

Capstone Project 2

Dataset Summary:

Dataset Title: Global Sugar Consumption Dataset

Source: Kaggle

Time Span: 1960–2023

Entities/Fields (26 total):

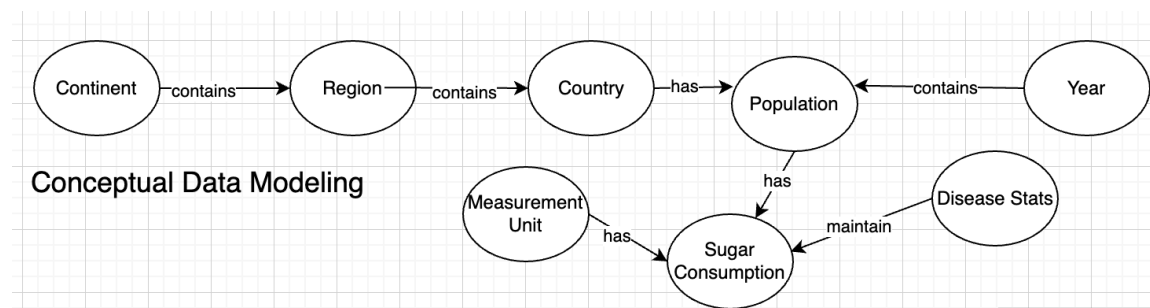
- Economic Indicators: GDP per capita, urbanization rate, retail sugar price
- Agricultural Data: Production, imports, exports, climate suitability
- Health Metrics: Obesity rate, diabetes prevalence, sugar intake
- Policy Data: Sugar taxes, subsidies, education efforts
- Country & Year-level granularity

Why a Relational Database?

- The dataset is structured, with consistent fields across country-year entries.

1.Relationships:

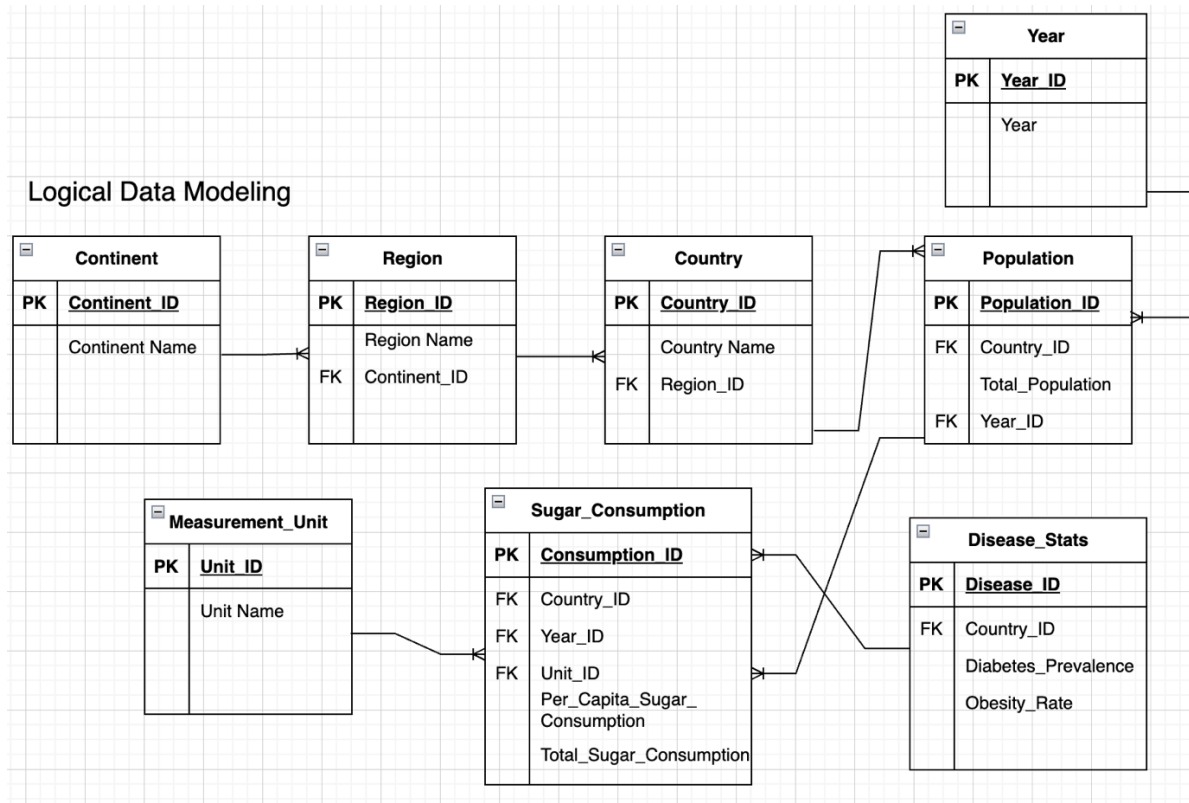
1. A continent has many regions.
2. A region has many countries.
3. A country has many population entries.
4. Disease Stats has many sugar consumption.
5. Year has many population records.
6. Population has many sugar consumption.
7. Measurement unit links to sugar consumption for normalization.



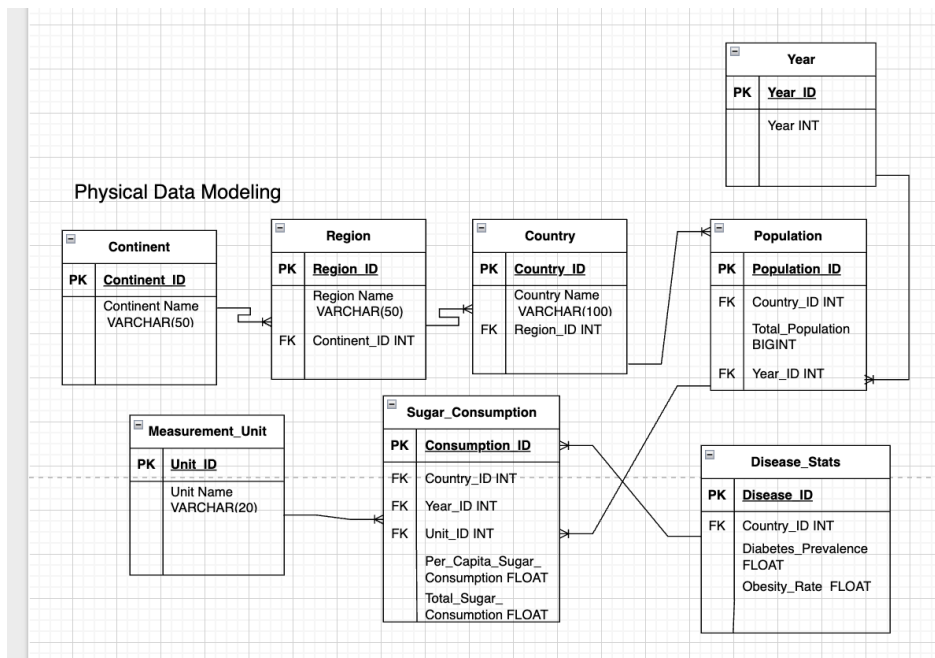
This structure accommodates global sugar consumption data efficiently by:

- Supporting country-year level granularity
- Enabling aggregated and trend-based analytics
- Keeping entities modular and normalized

2.



3.



CHAR(size)	A FIXED length string (can contain letters, numbers, and special characters). The <i>size</i> parameter specifies the column length in characters - can be from 0 to 255. Default is 1
VARCHAR(size)	A VARIABLE length string (can contain letters, numbers, and special characters). The <i>size</i> parameter specifies the maximum string length in characters - can be from 0 to 65535

INT(size)	A medium integer. Signed range is from -2147483648 to 2147483647. Unsigned range is from 0 to 4294967295. The <i>size</i> parameter specifies the maximum display width (which is 255)
INTEGER(size)	Equal to INT(size)
BIGINT(size)	A large integer. Signed range is from -9223372036854775808 to 9223372036854775807. Unsigned range is from 0 to 18446744073709551615. The <i>size</i> parameter specifies the maximum display width (which is 255)
FLOAT(size, d)	A floating point number. The total number of digits is specified in <i>size</i> . The number of digits after the decimal point is specified in the <i>d</i> parameter. This syntax is deprecated in MySQL 8.0.17, and it will be removed in future MySQL versions
FLOAT(p)	A floating point number. MySQL uses the <i>p</i> value to determine whether to use FLOAT or DOUBLE for the resulting data type. If <i>p</i> is from 0 to 24, the data type becomes FLOAT(). If <i>p</i> is from 25 to 53, the data type becomes DOUBLE()

4.

```

1  -- Drop database if it exists to avoid conflicts
2  • DROP DATABASE IF EXISTS capstone_project;
3
4  -- Create and use the database
5  • CREATE DATABASE capstone_project;
6  • USE capstone_project;
7
8  -- Continent table
9  • CREATE TABLE Continent (
10     Continent_ID INT PRIMARY KEY,
11     Continent_Name VARCHAR(50) NOT NULL UNIQUE
12 );
13
14 -- Region table
15 • CREATE TABLE Region (
16     Region_ID INT PRIMARY KEY,
17     Region_Name VARCHAR(100) NOT NULL UNIQUE,
18     Continent_ID INT,
19     FOREIGN KEY (Continent_ID) REFERENCES Continent(Continent_ID)
20 );
21
22 -- Country table
23 • CREATE TABLE Country (
24     Country_ID INT PRIMARY KEY,
25     Country_Name VARCHAR(100) NOT NULL,
26     Region_ID INT,
27     FOREIGN KEY (Region_ID) REFERENCES Region(Region_ID)
28 );
29
30
31 -- Developer table
32 • CREATE TABLE Developer (
33     Developer_ID INT PRIMARY KEY AUTO_INCREMENT,
34     Name VARCHAR(200) NOT NULL,
35     Country VARCHAR(100),
36     Founded_Year YEAR
37 );
38
39 -- Year_Dimension table (renamed from 'Year')
40 • CREATE TABLE Year_Dimension (
41     Year_ID INT PRIMARY KEY,
42     Year_Value INT NOT NULL UNIQUE
43 );
44
45 -- Measurement Unit table
46 • CREATE TABLE Measurement_Unit (
47     Unit_ID INT PRIMARY KEY,
48     Unit_Name VARCHAR(20) NOT NULL UNIQUE
49 );
50
51 -- Population table
52 • CREATE TABLE Population (
53     Population_ID INT PRIMARY KEY,
54     Country_ID INT,
55     Year_ID INT,
56     Total_Population BIGINT,
57     FOREIGN KEY (Country_ID) REFERENCES Country(Country_ID),
58     FOREIGN KEY (Year_ID) REFERENCES Year_Dimension(Year_ID)
59 );
60
61 -- Sugar Consumption table
62 • CREATE TABLE Sugar_Consumption (
63     Consumption_ID INT PRIMARY KEY,
64     Country_ID INT,
65     Year_ID INT,
66     Unit_ID INT,
67     Per_Capita_Sugar_Consumption FLOAT,
68     Total_Sugar_Consumption FLOAT,
69     FOREIGN KEY (Country_ID) REFERENCES Country(Country_ID),
70     FOREIGN KEY (Year_ID) REFERENCES Year_Dimension(Year_ID),
71     FOREIGN KEY (Unit_ID) REFERENCES Measurement_Unit(Unit_ID)
72 );
73
74 -- Disease Stats table
75 • CREATE TABLE Disease_Stats (
76     Disease_ID INT PRIMARY KEY,
77     Country_ID INT,
78     Year_ID INT,
79     Diabetes_Prevalence FLOAT,
80     Obesity_Rate FLOAT,
81     FOREIGN KEY (Country_ID) REFERENCES Country(Country_ID),
82     FOREIGN KEY (Year_ID) REFERENCES Year_Dimension(Year_ID)
83 );
84
85 • CREATE TABLE sugar_data (
86     Country VARCHAR(100),
87     Year INT,
88     Obesity_Rate FLOAT

```

5.

```

84
85 -- Total population per country in the latest year
86 • SELECT
87     c.Country_Name,
88     SUM(p.Total_Population) AS Total_Population
89 FROM Population p
90 JOIN Country c ON p.Country_ID = c.Country_ID
91 JOIN Year_Dimension y ON p.Year_ID = y.Year_ID
92 WHERE y.Year_Value = 2022
93 GROUP BY c.Country_Name
94 ORDER BY Total_Population DESC;
95

```

100% 1:150

Result Grid Filter Rows: Search Export:

Country_Name	Obesity_Increase
--------------	------------------

```

95
96 -- Average per capita sugar consumption by country
97 • SELECT
98     c.Country_Name,
99     AVG(s.Per_Capita_Sugar_Consumption) AS Avg_Consumption
100 FROM Sugar_Consumption s
101 JOIN Country c ON s.Country_ID = c.Country_ID
102 GROUP BY c.Country_Name
103 ORDER BY Avg_Consumption DESC;
104

```

```

-- Top 5 countries with highest total sugar consumption in 2021
SELECT
    c.Country_Name,
    r.Region_Name,
    SUM(s.Total_Sugar_Consumption) AS Total_Consumption
108 FROM Sugar_Consumption s
109 JOIN Country c ON s.Country_ID = c.Country_ID
110 JOIN Region r ON c.Region_ID = r.Region_ID
111 JOIN Year_Dimension y ON s.Year_ID = y.Year_ID
112 WHERE y.Year_Value = 2021
113 GROUP BY c.Country_Name, r.Region_Name
114 ORDER BY Total_Consumption DESC
115 LIMIT 5;
116

```

100% 26:114

Result Grid Filter Rows: Search Export:

Country_Name	Region_Name	Total_Consumpti...
--------------	-------------	--------------------

```

110
119 -- Total diabetes prevalence in Europe region
120 • SELECT
121     r.Region_Name,
122     COUNT(ds.Disease_ID) AS Records,
123     AVG(ds.Diabetes_Prevalence) AS Avg_Diabetes
124 FROM Disease_Stats ds
125 JOIN Country c ON ds.Country_ID = c.Country_ID
126 JOIN Region r ON c.Region_ID = r.Region_ID
127 WHERE r.Region_Name LIKE '%Europe%'
128 GROUP BY r.Region_Name;
129
130 -- Country with highest increase in obesity rate from 2015 to 2020

```

Result Grid Filter Rows: Search Export:

Region_Name	Records	Avg_Diabetes

```

130 -- Country with highest increase in obesity rate from 2015 to 2020
131 • SELECT
132     c.Country_Name,
133     (ds_2020.Obesity_Rate - ds_2015.Obesity_Rate) AS Obesity_Increase
134 FROM
135     Disease_Stats ds_2020
136 JOIN Year_Dimension y2020 ON ds_2020.Year_ID = y2020.Year_ID AND y2020.Year_Value = 2020
137 JOIN Disease_Stats ds_2015 ON ds_2020.Country_ID = ds_2015.Country_ID
138 JOIN Year_Dimension y2015 ON ds_2015.Year_ID = y2015.Year_ID AND y2015.Year_Value = 2015
139 JOIN Country c ON ds_2020.Country_ID = c.Country_ID
140 WHERE ds_2020.Obesity_Rate IS NOT NULL AND ds_2015.Obesity_Rate IS NOT NULL
141 ORDER BY Obesity_Increase DESC
142 LIMIT 1;

```

143 100% 9:142

Result Grid Filter Rows: Search Export:

Country_Name	Obesity_Increase