

Lottery Sales in Texas

Business Analysis Template

Gabriela Cretu

June 2025

Contents

1	Business Description	2
1.1	Business Background	2
1.2	Problems Due to Poor Data Management	2
1.3	Benefits from Implementing a Data Warehouse	2
1.4	Datasets Description	3
1.4.1	Scratch-Based Lottery	3
1.4.2	Draw-Based Lottery	5
1.4.3	Proposed Use Cases	8
1.4.4	Dimension Hierarchies of scratch Based Game	8
1.4.5	Dimension Hierarchies for Draw-Based Lottery	10
1.5	Grain. DIM. FACT	11
1.5.1	Scratch Based Game	11
1.5.2	Draw Based Game	13
1.5.3	Grain Selection Process	14

1 Business Description

1.1 Business Background

Lottery games, or luck-based games as we might call them, are part of a large and profitable industry around the world. People keep playing because of the strong desire to win big—sometimes just by scratching a ticket or picking a few lucky numbers. These games continue to be popular in many places.

In the United States, the lottery market is very large, and Texas stands out as one of the top states in terms of ticket sales. That’s why this project focuses specifically on the Texas lottery. There are still many important questions to explore: Which games are the most or least profitable? What types of games do people in Texas prefer? Are instant-win scratch games more popular, or do players prefer waiting for draw results?

To find answers, it is important to understand how consumers behave, how well different games perform, and how retailers contribute to sales. This can be done by collecting and analyzing structured sales data from across Texas, which will help reveal useful insights and support better business decisions in this growing industry.

1.2 Problems Due to Poor Data Management

Poor data management can significantly hinder success in the lottery business. Without proper data, it’s difficult to know which games are doing well and which ones are not. If you don’t use tools that help you collect and analyze sales information, you won’t be able to understand what players want or how retailers are performing.

In a competitive market like the Texas lottery, not having the right data can lead to missed opportunities, poor planning, and less profit. To stay competitive and make smart decisions, it’s important to manage data properly and use it to create effective strategies.

1.3 Benefits from Implementing a Data Warehouse

Using a data warehouse can help solve the problems mentioned above. Implementing a data warehouse can answer important questions like:

- Which lottery games generate the most revenue?
- Which games have the widest range of sales across different retailers?
- Are there clear patterns in player preferences for instant-win versus draw-based games?

Further analysis of the data can also help to:

- Understand how sales vary by region or retailer type.
- Identify trends in customer behavior over time.
- Improve marketing strategies by targeting popular games.
- And many other useful insights.

By collecting and organizing data in one place, a data warehouse makes it easier to analyze sales, support better decision-making, and improve overall business performance.

1.4 Datasets Description

1.4.1 Scratch-Based Lottery

This dataset contains detailed transactional data for instant-win scratch tickets sold in Texas. Unlike the draw-based lottery, the assumption here is that a customer can purchase up to three tickets per transaction but can only win once per transaction because prizes are determined by ticket sellers. The prize value is represented by the `base_prize_scaled` attribute in the `DimPrizeRule` table.

Fact Table: `ScratchTicketSalesFact`

- `sale_id`: unique identifier for the sales record (PK)
- `date_surrogate_id`: foreign key to `DimDate` (FK)
- `game_surrogate_id`: foreign key to `DimScratchGame` (FK)
- `retailer_surrogate_id`: foreign key to `DimRetailer` (FK)
- `customer_surrogate_id`: foreign key to `DimCustomer` (FK)
- `employee_surrogate_id`: foreign key to `DimEmployee` (FK)
- `ticket_price`: price of a single scratch ticket
- `tickets_bought`: number of tickets purchased in the transaction (up to 3)
- `sale`: total transaction amount ($\text{ticket_price} * \text{tickets_bought}$)

Dimension Table: `DimPrizeRule`

- `prize_rule_surrogate_id` (PK): unique surrogate key for the prize rule
- `prize_rule_id`: original identifier for the prize rule
- `game_id`: foreign key linking to the game this prize rule applies to
- `winning_type_id`: identifier for the type of winning condition
- `base_prize_scaled`: the scaled base prize amount for this rule

Dimension Table: `DimScratchGame`

- `game_surrogate_id` (PK): unique surrogate key for the scratch game
- `scratch_game_number`: identifier number of the scratch game
- `ticket_price`: standard ticket price for the game
- `average_odds`: average odds of winning in this game
- `average_odds_probs`: average probability associated with the odds

Dimension Table: DimDate

- **date_surrogate_id** (PK): unique surrogate key for the date
- **date_id**: unique date identifier (e.g., YYYYMMDD)
- **fiscal_year**: fiscal year number for the date
- **fiscal_month**: fiscal month number for the date
- **fiscal_month_name_number**: textual name or number representation of fiscal month

Dimension Table: DimRetailer

- **retailer_surrogate_id** (PK): unique surrogate key for the retailer
- **retailer_license_number**: license number issued to the retailer
- **retailer_name**: official name of the retailer
- **retailer_number_location_name**: combined retailer number and location name for identification
- **retailer_address_1**: primary street address of the retailer location
- **retailer_city**: city where the retailer is located
- **retailer_state**: state where the retailer is located
- **retailer_zip_code**: postal zip code for the retailer location
- **retailer_county**: county where the retailer is located
- **owning_entity_number**: identifier of the owning entity
- **owning_entity_name**: name of the owning entity for the retailer
- **owning_entity_chain_head_name**: head of the owning entity chain (if applicable)

Dimension Table: DimCustomer

- **customer_surrogate_id** (PK): surrogate key, unique identifier for the customer
- **customer_id**: original customer ID from source system
- **customer_name**: full name of the customer
- **customer_date_of_birth**: birth date of the customer
- **customer_gender**: gender of the customer
- **customer_city**: city of residence

-
- **customer_state**: state of residence
 - **customer_zip**: postal code / ZIP code

Dimension Table: DimEmployee

- **employee_surrogate_id** (PK): unique surrogate key for the employee
- **employee_id**: original employee ID
- **employee_name**: full name of the employee
- **employee_email**: email address of the employee
- **employee_phone_number**: contact phone number of the employee

1.4.2 Draw-Based Lottery

The dataset captures detailed information about draw-based lottery games. Unlike the previous scratch game, our assumption this time is that a player could win multiple times, as unlike the other one where his win is in the hands of the ticket seller, this time if he is sure of the right numbers (premonition), he could fill all three tickets and win big. **Fact Table: DrawTicketSalesFact**

- **sale_id**: unique identifier for the sales record (PK)
- **date_surrogate_id**: foreign key to DimDate (FK)
- **game_surrogate_id**: foreign key to DimDrawGame (FK)
- **retailer_surrogate_id**: foreign key to DimRetailer (FK)
- **customer_surrogate_id**: foreign key to DimCustomer (FK)
- **employee_surrogate_id**: foreign key to DimEmployee (FK)
- **chosen_numbers_combination_surrogate_id**: foreign key to DimArrayCombinations (FK)
- **chosen_numbers_combination_surrogate_ids**: IDs of individual number surrogates
- **winning_numbers_combination_surrogate_id**: foreign key to DimCombinations (FK)
- **winning_combination_id**: original flat identifier for winning combination

-
- **ticket_price**: price of each ticket
 - **ticket_sales**: total sale amount for the transaction
 - **tickets_bought**: number of tickets purchased in transaction
 - **transaction_date**: date and time when the transaction occurred
 - **chosen_ids**: raw list of selected number IDs in the ticket
 - **winning_ticket_count**: number of winning tickets in the transaction
 - **is_winning_ticket**: boolean flag indicating whether the ticket is a winner
 - **winning_prize**: prize tier/category won by the ticket
 - **amount_won**: total monetary amount won in the transaction

Dimension Table: DimDrawGame

- **game_surrogate_id** (PK): unique surrogate key for the draw game
- **game_id**: original game identifier
- **game_category**: category or type of game (e.g., “million”)
- **ticket_price**: default or standard price per ticket

Dimension Table: DimCombinations

- **winning_numbers_combination_surrogate_id** (PK): unique surrogate key for the combination
- **winning_combination_id**: original combination identifier
- **winning_numbers_combination**: the actual numbers combination that won
- **game_category**: category of the game associated with the combination

Dimension Table: DimArrayCombinations

- **chosen_numbers_combination_surrogate_id** (PK): unique surrogate key for the array of number combinations
- **chosen_numbers_combination_surrogate_ids**: list of surrogate IDs for each individual number combination
- **chosen_numbers_combination**: the chosen number combinations as a set/array
- **game_category**: category of the game for the chosen numbers

Dimension Table: DimDate

-
- **date_surrogate_id** (PK): unique surrogate key for the date
 - **date_id**: unique date identifier (e.g., YYYYMMDD)
 - **fiscal_year**: fiscal year number for the date
 - **fiscal_month**: fiscal month number for the date
 - **fiscal_month_name_number**: textual name or number representation of fiscal month

Dimension Table: DimRetailer

- **retailer_surrogate_id** (PK): unique surrogate key for the retailer
- **retailer_license_number**: license number issued to the retailer
- **retailer_name**: official name of the retailer
- **retailer_number_location_name**: combined retailer number and location name for identification
- **retailer_address_1**: primary street address of the retailer location
- **retailer_city**: city where the retailer is located
- **retailer_state**: state where the retailer is located
- **retailer_zip_code**: postal zip code for the retailer location
- **retailer_county**: county where the retailer is located
- **owning_entity_number**: identifier of the owning entity
- **owning_entity_name**: name of the owning entity for the retailer
- **owning_entity_chain_head_name**: head of the owning entity chain (if applicable)

Dimension Table: DimCustomer

- **customer_surrogate_id** (PK): surrogate key, unique identifier for the customer
- **customer_id**: original customer ID from source system
- **customer_name**: full name of the customer
- **customer_date_of_birth**: birth date of the customer
- **customer_gender**: gender of the customer
- **customer_city**: city of residence
- **customer_state**: state of residence

-
- **customer_zip**: postal code / ZIP code

Dimension Table: DimEmployee

- **employee_surrogate_id** (PK): unique surrogate key for the employee
- **employee_id**: original employee ID
- **employee_name**: full name of the employee
- **employee_email**: email address of the employee
- **employee_phone_number**: contact phone number of the employee

1.4.3 Proposed Use Cases

- Detect top-performing game types and retailers across counties.
- Build dashboards to compare scratch vs. draw-based sales performance.
- Track monthly growth, returns, and adjustments to enhance planning.
- Identify underperforming games and regions needing marketing support.

Hierarchies

1.4.4 Dimension Hierarchies of scratch Based Game

Hierarchy: Date Dimension This hierarchy defines the fiscal time structure for temporal analysis and aggregation in the **DimDate** table.

- **Level 1: Fiscal Year** — **fiscal_year**
- **Level 2: Fiscal Month (Number)** — **fiscal_month**
- **Level 3: Fiscal Month (Name or Number)** — **fiscal_month_name_number**
- **Level 4: Date ID** — **date_id**

Hierarchy: Retailer Dimension This hierarchy organizes retailers geographically and by corporate ownership using attributes in the **DimRetailer** table.

- **Level 1: State** — **retailer_state**
- **Level 2: County** — **retailer_county**
- **Level 3: City** — **retailer_city**
- **Level 4: Zip Code** — **retailer_zip_code**
- **Level 5: Retailer Address** — **retailer_address_1**

-
- **Level 6: Retailer Surrogate ID** — `retailer_surrogate_id`
 - **Level 1: Owning Entity Chain Head Name** — `owning_entity_chain_head_name`
 - **Level 2: Owning Entity Name** — `owning_entity_name`
 - **Level 3: Owning Entity Number** — `owning_entity_number`
 - **Level 4: Retailer (Surrogate ID, License Number, Name)** — `retailer_surrogate_id`, `retailer_license_number`, `retailer_name`

Hierarchy: Customer Dimension This hierarchy structures customers demographically for behavioral analysis, using fields in the `DimCustomer` table.

Customer Dimension Hierarchy Levels

- **Level 1: State** — `customer_state`
- **Level 2: City** — `customer_city`
- **Level 3: ZIP Code** — `customer_zip`
- **Level 4: Customer Name** — `customer_name`
- **Level 5: Customer IDs**
 - `customer_surrogate_id` (PK)
 - `customer_id` (Source System ID)

Hierarchy: Scratch Game Dimension This hierarchy enables grouping of games by pricing and winning characteristics, based on the `DimScratchGame` table.

- **Level 1: Ticket Price** — `ticket_price`
- **Level 2: Odds Bracket** — `average_odds` (can be bucketed)
- **Level 3: Game Identifier** — `scratch_game_number`

Hierarchy: Prize Rule Dimension This hierarchy supports analysis of prize structures from general rules to scaled values in the `DimPrizeRule` table.

- **Level 1: Game** — `game_id`
- **Level 2: Winning Type** — `winning_type_id`
- **Level 3: Scaled Prize** — `base_prize_scaled`

1.4.5 Dimension Hierarchies for Draw-Based Lottery

Draw Game Dimension Hierarchy Levels

- **Level 1: Game Category** — `game_category` (defines standard ticket price)
- **Level 2: Game IDs**
 - `game_surrogate_id` (PK)
 - `game_id` (Original ID)

Hierarchy of Chosen Number Combinations

- **Level 1: Array of Chosen Combinations** — `chosen_numbers_combination_surrogate_id`
- **Level 2: Individual Number Combinations (referenced via array attribute)**
 - `winning_numbers_combination_surrogate_id` (from DimCombinations)
 - `winning_numbers_combination`

Hierarchy: Date Dimension This hierarchy defines the fiscal time structure for temporal analysis and aggregation in the DimDate table.

- **Level 1: Fiscal Year** — `fiscal_year`
- **Level 2: Fiscal Month (Number)** — `fiscal_month`
- **Level 3: Fiscal Month (Name or Number)** — `fiscal_month_name_number`
- **Level 4: Date ID** — `date_id`

Hierarchy: Retailer Dimension This hierarchy organizes retailers geographically and by corporate ownership using attributes in the DimRetailer table.

- **Level 1: State** — `retailer_state`
- **Level 2: County** — `retailer_county`
- **Level 3: City** — `retailer_city`
- **Level 4: Zip Code** — `retailer_zip_code`
- **Level 5: Retailer Address** — `retailer_address_1`
- **Level 6: Retailer Surrogate ID** — `retailer_surrogate_id`
- **Level 1: Owning Entity Chain Head Name** — `owning_entity_chain_head_name`
- **Level 2: Owning Entity Name** — `owning_entity_name`

- **Level 3: Owning Entity Number** — `owning_entity_number`
- **Level 4: Retailer (Surrogate ID, License Number, Name)** — `retailer_surrogate_id`, `retailer_license_number`, `retailer_name`

Hierarchy: Customer Dimension This hierarchy structures customers demographically for behavioral analysis, using fields in the `DimCustomer` table.

Customer Dimension Hierarchy Levels

- **Level 1: State** — `customer_state`
- **Level 2: City** — `customer_city`
- **Level 3: ZIP Code** — `customer_zip`
- **Level 4: Customer Name** — `customer_name`
- **Level 5: Customer IDs**
 - `customer_surrogate_id` (PK)
 - `customer_id` (Source System ID)

1.5 Grain. DIM. FACT

1.5.1 Scratch Based Game

Column Name	PostgreSQL Type	Example	Description
<code>sale_id</code>	BIGINT	1	Unique identifier for each sale record
<code>fiscal_year</code>	SMALLINT	2021	Fiscal year of the transaction
<code>fiscal_month</code>	SMALLINT	7	Fiscal month number (1-12)
<code>fiscal_month_name_and_number</code>	VARCHAR(20)	07-March	Fiscal month as number and name
<code>scratch_game_number</code>	INTEGER	2053	Identifier for the scratch game
<code>ticket_price</code>	INTEGER	30	Price of one ticket
<code>retailer_license_number</code>	BIGINT	100663	License number of the retailer
<code>retailer_location_name</code>	VARCHAR(100)	ISI-KAT	Name of the retailer location
<code>retailer_number_and_location_name</code>	VARCHAR(150)	100663 - ISI-KAT	Combined retailer number and location name
<code>retailer_location_address_1</code>	VARCHAR(150)	1619 BLANCO RD	Street address of retailer location

retailer_location_city	VARCHAR(100)	San Antonio	City of retailer location
retailer_location_state	VARCHAR(5)	TX	State code of retailer location
retailer_location_zip_code	VARCHAR(10)	78212	Zip or postal code
retailer_location_county	VARCHAR(100)	Bexar	County of retailer location
owning_entity_retailer_number	BIGINT	100663	Retailer number of the owning entity
owning_entity_retailer_name	VARCHAR(100)	ISI-KAT	Name of the owning entity
ticket_bought	INTEGER	3	Number of tickets bought in the sale
transaction_date	DATE	2021-07-12	Date of transaction
date_id	BIGINT	20210712	Numeric date identifier (YYYYMMDD)
sale	INTEGER	90	Total sale amount
customer_id	VARCHAR(50)	CUST_100663_20	Unique identifier for customer
customer_name	VARCHAR(100)	melissa reed	Customer's full name
customer_email	VARCHAR(150)	donna03@example.net	Customer's email address
customer_phone_number	VARCHAR(30)	634-499-5460x869	Customer's phone number
customer_street_address	VARCHAR(150)	89673 Howard Village Apt 408	Customer's street address
customer_city	VARCHAR(100)	Robbinstown	Customer's city
customer_state	VARCHAR(5)	VI	Customer's state or region code
customer_zip_code	VARCHAR(10)	14401	Customer's postal code
customer_birthdate	DATE	2005-07-29	Customer's birth date
customer_gender	VARCHAR(10)	female	Customer's gender
customer_job_title	VARCHAR(150)	Accountant, chartered public finance	Customer's job title
customer_company	VARCHAR(100)	Griffin Inc	Customer's company name
average_odds	VARCHAR(20)	1:3.44	Average odds represented as ratio
average_odds_prob	NUMERIC(6,4)	0.2907	Probability of average odds
prize_rule_surrogate_id	INTEGER	3	Surrogate key for prize rule
winning_type_id	VARCHAR(50)	other_small_prize	Type identifier for winning
base_prize_scaled	INTEGER	300	Scaled base prize amount
customer_surrogate_id	INTEGER	1452	Surrogate key for customer
date_surrogate_id	INTEGER	193	Surrogate key for date
game_surrogate_id	INTEGER	3	Surrogate key for game
retailer_surrogate_id	INTEGER	28	Surrogate key for retailer
employee_id	VARCHAR(50)	E100663_0002	Unique identifier for employee
employee_surrogate_id	INTEGER	86	Surrogate key for employee

employee_name	VARCHAR(100)	denise_frederick	Employee full name
employee_email	VARCHAR(150)	mullinsashley@example.com	Employee email address
employee_phone_number	VARCHAR(30)	831.243.4060x871	Employee phone number

1.5.2 Draw Based Game

Column Name	PostgreSQL Type	Example	Description
sale_id	VARCHAR(50)	100663-23376-28	Unique identifier for each sale record
fiscal_year	SMALLINT	2021	Fiscal year of the transaction
fiscal_month	SMALLINT	5	Fiscal month number (1-12)
fiscal_month_name_number	VARCHAR(20)	05-January	Fiscal month as number and name
game_category	VARCHAR(100)	All or Nothing	Category of the game
ticket_price	NUMERIC(10,2)	N/A	Price of one ticket (may be missing)
retailer_license_number	BIGINT	100663	License number of the retailer
retailer_name	VARCHAR(100)	ISI-KAT	Name of the retailer
retailer_number_location_name	VARCHAR(150)	100663 - ISI-KAT	Combined retailer number and location name
retailer_address_1	VARCHAR(150)	1619 BLANCO RD	Street address of retailer location
retailer_city	VARCHAR(100)	San Antonio	City of retailer location
retailer_state	VARCHAR(5)	TX	State code of retailer location
retailer_zip_code	VARCHAR(10)	78212	Zip or postal code
retailer_county	VARCHAR(100)	Bexar	County of retailer location
owning_entity_number	BIGINT	100663	Retailer number of the owning entity
owning_entity_name	VARCHAR(100)	ISI-KAT	Name of the owning entity
owning_entity_chain_head_name	VARCHAR(150)	100663 - ISI-KAT	Chain head name of the owning entity
sales	INTEGER	3	ticket sales amount per transaction
tickets_bought	INTEGER	2	Number of tickets bought in the sale
transaction_date	DATE	2021-05-12	Date of transaction
date_id	BIGINT	20210512	Numeric date identifier (YYYYMMDD)
chosen_ids	JSONB	['ALLORNOTHING_000009', 'ALLORNOTHING_000010']	List of chosen IDs

winning_combination_id	VARCHAR(50)	ALLORNOTHING_000002	Identifier for winning combination
winning_numbers_combination_surrogate_id	INTEGER	2	Surrogate ID for winning numbers combination
game_id	VARCHAR(100)	all_or_nothing_05-january-100663	Unique game identifier
game_surrogate_id	INTEGER	1	Surrogate key for game
date_surrogate_id	INTEGER	6	Surrogate key for date
retailer_surrogate_id	INTEGER	1	Surrogate key for retailer
customer_surrogate_id	INTEGER	6	Surrogate key for customer
customer_id	VARCHAR(50)	CUST_100663_26	Unique identifier for customer
customer_name	VARCHAR(100)	Lawrence Brandt	Customer's full name
customer_date_of_birth	DATE	1975-07-18	Customer's birth date
customer_gender	VARCHAR(10)	M	Customer's gender
customer_city	VARCHAR(100)	New David	Customer's city
customer_state	VARCHAR(5)	FM	Customer's state or region code
customer_zip	VARCHAR(10)	75348	Customer's postal code
employee_id	VARCHAR(50)	E100663_0002	Unique identifier for employee
employee_name	VARCHAR(100)	katie.giles	Employee full name
employee_email	VARCHAR(150)	bullockbrad@example.com	Employee email address
employee_phone_number	VARCHAR(30)	(884)709-6208x4208	Employee phone number
employee_surrogate_id	INTEGER	2	Surrogate key for employee
chosen_numbers_combination	JSONB	['3-4-5-7-9-11-12-13-15-16-19-23', '3-8-9-12-13-14-16-18-19-21-22-24']	List of chosen number combinations
chosen_numbers_combination_surrogate_ids	JSONB	[9, 10]	List of surrogate IDs for chosen combinations
chosen_numbers_combination_surrogate_id	INTEGER	6	Surrogate key for chosen number combination
winning_numbers_combination	VARCHAR(100)	1-3-4-7-8-10-14-17-18-21-22-24	Winning number combination as string
winning_ticket_count	INTEGER	0	Count of winning tickets
is_winning_ticket	BOOLEAN	False	Whether the ticket is winning or not
winning_prize	NUMERIC(10,2)	0.0	Amount of prize won
amount_won	NUMERIC(10,2)	0.0	Amount of money won

1.5.3 Grain Selection Process

To ensure our fact tables accurately reflect the business processes of the Texas Lottery system, we have followed the classic four-step method for defining the grain of a fact table, as outlined by Ralph Kimball. The essence of this approach lies in performing the hard

work upfront—carefully identifying the highest-level variables to include in each fact table—so that querying becomes significantly easier later. By distilling the fact tables to their most granular, meaningful level and linking them to relevant dimension tables, we enable streamlined and intuitive analytics.

1. Choose the Business Process

We began by identifying the key business processes we aim to model:

- **Scratch-Based Lottery Sales:** Instant-win ticket purchases where outcomes are known at the time of purchase.
- **Draw-Based Lottery Sales:** Number-selection games where outcomes are determined in a later draw event.

These two distinct processes led us to create two separate fact tables: `ScratchTicketSalesFact` and `DrawTicketSalesFact`.

2. Declare the Grain

The grain defines the level of detail represented by each row in the fact table:

- For `ScratchTicketSalesFact`, the grain is **one scratch ticket sales transaction**—e.g., a customer purchasing 3 tickets for **Game #2085: \$500 Frenzy** in a single transaction.
- For `DrawTicketSalesFact`, the grain is **one draw-based ticket transaction**, capturing selected numbers and the corresponding winning combinations.

Clearly declaring the grain has guided our decisions about which dimensions and facts are appropriate for each table.

3. Identify the Dimensions

Next, we determined the necessary dimension tables to describe the context surrounding each fact:

- `ScratchTicketSalesFact` is linked to: `DimScratchGame`, `DimRetailer`, `DimCustomer`, `DimEmployee`, and `DimDate`.
- `DrawTicketSalesFact` connects to: `DimDrawGame`, `DimCombinations`, `DimArrayCombinations`, `DimRetailer`, `DimCustomer`, `DimEmployee`, and `DimDate`.

These dimensions enable rich analysis by game type, customer demographics, retailer performance, employee involvement, and time. For draw-based games, additional dimensions like `DimCombinations` and `DimArrayCombinations` support detailed tracking of selected and winning number combinations.

4. Determine the Facts

Finally, we identified the quantitative measures that are meaningful at each grain level:

- `ScratchTicketSalesFact`: `ticket_price`, `tickets_bought`, and `sale` (e.g., 3 tickets at \$5 = \$15).
- `DrawTicketSalesFact`: `ticket_price`, `tickets_bought`, `ticket_sales`, `winning_ticket_count`, `is_winning_ticket`, `winning_prize`, and `amount_won`.

These facts enable comprehensive insights into customer purchasing behavior, game profitability, and overall revenue trends.