E6760 DATA WAREHOUSING AND INTEGRATION

Behavioral Risk Factor Surveillance System: Leveraging Data for Public Health Insights
Project Report

Group 8

Student 1: Yash Bhadreshwara
Student 2: Siddhant Chavan

Percentage of Effort Contributed by Student 1: 50%
Percentage of Effort Contributed by Student 2: 50%

Signature of Student 1: Yash Bhadreshwara
Signature of Student 2: Siddhant Chavan

- CONTENT

1. Summary

The Behavioral Risk Factor Surveillance System (BRFSS) is one of the most comprehensive public health surveillance programs in the world, capturing data on health-related risk behaviors. This massive dataset is instrumental for understanding public health trends and shaping interventions aimed at improving population health. However, the complexity and volume of BRFSS data present challenges for effective analysis. To address these challenges, this project was designed with the goal of creating a scalable and efficient framework that could process and analyze BRFSS data comprehensively. Along with that robust data warehouse—a centralized repository was designed to store and manage the vast and varied BRFSS dataset in a structured manner. Alongside this, an Online Analytical Processing (OLAP) system was integrated, enabling multi-dimensional analysis and real-time querying capabilities.

Extract, Transform, Load (ETL) process was developed to ensure that raw data from diverse sources could be standardized, cleaned, and loaded into the data warehouse seamlessly. This process was designed to handle large volumes of data with minimal errors, ensuring that the dataset maintained its integrity and usability. Furthermore, the project implemented a Type 3 Slowly Changing Dimension (SCD) mechanism, which is essential for tracking and managing historical changes in data dimensions.

By deploying these systems and processes, the project successfully developed a framework that transforms BRFSS data into a highly accessible and actionable resource. The outcomes include improved data integration, which eliminates redundancies and inconsistencies; enhanced analytical capabilities, and insightful visualizations that empower decision-makers to tackle pressing health issues.

2. Background

The BRFSS collects data through surveys to monitor health-related risk behaviors, chronic health conditions, and the use of preventive services. With over 400,000 interviews conducted annually, BRFSS is the largest continuously conducted health survey system. However, utilizing this wealth of data for actionable insights is challenging due to its size, complexity, and variability. This project addresses these challenges by creating a comprehensive data warehouse and OLAP system, transforming how health data is stored, managed, and analyzed.

The system enables efficient integration of diverse datasets, ensuring consistency and accessibility across various domains of public health. By implementing advanced ETL processes, it resolves common issues such as missing values and inconsistent formats, making the data more reliable for decision-making. Additionally, the use of Type 3 Slowly Changing Dimensions ensures that both current and historical information is retained, allowing for longitudinal analysis of health trends. Through this framework, public health officials can identify patterns in risk behaviors, compare state and national trends, and tailor interventions to address specific health challenges. Ultimately, the project bridges the gap between raw data and actionable insights, fostering data-driven strategies to improve health outcomes.

3. Problem Definition:

Despite its importance, the BRFSS dataset is notoriously difficult to work with due to several limitations. The primary issues include data fragmentation across multiple sources, inconsistency in formats, and a lack of historical tracking mechanisms. These challenges impede the ability of analysts to derive meaningful insights. For instance, monitoring trends such as obesity rates or activity levels over time requires robust data structures and analytical tools. Without such systems, identifying and addressing public health priorities becomes a daunting task. This project aims to resolve these issues by building an integrated data warehouse and analytical framework.

Along with the creation of a robust ETL process that extracts raw data on nutrition, physical activity, and obesity from the BRFSS dataset, cleans and transforms the data into an analyzable format, and loads it into a structured database for further analysis. The project will utilize tools and techniques like Python and MS Excel for data processing, SQL/Postgres for database management, and visualizations on tools such as Tableau for presenting insights. The goal is to create a structured, accessible, and user-friendly data pipeline that allows public health stakeholders to easily query and visualize trends in dietary habits, physical activity, and obesity rates across various demographics and geographic regions.

4. Objectives

Data Extraction and Ingestion: Extracting raw data from the BRFSS dataset, focusing on health-related information such as diet, physical activity, and obesity rates. Ensure consistency and reliable data import from national and state sources.

Data Cleaning and Transformation: Cleaning and standardizing the data by handling missing values and inconsistencies. Transforming the raw data into a structured format, including generating new calculated fields like BMI categories.

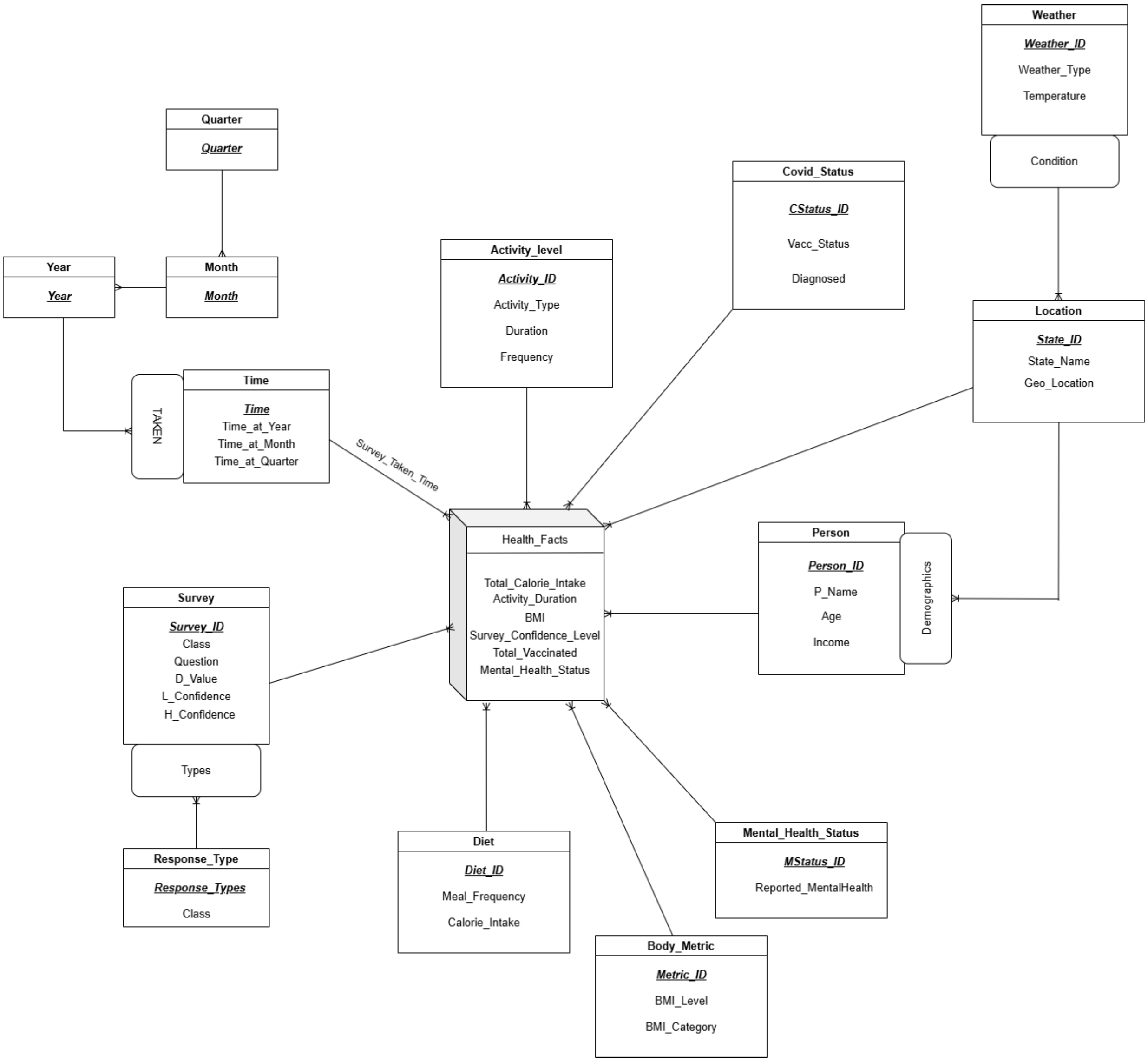
Trend and Comparative Analysis: Analyze the cleaned data to identify trends and patterns in obesity, nutrition, and physical activity across different demographics (e.g., age, gender, state) and time periods. Compare state-level trends to national averages.

Data Loading and Reporting: Loading the cleaned data into a structured database or data warehouse for further analysis. Develop reports and visualizations that summarize key findings and health insights.

Geospatial and Demographic Visualization: Creating interactive dashboards and maps that allow public health officials to visualize data trends by location, demographics, and health behaviors, supporting data-driven decisions.

5. Conceptual Model:

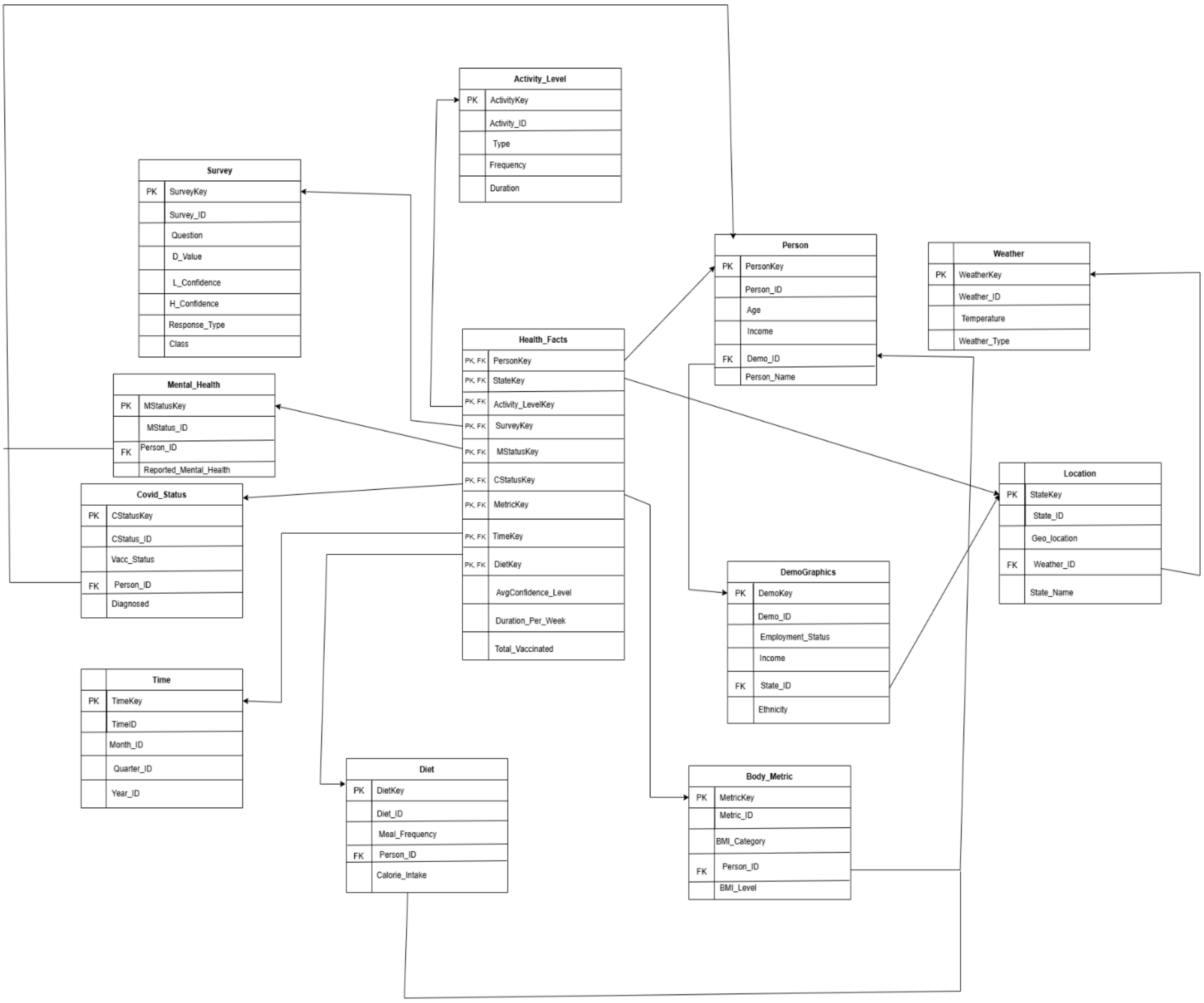
This conceptual model illustrates a data warehouse structure that integrates health-related data from the Behavioral Risk Factor Surveillance System. It shows how various dimensions and fact tables interact to provide a robust framework for analyzing public health metrics. The primary goal of this model is to enable multidimensional analysis, offering insights into public health behaviors, lifestyle factors, and environmental impacts on health.



At the core of the model lies the **Health_Facts** table, which acts as the fact table. It contains key quantitative metrics such as total calorie intake, activity duration, BMI, total vaccinated. These metrics are the central data points analyzed in the system. Surrounding the fact table are several dimension tables including Time, Survey, ResponseType, Activity_Level, Diet, Weather, Covid_Status, Location, Person, Mental_Health_Status, Body_Metric. Additionally, hierarchies are present within the model to enable multi-level analysis. In the Time dimension, a hierarchy exists as Quarter → Month → Year, allowing for drill-down or roll-up of data across different time granularities, such as quarterly, monthly, or yearly trends. Similarly, the Location dimension includes a hierarchy of State_ID → Geo_Location, facilitating geographic analysis at various levels.

6. Logical Model

The following logical data model is designed to facilitate health-related data analysis by organizing information into interrelated entities, following a star schema structure commonly used in data warehousing.

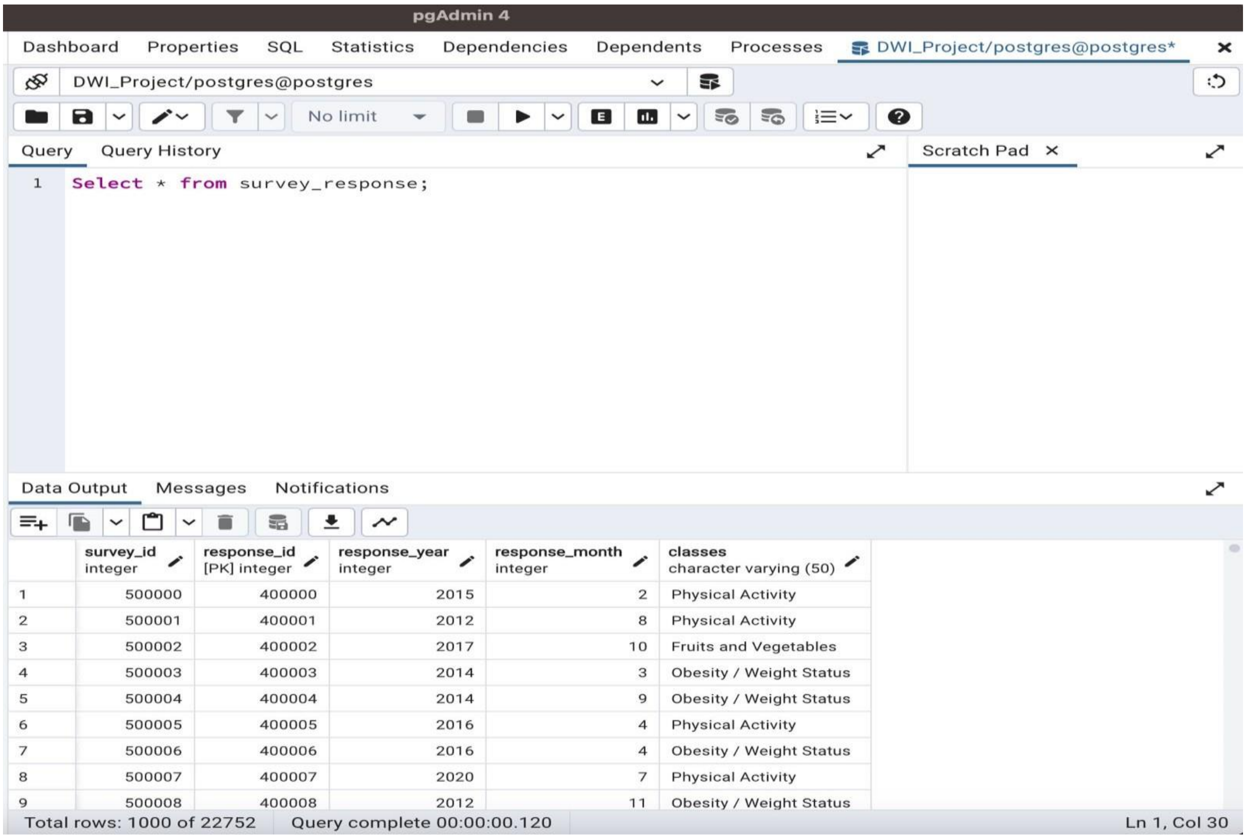


At the core are fact tables like Health_facts and this fact table is linked to various dimension tables, including Person, Activity_Level, Survey, Diet, Body_Metric, Weather, Location, Demographics, and Time. The dimensions provide context for the facts, capturing details about personal attributes (e.g., age, income), geographic locations, environmental factors like weather, lifestyle choices such as diet and activity levels, and time-based keys for temporal analysis. This comprehensive model enables multi-dimensional insights into health and wellness trends by integrating environmental, demographic, behavioral, and temporal data. Surrogate keys are present in almost every table, such as PersonKey, SurveyKey, and MStatusKey, to uniquely identify rows in the database. These keys are system-generated identifiers that simplify the management of relationships and ensure data integrity. This model enables a star-schema design suitable for data warehousing, with dimensions providing descriptive attributes and fact tables focusing on measurable events or metrics. It is well-suited for advanced analytics such as trend analysis, clustering, and predictive modeling in healthcare and demographic studies.

7. Loading Data into PostgresSQL Database

The process of loading data into a PostgreSQL database began with obtaining the dataset from data.gov, which provided information about the Behavioral Risk Factor Surveillance System (BRFSS). The dataset consisted of over 90,000 rows but included unstructured, irrelevant, and incomplete data that required extensive cleaning and preprocessing. To address these issues, Python libraries such as NumPy and Pandas were utilized for data manipulation, including filtering unnecessary columns, restructuring data formats, and handling inconsistencies. Additionally, advanced Microsoft Excel functions like conditional formatting, IF-ELSE logic, and pivot tables were employed to supplement the data cleaning process. Missing values were managed using data manipulation techniques, including logical imputation, removal of redundant records, and standardizing formats to ensure data quality.

Once the dataset was cleaned and structured, a PostgreSQL database was created to house the data. The database schema was carefully designed based on a logical data model, with tables structured to reflect key dimensions and facts. Primary and foreign keys were defined to maintain referential integrity, and attributes were appropriately assigned to ensure seamless data relationships. Following the schema creation, the cleaned data was exported into CSV files and imported into the corresponding tables within the database using PostgreSQL’s import utilities. During this step, table constraints and indexes were applied to enhance performance and data integrity. The final dataset was reduced to approximately 22,000 rows after cleaning and structuring, ensuring the removal of redundancies and retaining only meaningful information.

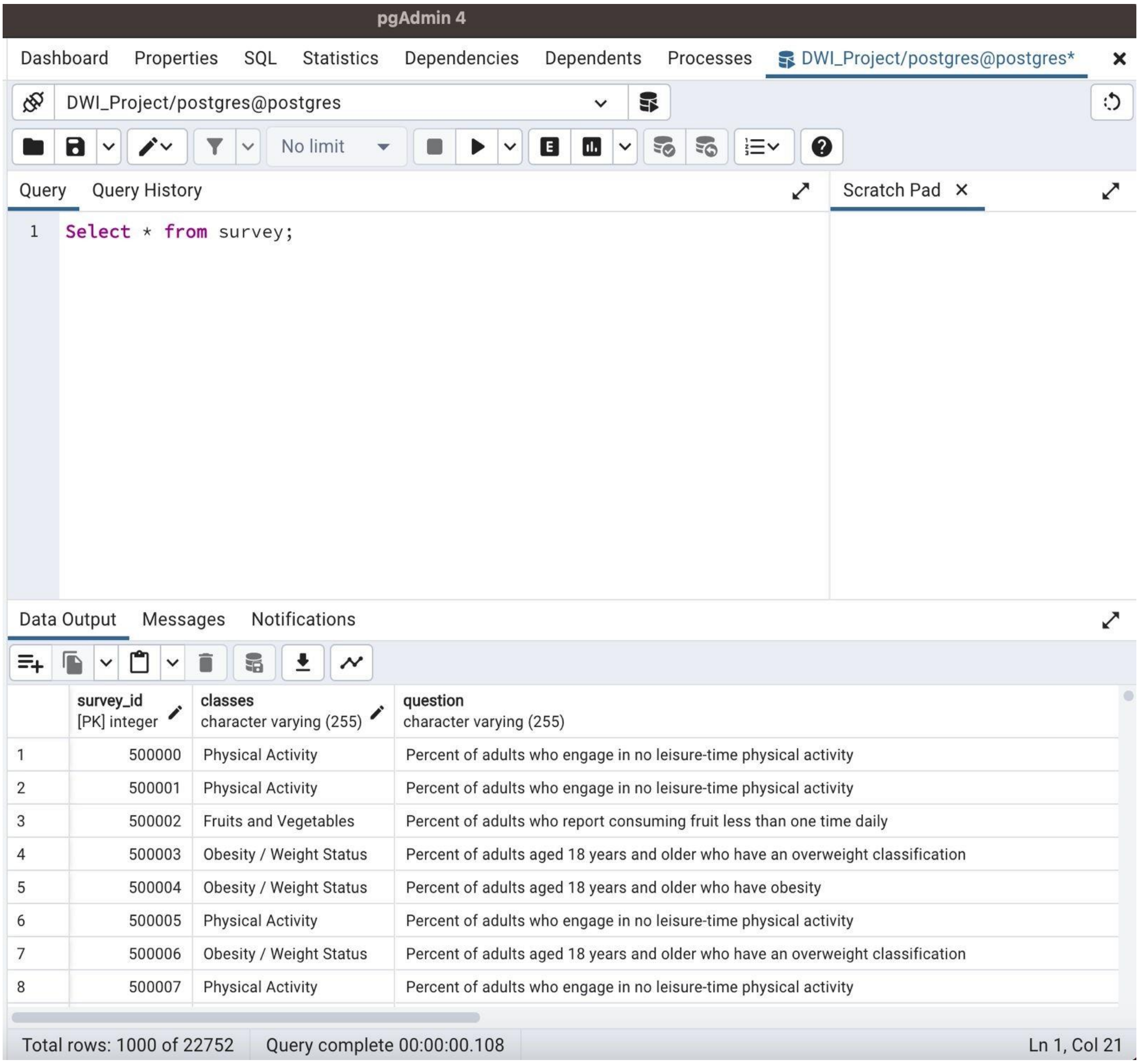


```
Select * from survey_response;
```

	survey_id integer	response_id [PK] integer	response_year integer	response_month integer	classes character varying (50)
1	500000	400000	2015	2	Physical Activity
2	500001	400001	2012	8	Physical Activity
3	500002	400002	2017	10	Fruits and Vegetables
4	500003	400003	2014	3	Obesity / Weight Status
5	500004	400004	2014	9	Obesity / Weight Status
6	500005	400005	2016	4	Physical Activity
7	500006	400006	2016	4	Obesity / Weight Status
8	500007	400007	2020	7	Physical Activity
9	500008	400008	2012	11	Obesity / Weight Status

Total rows: 1000 of 22752 Query complete 00:00:00.120 Ln 1, Col 30

After populating the tables, validation of the imported data was conducted using SQL queries to confirm data accuracy and consistency. Queries were written to extract specific rows, calculate aggregate statistics, and perform sample checks on columns to verify proper data alignment. This ensured that the imported data matched the intended schema and maintained its analytical value. The fully populated PostgreSQL database is now structured, clean, and ready for further analytical queries and reporting, providing a robust foundation for data-driven insights.



```
Select * from survey;
```

	survey_id [PK] integer	classes character varying (255)	question character varying (255)
1	500000	Physical Activity	Percent of adults who engage in no leisure-time physical activity
2	500001	Physical Activity	Percent of adults who engage in no leisure-time physical activity
3	500002	Fruits and Vegetables	Percent of adults who report consuming fruit less than one time daily
4	500003	Obesity / Weight Status	Percent of adults aged 18 years and older who have an overweight classification
5	500004	Obesity / Weight Status	Percent of adults aged 18 years and older who have obesity
6	500005	Physical Activity	Percent of adults who engage in no leisure-time physical activity
7	500006	Obesity / Weight Status	Percent of adults aged 18 years and older who have an overweight classification
8	500007	Physical Activity	Percent of adults who engage in no leisure-time physical activity

Total rows: 1000 of 22752 Query complete 00:00:00.108 Ln 1, Col 21

8. OLAP

OLAP is a powerful tool for transforming raw data into valuable insights, and its application in our BRFSS data warehouse project has enabled us to address the complex, multi-dimensional nature of public health data. OLAP systems allow for multidimensional analysis, which means data can be viewed and queried across multiple dimensions, such as time, geography, demographics, and other relevant categories. Unlike traditional databases that primarily support transactional data processing, OLAP is optimized for query-intensive tasks where users need to analyze trends, patterns, and insights from historical data.

For this project, we are leveraging OLAP to transform complex, multi-dimensional health data collected by the Behavioral Risk Factor Surveillance System (BRFSS) into actionable insights. The BRFSS collects data on a variety of health behaviors, chronic conditions, and preventive practices from many respondents annually. Given the vastness and complexity of this dataset, traditional data analysis methods often fall short in delivering quick, meaningful insights.

By implementing an OLAP system integrated with the data warehouse, we enable efficient querying and multidimensional analysis of the BRFSS data. It allowed us to gain insights on

- Health Behaviors (Physical activity, types, diet)
- Vaccinations (Covid Status)
- Mental Health Status
- Body Metrics

We performed some of the OLAP operations:

- ***Total Number of People Surveyed per State:***
  Surveyed_1 <- Rollup * (Survey, Locations, Person, State -> State_Name,count(P_ID))
  Result <- Surveyed_1

- ***Average Duration of Activities by Gender and Age Group***:
  Activity_1 <- Rollup * (Activity_Level, Person, Gender, Age -> Age_Group, avg(Duration))
  Result <- Activity_1

- ***Average Calorie Intake by State:***
  Calorie_1 <- Rollup * (Diet, Locations, Person, State -> State_Name, avg(Calorie_Intake))
  Result <- Calorie_1

- ***Calculate Average BMI per Age Group:***

```
BMI_1 <- Rollup * (BodyMetrics, Person, Age -> Age_Group, avg(BMI))
Result <- BMI_1
```

The OLAP system allows for efficient querying, multidimensional analysis, and the discovery of trends and correlations that were previously difficult to identify. Beyond our project, OLAP’s ability to analyze vast datasets from various perspectives makes it a versatile solution for a wide range of industries and use cases, empowering decision-makers with the data they need to drive change and improve outcomes.

9. ETL Implementation

The ETL implementation for this project was carried out using Talend to extract, transform, and load BRFSS data into a PostgreSQL data warehouse

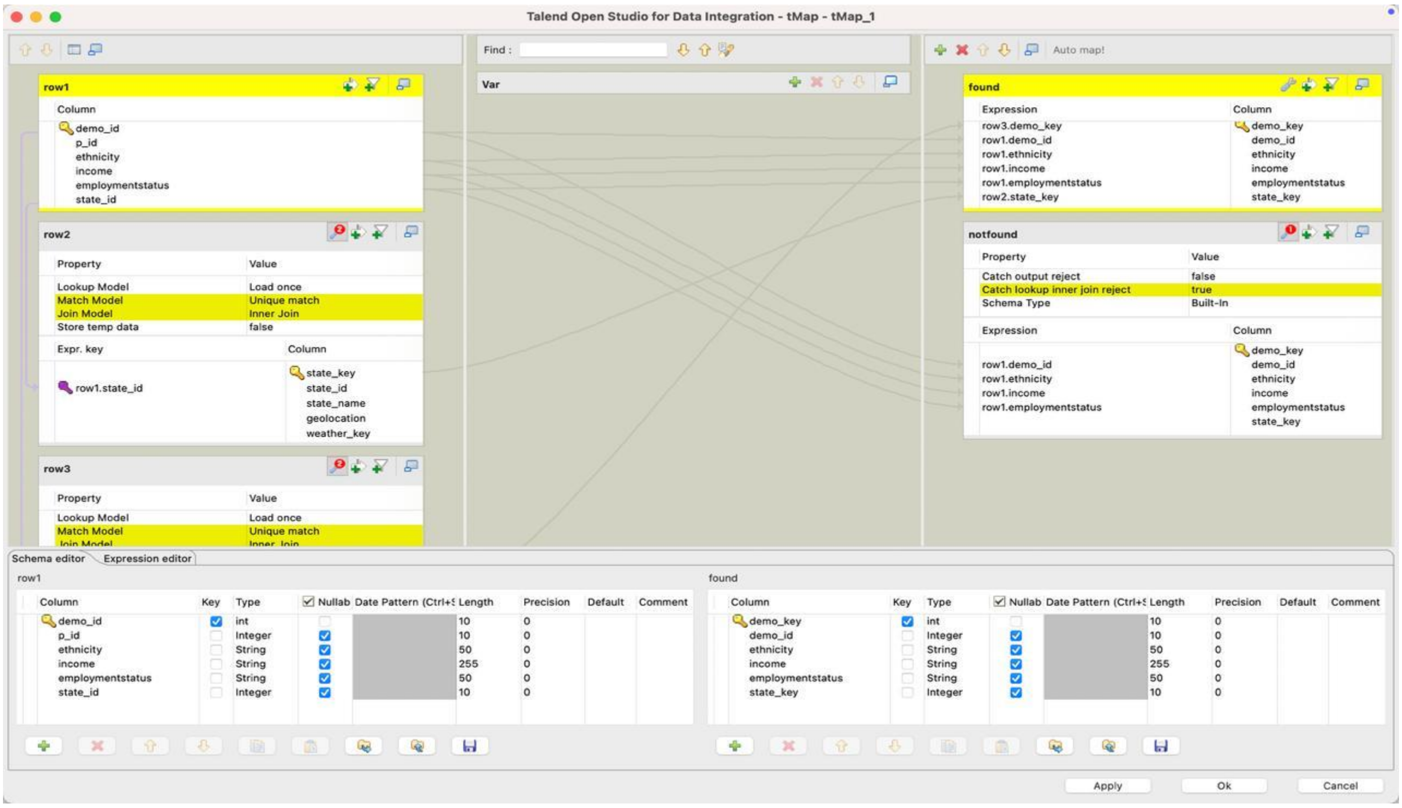
- Extraction

  The first step in the ETL process was extracting the source data. The raw data, primarily in CSV or Excel formats, contained information from the BRFSS, including survey responses, health metrics, demographics, and geographic data. Each dataset was read and imported into Talend, with source files linked to their respective tables in the data warehouse schema. For example:
- The **Person** dimension extracted demographic data like age, income, and gender.
- The **Diet** dimension sourced information on meal frequency and calorie intake.
- The **Activity Level** dimension captured activity type, duration, and frequency.

  Each source was linked to a corresponding staging table in Talend, where the raw data was temporarily stored for further processing.

- Transformation

  Transformation was carried out using Talend's **tMap** component to map and clean the data. This step ensured that the extracted data conformed to the schema of the target tables in PostgreSQL. Key steps in this phase included:
- **Key Mapping**: Data fields from source files were mapped to their respective attributes in the target scheme. For example, State_ID in the **Location** dimension was mapped to geographic attributes in the **Health_Facts** table.
- **Data Cleaning**: Missing values were handled by imputing defaults or using calculated averages, and inconsistent formats (e.g., dates) were standardized.
- **Lookup Operations**:
  - **Update Existing Records**: A lookup was performed against the target table to check if the data already existed. If a match was found, the record was updated with the latest information (e.g., changes in vaccination status in the **CovidStatus** dimension).
  - **Insert New Records**: If no match was found, the data was inserted into the target table as a new record. This ensured that the target table was always updated with fresh and accurate data.



row1

Column
demo_id
p_id
ethnicity
income
employmentstatus
state_id

row2

Property	Value
Lookup Model	Load once
Match Model	Unique match
Join Model	Inner Join
Store temp data	false

Expr. key	Column
row1.state_id	state_key
	state_id
	state_name
	geolocation
	weather_key

row3

Property	Value
Lookup Model	Load once
Match Model	Unique match
Join Model	Inner Join

found

Expression	Column
row3.demo_key	demo_key
row1.demo_id	demo_id
row1.ethnicity	ethnicity
row1.income	income
row1.employmentstatus	employmentstatus
row2.state_key	state_key

notfound

Property	Value
Catch output reject	false
Catch lookup inner join reject	true
Schema Type	Built-In

Expression	Column
	demo_key
row1.demo_id	demo_id
row1.ethnicity	ethnicity
row1.income	income
row1.employmentstatus	employmentstatus
	state_key

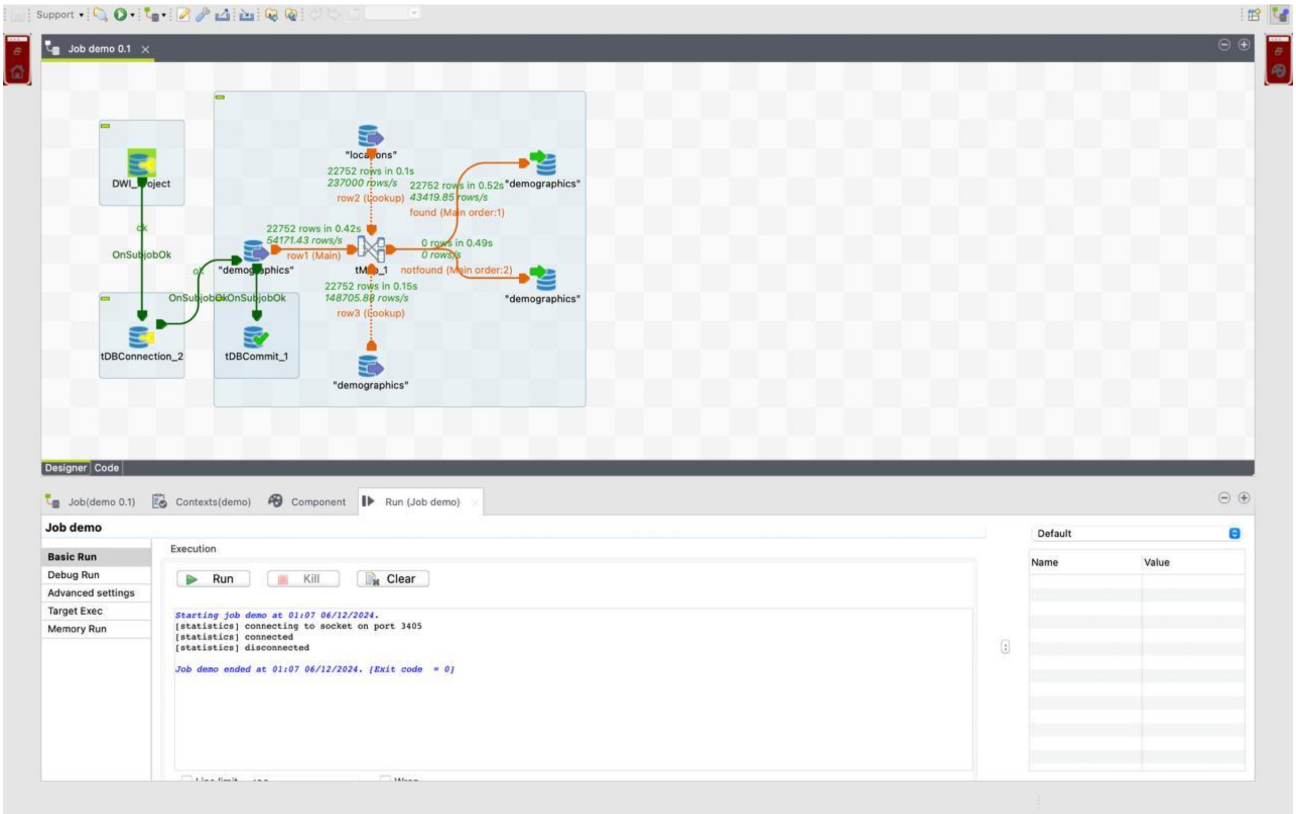
row1

Column	Key	Type	☑ Nullab	Date Pattern (Ctrl+S	Length	Precision	Default	Comment
demo_id	☑	int	□		10	0		
p_id	□	Integer	☑		10	0		
ethnicity	□	String	☑		50	0		
income	□	String	☑		255	0		
employmentstatus	□	String	☑		50	0		
state_id	□	Integer	☑		10	0		

found

Column	Key	Type	☑ Nullab	Date Pattern (Ctrl+S	Length	Precision	Default	Comment
demo_key	☑	int	□		10	0		
demo_id	□	Integer	☑		10	0		
ethnicity	□	String	☑		50	0		
income	□	String	☑		255	0		
employmentstatus	□	String	☑		50	0		
state_key	□	Integer	☑		10	0		

- <u>Loading</u>
  The transformed data was then loaded into the target tables in the PostgreSQL data warehouse. Each table was populated according to the relationships defined in the star schema. For instance:
- The **Health_Facts** table was populated with key metrics like total calorie intake, activity duration, BMI, and mental health status.
- Dimension tables such as **Person**, **Location**, and **Diet** were updated or expanded with the latest data.



```
Starting job demo at 01:07 06/12/2024.
[statistics] connecting to socket on port 3405
[statistics] connected
[statistics] disconnected

Job demo ended at 01:07 06/12/2024. [Exit code  = 0]
```

- Control Flow in the ETL Process

  Control flow is a crucial component of the ETL pipeline, managing the sequence and execution of individual ETL jobs. In Talend, control flow ensures that the data extraction, transformation, and loading processes occur in the correct order and under the appropriate conditions.

  Main task of Control flow:

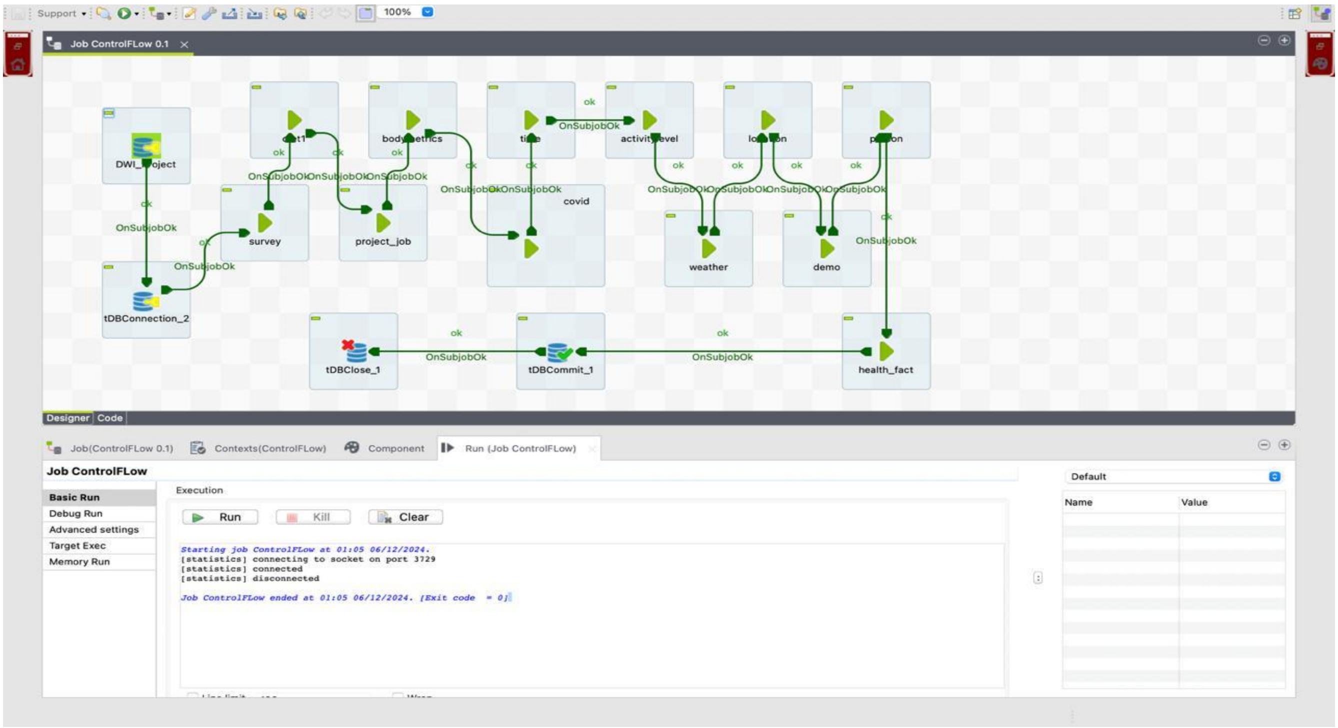
- ***Sequence Execution:*** Talend's control flow component organizes ETL jobs into a sequence. For example, dimension tables are populated first, followed by the Health_Facts table, ensuring foreign keys are resolved.

- ***Error Logging***: Errors encountered during any stage of the ETL process are logged for analysis, allowing for targeted debugging.

- ***Condition Handling***: Includes logic for handling conditional workflows, such as skipping a step if the required data isn't available or retrying a failed task.

- ***Truncation and Reloading***: The control flow automates the truncation of target tables before the data load. This ensures that outdated or duplicate data is cleared, and fresh data is loaded after each ETL run.

In this project, the control flow manages the execution of individual jobs, such as those for **Person**, **Diet**, **Covid Status**, and other dimensions, before consolidating them into the **Health_Facts** table. After all the ETL jobs are executed, the control flow ensures that the final populated tables are validated in PostgreSQL. This systematic approach ensures that the ETL pipeline is efficient, reliable, and capable of handling large datasets while maintaining data integrity.



```
Starting job ControlFLow at 01:05 06/12/2024.
[statistics] connecting to socket on port 3729
[statistics] connected
[statistics] disconnected

Job ControlFLow ended at 01:05 06/12/2024. [Exit code  = 0]
```

By using Talend’s ETL pipeline, the source data was successfully extracted, transformed, and loaded into the PostgreSQL database. The lookup operations ensured data consistency by updating existing records and inserting new ones where necessary. This approach allowed the data warehouse to maintain its accuracy and relevance. The fact and dimension tables were populated correctly, forming the foundation for advanced OLAP operations and multidimensional analysis.
Post-execution checks in PostgreSQL confirmed that the truncation and reloading process worked as intended, with tables reflecting the latest updates and clean data ready for analytical use. This efficient ETL process enables stakeholders to derive actionable insights into health trends, behaviors, and outcomes.

10. Slowly Changing Dimensions

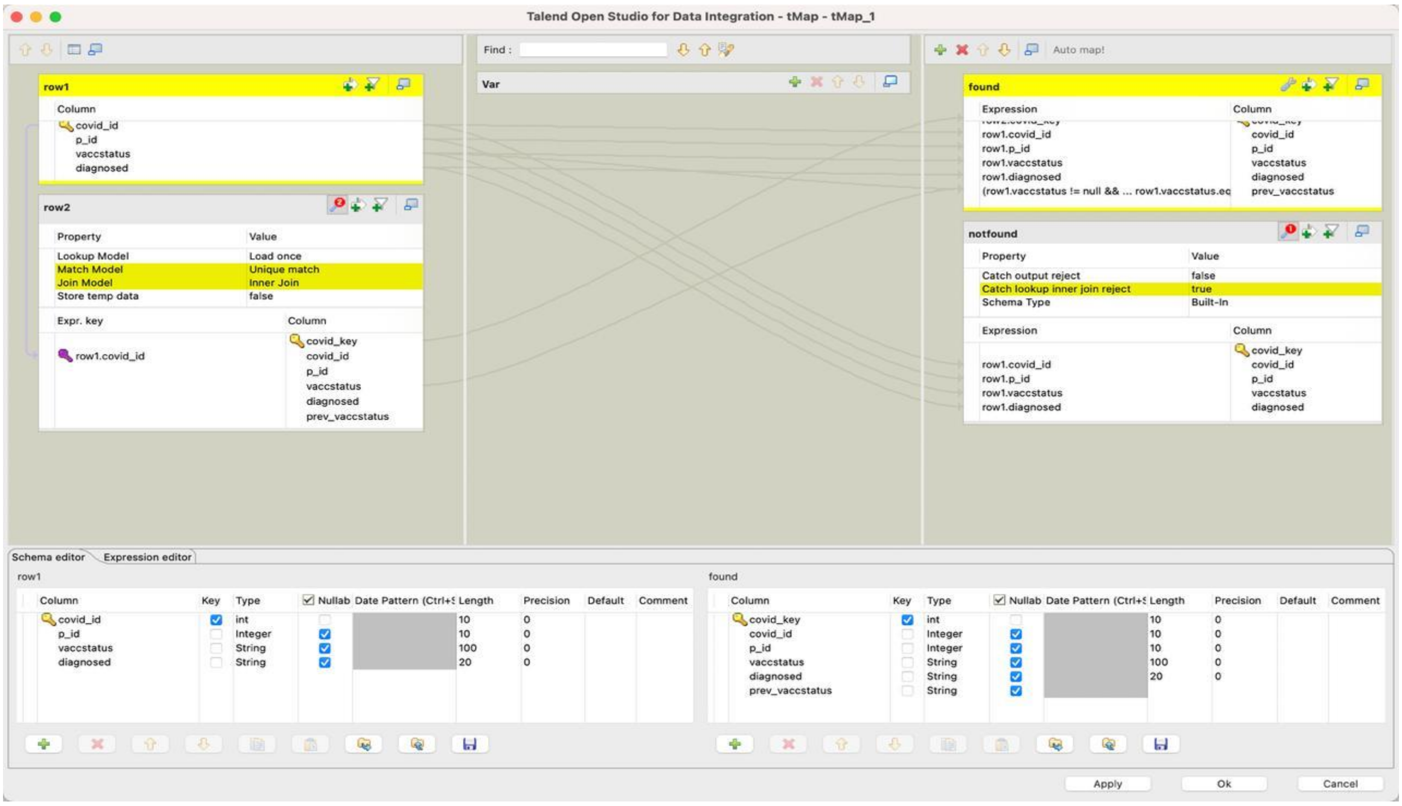
Slowly Changing Dimensions (SCDs) are used in data warehousing to manage and track changes in dimension attributes over time. These dimensions evolve slowly, as opposed to rapidly changing dimensions like transactional data. The challenge lies in maintaining historical data while updating the current state of the dimension.

In this project, we used the **Type 3 SCD method** to handle changes in specific dimensions, enabling the tracking of both historical and current values for selected attributes. Type 3 SCDs achieve this by adding new columns to the table for the historical value and the timestamp of the change. This approach is particularly effective when only a limited history of changes needs to be stored, making it both efficient and insightful.

- <u>Covid Status Dimension</u>
  One of the key dimensions where Type 3 SCD was implemented is the Covid Status Dimension. This table tracks an individual’s vaccination status, which can evolve over time. For example:
  An individual might initially be "Not Vaccinated" or "Partially Vaccinated."
  Once the individual completes their vaccination, the status changes to "Fully Vaccinated."

  To handle these updates while preserving historical data, the following approach was used:
  Columns Added for Historical Data:

  i. **Previous_Vacc_Status**: Stores the last recorded vaccination status.
  ii. **Current_Vacc_Status**: Stores the current vaccination status.



row1

Column
covid_id
p_id
vaccstatus
diagnosed

row2

Property	Value
Lookup Model	Load once
Match Model	Unique match
Join Model	Inner Join
Store temp data	false

Expr. key	Column
	covid_key
row1.covid_id	covid_id
	p_id
	vaccstatus
	diagnosed
	prev_vaccstatus

found

Expression	Column
row2.covid_key	covid_key
row1.covid_id	covid_id
row1.p_id	p_id
row1.vaccstatus	vaccstatus
row1.diagnosed	diagnosed
(row1.vaccstatus != null && ... row1.vaccstatus.eq	prev_vaccstatus

notfound

Property	Value
Catch output reject	false
Catch lookup inner join reject	true
Schema Type	Built-In

Expression	Column
	covid_key
row1.covid_id	covid_id
row1.p_id	p_id
row1.vaccstatus	vaccstatus
row1.diagnosed	diagnosed

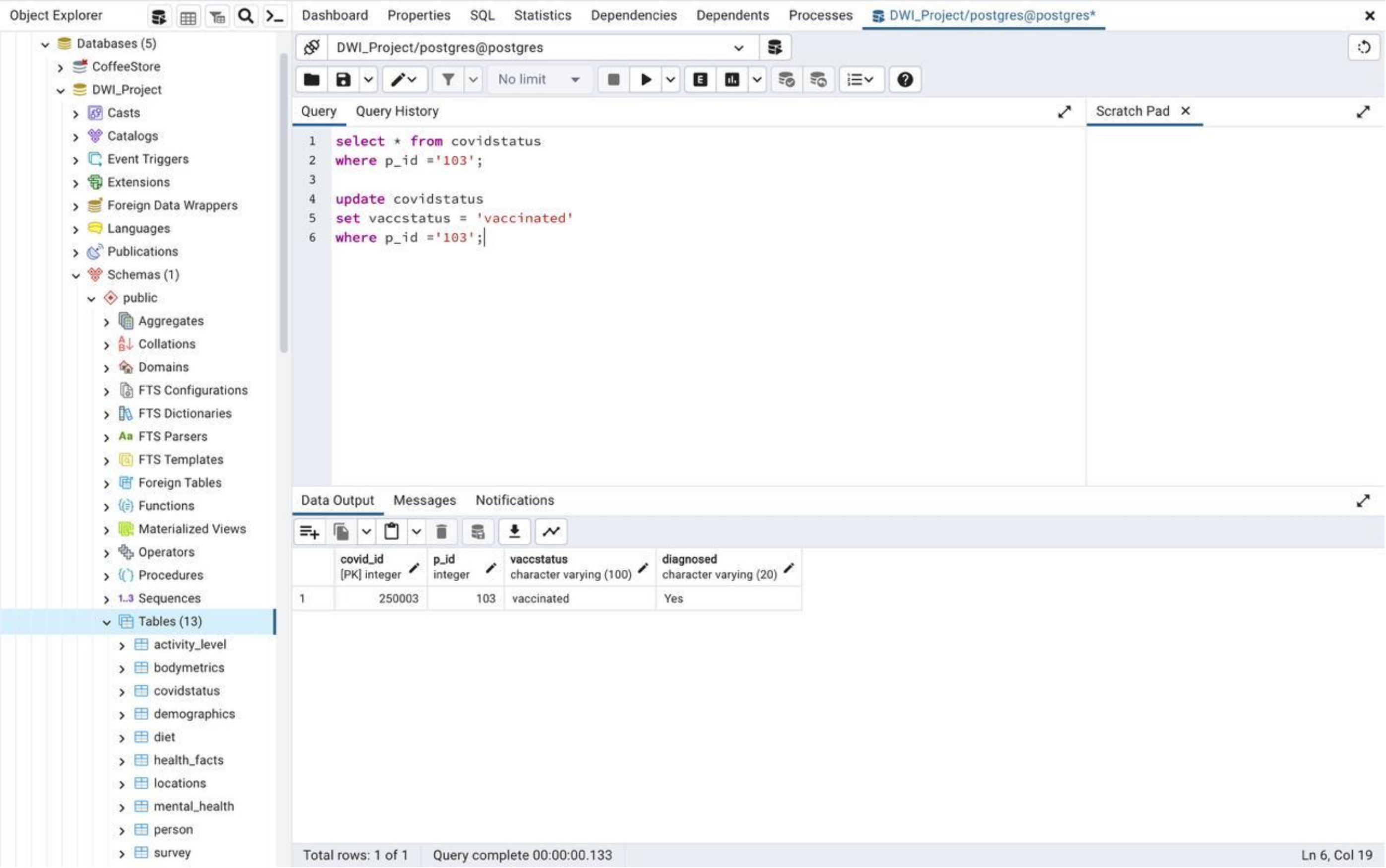
row1

Column	Key	Type	☑ Nullab	Date Pattern (Ctrl+S	Length	Precision	Default	Comment
covid_id	☑	int	□		10	0		
p_id	□	Integer	☑		10	0		
vaccstatus	□	String	☑		100	0		
diagnosed	□	String	☑		20	0		

found

Column	Key	Type	☑ Nullab	Date Pattern (Ctrl+S	Length	Precision	Default	Comment
covid_key	☑	int	□		10	0		
covid_id	□	Integer	☑		10	0		
p_id	□	Integer	☑		10	0		
vaccstatus	□	String	☑		100	0		
diagnosed	□	String	☑		20	0		
prev_vaccstatus	□	String	☑					

- <u>ETL Logic in Talend:</u>
  During the ETL process, the current vaccination status from the source data is compared with the existing data in the Covid Status Dimension.
  If the status has changed (e.g., from "Partially Vaccinated" to "Fully Vaccinated"), the Previous_Vacc_Status column is updated with the current value before the change.
  The Current_Vacc_Status column is then updated with the new status

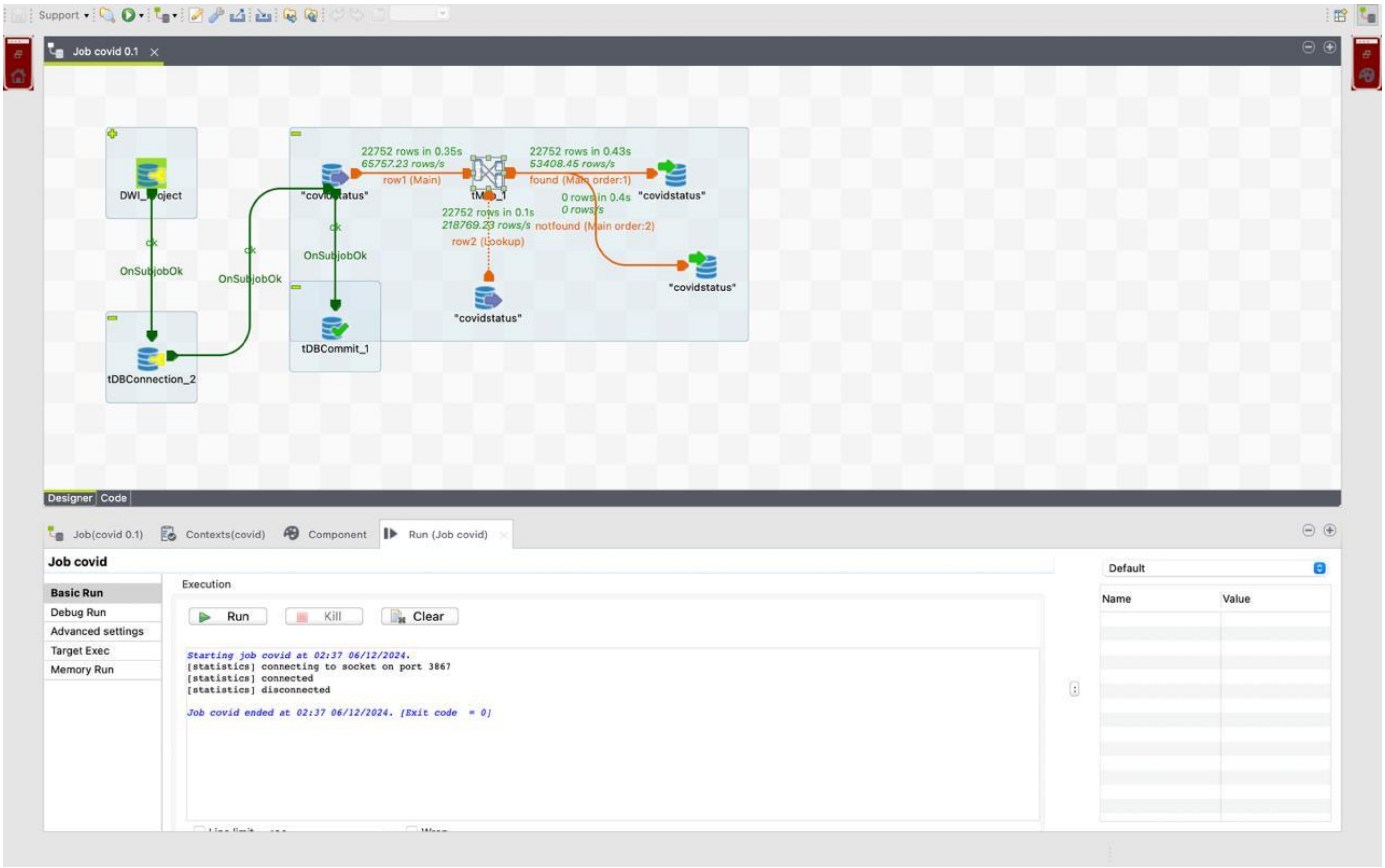


```
select * from covidstatus
where p_id ='103';

update covidstatus
set vaccstatus = 'vaccinated'
where p_id ='103';
```

	covid_id [PK] integer	p_id integer	vaccstatus character varying (100)	diagnosed character varying (20)
1	250003	103	vaccinated	Yes

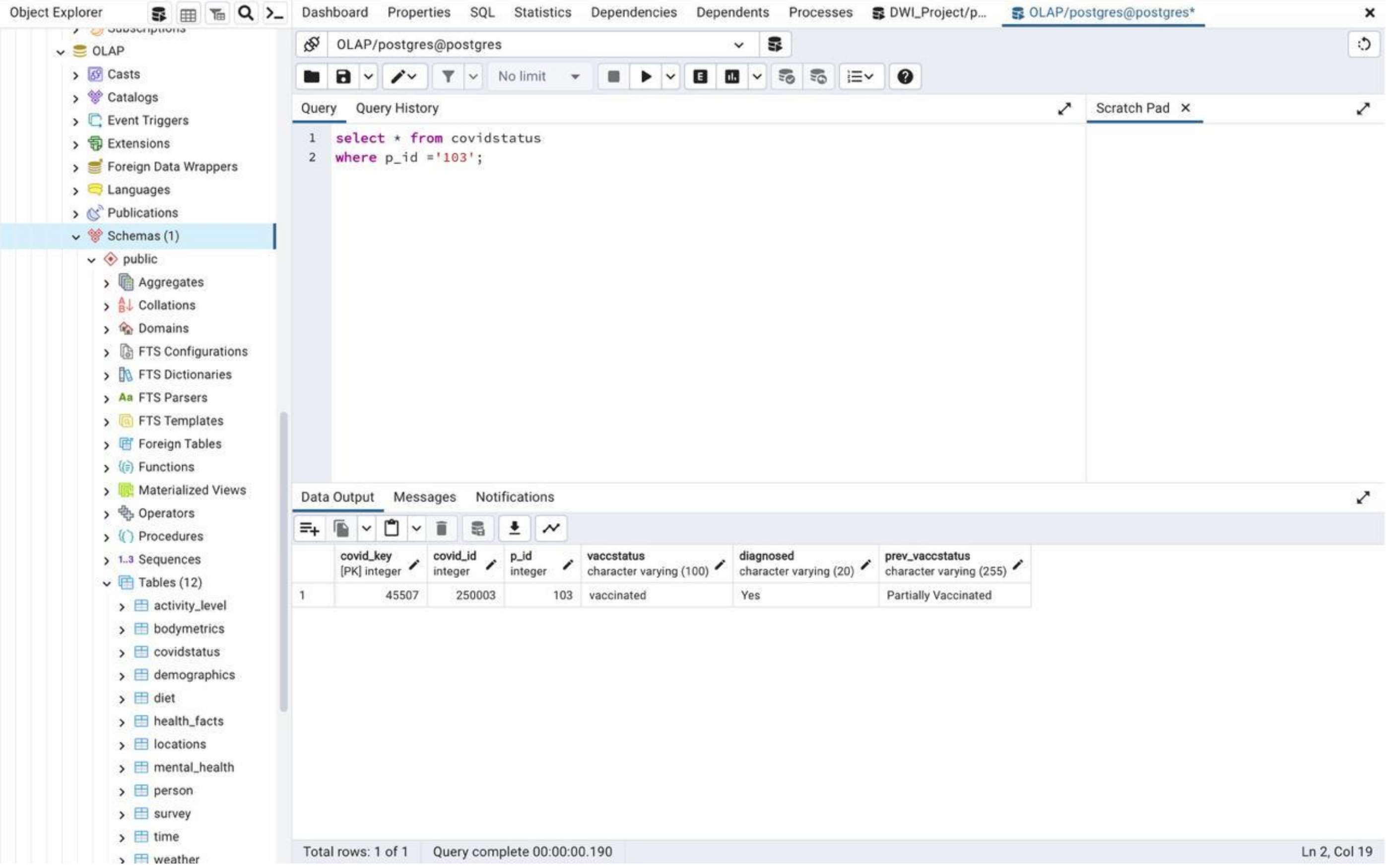
Total rows: 1 of 1 Query complete 00:00:00.133 Ln 6, Col 19



```
Starting job covid at 02:37 06/12/2024.
[statistics] connecting to socket on port 3867
[statistics] connected
[statistics] disconnected

Job covid ended at 02:37 06/12/2024. [Exit code  = 0]
```

After Updating the record, and running the job in Talend, the OLAP database gets updated, provides the previous as well as the current result of the status as given below -



```sql
select * from covidstatus
where p_id ='103';
```

	covid_key [PK] integer	covid_id integer	p_id integer	vaccstatus character varying (100)	diagnosed character varying (20)	prev_vaccstatus character varying (255)
1	45507	250003	103	vaccinated	Yes	Partially Vaccinated

Total rows: 1 of 1   Query complete 00:00:00.190   Ln 2, Col 19

This implementation ensures that both the latest vaccination status and the historical context are retained, supporting analyses that require time-based trends or status progression.

Benefits of Type 3 SCD Implementation

- Historical Context with Simplicity: Type 3 SCD provides a simple way to maintain historical values without creating multiple rows for each change, as in Type 2 SCD.

- Efficient Storage: By limiting historical tracking to one prior state, this method minimizes the storage overhead while still offering valuable insights.

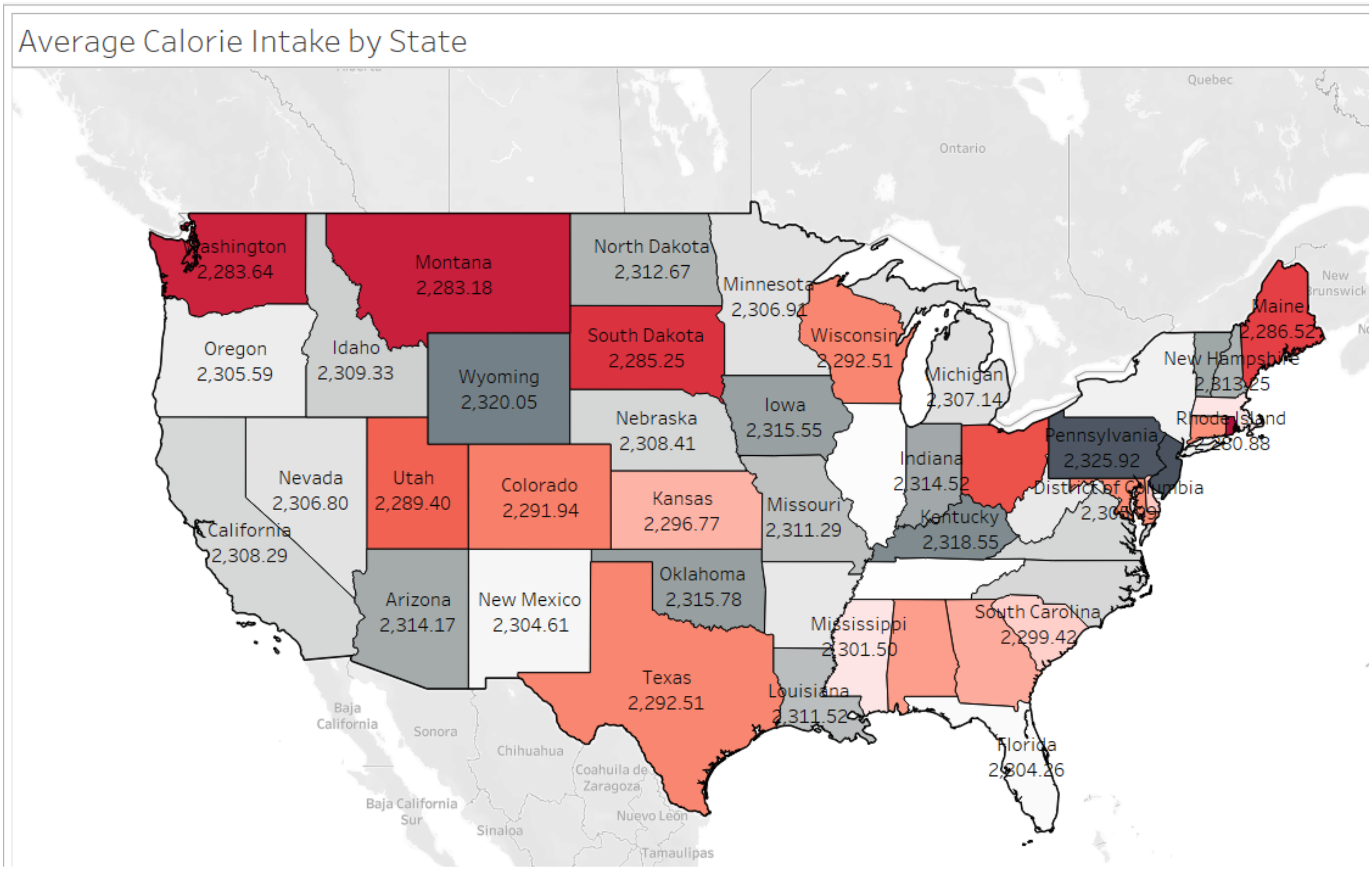
- Enhanced Analysis: Retaining both historical and current states allows for temporal analysis, such as examining vaccination progressions or age-related patterns over time.

The Type 3 SCD approach implemented in the Covid Status and Person dimensions ensures that the data warehouse remains both dynamic and insightful, supporting detailed analyses and trend monitoring in key public health metrics.

11. Insights & Recommendation

1. Average Calorie Intake by State

- **States with Lowest Average Calorie Intake**:
  - **Washington** and **Montana** had the lowest average calorie intake, each at **2,283 calories**. This could be attributed to lifestyle differences or dietary habits influenced by local culture and demographics.
- **States with Highest Average Calorie Intake**:
  - **Pennsylvania** recorded the highest average at **2,325 calories**, followed by **Wyoming** at **2,320 calories**. These higher values might reflect regional preferences for calorie-dense foods or differences in activity levels. Monitoring these trends can help tailor state-specific nutritional awareness programs.


Average Calorie Intake by State

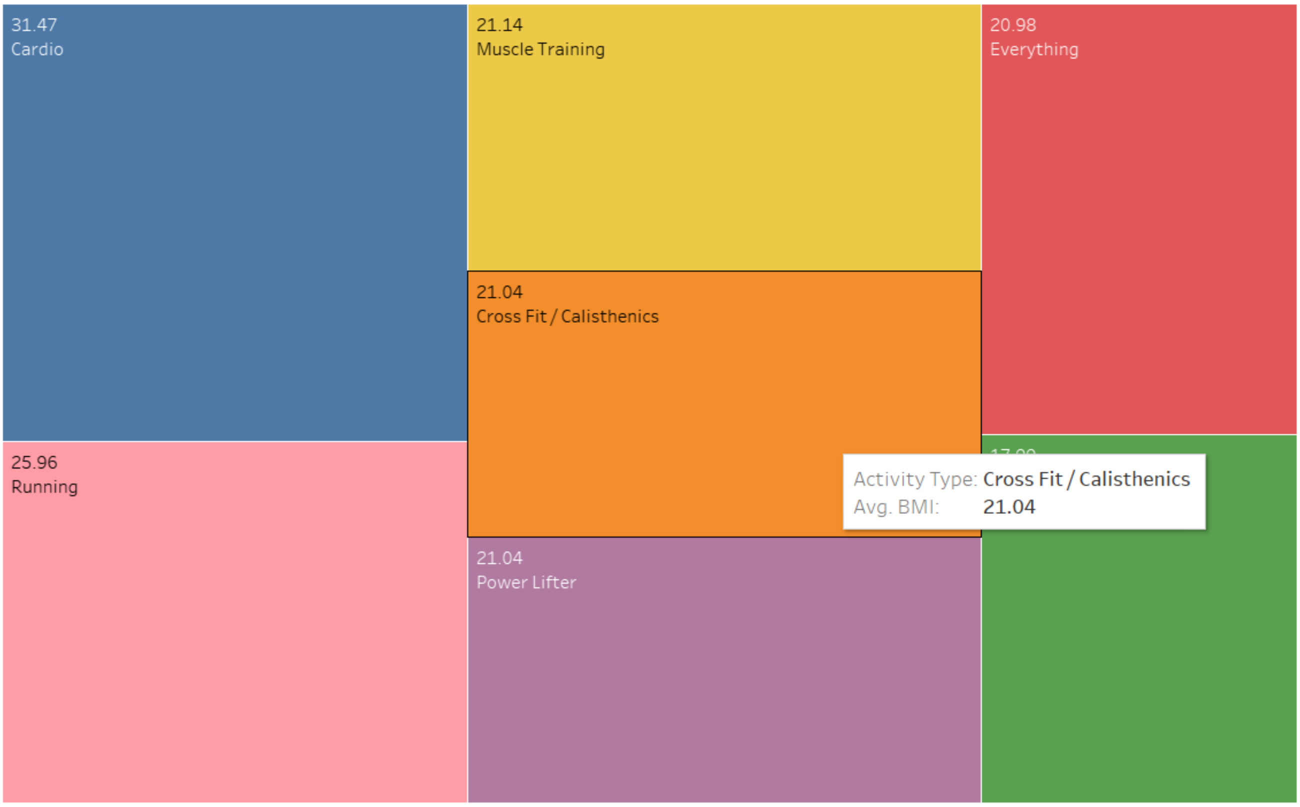
2. Average BMI by Age Group

- All age groups—**19–30 years**, **31–45 years**, and **45 years and above**—had BMIs within the **normal range (23–24 BMI)**. This consistency indicates that, on average, individuals are maintaining a healthy balance between calorie intake and activity levels across different life stages, likely reflecting greater awareness of health and fitness trends in recent years.

3. Relationship Between Activity Type and BMI

- Using a **treemap visualization**, we identified trends in BMI based on activity type:
  - **Cardio** and **Running** were associated with individuals having the highest BMI levels (**31 and 26**, respectively). This suggests that **overweight or obese individuals prefer these activities**, possibly because they are accessible, effective for weight loss, and do not require specialized equipment.
  - **High-Intensity Training (HIT)** participants had the **lowest BMI (18)**, which could be attributed to the rigorous nature of HIT workouts, often attracting already fit individuals aiming for advanced fitness goals.
  - These findings highlight the need for programs that encourage diverse activity types to ensure balanced fitness outcomes.

TreeMap for Activity Type & BMI



4. State-Level Focus: Massachusetts

- A drill-down analysis of Massachusetts revealed:
  - **HIT exercises** were the preferred activity type, likely due to the increasing popularity of fitness studios and group classes in urban areas.
  - **CrossFit** was the least preferred activity, which could be due to its demanding nature or limited accessibility in certain regions. Promoting varied options could encourage broader participation across different fitness levels.

5. Impact of Weather on Activity Preferences

- Comparing **sunny** and **rainy** weather conditions:
  - On sunny days, individuals gravitated towards **running** and **cardio**, likely driven by the appeal of outdoor activities and the psychological boost from sunlight exposure.
  - On rainy days, there was a noticeable increase in **muscle training**, potentially due to more people shifting to gyms or at-home workouts. This shift suggests an opportunity to promote adaptable workout routines that fit seasonal conditions.

6. Meal Frequency and Exercise Type

- The analysis of meal frequency based on exercise type highlighted the following:
  - Individuals involved in **cardio** and **running** had the **highest meal frequency (5–6 meals/day)**, reflecting the increased energy demands of these endurance-focused activities, especially for weight management.
  - Participants of **HIT**, **CrossFit**, and **muscle training** had the **lowest meal frequency (3 meals/day)**, which aligns with their goals of maintaining lean body mass while optimizing energy intake.
  - **Powerlifters** had a high meal frequency (5–6 meals/day), consistent with the energy needs required for strength-based activities. Understanding these patterns can help design personalized meal plans that enhance performance and health outcomes.

Recommendations

1. **Personalized Exercise Programs**:
   - Encourage individuals with higher BMI to adopt a mix of cardio and strength training, balancing weight loss with muscle building.
2. **Weather-Based Activity Promotions**:
   - Promote outdoor activities such as running and cardio during sunny weather while offering gym-based or home workout alternatives during rainy conditions.
3. **Meal Planning Assistance**:
   - Provide tailored meal plans for powerlifters and individuals engaged in cardio, ensuring their energy intake matches activity demands.
4. **State Specific Campaigns**:
   - In Massachusetts, focus on promoting diverse workout options to balance the preference for HIT exercises with other activities like CrossFit.

12. Conclusion

This project successfully addressed the complexities of analyzing the **Behavioral Risk Factor Surveillance System (BRFSS)** dataset by designing and implementing a robust data warehousing framework. Using tools like **Talend** and PostgreSQL **database**, the project transformed raw, fragmented data into a structured and actionable resource. This framework enabled multi-dimensional analysis and uncovered critical insights into public health trends, dietary habits, physical activity, and their relationship with health outcomes.

ETL Process with Talend:

The ETL pipeline was developed using Talend to extract raw BRFSS data, transform it into a structured format, and load it into the database.

Key transformations included handling missing values, standardizing inconsistent formats, and mapping data fields to the defined schema.

Slowly Changing Dimensions (Type 3 SCD) were implemented to retain historical data alongside current states, ensuring meaningful temporal analysis. For instance, vaccination statuses were tracked over time to capture both past and present data.

Conceptual and Logical Data Models:

The conceptual model established a star schema with a central fact table (Health_Facts) surrounded by dimension tables such as Demographics, Diet, Activity Level, and Covid Status. This facilitated a multi-dimensional view of public health metrics.

The logical model refined these relationships with surrogate keys for easier management and data integrity, ensuring that the database could efficiently handle queries and advanced analytics.

OLAP Database:

The OLAP database supported multi-dimensional querying, enabling stakeholders to analyze trends across dimensions such as time, geography, activity levels, and dietary habits.

Key OLAP operations included:

1) Roll-Up Analysis: Aggregating calorie intake by state or BMI by age group.
2) Drill-Down Analysis: Exploring specific trends within states or demographics.
3) Slice-and-Dice: Comparing activity preferences based on weather conditions.

Insights and Analysis

1. Dietary Patterns and BMI:

States like Washington and Montana had the lowest average calorie intake, while Pennsylvania recorded the highest, highlighting state-level dietary behavior differences. BMI values remained consistent across age groups, reflecting balanced calorie intake and activity levels.

2. Activity Preferences and Their Impact:
   Individuals with higher BMI gravitated toward accessible activities like cardio and running, while High-Intensity Training (HIT) was preferred by individuals with lower BMI.
   Weather conditions significantly influenced activity preferences, with sunny days favoring outdoor activities and rainy days increasing gym-based workouts.

This project demonstrated the power of integrating conceptual and logical models with OLAP systems and Talend ETL workflows to build a robust data warehousing solution. The structured framework provided actionable insights into public health behaviors, enabling decision-makers to craft targeted strategies for improving health outcomes. By bridging the gap between raw data and actionable intelligence, this project sets the foundation for future expansions and deeper explorations into public health trends.