp.**enc\_id** = total\_denominator\_temp.**enc\_id  
 LEFT JOIN** exclusion\_columns\_temp **ON** exclusion\_columns\_temp.**patient\_id** = total\_denominator\_temp.**patient\_id  
 LEFT JOIN** exclusion\_resolved\_columns\_temp **ON** exclusion\_resolved\_columns\_temp.**patient\_id** = total\_denominator\_temp.**patient\_id  
ORDER BY** total\_denominator\_temp.memberlastname, total\_denominator\_temp.memberfirstname,  
 total\_denominator\_temp.dob, total\_denominator\_temp.enc\_date

# Appendix D: SQL Code for the Report “ECDS: Unhealthy Alcohol Use Screening and Follow Up (ASF-E)” and Related Transformers/Data Elements

NEW Transformer/Data Element Pair(s) for Alcohol Screens

Configure depending on which screen(s) are used at your health center. There are three possible screens and three possible Transformer/Data Element pairs. Note that the ID numbers for catid, itemid, and detailed will be unique at each health center. One option to find these ID numbers is the report “RCHC List All Structured Data Items” (available on the RCHC Aggregate for eCW health centers).

The sample code for the Transformer/Data Element pair below is for the Audit C screen. The Audit screen and the single-question screen should take a similar approach. See the above section for this measure for a table with Transformer and Data Element names for the three screens. In the code below, the screen is located in Social History, but it may be in another structured data location at your health center.

**Transformer**: Build relevant\_audit\_c\_screens

**DROP TABLE IF EXISTS** relevant**\_**audit\_c\_screens;  
**CREATE TABLE** relevant**\_**audit\_c\_screens **AS  
SELECT DISTINCT**  
 enc.**patientid AS** patient\_id,  
 enc.**date** :: **DATE AS** performed\_on,  
 audit\_c\_score.**value AS** score,  
 **CASE WHEN** audit\_c\_interpret.**value** = **'Positive' THEN TRUE ELSE FALSE END AS** positive  
**FROM** enc  
 **INNER JOIN** structsocialhistory **AS** audit\_c\_score **ON** audit\_c\_score.**encounterid** = enc.**encounterid  
 AND** audit\_c\_score.**catid** = 262111  
 **AND** audit\_c\_score.**itemid** = 262128  
 **AND** audit\_c\_score.**detailid** = 1004 *--Use a unique combination of codes to identify the AUDIT-C score* **AND** audit\_c\_score.**value** ~ **'^\d'  
 LEFT JOIN** structsocialhistory **AS** audit\_c\_interpret **ON** audit\_c\_interpret.**encounterid** = enc.**encounterid  
 AND** audit\_c\_interpret.**catid** = 262111  
 **AND** audit\_c\_interpret.**itemid** = 262128  
 **AND** audit\_c\_interpret.**detailid** = 1005 *--Use a unique combination of codes to identify the AUDIT-C interpretation* **AND NOT** audit\_c\_interpret.**value IS NULL  
 AND NOT** audit\_c\_interpret.**value** = **' '  
WHERE** enc.**deleteflag** = 0

**AND** enc.**status** = **'CHK'**;

**Data Element:** AUDIT C Screens

**SELECT DISTINCT  
 patient\_id**,  
 **performed\_on**,  
 **score**,  
 **positive  
FROM** relevant**\_**audit\_c\_screens

NEW Transformer/Data Element Pair for Alcohol Follow-up Counseling

The report output displays alcohol follow-up counseling entered as a CPT code, a HCPCS code, an ICD-10 code, or into structured data. The CPT and HCPCS codes are extracted directly from claims and the ICD-10 codes from visit assessments by the report so no separate Transformers or Data Elements are needed for them. The alcohol follow-up Transformer/Data Element pair identifies alcohol counseling only from structured data. Similar to the alcohol screening Transformer, the location and ID numbers for the counseling item will be unique at each health center.

**Transformer:** Build relevant\_alcohol\_counseling

**DROP TABLE IF EXISTS** relevant\_alcohol\_counseling;  
**CREATE TABLE** relevant\_alcohol\_counseling **AS  
SELECT DISTINCT** enc.**patientid AS** patient\_id,  
 enc.**date** :: **DATE AS** performed\_on  
**FROM** enc  
 **INNER JOIN** structpreventive **ON** structpreventive.**encounterid** = enc.**encounterid  
WHERE** structpreventive.**catid** = 306460  
 **AND** structpreventive.**itemid** = 306462  
 **AND** structpreventive.**detailid** = 1021 *--Use a unique combination of codes to identify the alcohol counseling item* **AND NOT** structpreventive.**value IS NULL  
 AND NOT** structpreventive.**value** = **' '  
 AND NOT** structpreventive.**value ILIKE 'No%'  
 AND** enc.**deleteflag** = 0  
 **AND** enc.**status** = **'CHK'**;

**Data Element:** Alcohol Counseling Or Other Followups

**SELECT  
 patient\_id**,  
 **performed\_on  
FROM** relevant\_alcohol\_counseling

Report SQL

**Database:** Production Database

**Description**: This report displays data for the ECDS Unhealthy Alcohol Use Screening and Follow Up (ASF-E) measure that could be submitted to Partnership HealthPlan of California (2022 Edition)

**Parameters**: {{start\_date}} and {{end\_date}}

**Report SQL Code:**

*/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
Report Name: ECDS: Unhealthy Alcohol Use Screening and Follow Up (ASF-E)  
  
Code Edited By: Ben Fouts for RCHC  
  
Description: This report displays data for submission to Partnership Healthplan. It should not be used to  
 directly evaluate a Quality Measure or for case management purposes  
  
RCHC Started: June 2022 (based on code from the Partnership SQL specifications document)  
  
Version Date: July 25, 2022  
  
Revision History: 2022 Version 1  
  
Resources Used: patients, visits, providers, locations, genders, hedis\_value\_set\_codes, visit\_billing\_codes,   
 billing\_codes, visit\_diagnosis\_codes, patient\_diagnoses, diagnosis\_codes, audit\_c\_screens, audit\_screens,   
 single\_question\_alcohol\_use\_screens, alcohol\_counseling\_or\_other\_followups  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
  
  
-- Universe Definition:  
-- 1) Visit date in the measurement period  
-- 2) Patient age 12 years and older at the start of the measurement period  
-- 3) Patient currently has Partnership HealthPlan insurance -- this is the default, but can be changed  
-- by the programmer to join to internal tables containing Partnership patients from  
-- monthly membership files during the measurement period***DROP TABLE IF EXISTS** universe\_temp\_raw;  
**CREATE TEMPORARY TABLE** universe\_temp\_raw **AS  
SELECT DISTINCT** patients.**id AS** patient\_id,  
 patients.**mrn**,  
 *LEFT*(patients.**subscriber\_number**,10) **AS** Member\_Key,  
 patients.**first\_name AS** Member\_First\_Name,  
 patients.**middle\_name AS** Member\_Middle\_Name,  
 patients.**last\_name AS** Member\_Last\_Name,  
 patients.**date\_of\_birth** :: **DATE AS** DOB,  
 *EXTRACT*(**YEAR FROM** *age*({{start\_date}}, patients.date\_of\_birth :: DATE)) **AS** Age\_startMP,  
 **CASE WHEN** genders.**name** = **'Female' THEN 'F'  
 WHEN** genders.**name** = **'Male' THEN 'M'  
 WHEN** genders.**name** = **'Unknown' OR** genders.**name** = **'Choose not to disclose' THEN 'U'  
 WHEN** genders.**name IS NULL THEN 'U'  
 ELSE 'O' END AS** Sex,  
 visits.**id AS** encounter\_id,  
 visits.**visit\_date** :: **DATE AS** Date\_of\_Service,  
 locations.**name AS** Site\_Name,  
 providers.**first\_name AS** Clinician\_First\_Name,  
 providers.**middle\_name AS** Clinician\_Middle\_Name,  
 providers.**last\_name AS** Clinician\_Last\_Name,  
 providers.**npi AS** Clinician\_NPI\_Number  
**FROM** patients  
 **INNER JOIN** visits **ON** visits.**patient\_id** = patients.**id  
 INNER JOIN** providers **ON** providers.**id** = visits.**provider\_id  
 LEFT JOIN** genders **ON** genders.**id** = patients.**gender\_id  
 LEFT JOIN** locations **ON** locations.**id** = visits.**location\_id  
WHERE** *EXTRACT*(**YEAR FROM** *age*({{start\_date}}, patients.date\_of\_birth :: DATE)) >=12  
 **AND** visits.**visit\_date BETWEEN** {{start\_date}} **AND** {{end\_date}}  
 **AND** patients.primary\_insurance\_group\_id = 17; *-- "PARTNERSHIP MANAGED CARE"  
  
  
-- List alcohol counseling and other followup in structured data (one code per encounter ID)***DROP TABLE IF EXISTS** alc\_followup\_struct\_raw\_temp;  
**CREATE TEMPORARY TABLE** alc\_followup\_struct\_raw\_temp **AS  
SELECT DISTINCT** universe\_temp\_raw.**patient\_id**,  
 alcohol\_counseling\_or\_other\_followups.**performed\_on**,  
 **'413473000'** :: **VARCHAR AS** snomed\_code *--This is the SNOMED code that most closely describes health center counseling***FROM** alcohol\_counseling\_or\_other\_followups  
 **INNER JOIN** universe\_temp\_raw **ON** universe\_temp\_raw.**patient\_id** = alcohol\_counseling\_or\_other\_followups.**patient\_id**;  
  
  
*--List alcohol counseling and other followup by Value Set CPT or HCPCS Value Set Codes  
-- (one code per type per encounter ID)***DROP TABLE IF EXISTS** alc\_followup\_claim\_raw\_temp;  
**CREATE TEMPORARY TABLE** alc\_followup\_claim\_raw\_temp **AS  
SELECT DISTINCT ON** (universe\_temp\_raw.**encounter\_id**, hedis\_value\_set\_codes.**code\_system\_name**)  
 universe\_temp\_raw.**encounter\_id**,  
 universe\_temp\_raw.**patient\_id**,  
 hedis\_value\_set\_codes.**code\_system\_name AS** code\_system,  
 billing\_codes.**code AS** claim\_code  
**FROM** universe\_temp\_raw  
 **INNER JOIN** visits **ON** visits.**patient\_id** = universe\_temp\_raw.**patient\_id  
 INNER JOIN** visit\_billing\_codes **ON** visit\_billing\_codes.**visit\_id** = visits.**id  
 INNER JOIN** billing\_codes **ON** billing\_codes.**id** = visit\_billing\_codes.**billing\_code\_id  
 INNER JOIN** hedis\_value\_set\_codes **ON** hedis\_value\_set\_codes.**code\_value** = billing\_codes.**code  
WHERE** visits.**visit\_date** :: **DATE BETWEEN** {{start\_date}} **AND** {{end\_date}}  
 **AND** hedis\_value\_set\_codes.value\_set\_name = **'Alcohol Counseling or Other Follow Up Care'  
 AND** hedis\_value\_set\_codes.latest = **'TRUE'  
ORDER BY** universe\_temp\_raw.encounter\_id, hedis\_value\_set\_codes.code\_system\_name, billing\_codes.code;  
  
  
*-- List alcohol counseling and other followup by Value Set ICD-10 Value Set Codes (one code per encounter ID)***DROP TABLE IF EXISTS** alc\_followup\_assess\_raw\_temp;  
**CREATE TEMPORARY TABLE** alc\_followup\_assess\_raw\_temp **AS  
SELECT DISTINCT ON** (universe\_temp\_raw.**encounter\_id**)  
 universe\_temp\_raw.**encounter\_id**,  
 universe\_temp\_raw.**patient\_id**,  
 diagnosis\_codes.**code AS** assess\_code  
**FROM** universe\_temp\_raw  
 **INNER JOIN** visits **ON** visits.**patient\_id** = universe\_temp\_raw.**patient\_id  
 INNER JOIN** visit\_diagnosis\_codes **ON** visit\_diagnosis\_codes.**visit\_id** = visits.**id  
 INNER JOIN** diagnosis\_codes **ON** diagnosis\_codes.**id** = visit\_diagnosis\_codes.**diagnosis\_code\_id  
 INNER JOIN** hedis\_value\_set\_codes **ON** hedis\_value\_set\_codes.**code\_value** = diagnosis\_codes.**code  
WHERE** visits.**visit\_date** :: **DATE BETWEEN** {{start\_date}} **AND** {{end\_date}}  
 **AND** hedis\_value\_set\_codes.value\_set\_name = **'Alcohol Counseling or Other Follow Up Care'  
 AND** hedis\_value\_set\_codes.latest = **'TRUE'  
ORDER BY** universe\_temp\_raw.encounter\_id, hedis\_value\_set\_codes.code\_system\_name, diagnosis\_codes.code;  
  
  
*-- Identify start date of exclusion diagnoses from the Problem List and Assessments***DROP TABLE IF EXISTS** exclusions\_raw\_temp;  
**CREATE TEMPORARY TABLE** exclusions\_raw\_temp **AS  
SELECT** patient\_id,  
 code\_group,  
 exclude\_code,  
 *MIN*(dx\_date) **AS** first\_dx\_date,  
 *MAX*(ended\_on) **AS** resolution\_date  
**FROM** (  
 **SELECT** universe\_temp\_raw.**patient\_id AS** patient\_id,  
 hedis\_value\_set\_codes.**value\_set\_name AS** code\_group,  
 diagnosis\_codes.**code AS** exclude\_code,  
 patient\_diagnoses.**start\_date AS** dx\_date,  
 **CASE WHEN** patient\_diagnoses.**end\_date** <= {{end\_date}}  
 **THEN** patient\_diagnoses.**end\_date END AS** ended\_on  
 **FROM** universe\_temp\_raw  
 **INNER JOIN** patient\_diagnoses **ON** patient\_diagnoses.**patient\_id** = universe\_temp\_raw.**patient\_id  
 INNER JOIN** diagnosis\_codes **ON** diagnosis\_codes.**id** = patient\_diagnoses.**diagnosis\_code\_id  
 INNER JOIN** hedis\_value\_set\_codes **ON** hedis\_value\_set\_codes.**code\_value** = diagnosis\_codes.**code  
 WHERE** hedis\_value\_set\_codes.**value\_set\_name IN**(**'Alcohol Use Disorder'**, **'Dementia'**)  
 **AND** hedis\_value\_set\_codes.**latest** = **'TRUE'  
 UNION  
 SELECT** universe\_temp\_raw.**patient\_id AS** patient\_id,  
 hedis\_value\_set\_codes.**value\_set\_name AS** code\_group,  
 diagnosis\_codes.**code AS** exclude\_code,  
 visits.**visit\_date** :: **DATE AS** dx\_date,  
 **NULL AS** ended\_on  
 **FROM** universe\_temp\_raw  
 **INNER JOIN** visits **ON** visits.**patient\_id** = universe\_temp\_raw.**patient\_id  
 INNER JOIN** visit\_diagnosis\_codes **ON** visit\_diagnosis\_codes.**visit\_id** = visits.**id  
 INNER JOIN** diagnosis\_codes **ON** diagnosis\_codes.**id** = visit\_diagnosis\_codes.**diagnosis\_code\_id  
 INNER JOIN** hedis\_value\_set\_codes **ON** hedis\_value\_set\_codes.**code\_value** = diagnosis\_codes.**code  
 WHERE** visits.**visit\_date** :: **DATE BETWEEN** {{start\_date}} **AND** {{end\_date}}  
 **AND** hedis\_value\_set\_codes.value\_set\_name **IN**(**'Alcohol Use Disorder'**, **'Dementia'**)  
 **AND** hedis\_value\_set\_codes.latest = **'TRUE'** ) **AS** raw\_exclusions  
**WHERE** dx\_date <= {{end\_date}}  
**GROUP BY** patient\_id, code\_group, exclude\_code;  
  
  
*-- Add row numbers to exclusions so they can be put into output columns***DROP TABLE IF EXISTS** exclusions\_raw\_final;  
**CREATE TEMPORARY TABLE** exclusions\_raw\_final **AS  
SELECT  
 patient\_id**,  
 **code\_group**,  
 **exclude\_code**,  
 **first\_dx\_date**,  
 **resolution\_date**,  
 *ROW\_NUMBER* () **OVER** (  
 **PARTITION BY patient\_id  
 ORDER BY code\_group DESC**, **resolution\_date DESC**, **first\_dx\_date**) **AS** row\_excl  
**FROM** exclusions\_raw\_temp;  
  
  
*-- Results query  
-- First part displays individual visits with unhealthy alcohol screening and/or counseling/followup  
-- Second part (after UNION) displays the exclusions independent of any visit***SELECT** *\** **FROM** (  
**SELECT** universe\_temp\_raw.**Date\_of\_Service**,  
 universe\_temp\_raw.**Site\_Name**,  
 **NULL AS** Site\_ID\_Number, *--Needs to be added from service provider's system* **NULL AS** Provider\_Key, *--Needs to be added from service provider's system* universe\_temp\_raw.**Clinician\_NPI\_Number**,  
 **NULL AS** Clinician\_Name, *--Optional* universe\_temp\_raw.**Clinician\_First\_Name**,  
 universe\_temp\_raw.**Clinician\_Middle\_Name**,  
 universe\_temp\_raw.**Clinician\_Last\_Name**,  
 universe\_temp\_raw.**Member\_Key**,  
 **NULL AS** Member\_Name, *--Optional* universe\_temp\_raw.**Member\_First\_Name**,  
 universe\_temp\_raw.**Member\_Middle\_Name**,  
 universe\_temp\_raw.**Member\_Last\_Name**,  
 universe\_temp\_raw.**DOB**,  
 universe\_temp\_raw.**Sex**,  
 audit\_score\_temp.Audit\_Test\_Score,  
 audit\_score\_temp.Audit\_Score\_Interpretation,  
 auditc\_score\_temp.AuditC\_Test\_Score,  
 auditc\_score\_temp.AuditC\_Score\_Interpretation,  
 single\_q\_score\_temp.Single\_Q\_Score,  
 single\_q\_score\_temp.Single\_Q\_Interpretation,  
 couns\_cpt\_temp.Counseling\_CPT,  
 couns\_hcpcs\_temp.Counseling\_HCPCS,  
 couns\_icd\_temp.Counseling\_ICD10,  
 couns\_struct\_temp.Counseling\_SNOMED,  
 **NULL AS** Exclusions\_ICDDX10\_1,  
 **NULL AS** Exclusions\_ICDDX10\_1\_DOS,  
 **NULL AS** Exclusions\_ICDDX10\_2,  
 **NULL AS** Exclusions\_ICDDX10\_2\_DOS,  
 **NULL AS** Exclusions\_ICDDX10\_3,  
 **NULL AS** Exclusions\_ICDDX10\_3\_DOS,  
 **NULL AS** Exclusions\_ICDDX10\_4,  
 **NULL AS** Exclusions\_ICDDX10\_4\_DOS,  
 **NULL AS** Exclusions\_ICDDX10\_5,  
 **NULL AS** Exclusions\_ICDDX10\_5\_DOS,  
 **NULL AS** Exclusions\_ICDDX10\_1\_Resolution,  
 **NULL AS** Exclusions\_ICDDX10\_2\_Resolution,  
 **NULL AS** Exclusions\_ICDDX10\_3\_Resolution,  
 **NULL AS** Exclusions\_ICDDX10\_4\_Resolution,  
 **NULL AS** Exclusions\_ICDDX10\_5\_Resolution  
**FROM** universe\_temp\_raw  
 **LEFT JOIN** (**SELECT patient\_id**,  
 **performed\_on**,  
 **score AS** Audit\_Test\_Score,  
 **CASE WHEN positive THEN 'Positive'  
 ELSE 'Negative' END AS** Audit\_Score\_Interpretation  
 **FROM** audit\_screens) **AS** audit\_score\_temp  
 **ON** audit\_score\_temp.**patient\_id** = universe\_temp\_raw.**patient\_id  
 AND** audit\_score\_temp.**performed\_on** :: **DATE** = universe\_temp\_raw.**Date\_of\_Service** :: **DATE  
 LEFT JOIN** (**SELECT patient\_id**,  
 **performed\_on**,  
 **score AS** AuditC\_Test\_Score,  
 **CASE WHEN positive THEN 'Positive'  
 ELSE 'Negative' END AS** AuditC\_Score\_Interpretation  
 **FROM** audit\_c\_screens) **AS** auditc\_score\_temp  
 **ON** auditc\_score\_temp.**patient\_id** = universe\_temp\_raw.**patient\_id  
 AND** auditc\_score\_temp.**performed\_on** :: **DATE** = universe\_temp\_raw.**Date\_of\_Service** :: **DATE  
 LEFT JOIN** (**SELECT patient\_id**,  
 **performed\_on**,  
 **score AS** Single\_Q\_Score,  
 **CASE WHEN positive THEN 'Positive'  
 ELSE 'Negative' END AS** Single\_Q\_Interpretation  
 **FROM** single\_question\_alcohol\_use\_screens) **AS** single\_q\_score\_temp  
 **ON** single\_q\_score\_temp.**patient\_id** = universe\_temp\_raw.**patient\_id  
 AND** single\_q\_score\_temp.**performed\_on** :: **DATE** = universe\_temp\_raw.**Date\_of\_Service** :: **DATE  
 LEFT JOIN** (**SELECT encounter\_id**,  
 **claim\_code AS** Counseling\_CPT  
 **FROM** alc\_followup\_claim\_raw\_temp  
 **WHERE code\_system** = **'CPT'**) **AS** couns\_cpt\_temp  
 **ON** couns\_cpt\_temp.**encounter\_id** = universe\_temp\_raw.**encounter\_id  
 LEFT JOIN** (**SELECT encounter\_id**,  
 **claim\_code AS** Counseling\_HCPCS  
 **FROM** alc\_followup\_claim\_raw\_temp  
 **WHERE code\_system** = **'HCPCS'**) **AS** couns\_hcpcs\_temp  
 **ON** couns\_hcpcs\_temp.**encounter\_id** = universe\_temp\_raw.**encounter\_id  
 LEFT JOIN** (**SELECT encounter\_id**,  
 **assess\_code AS** Counseling\_ICD10  
 **FROM** alc\_followup\_assess\_raw\_temp) **AS** couns\_icd\_temp  
 **ON** couns\_icd\_temp.**encounter\_id** = universe\_temp\_raw.**encounter\_id  
 LEFT JOIN** (**SELECT patient\_id**,  
 **performed\_on**,  
 **snomed\_code AS** Counseling\_SNOMED  
 **FROM** alc\_followup\_struct\_raw\_temp) **AS** couns\_struct\_temp  
 **ON** couns\_struct\_temp.**patient\_id** = universe\_temp\_raw.**patient\_id  
 AND** couns\_struct\_temp.**performed\_on** :: **DATE** = universe\_temp\_raw.**Date\_of\_Service** :: **DATE  
WHERE** audit\_score\_temp.**patient\_id IS NOT NULL  
 OR** auditc\_score\_temp.**patient\_id IS NOT NULL  
 OR** single\_q\_score\_temp.**patient\_id IS NOT NULL  
 OR** couns\_cpt\_temp.**encounter\_id IS NOT NULL  
 OR** couns\_hcpcs\_temp.**encounter\_id IS NOT NULL  
 OR** couns\_icd\_temp.**encounter\_id IS NOT NULL  
 OR** couns\_struct\_temp.**patient\_id IS NOT NULL  
UNION  
SELECT** *-- This part is for exclusions that are not attatched to a specific visit* **NULL AS** Date\_of\_Service,  
 **NULL AS** Site\_Name,  
 **NULL AS** Site\_ID\_Number,  
 **NULL AS** Provider\_Key,  
 **NULL AS** Clinician\_NPI\_Number,  
 **NULL AS** Clinician\_Name,  
 **NULL AS** Clinician\_First\_Name,  
 **NULL AS** Clinician\_Middle\_Name,  
 **NULL AS** Clinician\_Last\_Name,  
 undup\_pts.**Member\_Key**,  
 **NULL AS** Member\_Name, *--Optional* undup\_pts.**Member\_First\_Name**,  
 undup\_pts.**Member\_Middle\_Name**,  
 undup\_pts.**Member\_Last\_Name**,  
 undup\_pts.**DOB**,  
 undup\_pts.**Sex**,  
 **NULL AS** Audit\_Test\_Score,  
 **NULL AS** Audit\_Score\_Interpretation,  
 **NULL AS** AuditC\_Test\_Score,  
 **NULL AS** AuditC\_Score\_Interpretation,  
 **NULL AS** Single\_Q\_Score,  
 **NULL AS** Single\_Q\_Interpretation,  
 **NULL AS** Counseling\_CPT,  
 **NULL AS** Counseling\_HCPCS,  
 **NULL AS** Counseling\_ICD10,  
 **NULL AS** Counseling\_SNOMED,  
 exclusions\_1a.Exclusions\_ICDDX10\_1,  
 exclusions\_1a.Exclusions\_ICDDX10\_1\_DOS,  
 exclusions\_2a.Exclusions\_ICDDX10\_2,  
 exclusions\_2a.Exclusions\_ICDDX10\_2\_DOS,  
 exclusions\_3a.Exclusions\_ICDDX10\_3,  
 exclusions\_3a.Exclusions\_ICDDX10\_3\_DOS,  
 exclusions\_4a.Exclusions\_ICDDX10\_4,  
 exclusions\_4a.Exclusions\_ICDDX10\_4\_DOS,  
 exclusions\_5a.Exclusions\_ICDDX10\_5,  
 exclusions\_5a.Exclusions\_ICDDX10\_5\_DOS,  
 exclusions\_1a.Exclusions\_ICDDX10\_1\_Resolution,  
 exclusions\_2a.Exclusions\_ICDDX10\_2\_Resolution,  
 exclusions\_3a.Exclusions\_ICDDX10\_3\_Resolution,  
 exclusions\_4a.Exclusions\_ICDDX10\_4\_Resolution,  
 exclusions\_5a.Exclusions\_ICDDX10\_5\_Resolution  
**FROM** (**SELECT DISTINCT patient\_id**,  
 **Member\_Key**,  
 **Member\_First\_Name**,  
 **Member\_Middle\_Name**,  
 **Member\_Last\_Name**,  
 **DOB**,  
 **Sex  
 FROM** universe\_temp\_raw) **AS** undup\_pts  
 **LEFT JOIN** (**SELECT patient\_id**,  
 **exclude\_code AS** Exclusions\_ICDDX10\_1,  
 **first\_dx\_date AS** Exclusions\_ICDDX10\_1\_DOS,  
 **resolution\_date AS** Exclusions\_ICDDX10\_1\_Resolution  
 **FROM** exclusions\_raw\_final  
 **WHERE row\_excl** = 1) **AS** exclusions\_1a  
 **ON** exclusions\_1a.**patient\_id** = undup\_pts.**patient\_id  
 LEFT JOIN** (**SELECT patient\_id**,  
 **exclude\_code AS** Exclusions\_ICDDX10\_2,  
 **first\_dx\_date AS** Exclusions\_ICDDX10\_2\_DOS,  
 **resolution\_date AS** Exclusions\_ICDDX10\_2\_Resolution  
 **FROM** exclusions\_raw\_final  
 **WHERE row\_excl** = 2) **AS** exclusions\_2a  
 **ON** exclusions\_2a.**patient\_id** = undup\_pts.**patient\_id  
 LEFT JOIN** (**SELECT patient\_id**,  
 **exclude\_code AS** Exclusions\_ICDDX10\_3,  
 **first\_dx\_date AS** Exclusions\_ICDDX10\_3\_DOS,  
 **resolution\_date AS** Exclusions\_ICDDX10\_3\_Resolution  
 **FROM** exclusions\_raw\_final  
 **WHERE row\_excl** = 3) **AS** exclusions\_3a  
 **ON** exclusions\_3a.**patient\_id** = undup\_pts.**patient\_id  
 LEFT JOIN** (**SELECT patient\_id**,  
 **exclude\_code AS** Exclusions\_ICDDX10\_4,  
 **first\_dx\_date AS** Exclusions\_ICDDX10\_4\_DOS,  
 **resolution\_date AS** Exclusions\_ICDDX10\_4\_Resolution  
 **FROM** exclusions\_raw\_final  
 **WHERE row\_excl** = 4) **AS** exclusions\_4a  
 **ON** exclusions\_4a.**patient\_id** = undup\_pts.**patient\_id  
 LEFT JOIN** (**SELECT patient\_id**,  
 **exclude\_code AS** Exclusions\_ICDDX10\_5,  
 **first\_dx\_date AS** Exclusions\_ICDDX10\_5\_DOS,  
 **resolution\_date AS** Exclusions\_ICDDX10\_5\_Resolution  
 **FROM** exclusions\_raw\_final  
 **WHERE row\_excl** = 5) **AS** exclusions\_5a  
 **ON** exclusions\_5a.**patient\_id** = undup\_pts.**patient\_id  
WHERE** exclusions\_1a.**patient\_id IS NOT NULL  
 OR** exclusions\_2a.**patient\_id IS NOT NULL  
 OR** exclusions\_3a.**patient\_id IS NOT NULL  
 OR** exclusions\_4a.**patient\_id IS NOT NULL  
 OR** exclusions\_5a.**patient\_id IS NOT NULL** ) **AS** all\_records  
**ORDER BY Member\_Last\_Name**, **Member\_First\_Name**, **Member\_Middle\_Name**, **Member\_Key**, Exclusions\_ICDDX10\_1\_DOS **DESC**, **Date\_of\_Service**

1. http://www.partnershiphp.org/Providers/Quality/Pages/Perinatal-QIP.aspx [↑](#footnote-ref-1)
2. https://www.rchc.net/population-health/data-analytics-and-governance/#toggle-id-2 [↑](#footnote-ref-2)
3. https://www.rchc.net/population-health/data-analytics-and-governance/#toggle-id-3 [↑](#footnote-ref-3)
4. You can use this SQL to look for the group number(s) at your health center: SELECT \* FROM payer\_groups [↑](#footnote-ref-4)
5. Such as procedure (CPT) and diagnosis (ICD) codes [↑](#footnote-ref-5)
6. The sample SQL code was designed in the Santa Rosa instance. The Transformers are named relevant\_phq2 and relevant\_phq9 [↑](#footnote-ref-6)
7. Health centers with eCW can use the report “RCHC List All Structured Data Items” to identify structured data items, along with the unique ID numbers that will be used in the SQL code [↑](#footnote-ref-7)