Exploring Insights into Walmart Sales

Project Details

TOOLS

I have used R, MySQL, Tableau, Lucid Charts and Microsoft Excel in this project

SCOPE

Foundations of database using MySQL

COMPONENTS

Research, Design, Data loading, Data cleaning, DataAnalysis, Insights

DATASET

https://www.kaggle.com/datasets/aslanahmedov/walmart-sales-forecast

DURATION

2 months of the first semester

Project Index:

1. Introduction
2. Conceptual Model
3. Logical Model
4. Physical Model
5. Data loading concept
6. Data analysis and Insights
7. Data Visualizations
8. Conclusion

Project Introduction

01

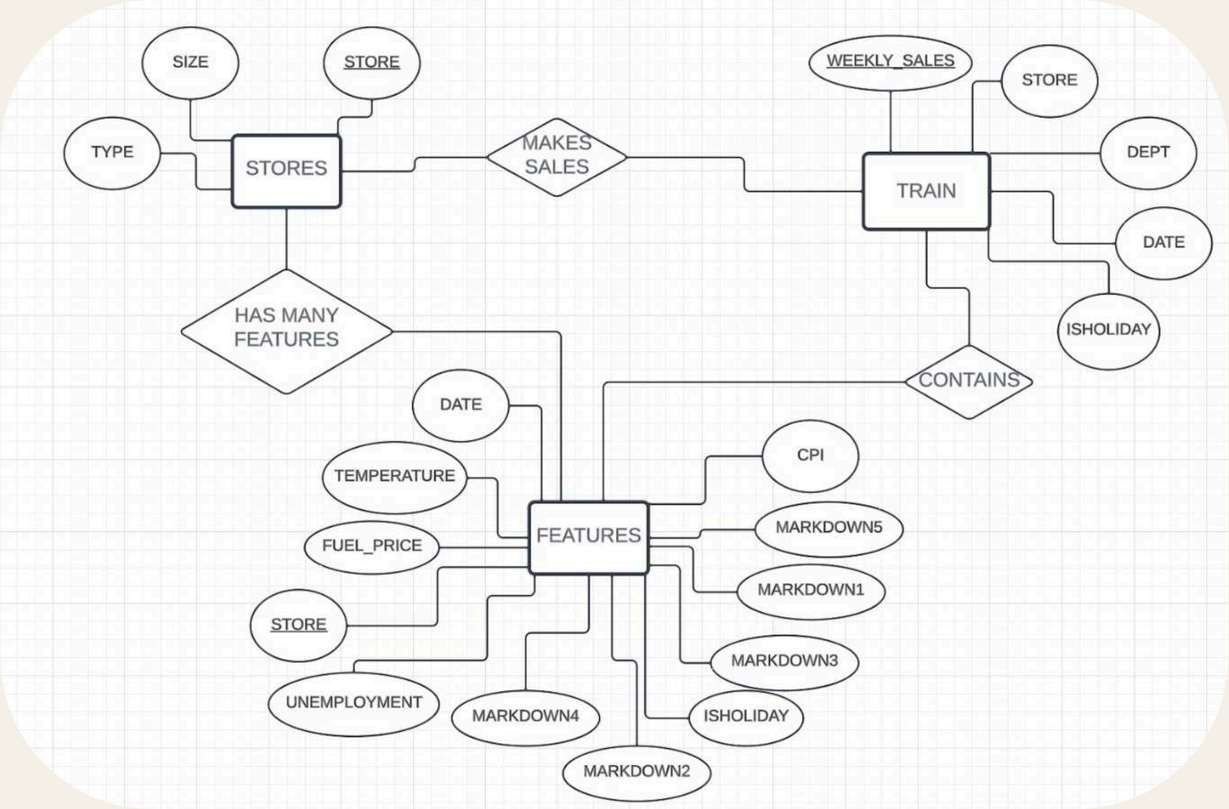
As students new to the world of large scale shopping centers such as Walmart, target, etc. we were curious to better our understanding of how the business works. How the products are placed, the thought behind every small detail in a store/on the website and how our inights could increase the revenue (obviously backed up by data)

Large-scale shopping centers are a highly competitive market. As a business analytics student, it is important to understand why one business stands out from another, the uniqueness of the business model, and how to better increase the impact of a company.

I analyzed the dataset with the goal to find insights that Walmart can follow to enhance their sales.

Conceptual Diagram

02



ENTITIES

The main boxes in the diagram represent entities—key components or objects in the data model. In this case, the entities are:

- STORES: Represents individual retail stores.
- FEATURES: Represents various external features or attributes related to stores (like temperature, fuel prices, etc.).
- TRAIN: Represents sales data, capturing weekly sales figures, the department, the date, and whether the sales week included a holiday.

RELATIONSHIPS

The diamond shapes between entities represent relationships—how the entities interact or relate to each other.

In this diagram:

- HAS MANY FEATURES: This indicates that each STORE can have multiple FEATURES associated with it. For example, a store may have data on different temperatures, fuel prices, and markdowns over time.
- MAKES SALES: This relationship shows that STORES make SALES, which are captured in the TRAIN entity.
- CONTAINS: This relationship connects the TRAIN entity with the FEATURES, indicating that the sales data in TRAIN may be analyzed in relation to the external features in FEATURES.

ATTRIBUTES

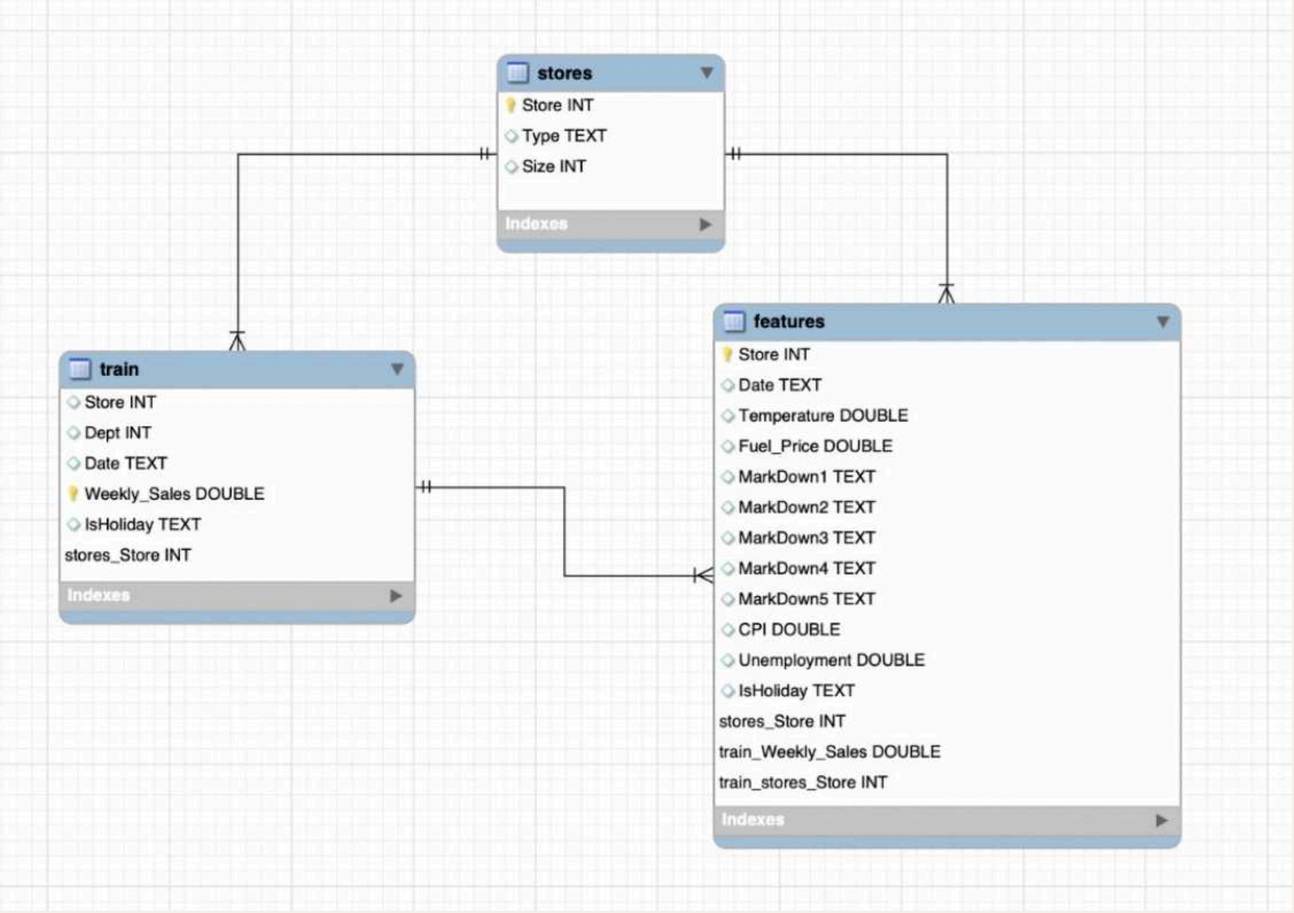
The ovals connected to the entities represent attributes—specific pieces of information associated with each entity.

- STORES has attributes like Store, Type, and Size.
- FEATURES has attributes like Date, Temperature, Fuel_Price, and various MarkDown values.
- TRAIN has attributes like Store, Dept, Date, Weekly_Sales, and IsHoliday.

Logical Diagram

03

The logical model represents a more detailed and structured version of the conceptual model, specifying the actual tables and relationships in a database management system (DBMS) without delving into the physical implementation (like storage details). Here's an explanation of what this logical model represents:



RELATIONSHIPS:

- The STORES table has a 1-to-many relationship with both the FEATURES and TRAIN tables. This means each store can have multiple entries in the FEATURES table (capturing different data points like temperature, markdowns, etc., over time) and multiple sales records in the TRAIN table.

- The TRAIN table also has a 1-to-many relationship with the FEATURES table, indicating that each sales record can be related to multiple features recorded during that period.

Physical Model

04

```
-- MySQL Workbench Forward Engineering

SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR
_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';

-- -----------------------------------------------------
-- Schema mydb
-- -----------------------------------------------------
-- -----------------------------------------------------
-- Schema walmart
-- -----------------------------------------------------

-- -----------------------------------------------------
-- Schema walmart
-- -----------------------------------------------------
```

```
CREATE SCHEMA IF NOT EXISTS `walmart` DEFAULT CHARACTER SET utf8mb4 COLLATE
utf8mb4_0900_ai_ci ;
USE `walmart` ;


-- -----------------------------------------------------
-- Table `walmart`.`stores`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `walmart`.`stores` (
  `Store` INT NOT NULL,
  `Type` TEXT NULL DEFAULT NULL,
  `Size` INT NULL DEFAULT NULL,
  PRIMARY KEY (`Store`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;



-- -----------------------------------------------------
-- Table `walmart`.`train`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `walmart`.`train` (
  `Store` INT NULL DEFAULT NULL,
  `Dept` INT NULL DEFAULT NULL,
  `Date` TEXT NULL DEFAULT NULL,
  `Weekly_Sales` DOUBLE NOT NULL,
  `IsHoliday` TEXT NULL DEFAULT NULL,
  `stores_Store` INT NOT NULL,
  PRIMARY KEY (`Weekly_Sales`, `stores_Store`),
  INDEX `fk_train_stores1_idx` (`stores_Store` ASC) VISIBLE,
  CONSTRAINT `fk_train_stores1`
    FOREIGN KEY (`stores_Store`)
    REFERENCES `walmart`.`stores` (`Store`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;
```

```
-- -----------------------------------------------------
-- Table `walmart`.`features`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `walmart`.`features` (
  `Store` INT NOT NULL,
  `Date` TEXT NULL DEFAULT NULL,
  `Temperature` DOUBLE NULL DEFAULT NULL,
  `Fuel_Price` DOUBLE NULL DEFAULT NULL,
  `MarkDown1` TEXT NULL DEFAULT NULL,
  `MarkDown2` TEXT NULL DEFAULT NULL,
  `MarkDown3` TEXT NULL DEFAULT NULL,
  `MarkDown4` TEXT NULL DEFAULT NULL,
  `MarkDown5` TEXT NULL DEFAULT NULL,
  `CPI` DOUBLE NULL DEFAULT NULL,
  `Unemployment` DOUBLE NULL DEFAULT NULL,
  `IsHoliday` TEXT NULL DEFAULT NULL,
  `stores_Store` INT NOT NULL,
  `train_Weekly_Sales` DOUBLE NOT NULL,
  `train_stores_Store` INT NOT NULL,
  PRIMARY KEY (`Store`, `stores_Store`, `train_Weekly_Sales`, `train_stores_Store`),
  INDEX `fk_features_stores_idx` (`stores_Store` ASC) VISIBLE,
  INDEX `fk_features_train1_idx` (`train_Weekly_Sales` ASC, `train_stores_Store` ASC) VISIBLE,
  CONSTRAINT `fk_features_stores`
    FOREIGN KEY (`stores_Store`)
    REFERENCES `walmart`.`stores` (`Store`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_features_train1`
    FOREIGN KEY (`train_Weekly_Sales` , `train_stores_Store`)
    REFERENCES `walmart`.`train` (`Weekly_Sales` , `stores_Store`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;


SET SQL_MODE=@OLD_SQL_MODE;
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```

Data Loading Concept

05

```
--
-- Table structure for table `features`
--

DROP TABLE IF EXISTS `features`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `features` (
  `Store` int DEFAULT NULL,
  `Date` text,
  `Temperature` double DEFAULT NULL,
  `Fuel_Price` double DEFAULT NULL,
  `MarkDown1` text,
  `MarkDown2` text,
  `MarkDown3` text,
  `MarkDown4` text,
  `MarkDown5` text,
  `CPI` double DEFAULT NULL,
  `Unemployment` double DEFAULT NULL,
  `IsHoliday` text
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;
```

```
/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!50503 SET NAMES utf8 */;
/*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
/*!40103 SET TIME_ZONE='+00:00' */;
/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */;
/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
/*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;
```

This is the data loading concept used

First, I created schema called "Walmart"

Then, I ran code for each table in dataset

This shows example for table "features"

```
42    -- Dumping data for table `features`
43    --
44
45    LOCK TABLES `features` WRITE;
46    /*!40000 ALTER TABLE `features` DISABLE KEYS */;
47    INSERT INTO `features` VALUES (1,'2010-02-05',42.31,2.572,'NA','NA','NA','NA','NA',211.0963582,8.106,'FALSE'),(1,'2010-02-12',38.
48    /*!40000 ALTER TABLE `features` ENABLE KEYS */;
49    UNLOCK TABLES;
```

Data Analysis and Insights

06

Basic insights

I conducted data anaysis using MySQL queries

3

types of stores

45

total number of stores

A,B,C

categories of the stores

Initial analysis

#13

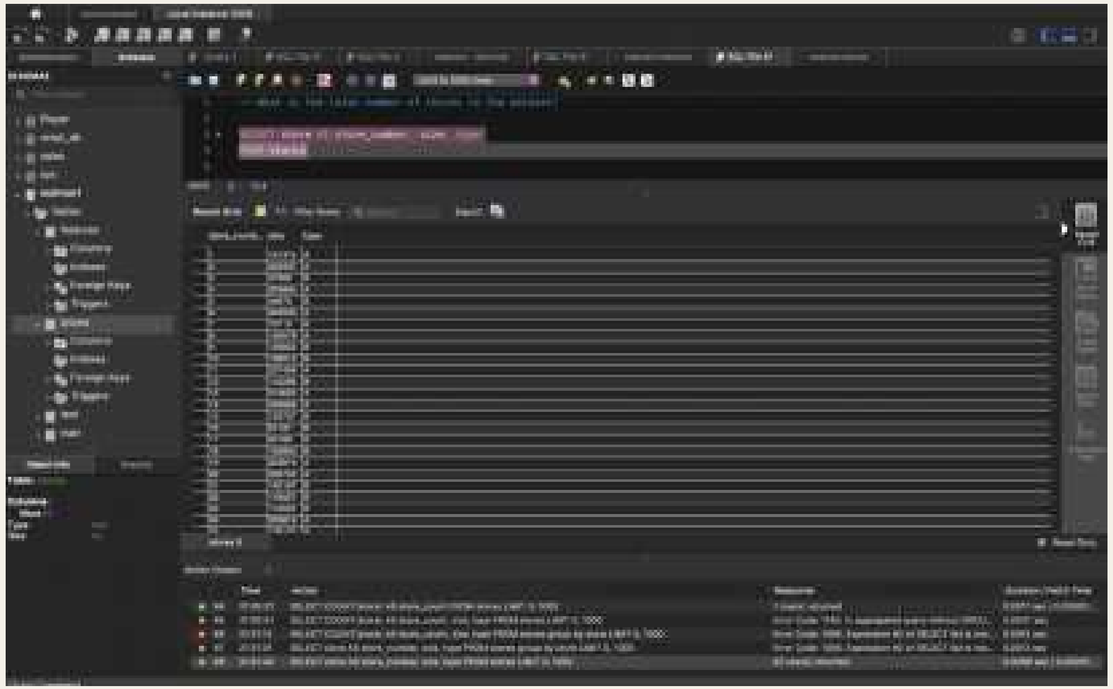
avg weekly sales $27,355.14
size 219,622,

#32

avg weekly sales $16,351.62
size 203,007

despite having a substantial size of 203,007, #32 has a comparatively lower average weekly sales. Business managers should work on optimizinf the space to increase products with high demand in #32 by using data from stores such as #13

Initial analysis:



#20 & #4

stores with highest sales

#33 & #44

stores with lowest sales

What were stores #20 and #4 doing right?

What were stores #33 and #44 doing wrong?

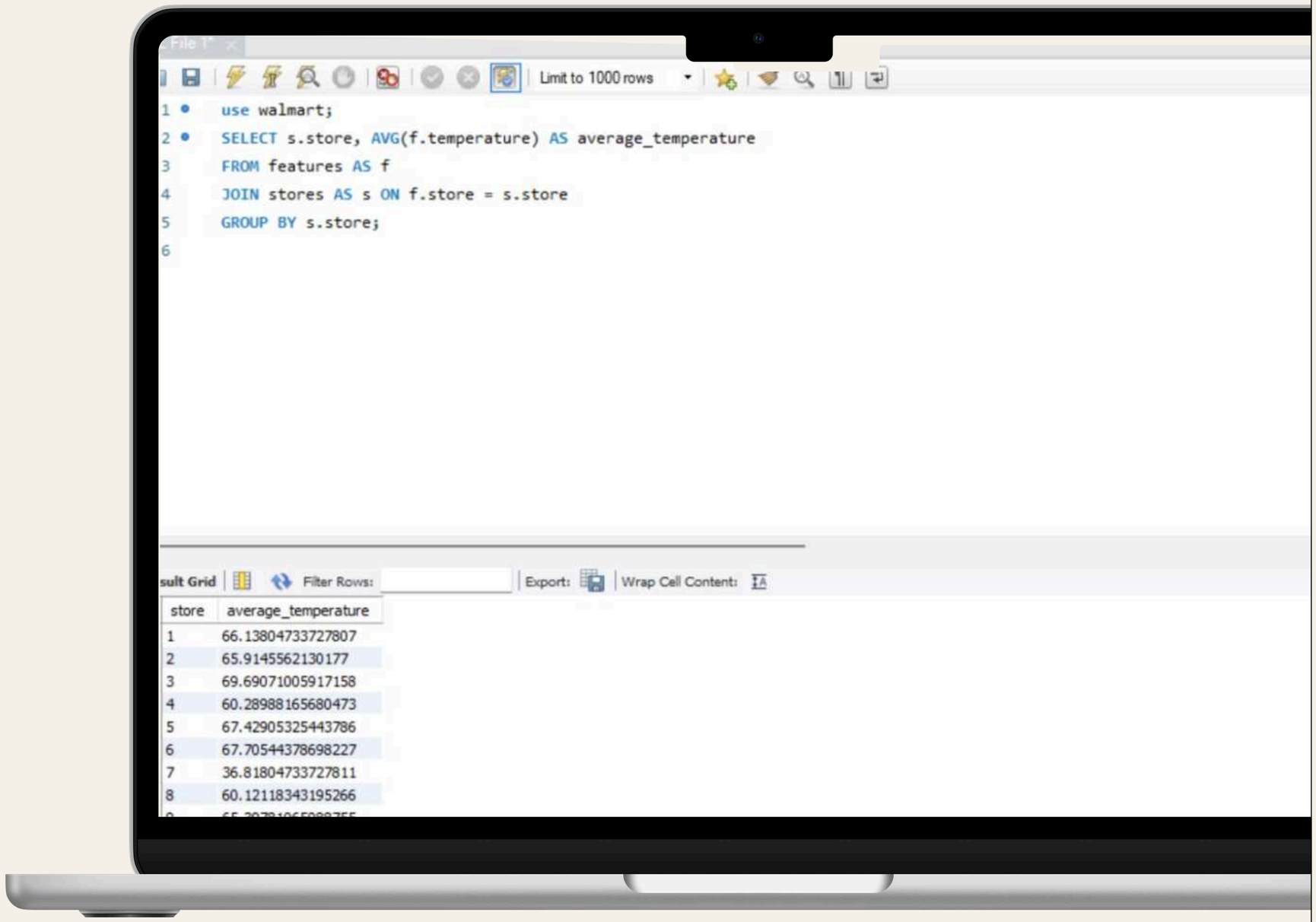
Average temperature

Average temperature of each store ranges from 44 degrees to 74 degrees.

This information can help with inventory management - certain products may be more in demand in regions with specific temperature ranges

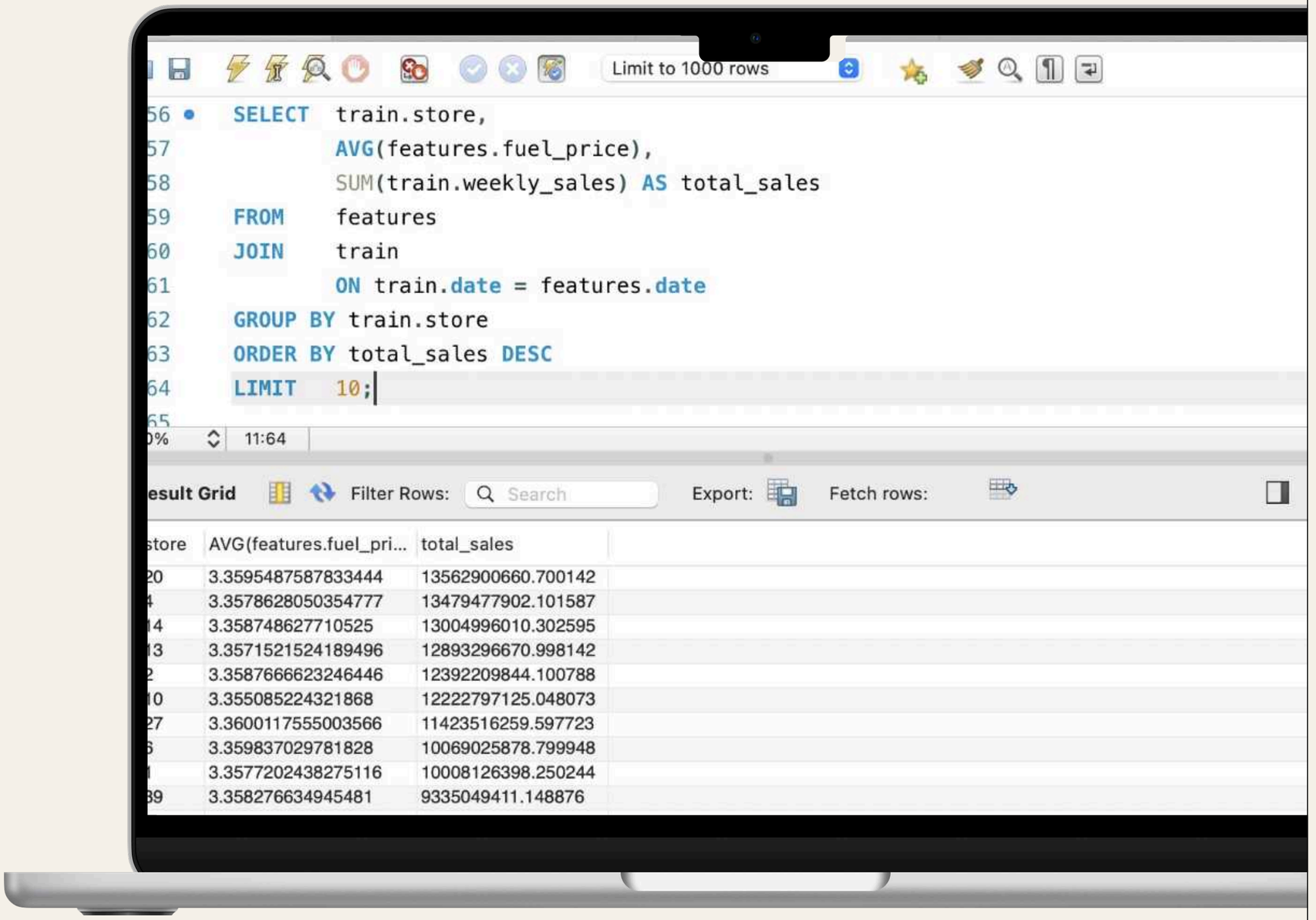
Additionally, insights into temperature patterns can inform energy consumption strategies, helping stores optimize heating or cooling systems based on the climate

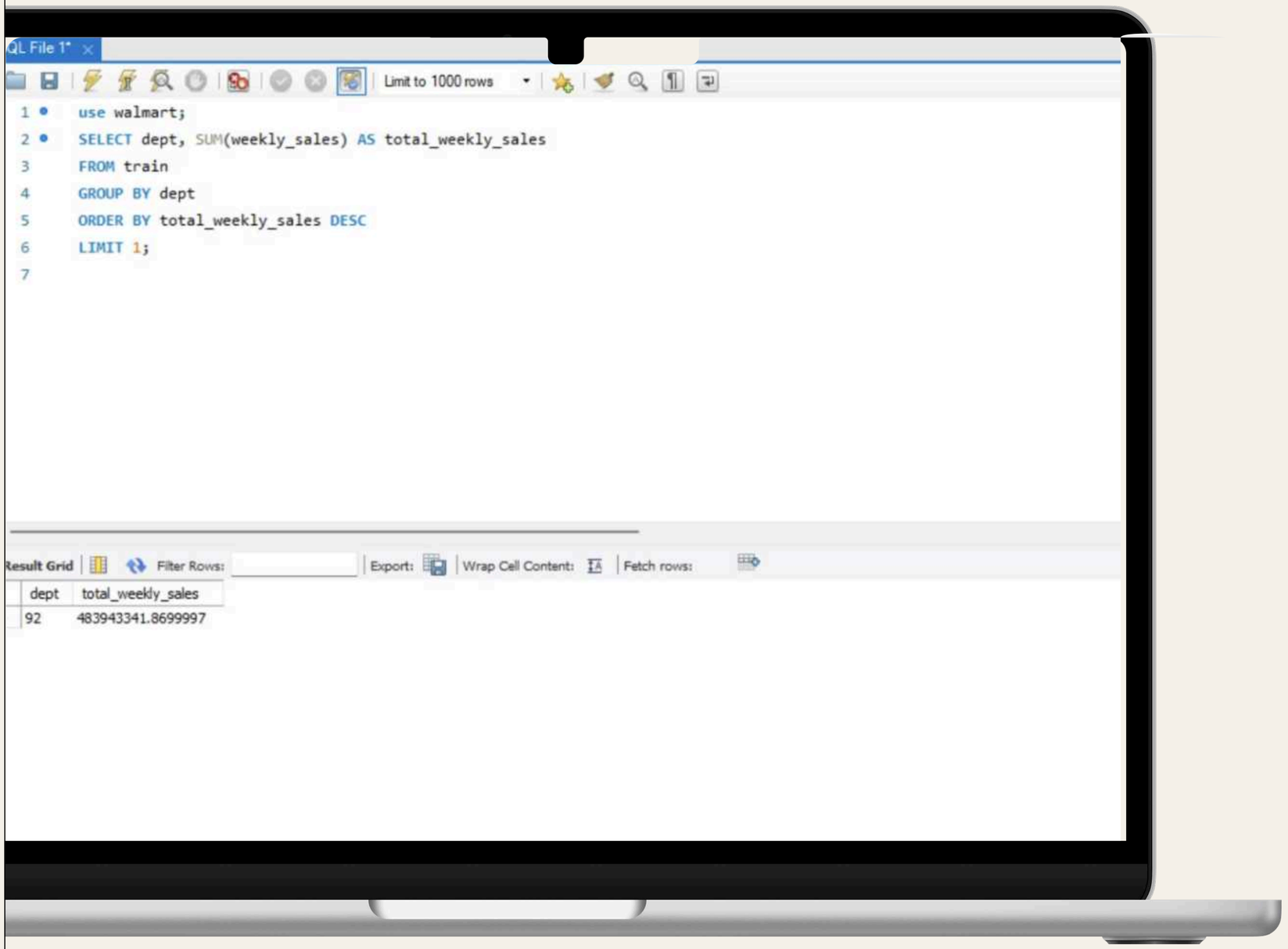
The data can also be valuable for marketing and product placement decisions, Walmart can stock products that align with the local climate preferences of each store's customer base



Average fuel price

- The average fuel price is currently around **$3.35,** which is a key figure for our analysis.
- Notably, the top 10 sales occurred at these price points, indicating a correlation between fuel prices and sales performance.
- It's crucial for Walmart to monitor fuel prices in their area to stay competitive.
- We should be aware that sales may decrease if fuel costs increase, impacting overall revenue.





Department 92 has the highest weekly sales

- Department 92 has shown outstanding performance by achieving the highest weekly sales overall.
- This success serves as a significant indicator of the department's effectiveness.
- Walmart should look into expanding the marketing strategies that have worked well in Department 92.
- By implementing these successful practices across other departments, Walmart can boost overall business performance.
- It's essential to identify areas for improvement in other departments to ensure sustained growth and efficiency.

```sql
SELECT  *
FROM    (
    SELECT  train.date,
            train.weekly_sales,
            train.store,
            RANK() OVER(PARTITION BY train.store ORDER BY train.weekly_sales DESC) AS sales_rank
    FROM    train
        ) rnk_table
WHERE   sales_rank = 1
ORDER BY store;
```

17:67

rid Filter Rows: Search Export:

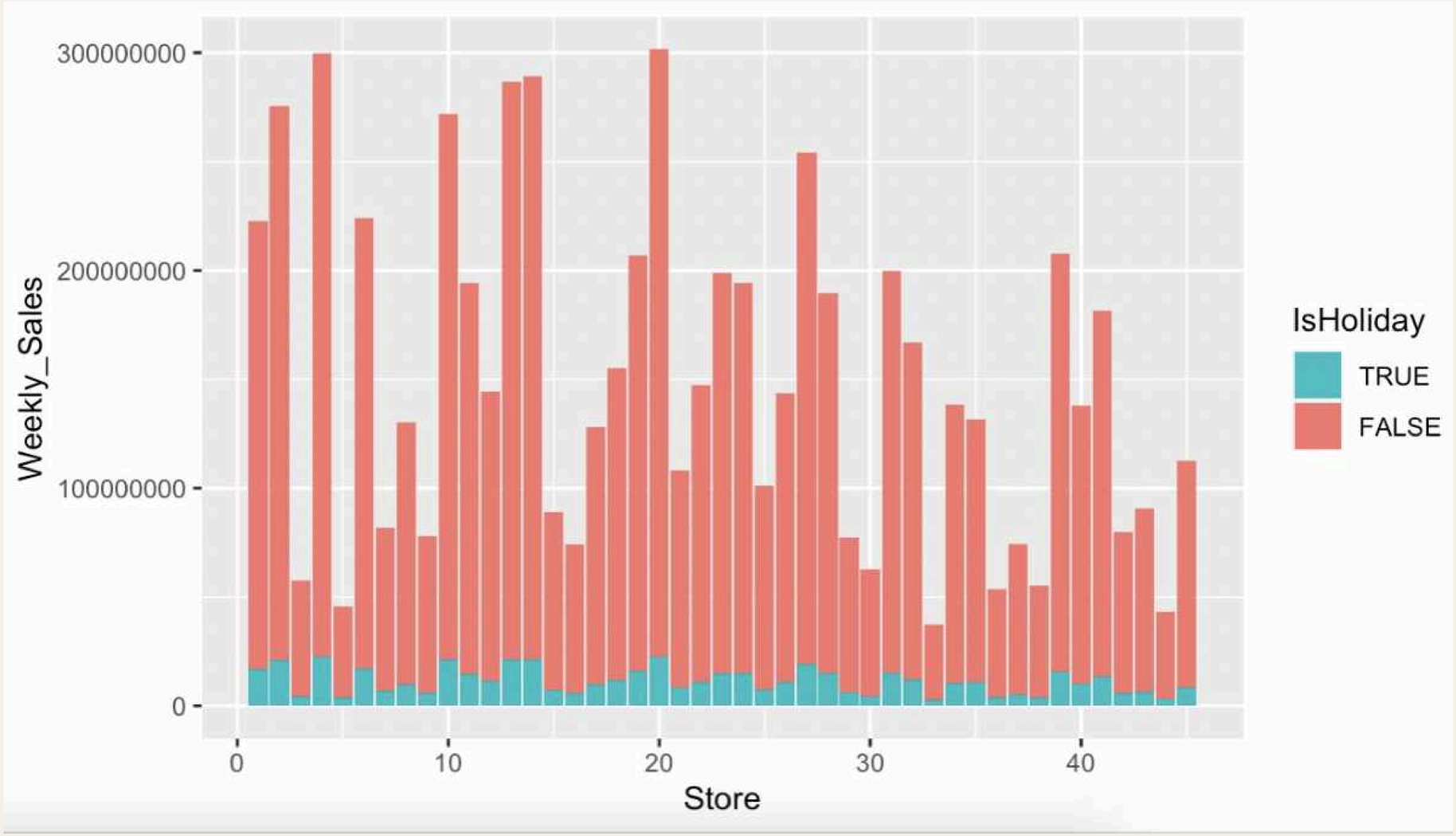
	weekly_sales	store	sales_rank
-25	203670.47	1	1
-26	285353.53	2	1
-05	155897.94	3	1
-25	385051.04	4	1
-25	93517.72	5	1
-26	342578.65	6	1
-25	222921.09	7	1
-26	153431.69	8	1
-25	139427.43	9	1
-26	693099.36	10	1
-26	245767.47	11	1

Highest weekly sales

- The highest weekly sales observed, particularly noting that the majority occur at the end of November.
- This trend is consistent across most stores, regardless of the year.
- We typically see a surge in sales around Black Friday and thanksgiving, which is a key shopping period.
- Walmart has opportunities to further promote sales during this time to maintain or even increase sales figures.

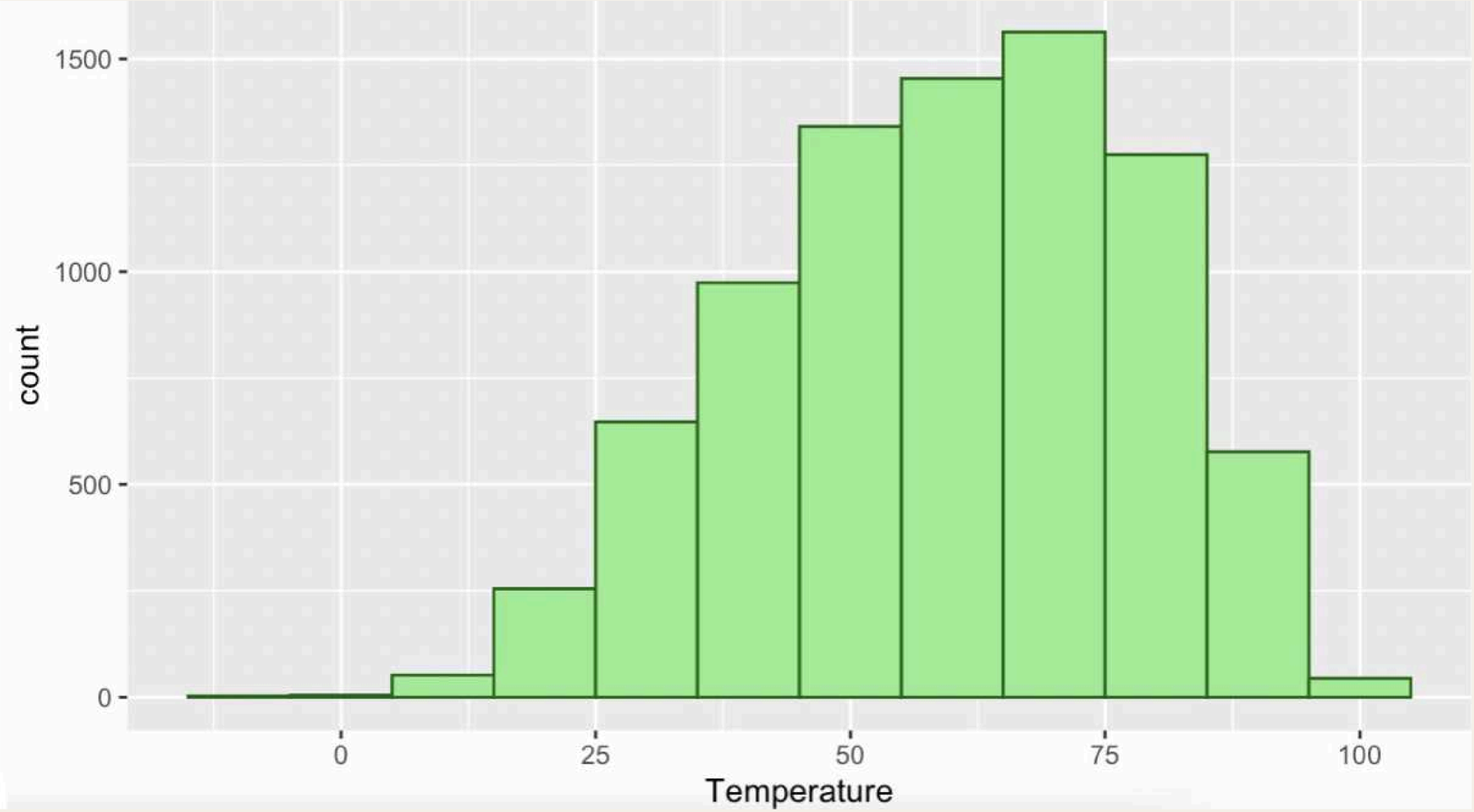
Data Visualization

07

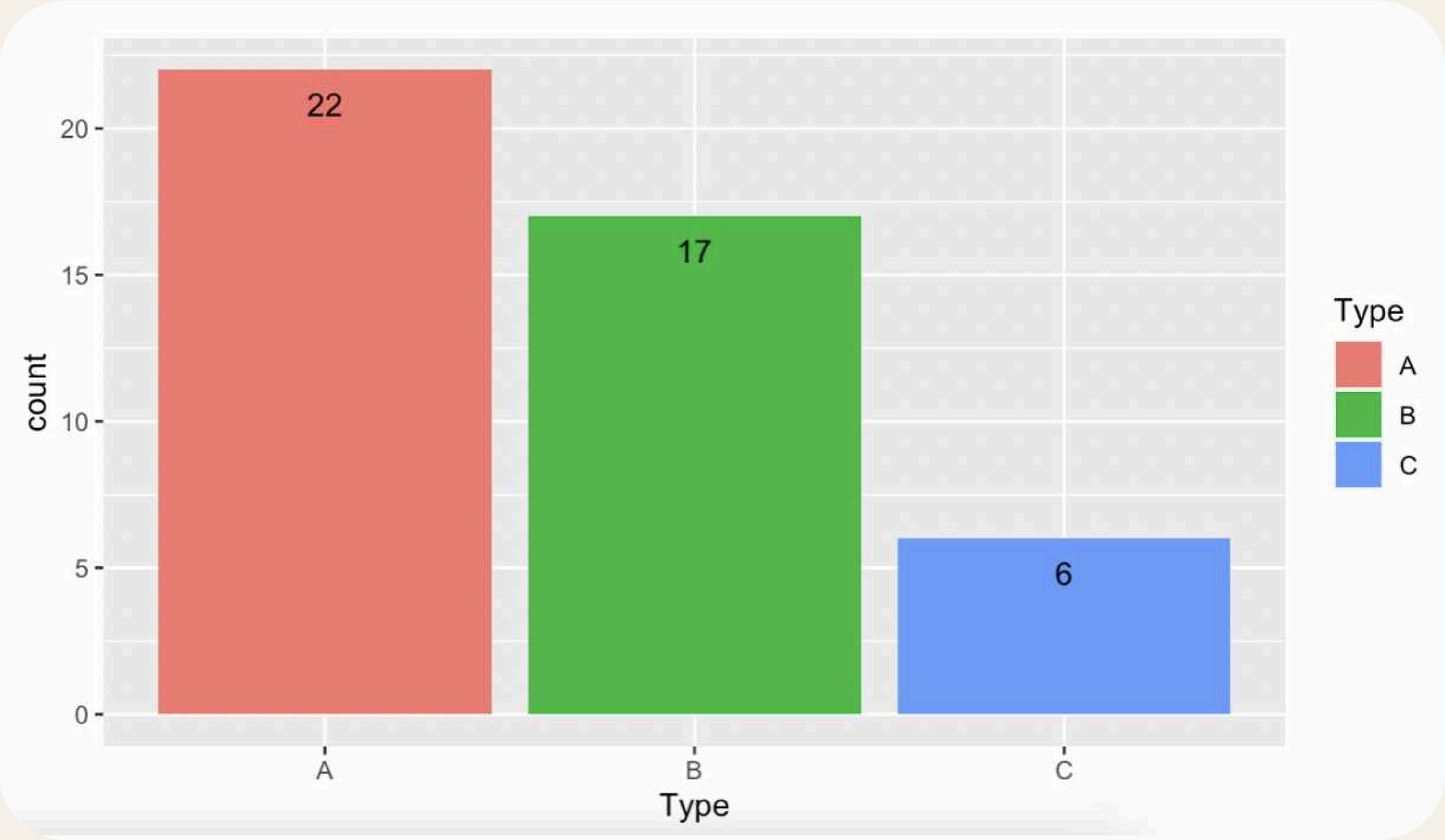


Non-holiday weeks contribute most to each store's weekly sales, with substantial variability in total weekly sales between different stores.

Some stores show exceptionally high weekly sales, reaching up to 30,000,000, while others have lower weekly sales. This variation could reflect differences in store size, location, or customer base.



The graph shows that most stores experience temperatures in the 50–75 range, with fewer stores at very low or very high temperatures.



The graph shows that there are

22 stores of type A, 17 of type B and 6 of type C

Conclusion

08

- **Consumer Spending Trends:** Seasonal peaks (e.g., holidays, back-to-school) and category-specific surges guide inventory and marketing strategies.

- **Holiday & Time-Based Spikes:** Major holidays and external events drive sales spikes, requiring tailored promotions and staffing adjustments.

- **Store Size & Temperature Impact:** Larger stores see higher sales; weather influences product demand (e.g., winter gear vs. summer items).

- **Marketing & Promo Guidance:** Insights optimize campaign timing, target demographics, and promotional effectiveness.

- **Revenue Growth Potential:** Findings enable dynamic pricing, better stock management, and tailored store-level decisions to boost sales and customer satisfaction.

Thank you!