-- icnarc\_\_04\_\_estimate\_discharge\_date\_and\_status

-- This script creates 'icnarc\_04e\_unit\_discharge\_status\_estimated', where the date (and datetime) of unit discharge have been estimated by adding the length of unit stay to the unit admission datetime, the earliest and latest visit dates have been calculated for each admission record, and the status at discharge (alive/dead) has been estimated.

-- Note that tables are created with the naming convention

-- icnarc\_XXy\_description, where

-- XX is the number, e.g. '01' relating to the script number

-- y is a letter, e.g. 'a' relating to the order of table creation within the script

-- Tables are dropped when no longer needed

-- Steps:

-- 1) Concatenate unit admission date and time, and extract unit stay days and hours

-- 2) Estimate unit discharge date time by adding the length of unit stay to the unit admission datetime

-- 3) Extract just the date (also keep the datetime)

-- 4) Add a column containing the earliest and latest visit date of each record which will be used when the observation\_period table is built later

-- 5) Estimate unit discharge status by checking for entries in the death-related columns

-- The discharge date would be helpful for the visits table but we don't have it for whatever reason

-- We do, however, have length\_of\_unit\_stay (string of days and hours)

-- So convert length\_of\_unit\_stay to a date(time)

-- 1) Concatenate unit admission date and time

-- Extract unit stay days and hours (the only entry of 'mins' was 0)

CREATE TABLE `yhcr-prd-phm-bia-core.CY\_MYSPACE\_EmW.icnarc\_04a\_unit\_stay\_extracted`

AS

SELECT \*,

-- put unit admission date and time together to make a datetime

CAST(CONCAT(unit\_admission\_date,'T',unit\_admission\_time) AS DATETIME) AS unit\_admission\_datetime,

-- from length of unit stay, grab the days component (allow for 1 or more digits), cast as an int

CAST(REGEXP\_EXTRACT(length\_of\_unit\_stay, r'([0-9]+) days?') AS INT64) AS length\_of\_unit\_stay\_days\_component,

-- from length of unit stay, grab the hours component (allow for 2 digits), cast as an int

CAST(REGEXP\_EXTRACT(length\_of\_unit\_stay, r'([0-9]{1,2}) hours?') AS INT64) AS length\_of\_unit\_stay\_hours\_component

FROM `yhcr-prd-phm-bia-core.CY\_MYSPACE\_EmW.icnarc\_03h\_renamed\_columns`;

DROP TABLE `yhcr-prd-phm-bia-core.CY\_MYSPACE\_EmW.icnarc\_03h\_renamed\_columns`;

-- 2) Estimate discharge datetime

CREATE TABLE `yhcr-prd-phm-bia-core.CY\_MYSPACE\_EmW.icnarc\_04b\_unit\_discharge\_datetime\_estimated`

AS

SELECT \*

-- remove these two variables I created in the previous command

EXCEPT(

length\_of\_unit\_stay\_days\_component,

length\_of\_unit\_stay\_hours\_component

),

CASE

-- if neither are null, add both length of unit stay days and hours to unit admission date time

WHEN length\_of\_unit\_stay\_days\_component IS NOT NULL AND length\_of\_unit\_stay\_hours\_component IS NOT NULL

THEN DATETIME\_ADD(DATETIME\_ADD(unit\_admission\_datetime, INTERVAL length\_of\_unit\_stay\_days\_component DAY), INTERVAL length\_of\_unit\_stay\_hours\_component HOUR)

-- add just days if hours is null

WHEN length\_of\_unit\_stay\_days\_component IS NOT NULL AND length\_of\_unit\_stay\_hours\_component IS NULL

THEN DATETIME\_ADD(unit\_admission\_datetime, INTERVAL length\_of\_unit\_stay\_days\_component DAY)

-- add just hours if days is null

WHEN length\_of\_unit\_stay\_days\_component IS NULL AND length\_of\_unit\_stay\_hours\_component IS NOT NULL

THEN DATETIME\_ADD(unit\_admission\_datetime, INTERVAL length\_of\_unit\_stay\_hours\_component HOUR)

END AS estimated\_unit\_discharge\_datetime

FROM `yhcr-prd-phm-bia-core.CY\_MYSPACE\_EmW.icnarc\_04a\_unit\_stay\_extracted`;

DROP TABLE `yhcr-prd-phm-bia-core.CY\_MYSPACE\_EmW.icnarc\_04a\_unit\_stay\_extracted`;

-- 3) extract just the date of unit discharge from the new datetime

CREATE TABLE `yhcr-prd-phm-bia-core.CY\_MYSPACE\_EmW.icnarc\_04c\_unit\_discharge\_date\_estimated`

AS

SELECT \*,

EXTRACT(DATE FROM estimated\_unit\_discharge\_datetime) AS estimated\_unit\_discharge\_date

FROM `yhcr-prd-phm-bia-core.CY\_MYSPACE\_EmW.icnarc\_04b\_unit\_discharge\_datetime\_estimated`;

DROP TABLE `yhcr-prd-phm-bia-core.CY\_MYSPACE\_EmW.icnarc\_04b\_unit\_discharge\_datetime\_estimated`;

-- 4) Add a column containing the earliest and latest visit date of each record which will be used when the observation\_period table is built later

-- Add a column for the latest date for each person

-- Couldn't just use 'Greatest()' here as it would return null if any columns are null

-- Solution adapted from: https://justrocketscience.com/post/bigquery\_least\_nulls/

CREATE TEMP FUNCTION GREATEST\_ARRAY(arr ANY TYPE) AS ((

SELECT max(a) FROM UNNEST(arr) a WHERE a is not NULL

));

CREATE TEMP FUNCTION SMALLEST\_ARRAY(arr ANY TYPE) AS ((

SELECT min(a) FROM UNNEST(arr) a WHERE a is not NULL

));

-- Only include the dates that will be present in the visits table

CREATE TABLE `yhcr-prd-phm-bia-core.CY\_MYSPACE\_EmW.icnarc\_04d\_earliest\_and\_latest\_visit\_date`

AS

SELECT \*,

SMALLEST\_ARRAY([

--Date\_of\_birth,

--actual\_date\_of\_delivery,

--expected\_date\_of\_delivery,

--date\_1st\_managed\_by\_unit\_team,

--date\_of\_last\_critical\_care,

hospital\_admission\_date,

original\_hosp\_admission\_date,

original\_unit\_admission\_date,

--outreach\_last\_visit\_date,

unit\_admission\_date,

-----event\_date, (no one knows what this is so don't include it here - won't be shared with researchers)

date\_body\_removed,

--date\_of\_first\_critical\_care,

date\_ult\_unit\_discharge,

estimated\_unit\_discharge\_date]) earliest\_visit\_date,

GREATEST\_ARRAY([

--Date\_of\_birth,

--actual\_date\_of\_delivery,

--expected\_date\_of\_delivery,

--date\_1st\_managed\_by\_unit\_team,

--date\_of\_last\_critical\_care,

hospital\_admission\_date,

original\_hosp\_admission\_date,

original\_unit\_admission\_date,

--outreach\_last\_visit\_date,

unit\_admission\_date,

-----event\_date, (no one knows what this is so don't include it here - won't be shared with researchers)

date\_body\_removed,

--date\_of\_first\_critical\_care,

date\_ult\_unit\_discharge,

estimated\_unit\_discharge\_date]) latest\_visit\_date

FROM `yhcr-prd-phm-bia-core.CY\_MYSPACE\_EmW.icnarc\_04c\_unit\_discharge\_date\_estimated`;

drop table `yhcr-prd-phm-bia-core.CY\_MYSPACE\_EmW.icnarc\_04c\_unit\_discharge\_date\_estimated`;

-- 5) Estimate unit discharge status by checking for entries in the death-related columns

CREATE TABLE `yhcr-prd-phm-bia-core.CY\_MYSPACE\_EmW.icnarc\_04e\_unit\_discharge\_status\_estimated`

AS

SELECT

\*,

CASE

-- seemingly alive:

WHEN

-- patient was dicharged or is not in unit

(patient\_discharged = "Yes" OR patient\_in\_unit = "No") AND

-- no entries for any of the death columns

(brainstem\_death\_declared IS NULL

AND assent\_for\_solid\_organ IS NULL

AND date\_body\_removed IS NULL

AND demise IS NULL)

THEN "Alive"

-- seemingly dead:

WHEN

-- patient was discharged or is not in unit

(patient\_discharged = "Yes" OR patient\_in\_unit = "No") AND

-- any entries for any of the death columns

(brainstem\_death\_declared IS NOT NULL

OR assent\_for\_solid\_organ IS NOT NULL

OR date\_body\_removed IS NOT NULL

OR demise IS NOT NULL)

THEN "Dead"

-- seemingly in unit at export

WHEN

-- patient was discharged or is not in

(patient\_discharged = "No" OR patient\_in\_unit = "Yes") AND

-- any entries for any of the death columns

(brainstem\_death\_declared IS NULL

AND assent\_for\_solid\_organ IS NULL

AND date\_body\_removed IS NULL

AND demise IS NULL)

THEN "Not yet discharged at export"

END AS estimated\_unit\_discharge\_status

FROM `yhcr-prd-phm-bia-core.CY\_MYSPACE\_EmW.icnarc\_04d\_earliest\_and\_latest\_visit\_date`;

DROP TABLE `yhcr-prd-phm-bia-core.CY\_MYSPACE\_EmW.icnarc\_04d\_earliest\_and\_latest\_visit\_date`;