

Documented Report for Power BI Capstone Project

- Vibhore Aggarwal (S5289)

1. SQL Data Source

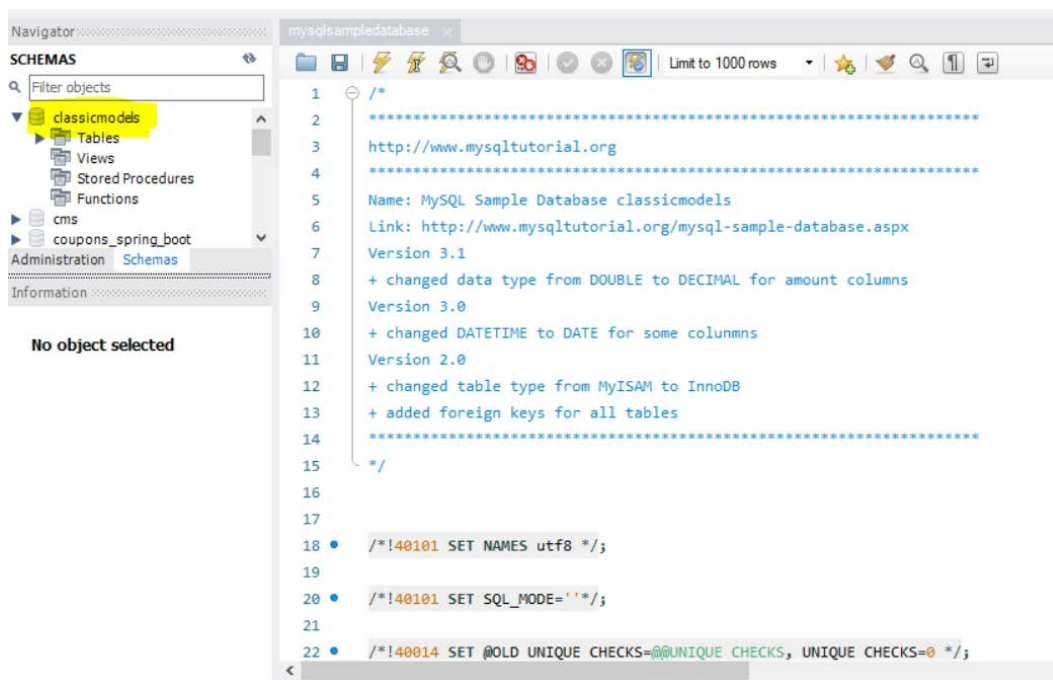
```
mysqlsampledatabase.sql

/*
*****
http://www.mysqltutorial.org
*****
Name: MySQL Sample Database classicmodels
Link: http://www.mysqltutorial.org/mysql-sample-database.aspx
Version 3.1
+ changed data type from DOUBLE to DECIMAL for amount columns
Version 3.0
+ changed DATETIME to DATE for some columns
Version 2.0
+ changed table type from MyISAM to InnoDB
+ added foreign keys for all tables
*****
*/

/*!40101 SET NAMES utf8 */;
/*!40101 SET SQL_MODE='';*/

/*!40104 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
/*!40104 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 */;
CREATE DATABASE /*!32312 IF NOT EXISTS*/ 'classicmodels' /*!40100 DEFAULT CHARACTER SET latin1 */;
USE 'classicmodels';

/*Table structure for table 'customers' */
```



| | | | |
|--------------------------------|------------------|------------------|---------------|
| Microsoft Update Health Ser... | Maintains U... | Disabled | Local System |
| Microsoft Windows SMS Ro... | Routes mess... | Manual (Trigg... | Local Service |
| Mozilla Maintenance Service | The Mozilla ... | Manual | Local System |
| MYSQL80 | Running | Manual | Network Se... |
| Natural Authentication | Signal aggre... | Manual (Trigg... | Local System |
| Net.Tcp Port Sharing Service | Provides abil... | Disabled | Local Service |
| Netlogon | Maintains a ... | Manual | Local System |

1. Downloaded the "mysqlsampledatabase.sql" file from LMS
2. Downloaded the MYSQL Server and Workbench
3. Made sure "MYSQL80" service is running in background services
4. Imported the sql file in the workbench and ran it to create "classicmodels" database
5. Downloaded the mysql connector/net and installed it.

2. Power BI: Data Cleaning and Loading

Get Data

All
File
Database
Microsoft Fabric (Preview)
Power Platform
Azure

All
IBM Db2 database
IBM Informix database (Beta)
IBM Netezza
MySQL database
PostgreSQL database
Sybase database

MySQL database

Server

Database

Advanced options

OK

Cancel

Navigator

Display Options

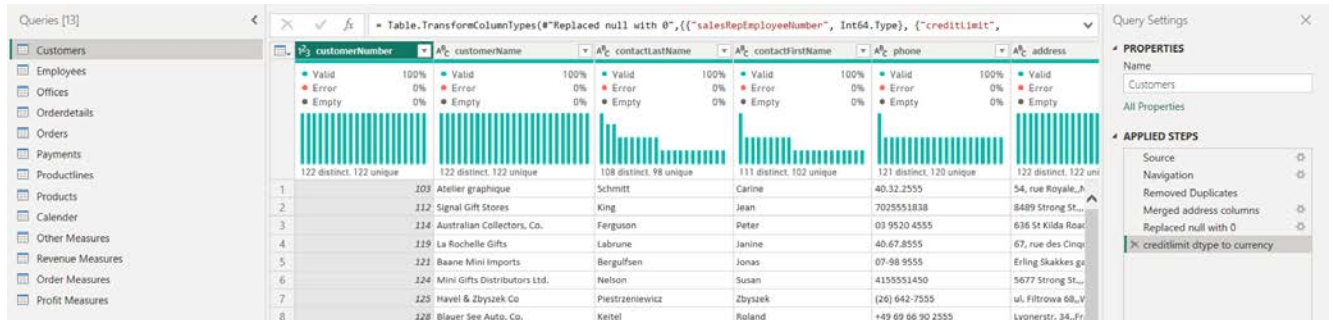
127.0.0.1:3306: classicmodels [8]

- classicmodels.customers
- classicmodels.employees
- classicmodels.offices
- classicmodels.orderdetails
- classicmodels.orders
- classicmodels.payments
- classicmodels.productlines
- classicmodels.products

classicmodels.payments
Preview downloaded on Thursday, 23 November, 2023

| customerNumber | checkNumber | paymentDate | amount | classicmodels.custo |
|----------------|-------------|-------------|----------|---------------------|
| 103 | HQ336336 | 19-10-2004 | 6066.78 | Value |
| 103 | JM555205 | 05-06-2003 | 14571.44 | Value |
| 103 | OM314933 | 18-12-2004 | 1676.14 | Value |
| 112 | BO864823 | 17-12-2004 | 14191.12 | Value |
| 112 | HQ55022 | 06-06-2003 | 32641.98 | Value |
| 112 | ND748579 | 20-08-2004 | 33347.88 | Value |
| 114 | GG31455 | 20-05-2003 | 45864.03 | Value |
| 114 | MA765515 | 15-12-2004 | 82261.22 | Value |
| 114 | NP603840 | 31-05-2003 | 7565.08 | Value |
| 114 | NR27552 | 10-03-2004 | 44894.74 | Value |
| 119 | DR933704 | 14-11-2004 | 19501.82 | Value |

1. Downloaded and installed the Power BI Desktop.
2. Get Data → MySQL Database → Server and Database options
3. Fetched the data from "classicmodels" database hosted on MySQL server
4. We could see all the tables on Navigator screen → Transform Data



5. In Power Query Editor, Used column profiling tools such as "Column Quality and Column Distribution" to identify the error and empty/missing values. Also checked the unique and distinct values.
6. Applied "First Column as Headers", removed the duplicates, merged address columns, and replaced text null values with "NA" and numerical null values with "0". Also, assigned the correct dtype to columns such as fixed decimal number for currency.
7. Finally once the data is cleaned, it was loaded into the Power BI front end/data model

3. Power BI: Creating of Calendar(Date) Table



FileHomeHelp

Table tools

Column tools

Name

Calendar

Mark as date table

Calendars

Manage relationships

Relationships

New measure

Quick measure

New column

New table

Calculations

Structure

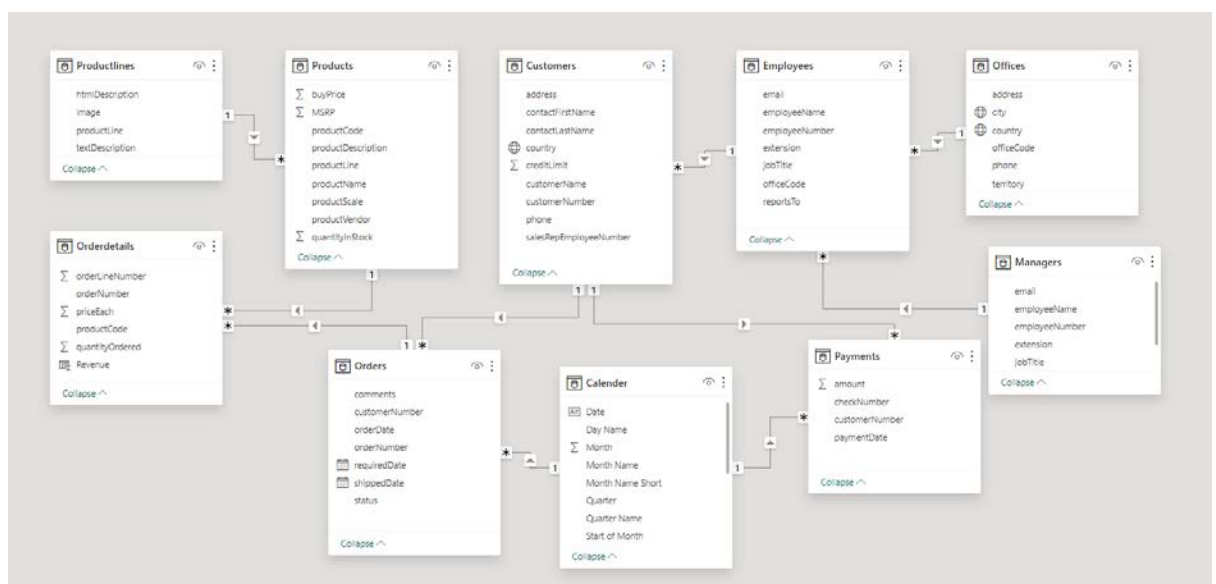
×

✓

| Date | Day Name | Quarter | Month Name | Year | Month Name Short | Quarter Name | Weekend | Year Month | Month | Start of Month | Start of Week |
|-----------------------------|-----------|---------|------------|------|------------------|--------------|---------|------------|-------|---------------------------|---------------------------|
| Thursday, 1 January, 2004 | Thursday | 1 | January | 2004 | Jan | Q1 | Weekday | Jan 2004 | 1 | Thursday, 1 January, 2004 | Monday, 29 December, 2003 |
| Friday, 2 January, 2004 | Friday | 1 | January | 2004 | Jan | Q1 | Weekday | Jan 2004 | 1 | Thursday, 1 January, 2004 | Monday, 29 December, 2003 |
| Monday, 5 January, 2004 | Monday | 1 | January | 2004 | Jan | Q1 | Weekday | Jan 2004 | 1 | Thursday, 1 January, 2004 | Monday, 5 January, 2004 |
| Tuesday, 6 January, 2004 | Tuesday | 1 | January | 2004 | Jan | Q1 | Weekday | Jan 2004 | 1 | Thursday, 1 January, 2004 | Monday, 5 January, 2004 |
| Wednesday, 7 January, 2004 | Wednesday | 1 | January | 2004 | Jan | Q1 | Weekday | Jan 2004 | 1 | Thursday, 1 January, 2004 | Monday, 5 January, 2004 |
| Thursday, 8 January, 2004 | Thursday | 1 | January | 2004 | Jan | Q1 | Weekday | Jan 2004 | 1 | Thursday, 1 January, 2004 | Monday, 5 January, 2004 |
| Friday, 9 January, 2004 | Friday | 1 | January | 2004 | Jan | Q1 | Weekday | Jan 2004 | 1 | Thursday, 1 January, 2004 | Monday, 5 January, 2004 |
| Monday, 12 January, 2004 | Monday | 1 | January | 2004 | Jan | Q1 | Weekday | Jan 2004 | 1 | Thursday, 1 January, 2004 | Monday, 12 January, 2004 |
| Tuesday, 13 January, 2004 | Tuesday | 1 | January | 2004 | Jan | Q1 | Weekday | Jan 2004 | 1 | Thursday, 1 January, 2004 | Monday, 12 January, 2004 |
| Wednesday, 14 January, 2004 | Wednesday | 1 | January | 2004 | Jan | Q1 | Weekday | Jan 2004 | 1 | Thursday, 1 January, 2004 | Monday, 12 January, 2004 |
| Thursday, 15 January, 2004 | Thursday | 1 | January | 2004 | Jan | Q1 | Weekday | Jan 2004 | 1 | Thursday, 1 January, 2004 | Monday, 12 January, 2004 |
| Friday, 16 January, 2004 | Friday | 1 | January | 2004 | Jan | Q1 | Weekday | Jan 2004 | 1 | Thursday, 1 January, 2004 | Monday, 12 January, 2004 |

1. After loading the cleaned data in model, we found there was no separate date table.
2. Selected a blank query in Power Query Editor and created a list of dates by providing the start and end date. These ranges covers the dates in the data model. Also, the calendar should have full dates starting from 1st of January to 31st of December.
3. Converted the list to a table format and changed the dtype to "Date".
4. Under Add Column tab, created many columns such as "Day Name", "Start of Month", "Quarter", "Year", "Start of Week", "Weekday/Weekend" etc.
5. Finally applied it to the data model and "Marked it as Official Data Table" in Data View.

4. Power BI: Creating Relationships



1. Identified that there are :
3 Fact/Data Tables – Order, Order Details, Payment
3 Dimension/Lookup Tables – Product, Customer, Employee
2 Sub Dimension Tables – ProductLines, Offices
1 Date Table – Calendar
2. Connected all foreign keys with their respective Primary keys to create multiple 1-to-Many relationships between the tables in the Manage Relationships in section.
3. This led to imitate the tables just like in a relational database. The relationships were falling in the “Snow Flake Schema” paradigm.
4. To eliminate the possibility of confusion/ambiguity, we hide the foreign key columns from the end user visibility so that they select only primary key columns to slice and dice the data while visualizing and reducing the need to for bi-direction cross filtering.
5. We also avoiding creating any Many-to-Many relationship wherever possible.


5. Power BI: Using DAX to explore and analyze data

| Order Measures | Other Measures | Revenue Measures | Orderdetails |
|--|--|---|---|
| <input type="checkbox"/> Average Orders per Customer | <input type="checkbox"/> Cost YTD | <input type="checkbox"/> Adjusted Revenue | <input type="checkbox"/> \sum orderLineNumber |
| <input type="checkbox"/> Average Orders per Employee | <input type="checkbox"/> Total Cost | <input type="checkbox"/> Average Revenue | <input type="checkbox"/> orderNumber |
| <input type="checkbox"/> Cancelled Orders | <input type="checkbox"/> Total Customers | <input type="checkbox"/> Average Revenue per Customer | <input type="checkbox"/> \sum priceEach |
| <input type="checkbox"/> Disputed Orders | <input type="checkbox"/> Total Debt | <input type="checkbox"/> Average Revenue per Employee | <input type="checkbox"/> productCode |
| <input type="checkbox"/> In Process Orders | <input type="checkbox"/> Total Employees | <input type="checkbox"/> Prev Month Revenue | <input type="checkbox"/> \sum quantityOrdered |
| <input type="checkbox"/> On-Hold Orders | <input type="checkbox"/> Total Payment | <input type="checkbox"/> Prev Quarter Revenue | <input type="checkbox"/> Revenue |
| <input type="checkbox"/> Order Target | <input type="checkbox"/> Total Products | <input type="checkbox"/> Prev Year Revenue | |
| <input type="checkbox"/> Order Target Gap | <input type="checkbox"/> Profit Measures | <input type="checkbox"/> Revenue MoM% | |
| <input type="checkbox"/> Orders YTD | <input type="checkbox"/> Adjusted Profit | <input type="checkbox"/> Revenue MTD | |
| <input type="checkbox"/> Prev Year Orders | <input type="checkbox"/> Prev Year Profit | <input type="checkbox"/> Revenue QoQ% | |
| <input type="checkbox"/> Resolved Orders | <input type="checkbox"/> Profit Margin % | <input type="checkbox"/> Revenue QTD | |
| <input type="checkbox"/> Shipped Orders | <input type="checkbox"/> Profit Margin YTD | <input type="checkbox"/> Revenue Target | |
| <input type="checkbox"/> Total Orders | <input type="checkbox"/> Profit Target | <input type="checkbox"/> Revenue Target Gap | |
| | <input type="checkbox"/> Profit Target Gap | <input type="checkbox"/> Revenue YoY% | |
| | <input type="checkbox"/> Profit YTD | <input type="checkbox"/> Revenue YTD | |
| | <input type="checkbox"/> Total Profit | <input type="checkbox"/> Total Revenue | |

1. Using Data Analysis Expressions, I created calculated columns like “Revenue”. I know we should limit creating these columns as they can be redundant and also contribute to file size.
2. Hence I created Measures using DAX to calculate numerous metrics like Total Revenue, Total Cost, Total Profit, Total Orders, Profit Margin% etc.
3. Made use of Time Intelligence functions to create measures like Sales YTD, MTD, QTD, Sales YoY%, Profit MoM%, Target metrics for performance analysis etc.
4. Also created measures for performing the “What-If” Analysis for experiencing the change in Profit by adjusting the selling price etc.

6. Power BI: Visualizing the sales data via reports, charts & Dashboards

1. Home Page



PROBLEM
Axon is a small retailer selling diecast classic cars. It is facing issues in managing and analyzing their sales data. The sales team is struggling to make sense of the data and they do not have a centralized system to manage and analyze the data. The management is unable to get accurate and up-to-date sales reports, which is affecting the decision-making process.

SOLUTION
Use the data source provided: Use the MySQL database provided as a data source.
Extract and clean the data: The next step is to extract the data from the identified sources and clean it to make it ready for analysis. This may involve tasks such as removing duplicates, handling missing values, and ensuring data consistency.
Load the data into a PowerBI: The cleaned data can then be loaded into a centralized database.
Design the dashboards and reports: Using PowerBI, data can be visualized in the form of interactive dashboards and reports. These dashboards and reports can be designed to provide useful insights and information to the management.

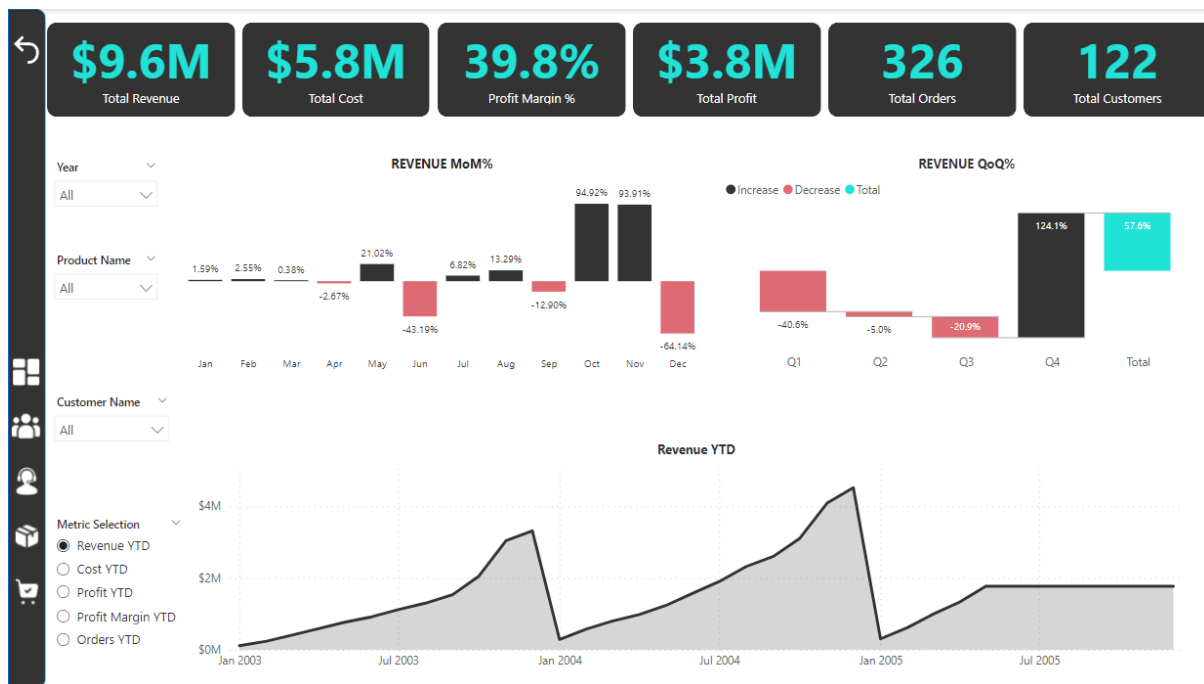
Sales Dashboard Customer Details Order Details
Employee Details Product Details

→

By Vibhore Aggarwal (S5289)

Home Page is the landing page where we have company logo, problem statement and solution and navigation menus to traverse to different report pages.

2. Sales Dashboard



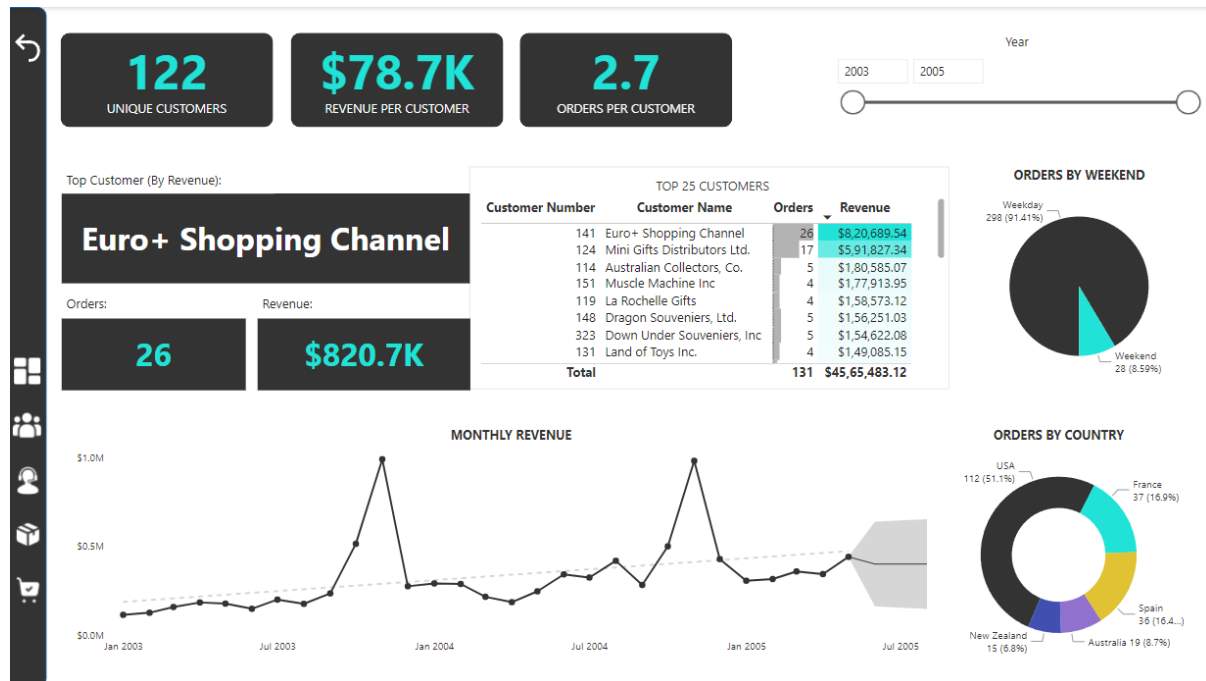
Sales Dashboard is for the Sales management to view the Total Revenue, Total Cost, Total Profit, Profit Margin%, Total Orders and Total customers at a glance.

They analyze sales growth Month over Month and Quarter over Quarter. Select the metrics to view the cumulative values Year To-Date for revenue, cost, profit, orders etc.

On the top of it they can slice and dice the whole data at year level, product level and customer level.

There is a trend that October and November month see a great growth in sales of 93% plus back to back. We must make sure of the inventory and delivery to carry out smooth operations during the peak seasons.

3. Customer Details



Customer details page take a jibe at the customer level details like Average revenue per customer, Average orders per customer and total unique customers.

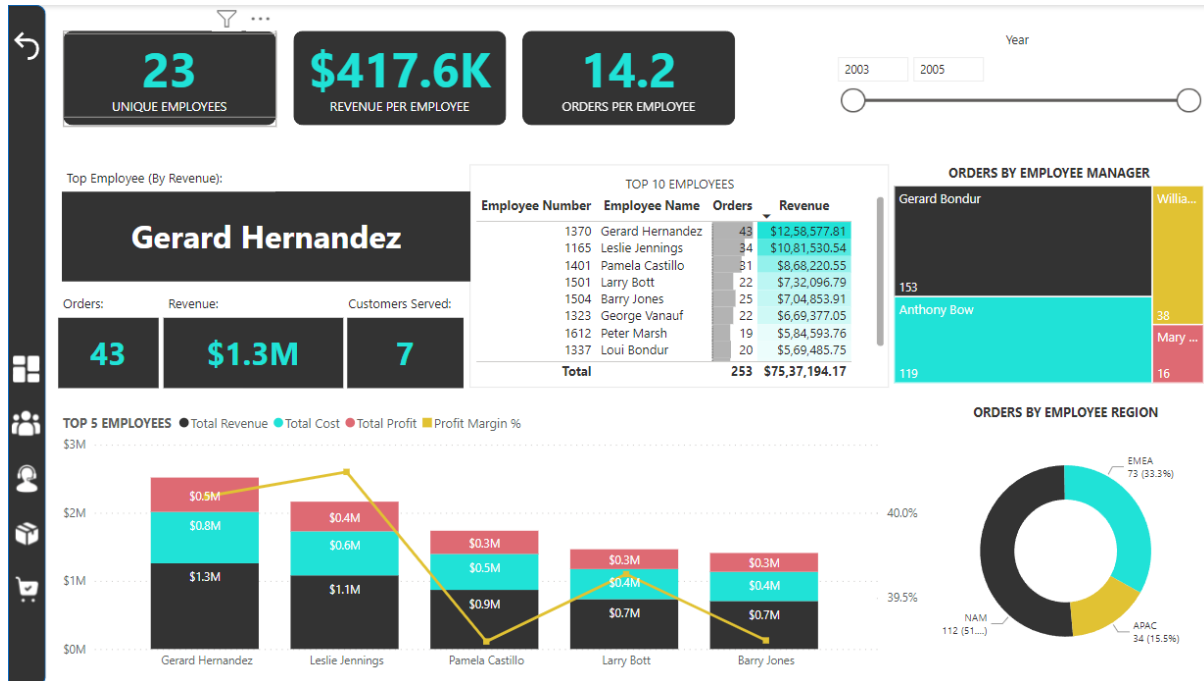
It highlights the Top Customer by revenue, there total orders and revenue provided.

Revenue is also divided on monthly basis to give a picture of monthly engagement.

We can the breakdown of orders by Customer Country from where they belong from and do they tend to order more over weekend or not.

Here is list of top 25 Customers by revenue which the company should keep them engaged so that they keep coming back.

4. Employee Details



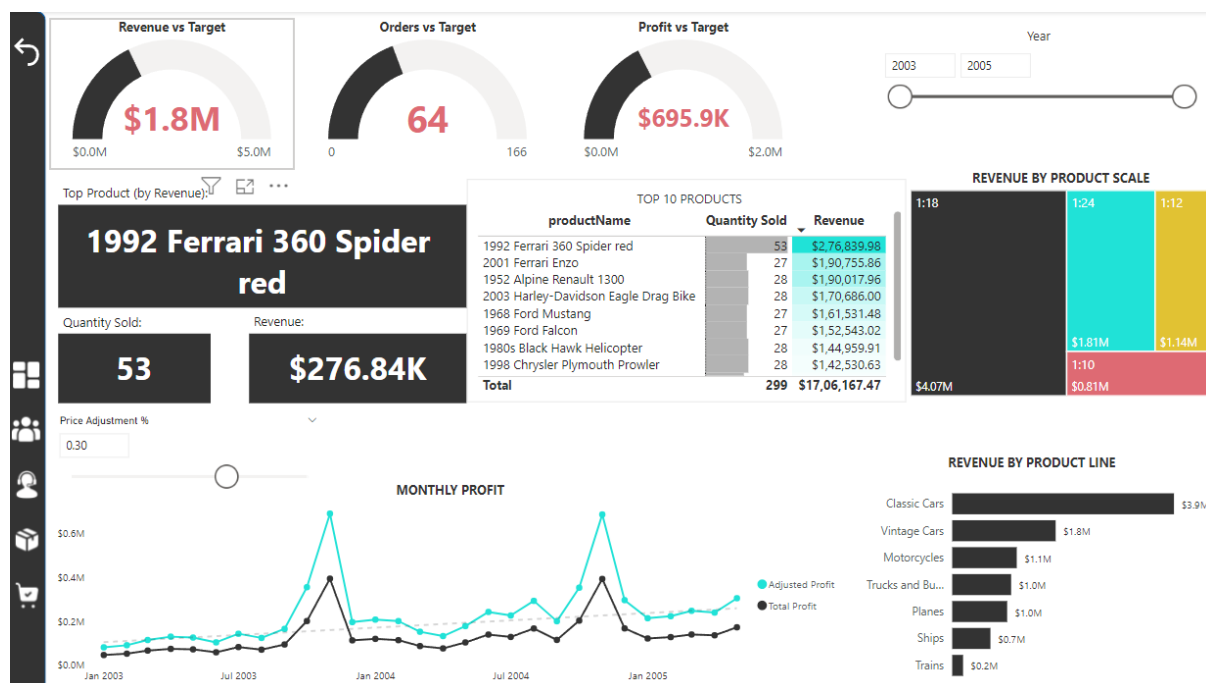
Employee Details report page shows the important employee level metric at a glance like Total unique employees, Average Revenue grossed per employee and average orders created per employee.

It highlights the top employee by revenue, number of orders created, number of customers served and total revenue grossed.

We can compare the Total Revenue, Cost and Profit and margin% for top 5 employees in the company.

Most employees from EMEA territory region created the most number of orders and grossed highest revenue. These sales persons are directly managed by Gerard Bondur.

5. Product Details



Product details report page clearly shows the metrics against their respective targets.

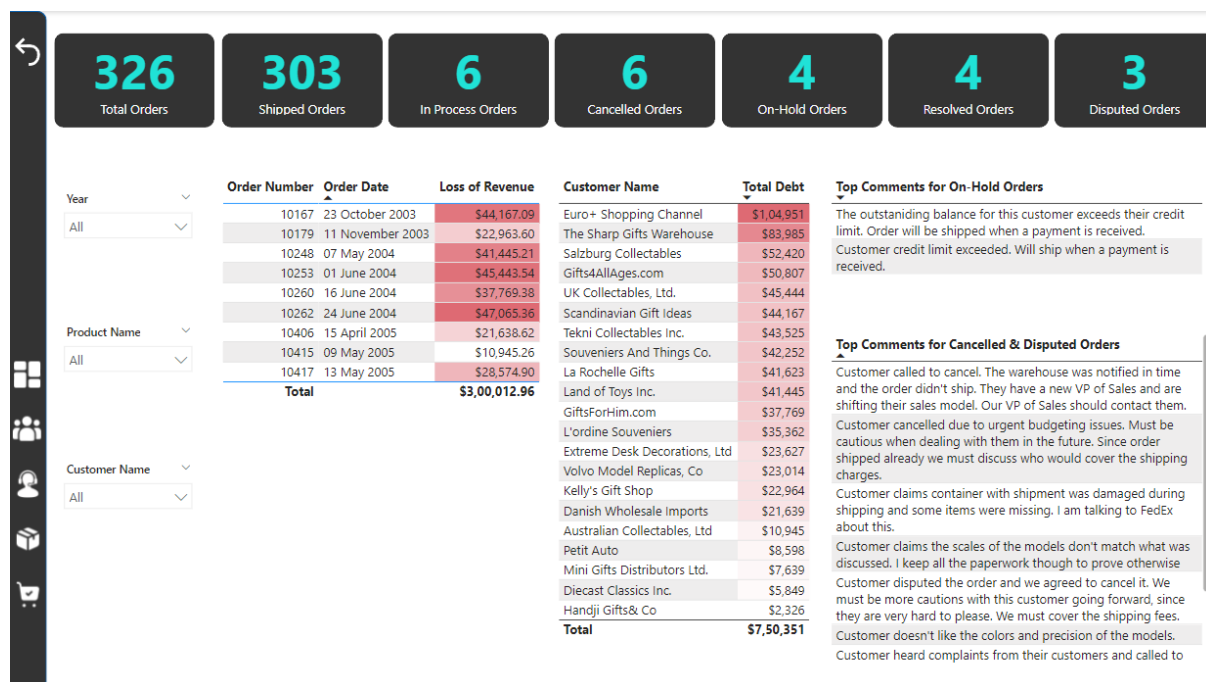
Target are defined as the previous month's performance with an increment of 10% growth. This helps the management to take decisions whether the business is growing in positive direction or not.

Targets like revenue, orders and profit are crucial for the company future aspects and growth.

We can also closely monitor the change in Profit by adjusting the product price. This is called what-If analysis and it helps management decide to whether drop or increase the product price on monthly trend.

Lastly, 1:18 scale die-cast cars falling under classic cars product line are the bestselling category which is bringing good business and profits to the company.

6. Order Details



Order details report page gives holistic breakdown of all the orders based upon the status like "Shipped", "In-Process", "Cancelled", "On-Hold", "Resolved" and "Disputed" orders.

It highlights the top comments on why the orders were cancelled or disputed or on-hold by the customers. By going through these comments taking them as a feedback for the scope for improvement in the company operations. We need to meet the growing demand and expectations of the vendors and the customers.

We can also take a look at the revenue lost by the company due to cancelled and disputed orders and how we can improve upon them.

Also, few customers have also been upon debt by the company. They have exhausted their credit limits set by the company. Once they settle the debt then only they are eligible for further credits.