DRAFT SQL Code - Local Food Infrastructure and Population Health - Victor Russak

```
/*
WORK IN PROGRESS DRAFT
Analyst: Victor Russak
Email: vic@vicrussak.com
*/

/*
EXTRACT AND LOAD PHASE
        Utilize ELT methodology: Import raw data as TEXT with three tables; tables will be cleaned/transformed as new tables in the data warehouse.
        Goal: Create new tables and import data from the Food Environment Atlas (my alias: FEA) and NYS Community Health Indicators Report (CHIR).
*/

-- Create Food Environment Atlas Access Table.
CREATE TABLE raw_fea_access (

fips TEXT,
"state" TEXT,
county TEXT,
laccess_pop15 TEXT,
laccess_pop19 TEXT,
pch_laccess_pop_15_19 TEXT,
pct_laccess_pop15 TEXT,
pct_laccess_pop19 TEXT,
laccess_lowi15 TEXT,
laccess_lowi19 TEXT,
pch_laccess_lowi_15_19 TEXT,
pct_laccess_lowi15 TEXT,
pct_laccess_lowi19 TEXT,
laccess_hhnv15 TEXT,
laccess_hhnv19 TEXT,
pch_laccess_hhnv_15_19 TEXT,
pct_laccess_hhnv15 TEXT,
pct_laccess_hhnv19 TEXT,
laccess_snap15 TEXT,
laccess_snap19 TEXT,
pch_laccess_snap_15_19 TEXT,
pct_laccess_snap15 TEXT,
pct_laccess_snap19 TEXT,
laccess_child15 TEXT,
laccess_child19 TEXT,
pch_laccess_child_15_19 TEXT,
pct_laccess_child15 TEXT,
pct_laccess_child19 TEXT,
laccess_seniors15 TEXT,
laccess_seniors19 TEXT,
pch_laccess_seniors_15_19 TEXT,
pct_laccess_seniors15 TEXT,
```

```
pct_laccess_seniors19 TEXT,
laccess_white15 TEXT,
laccess_white19 TEXT,
pch_laccess_white_15_19 TEXT,
pct_laccess_white15 TEXT,
pct_laccess_white19 TEXT,
laccess_black15 TEXT,
laccess_black19 TEXT,
pch_laccess_black_15_19 TEXT,
pct_laccess_black15 TEXT,
pct_laccess_black19 TEXT,
laccess_hisp15 TEXT,
laccess_hisp19 TEXT,
pch_laccess_hisp_15_19 TEXT,
pct_laccess_hisp15 TEXT,
pct_laccess_hisp19 TEXT,
laccess_asian15 TEXT,
laccess_asian19 TEXT,
pch_laccess_asian_15_19 TEXT,
pct_laccess_asian15 TEXT,
pct_laccess_asian19 TEXT,
laccess_aian15 TEXT,
laccess_aian19 TEXT,
pch_laccess_aian_15_19 TEXT,
pct_laccess_aian15 TEXT,
pct_laccess_aian19 TEXT,
laccess_hopi15 TEXT,
laccess_hopi19 TEXT,
pch_laccess_hopi_15_19 TEXT,
pct_laccess_hopi15 TEXT,
pct_laccess_hopi19 TEXT,
laccess_multir15 TEXT,
laccess_multir19 TEXT,
pch_laccess_multir_15_19 TEXT,
pct_laccess_multir15 TEXT,
pct_laccess_multir19 TEXT

);

-- Create Community Health Indicators County Trends Table.
CREATE TABLE raw_chir_county_trends (

topic_area TEXT,
indicator_title TEXT,
geographic_area TEXT,
year_type TEXT,
"year" TEXT,
measurement TEXT,
rate_or_percent TEXT,
data_source TEXT,
data_notes TEXT

);

```

```sql
-- Create the Community Health Indicators Latest Data table as a helper table. This table includes the indicator_numbers which can be used as a shortened primary key (3-5 characters instead of 50-80 chararcers) later on.
CREATE TABLE raw_chir_latest_data (

county_name TEXT,
health_topic_number     TEXT,
health_topic TEXT,
indicator_number TEXT,
"indicator" TEXT,
event_count TEXT,
average_number_of_denominator TEXT,
measure_unit TEXT,
percent_or_rate TEXT,
lower_limit_of_95pct_ci TEXT,
upper_limit_of_95pct_ci TEXT,
data_comments TEXT,
quartile TEXT,
data_years TEXT,
data_source TEXT,
mapping_distribution TEXT,
"location" TEXT

);

-- Create nysc_fips table to import a master list of FIPS codes.
CREATE TABLE raw_us_fips (
        state_name TEXT,
        county_name TEXT,
        city_name TEXT,
        state_code TEXT,
        state_fips_code TEXT,
        county_code TEXT,
        stcnty_fips_code TEXT,
        city_code TEXT,
        stcntycity_fips_code TEXT

);

--Import raw .csv files for the four tables. The .csv file exports from the Economic Research Service and the NYS Dept. of Health are encoded in WIN1252.
```

```sql
COPY raw_fea_access(fips, "state", county, laccess_pop15, laccess_pop19,
pch_laccess_pop_15_19, pct_laccess_pop15, pct_laccess_pop19, laccess_lowi15,
laccess_lowi19, pch_laccess_lowi_15_19, pct_laccess_lowi15, pct_laccess_lowi19,
laccess_hhnv15, laccess_hhnv19, pch_laccess_hhnv_15_19, pct_laccess_hhnv15,
pct_laccess_hhnv19, laccess_snap15, laccess_snap19, pch_laccess_snap_15_19,
pct_laccess_snap15, pct_laccess_snap19, laccess_child15, laccess_child19,
pch_laccess_child_15_19, pct_laccess_child15, pct_laccess_child19, laccess_seniors15,
laccess_seniors19, pch_laccess_seniors_15_19, pct_laccess_seniors15,
pct_laccess_seniors19, laccess_white15, laccess_white19, pch_laccess_white_15_19,
pct_laccess_white15, pct_laccess_white19, laccess_black15, laccess_black19,
pch_laccess_black_15_19, pct_laccess_black15, pct_laccess_black19, laccess_hisp15,
laccess_hisp19, pch_laccess_hisp_15_19, pct_laccess_hisp15, pct_laccess_hisp19,
laccess_asian15, laccess_asian19, pch_laccess_asian_15_19, pct_laccess_asian15,
pct_laccess_asian19, laccess_aian15, laccess_aian19, pch_laccess_aian_15_19,
pct_laccess_aian15,     pct_laccess_aian19, laccess_hopi15, laccess_hopi19,
pch_laccess_hopi_15_19, pct_laccess_hopi15,     pct_laccess_hopi19, laccess_multir15,
laccess_multir19,       pch_laccess_multir_15_19, pct_laccess_multir15,
pct_laccess_multir19)
        FROM
                'C:\Users\Public\fea_access.csv'
                WITH
                (FORMAT CSV, HEADER TRUE, DELIMITER ',', ENCODING 'WIN1252');

COPY raw_chir_county_trends(topic_area, indicator_title, geographic_area, year_type,
"year", measurement, rate_or_percent, data_source, data_notes)
        FROM
                'C:\Users\Public\chir_county_trend.csv'
        WITH
                (FORMAT CSV, HEADER TRUE, DELIMITER ',', ENCODING 'WIN1252');

COPY raw_chir_latest_data(county_name, health_topic_number, health_topic,
indicator_number, "indicator", event_count, average_number_of_denominator,
measure_unit, percent_or_rate, lower_limit_of_95pct_ci, upper_limit_of_95pct_ci,
data_comments, quartile, data_years, data_source, mapping_distribution, "location")
        FROM

'C:\Users\Public\Community_Health_Indicator_Reports__CHIRS___Latest_Data_20250820.csv'
        WITH
                (FORMAT CSV, HEADER TRUE, DELIMITER ',', ENCODING 'WIN1252');

COPY raw_us_fips(state_name, county_name, city_name, state_code, state_fips_code,
county_code, stcnty_fips_code, city_code, stcntycity_fips_code)
        FROM
        'C:\Users\Public\State,_County_and_City_FIPS_Reference_Table_20260119.csv'
        WITH
                (FORMAT CSV, HEADER TRUE, DELIMITER ',', ENCODING 'WIN1252');

-- Runtime of script for the EXTRACT AND LOAD PHASE: 6.043 seconds

/* DATA EXPLORATION PHASE
        Goal 1: Check that the data was loaded correctly.
        Goal 2: Check Data Years in both tables to find common years between them.
*/

```

```sql
-- Check fips codes were imported correctly (no loss of leading zeros) and there are no entry errors.
SELECT fips, COUNT(*)
FROM raw_fea_access
WHERE LENGTH(fips) != 5 OR fips !~ '^[0-9]+$'
GROUP BY fips
ORDER BY fips; -- Found 318 instances of lost leading zeros. Will need to address in transformation.

-- Data years in Food Environment Atlas are in the column names.
SELECT *
FROM raw_fea_access
LIMIT 0 -- Result: raw_fea_access only contains 2015 and 2019 data.

--Checking if CHIR has 2015 and 2019 data.
SELECT "year"
FROM
        raw_chir_county_trends
WHERE
        "year" = '2015' OR
        "year" = '2019'
GROUP BY
        "year"
ORDER BY
        "year" ASC; -- Result: CHIR has both 2015 and 2019. This analysis will compare both years.

/* TRANSFORMATION PHASE
        Goal 1: Create a clean fea_access19 table (2019 data) and a clean fea_access15 (2015 data) table from the raw table.
        Goal 2: Create a clean chir_county_trends table joined with chir_latest_data so that the Indicator Number can be used as primary key.
        Goal 3: Add FIPS codes to the new CHIR table for use as a foreign key.
*/

--Create clean fea_access2019 table from the raw table. Only bringing in population counts and percentages, not percent change, nor demographic data.
CREATE TABLE fea_access19 AS
SELECT
    CAST(LPAD(fips, 5, '0') AS VARCHAR(5)) AS fips, -- LPAD fixes the loss of leading zeros due to passing the data through Excel. Constrains to 5 characters.
        CAST("state" AS VARCHAR(2)) AS "state",
        CAST(county AS VARCHAR(50)) AS county,
        CASE -- Checks for alpha values due to possible entry error and checks for null indicators '-9999' and '-8888' as specified in data documentation and changes these to null.
                WHEN laccess_pop19 ~ '^[0-9.]+$' THEN CAST(laccess_pop19 AS NUMERIC)
                ELSE NULL
                END AS laccess_pop19,
        CASE
                WHEN pct_laccess_pop19 ~ '^[0-9.]+$' THEN CAST(pct_laccess_pop19 AS NUMERIC)
                ELSE NULL
                END AS pct_laccess_pop19,
```

```sql
        CASE
                WHEN laccess_lowi19 ~ '^[0-9.]+$' THEN CAST(laccess_lowi19 AS NUMERIC)
                ELSE NULL
                END AS laccess_lowi19,
        CASE
                WHEN pct_laccess_lowi19 ~ '^[0-9.]+$' THEN CAST(pct_laccess_lowi19 AS
NUMERIC)
                ELSE NULL
                END AS pct_laccess_lowi19,
        CASE
                WHEN laccess_hhnv19 ~ '^[0-9.]+$' THEN CAST(laccess_hhnv19 AS NUMERIC)
                ELSE NULL
                END AS laccess_hhnv19,
        CASE
                WHEN pct_laccess_hhnv19 ~ '^[0-9.]+$' THEN CAST(pct_laccess_hhnv19 AS
NUMERIC)
                ELSE NULL
                END AS pct_laccess_hhnv19,
        CASE
                WHEN laccess_snap19 ~ '^[0-9.]+$' THEN CAST(laccess_snap19 AS NUMERIC)
                ELSE NULL
                END AS laccess_snap19,
        CASE
                WHEN pct_laccess_snap19 ~ '^[0-9.]+$' THEN CAST(pct_laccess_snap19 AS
NUMERIC)
                ELSE NULL
                END AS pct_laccess_snap19
FROM raw_fea_access;

-- Add primary key
ALTER TABLE fea_access19 ADD PRIMARY KEY (fips);

-- How many null indicators were removed?
SELECT *
FROM
        fea_access19
WHERE
        laccess_pop19 IS NULL; -- Result: Three rows had null indicators: fips 02063,
02066, 02261.

--Create clean fea_access2015 table from the raw table. Only bringing in population
counts and percentages, not percent change, nor demographic data.
CREATE TABLE fea_access15 AS
SELECT
    CAST(LPAD(fips, 5, '0') AS VARCHAR(5)) AS fips, -- LPAD fixes the loss of leading
zeros due to passing the data through Excel. Constrains to 5 characters.
        CAST("state" AS VARCHAR(2)) AS "state",
        CAST(county AS VARCHAR(50)) AS county,
        CASE -- Checks for alpha values due to possible entry error and checks for
null indicators '-9999' and '-8888' as specified in data documentation and changes
these to null.
                WHEN laccess_pop15 ~ '^[0-9.]+$' THEN CAST(laccess_pop15 AS NUMERIC)
                ELSE NULL
                END AS laccess_pop15,
```

```sql
        CASE
                WHEN pct_laccess_pop15 ~ '^[0-9.]+$' THEN CAST(pct_laccess_pop15 AS NUMERIC)
                ELSE NULL
                END AS pct_laccess_pop15,
        CASE
                WHEN laccess_lowi15 ~ '^[0-9.]+$' THEN CAST(laccess_lowi15 AS NUMERIC)
                ELSE NULL
                END AS laccess_lowi15,
        CASE
                WHEN pct_laccess_lowi15 ~ '^[0-9.]+$' THEN CAST(pct_laccess_lowi15 AS NUMERIC)
                ELSE NULL
                END AS pct_laccess_lowi15,
        CASE
                WHEN laccess_hhnv15 ~ '^[0-9.]+$' THEN CAST(laccess_hhnv15 AS NUMERIC)
                ELSE NULL
                END AS laccess_hhnv15,
        CASE
                WHEN pct_laccess_hhnv15 ~ '^[0-9.]+$' THEN CAST(pct_laccess_hhnv15 AS NUMERIC)
                ELSE NULL
                END AS pct_laccess_hhnv15,
        CASE
                WHEN laccess_snap15 ~ '^[0-9.]+$' THEN CAST(laccess_snap15 AS NUMERIC)
                ELSE NULL
                END AS laccess_snap15,
        CASE
                WHEN pct_laccess_snap15 ~ '^[0-9.]+$' THEN CAST(pct_laccess_snap15 AS NUMERIC)
                ELSE NULL
                END AS pct_laccess_snap15
FROM raw_fea_access;

-- Add primary key
ALTER TABLE fea_access15 ADD PRIMARY KEY (fips);

-- How many null indicators were removed?
SELECT *
FROM
        fea_access15
WHERE
        laccess_pop15 IS NULL; -- Result: Two rows had null indicators: fips 02063, 02066.

-- Create clean us_fips master table with state and county FIPS only.
CREATE TABLE us_fips AS
SELECT
    state_name,
        county_name,
        CAST(state_code AS VARCHAR(2)) AS state_code,
        CAST(LPAD( state_fips_code, 2, '0') AS VARCHAR(2)) AS  state_fips_code, -- LPAD fixes the loss of leading zeros due to passing the data through Excel. Constrains to 5 characters.
```

```sql
        county_code,
        CAST(LPAD(stcnty_fips_code, 5, '0') AS VARCHAR(5)) AS stcnty_fips_code -- LPAD fixes the loss of leading zeros due to passing the data through Excel. Constrains to 5 characters.
FROM
        raw_us_fips
WHERE
        stcnty_fips_code IS NOT NULL -- Puerto Rico does not have a county code and is not needed for this analysis.
GROUP BY -- removing duplicate stcnty_fips so it can act as primary key
        state_name,
        county_name,
        state_code,
        state_fips_code,
        county_code,
        stcnty_fips_code;

ALTER TABLE us_fips ADD PRIMARY KEY (stcnty_fips_code);

--Create master table for the Indicator Numbers in raw_chir_latest_data. Will use later for join and primary key creation.
CREATE TABLE indicator_master AS
SELECT
        indicator_number,
        "indicator"
FROM
        raw_chir_latest_data
GROUP BY
        indicator_number,
        "indicator";

ALTER TABLE indicator_master ADD PRIMARY KEY (indicator_number);

-- Create chir_trends_1519 table and join county FIPS codes and Indicator Numbers from helper tables.
CREATE TABLE chir_trends_1519 AS
SELECT DISTINCT ON (us.stcnty_fips_code, im.indicator_number, raw.year) -- The PK Columns
        raw.topic_area AS topic_area,
        UPPER(TRIM(REPLACE(REPLACE(raw.geographic_area, 'County', ''), '.', ''))) AS geographic_area, -- Removes the string "County" and '.' from the county names to match FIPS master table.
        raw.year,
        raw.measurement,
        CASE -- Checks for alpha values due to possible entry error and checks for null indicators '-9999' and '-8888' as specified in data documentation and changes these to null.
                WHEN raw.rate_or_percent ~ '^[0-9.]+$' THEN CAST(raw.rate_or_percent AS NUMERIC)
                ELSE NULL
                END AS rate_or_percent,
        'NY' AS state_code, -- adding state code so FIPS can be joined on state and county names
        us.stcnty_fips_code,
```

```
        im.indicator_number,
        im.indicator
FROM
        raw_chir_county_trends AS raw
LEFT JOIN us_fips AS us
        ON us.state_code = 'NY'
        AND UPPER(TRIM(REPLACE(REPLACE(raw.geographic_area, 'County', ''), '.', ''))) = TRIM(us.county_name)
LEFT JOIN indicator_master AS im
        ON TRIM(im.indicator) = TRIM(raw.indicator_title)
WHERE
        "year" IN ('2015', '2019') AND
        topic_area IN ('Cardiovascular Disease Indicators', 'Cirrhosis, Diabetes, and Kidney Indicators') -- Only including relevant chronic disease indicators. Excluding Obesity Topic Area since the general population obesity indicators exist as duplicates in the Cardiovascular topic area.
ORDER BY
        us.stcnty_fips_code,
    im.indicator_number,
    raw.year,
    raw.topic_area DESC -- Drops duplicates from the Cardiovascular topic area and keeps them in Cirrhosis since all three duplicates are related to kidney disease
        ;

ALTER TABLE chir_trends_1519 ADD PRIMARY KEY("year", stcnty_fips_code, indicator_number);

/*

Next Steps:
I've successfully created clean copies of chir_trends_1519 which contains 2015 and 2019 data
for the topic areas Cardiovascular and Kidney Disease. These include Obesity Indicators.

I also successfully created fea_access15 and fea_access19.

I now need to create a table that combines chir_trends_1519 with fea_access15 and fea_access19.
The fea tables are in Wide format and will need to be converted to long format before continuing.

*/
```