Data Analyst | Statistics

2019 - 2023

portfolio.

Muhammad Nafis Sunaryo

hello.

My name is **Muhammad Nafis Sunaryo**

Nice to meet you

Muhammad Nafis Sunaryo

Data Analyst | Business Analyst

I am a recent graduate with a degree in Statistics. Have enthusiasm and strong interest in data analysis. with analytical skills, problem solving ability, fast learning ability, and dedication

I learned a lot about data while in college and I am very ready to help your company with data, creating dashboards, helping in making decisions, and ensuring teams work well together

About Me

After graduation, I have also successfully completed several projects, mastered the use of various data analysis tools, and coordinated various project elements smoothly. Currently, I am looking for new opportunities to develop skills and experience in the field of data analysis. I am ready to face bigger challenges and contribute to the world of data analysis.

My Vision

“

Develop skills and become a true inspiration for data analyst

Education

2019 - 2023

Brawijaya University

Bachelor of Statistics

Brawijaya University is a state campus located in Malang City and based on the world university website ranking agency 4ICU, in 2023 UB will be ranked 5th in Indonesia and 756th in the world.

2023

Coding Studio

Online Bootcamp

Coding Studio is an online-based academy that prepares digital talents throughout Indonesia for technology experts who are currently most in demand by companies

Expertise Skill

SQL

Python

Excel

Power BI

This project was assisted using SQL Workbench, Python, Excel, and Power BI tools. With different case studies but the aim is to master the four data analysis tools

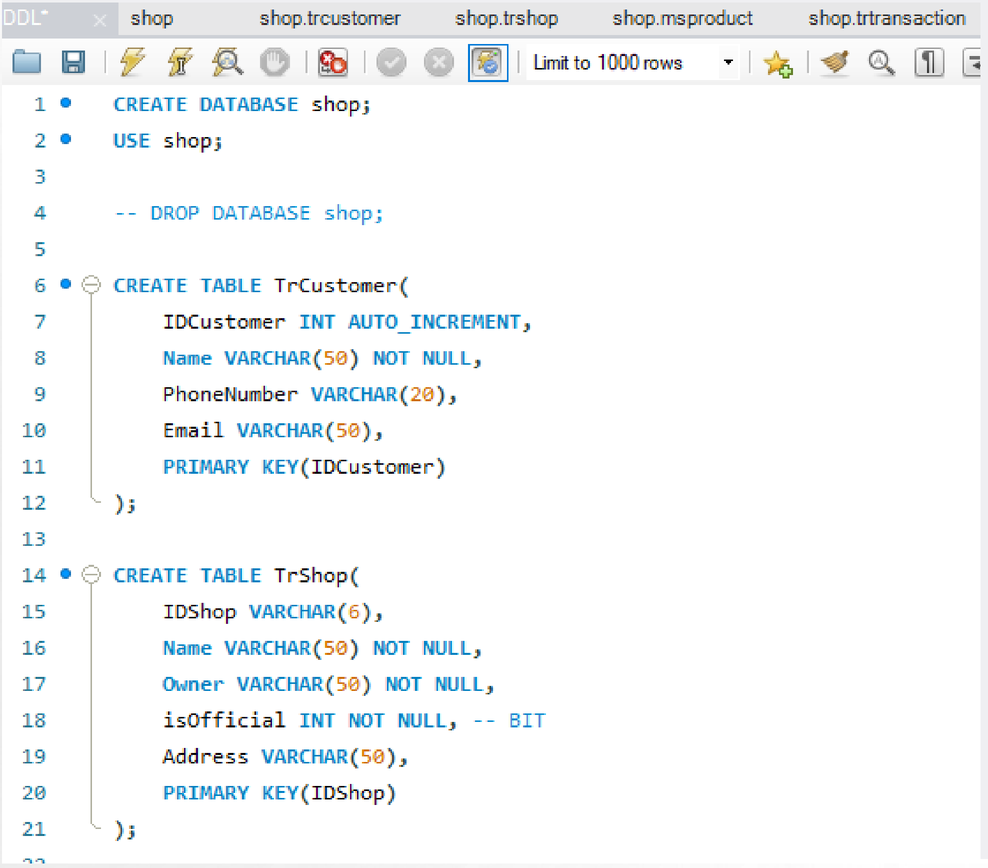
PROJECT 01

SQL database.

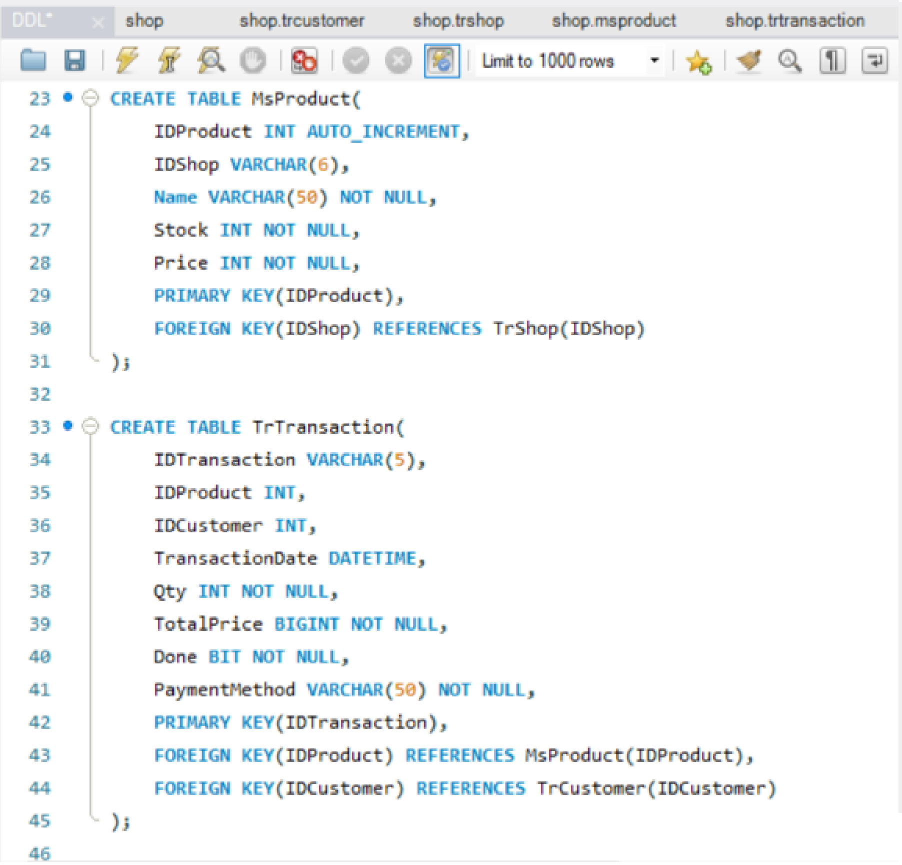
SQL WorkBench

Create tables and Insert data

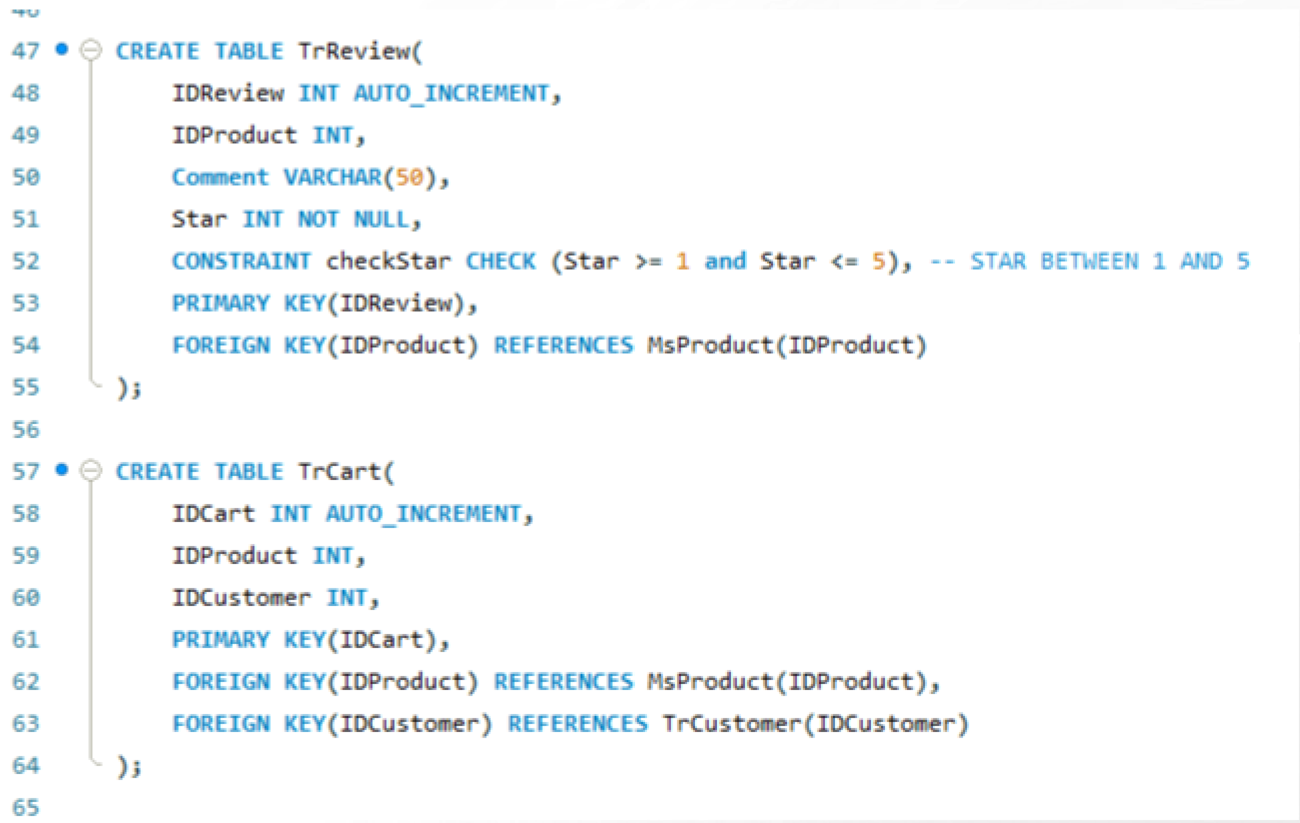
Syntax Create Table



```sql
CREATE DATABASE shop;
USE shop;

-- DROP DATABASE shop;

CREATE TABLE TrCustomer(
    IDCustomer INT AUTO_INCREMENT,
    Name VARCHAR(50) NOT NULL,
    PhoneNumber VARCHAR(20),
    Email VARCHAR(50),
    PRIMARY KEY(IDCustomer)
);

CREATE TABLE TrShop(
    IDShop VARCHAR(6),
    Name VARCHAR(50) NOT NULL,
    Owner VARCHAR(50) NOT NULL,
    isOfficial INT NOT NULL, -- BIT
    Address VARCHAR(50),
    PRIMARY KEY(IDShop)
);
```

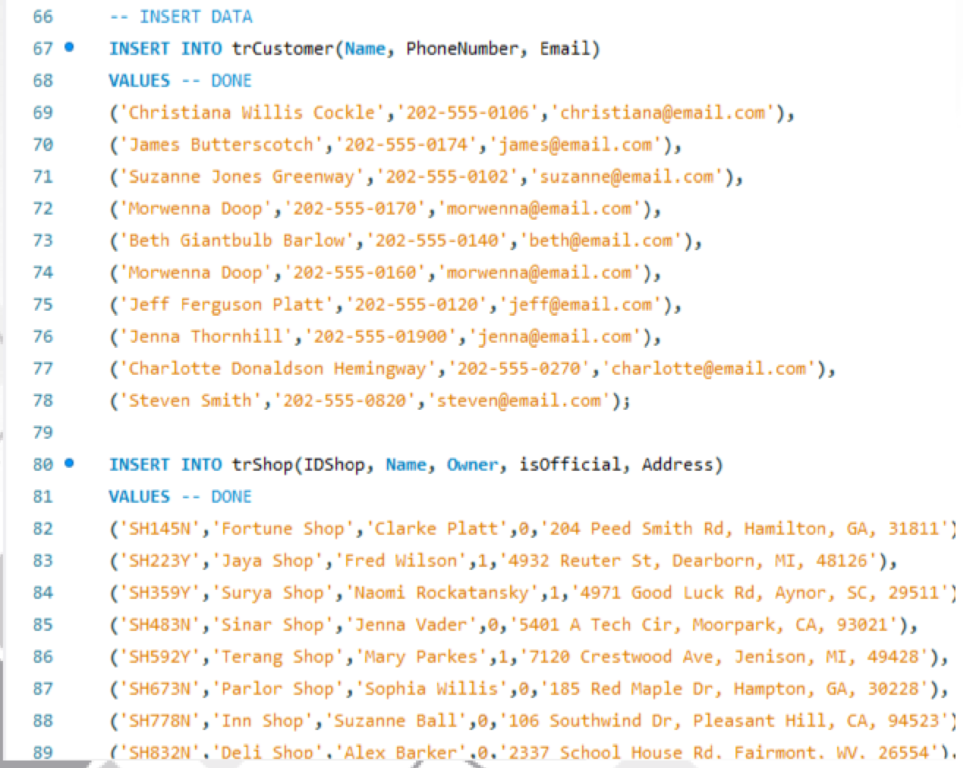


```sql
CREATE TABLE MsProduct(
    IDProduct INT AUTO_INCREMENT,
    IDShop VARCHAR(6),
    Name VARCHAR(50) NOT NULL,
    Stock INT NOT NULL,
    Price INT NOT NULL,
    PRIMARY KEY(IDProduct),
    FOREIGN KEY(IDShop) REFERENCES TrShop(IDShop)
);

CREATE TABLE TrTransaction(
    IDTransaction VARCHAR(5),
    IDProduct INT,
    IDCustomer INT,
    TransactionDate DATETIME,
    Qty INT NOT NULL,
    TotalPrice BIGINT NOT NULL,
    Done BIT NOT NULL,
    PaymentMethod VARCHAR(50) NOT NULL,
    PRIMARY KEY(IDTransaction),
    FOREIGN KEY(IDProduct) REFERENCES MsProduct(IDProduct),
    FOREIGN KEY(IDCustomer) REFERENCES TrCustomer(IDCustomer)
);
```

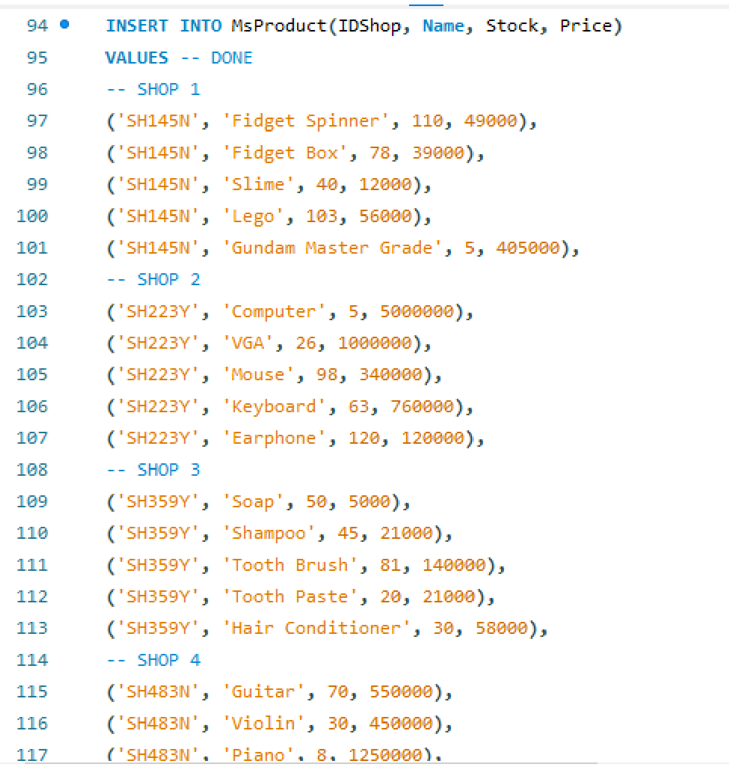


```sql
CREATE TABLE TrReview(
    IDReview INT AUTO_INCREMENT,
    IDProduct INT,
    Comment VARCHAR(50),
    Star INT NOT NULL,
    CONSTRAINT checkStar CHECK (Star >= 1 and Star <= 5), -- STAR BETWEEN 1 AND 5
    PRIMARY KEY(IDReview),
    FOREIGN KEY(IDProduct) REFERENCES MsProduct(IDProduct)
);

CREATE TABLE TrCart(
    IDCart INT AUTO_INCREMENT,
    IDProduct INT,
    IDCustomer INT,
    PRIMARY KEY(IDCart),
    FOREIGN KEY(IDProduct) REFERENCES MsProduct(IDProduct),
    FOREIGN KEY(IDCustomer) REFERENCES TrCustomer(IDCustomer)
);
```

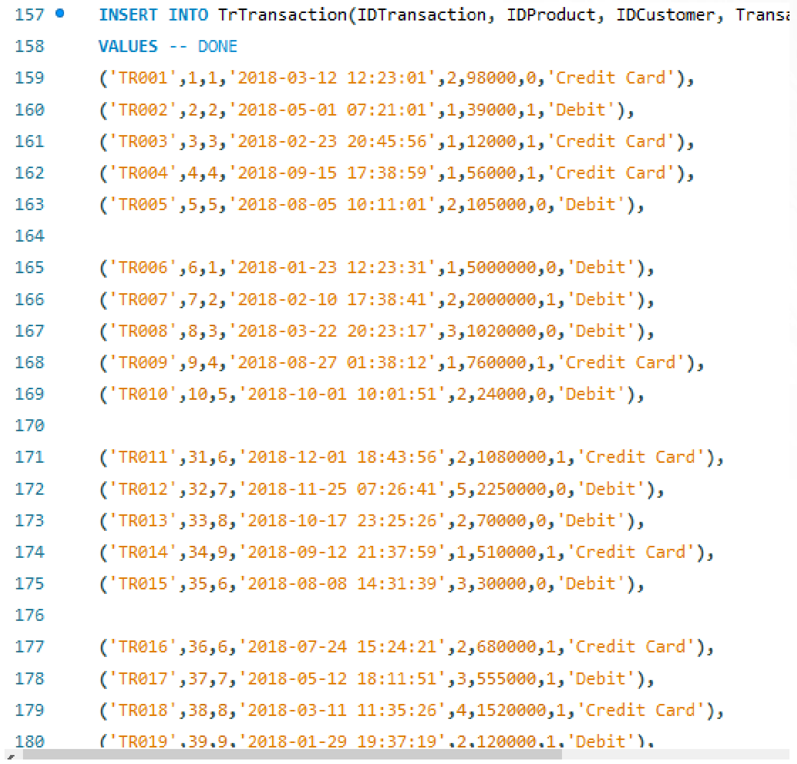
Syntax Insert Data



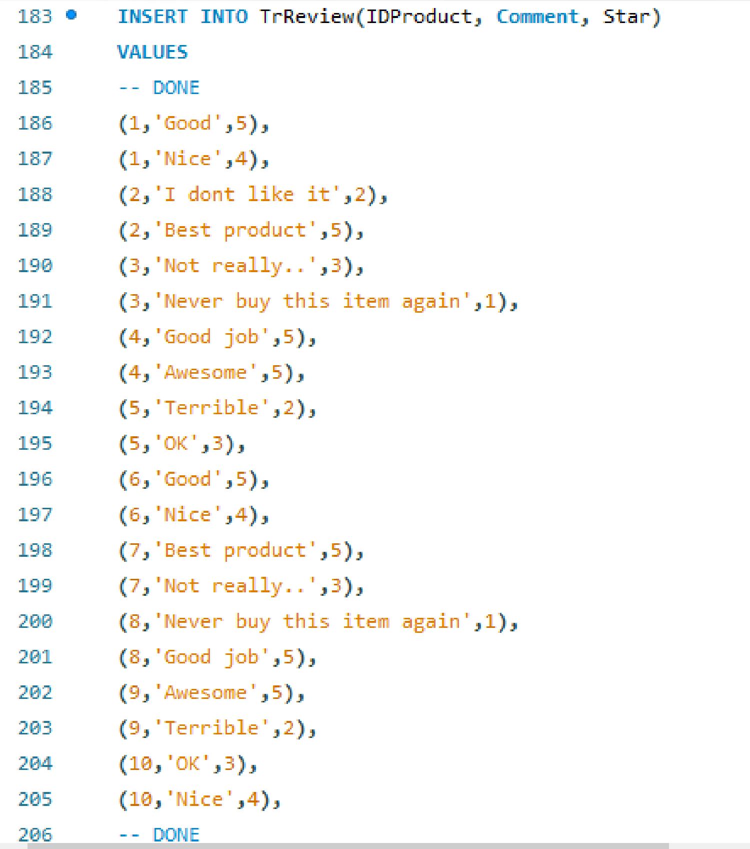
```sql
-- INSERT DATA
INSERT INTO trCustomer(Name, PhoneNumber, Email)
VALUES -- DONE
('Christiana Willis Cockle','202-555-0106','christiana@email.com'),
('James Butterscotch','202-555-0174','james@email.com'),
('Suzanne Jones Greenway','202-555-0102','suzanne@email.com'),
('Morwenna Doop','202-555-0170','morwenna@email.com'),
('Beth Giantbulb Barlow','202-555-0140','beth@email.com'),
('Morwenna Doop','202-555-0160','morwenna@email.com'),
('Jeff Ferguson Platt','202-555-0120','jeff@email.com'),
('Jenna Thornhill','202-555-01900','jenna@email.com'),
('Charlotte Donaldson Hemingway','202-555-0270','charlotte@email.com'),
('Steven Smith','202-555-0820','steven@email.com');

INSERT INTO trShop(IDShop, Name, Owner, isOfficial, Address)
VALUES -- DONE
('SH145N','Fortune Shop','Clarke Platt',0,'204 Peed Smith Rd, Hamilton, GA, 31811')
('SH223Y','Jaya Shop','Fred Wilson',1,'4932 Reuter St, Dearborn, MI, 48126'),
('SH359Y','Surya Shop','Naomi Rockatansky',1,'4971 Good Luck Rd, Aynor, SC, 29511')
('SH483N','Sinar Shop','Jenna Vader',0,'5401 A Tech Cir, Moorpark, CA, 93021'),
('SH592Y','Terang Shop','Mary Parkes',1,'7120 Crestwood Ave, Jenison, MI, 49428'),
('SH673N','Parlor Shop','Sophia Willis',0,'185 Red Maple Dr, Hampton, GA, 30228'),
('SH778N','Inn Shop','Suzanne Ball',0,'106 Southwind Dr, Pleasant Hill, CA, 94523')
('SH832N','Deli Shop','Alex Barker',0,'2337 School House Rd, Fairmont, WV, 26554'),
```



```sql
INSERT INTO MsProduct(IDShop, Name, Stock, Price)
VALUES -- DONE
-- SHOP 1
('SH145N', 'Fidget Spinner', 110, 49000),
('SH145N', 'Fidget Box', 78, 39000),
('SH145N', 'Slime', 40, 12000),
('SH145N', 'Lego', 103, 56000),
('SH145N', 'Gundam Master Grade', 5, 405000),
-- SHOP 2
('SH223Y', 'Computer', 5, 5000000),
('SH223Y', 'VGA', 26, 1000000),
('SH223Y', 'Mouse', 98, 340000),
('SH223Y', 'Keyboard', 63, 760000),
('SH223Y', 'Earphone', 120, 120000),
-- SHOP 3
('SH359Y', 'Soap', 50, 5000),
('SH359Y', 'Shampoo', 45, 21000),
('SH359Y', 'Tooth Brush', 81, 140000),
('SH359Y', 'Tooth Paste', 20, 21000),
('SH359Y', 'Hair Conditioner', 30, 58000),
-- SHOP 4
('SH483N', 'Guitar', 70, 550000),
('SH483N', 'Violin', 30, 450000),
('SH483N', 'Piano', 8, 1250000),
```



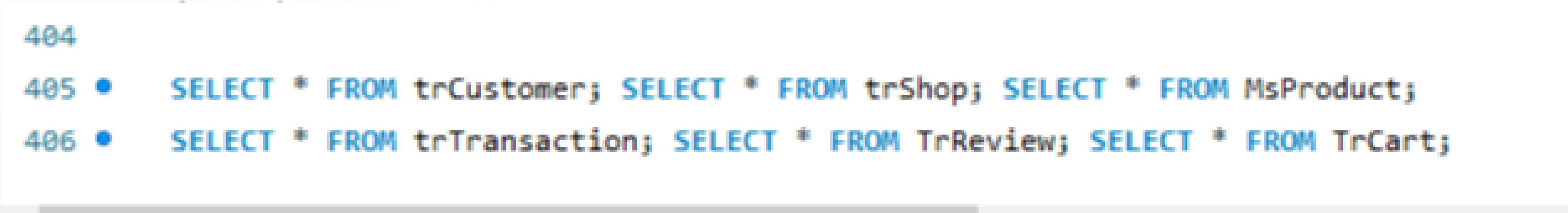
```sql
INSERT INTO TrTransaction(IDTransaction, IDProduct, IDCustomer, Transa
VALUES -- DONE
('TR001',1,1,'2018-03-12 12:23:01',2,98000,0,'Credit Card'),
('TR002',2,2,'2018-05-01 07:21:01',1,39000,1,'Debit'),
('TR003',3,3,'2018-02-23 20:45:56',1,12000,1,'Credit Card'),
('TR004',4,4,'2018-09-15 17:38:59',1,56000,1,'Credit Card'),
('TR005',5,5,'2018-08-05 10:11:01',2,105000,0,'Debit'),

('TR006',6,1,'2018-01-23 12:23:31',1,5000000,0,'Debit'),
('TR007',7,2,'2018-02-10 17:38:41',2,2000000,1,'Debit'),
('TR008',8,3,'2018-03-22 20:23:17',3,1020000,0,'Debit'),
('TR009',9,4,'2018-08-27 01:38:12',1,760000,1,'Credit Card'),
('TR010',10,5,'2018-10-01 10:01:51',2,24000,0,'Debit'),

('TR011',31,6,'2018-12-01 18:43:56',2,1080000,1,'Credit Card'),
('TR012',32,7,'2018-11-25 07:26:41',5,2250000,0,'Debit'),
('TR013',33,8,'2018-10-17 23:25:26',2,70000,0,'Debit'),
('TR014',34,9,'2018-09-12 21:37:59',1,510000,1,'Credit Card'),
('TR015',35,6,'2018-08-08 14:31:39',3,30000,0,'Debit'),

('TR016',36,6,'2018-07-24 15:24:21',2,680000,1,'Credit Card'),
('TR017',37,7,'2018-05-12 18:11:51',3,555000,1,'Debit'),
('TR018',38,8,'2018-03-11 11:35:26',4,1520000,1,'Credit Card'),
('TR019',39,9,'2018-01-29 19:37:19',2,120000,1,'Debit'),
```



```sql
INSERT INTO TrReview(IDProduct, Comment, Star)
VALUES
-- DONE
(1,'Good',5),
(1,'Nice',4),
(2,'I dont like it',2),
(2,'Best product',5),
(3,'Not really..',3),
(3,'Never buy this item again',1),
(4,'Good job',5),
(4,'Awesome',5),
(5,'Terrible',2),
(5,'OK',3),
(6,'Good',5),
(6,'Nice',4),
(7,'Best product',5),
(7,'Not really..',3),
(8,'Never buy this item again',1),
(8,'Good job',5),
(9,'Awesome',5),
(9,'Terrible',2),
(10,'OK',3),
(10,'Nice',4),
-- DONE
```

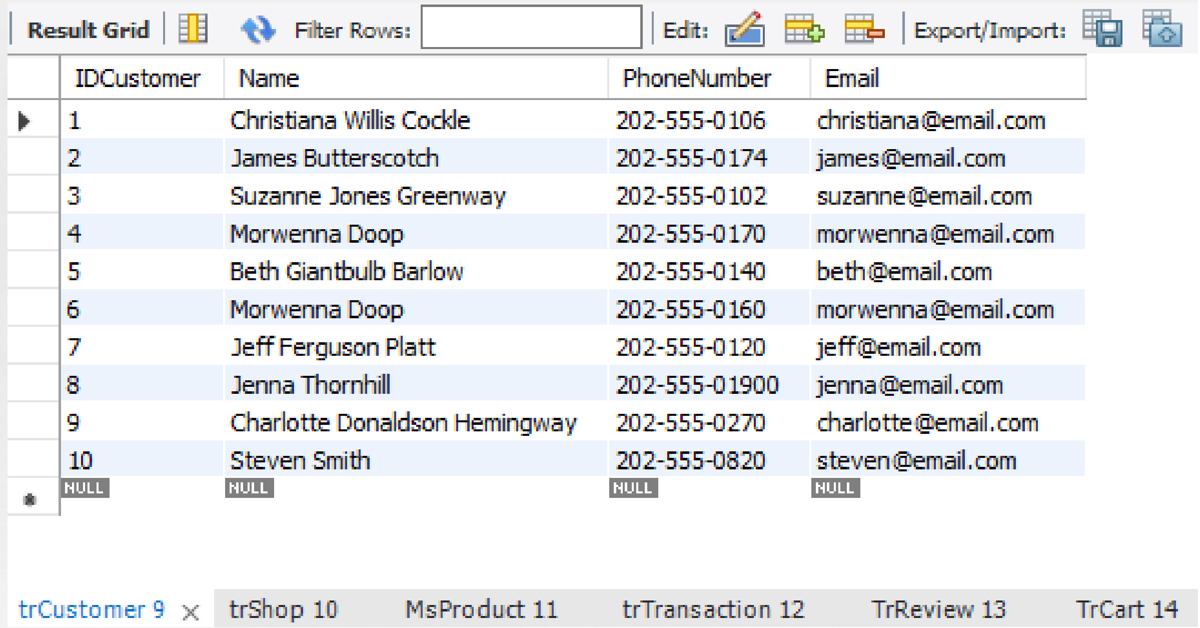
Displays the data that has been entered

Syntax

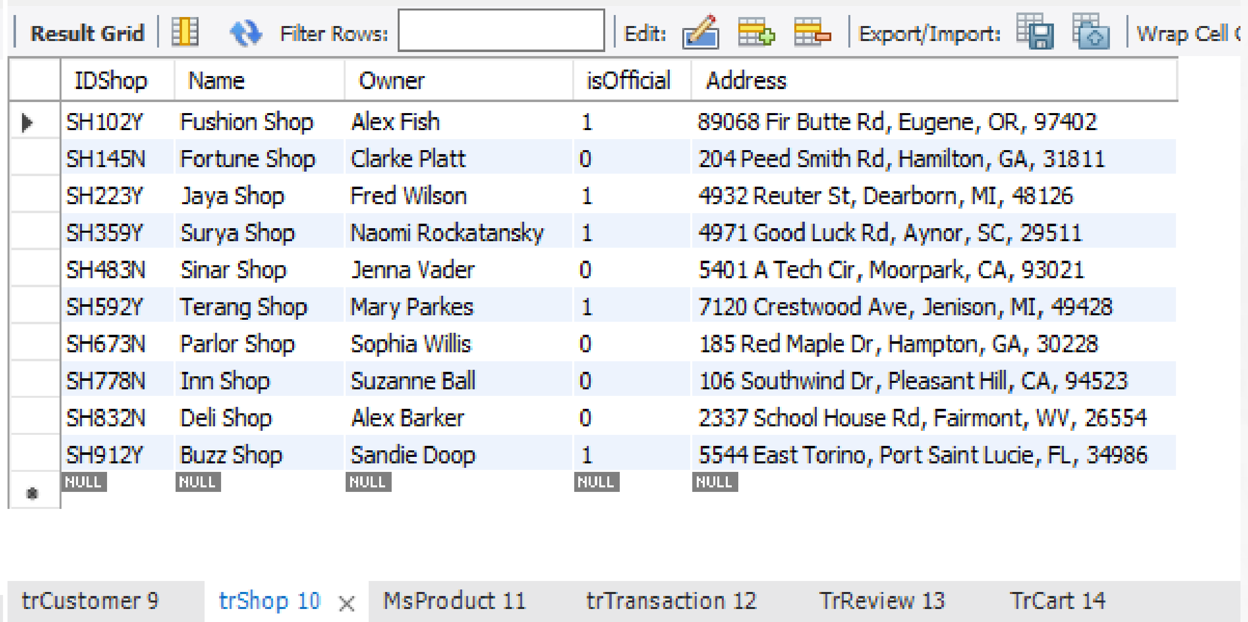


```
SELECT * FROM trCustomer; SELECT * FROM trShop; SELECT * FROM MsProduct;
SELECT * FROM trTransaction; SELECT * FROM TrReview; SELECT * FROM TrCart;
```

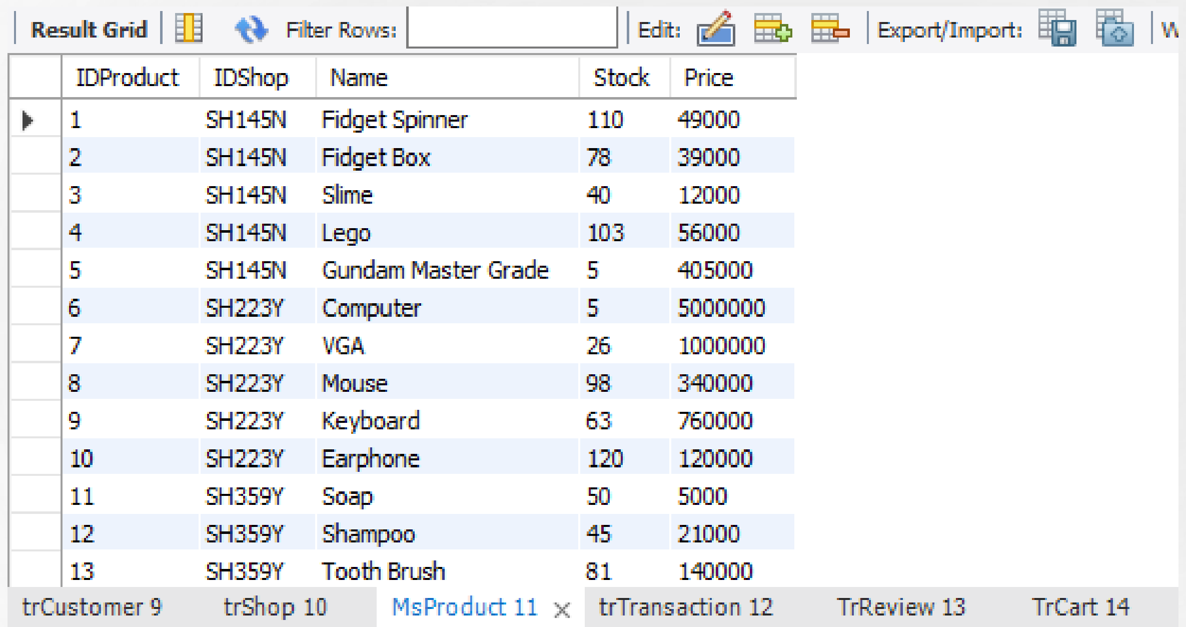
Query Result



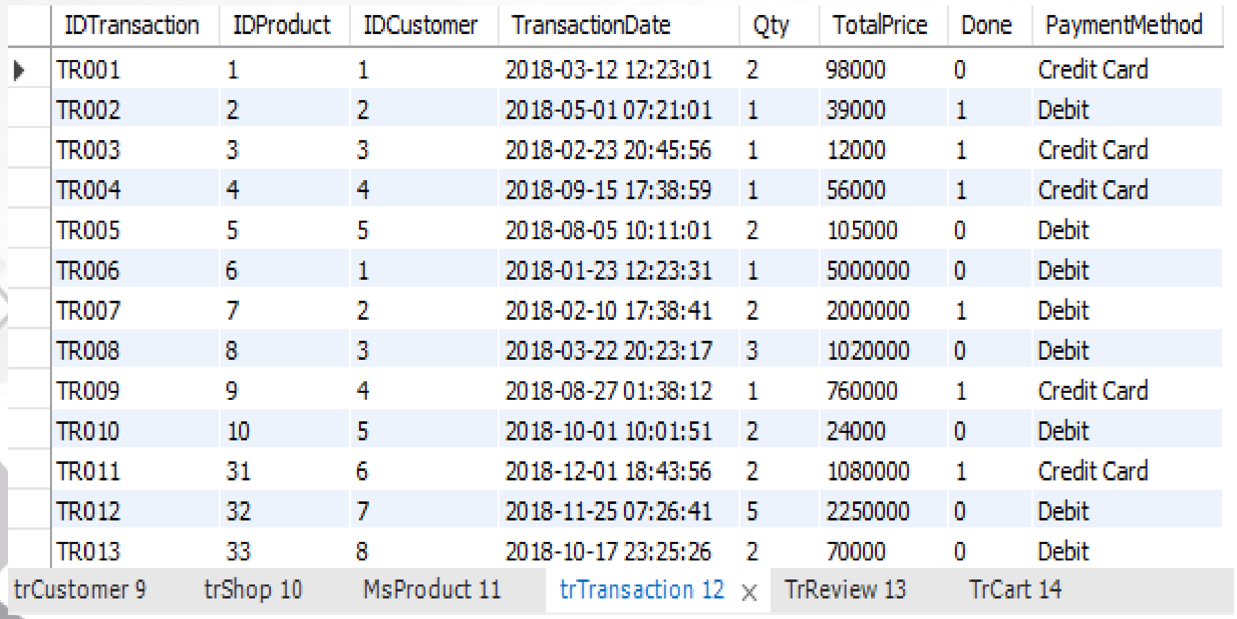
IDCustomer	Name	PhoneNumber	Email
1	Christiana Willis Cockle	202-555-0106	christiana@email.com
2	James Butterscotch	202-555-0174	james@email.com
3	Suzanne Jones Greenway	202-555-0102	suzanne@email.com
4	Morwenna Doop	202-555-0170	morwenna@email.com
5	Beth Giantbulb Barlow	202-555-0140	beth@email.com
6	Morwenna Doop	202-555-0160	morwenna@email.com
7	Jeff Ferguson Platt	202-555-0120	jeff@email.com
8	Jenna Thornhill	202-555-01900	jenna@email.com
9	Charlotte Donaldson Hemingway	202-555-0270	charlotte@email.com
10	Steven Smith	202-555-0820	steven@email.com
NULL	NULL	NULL	NULL



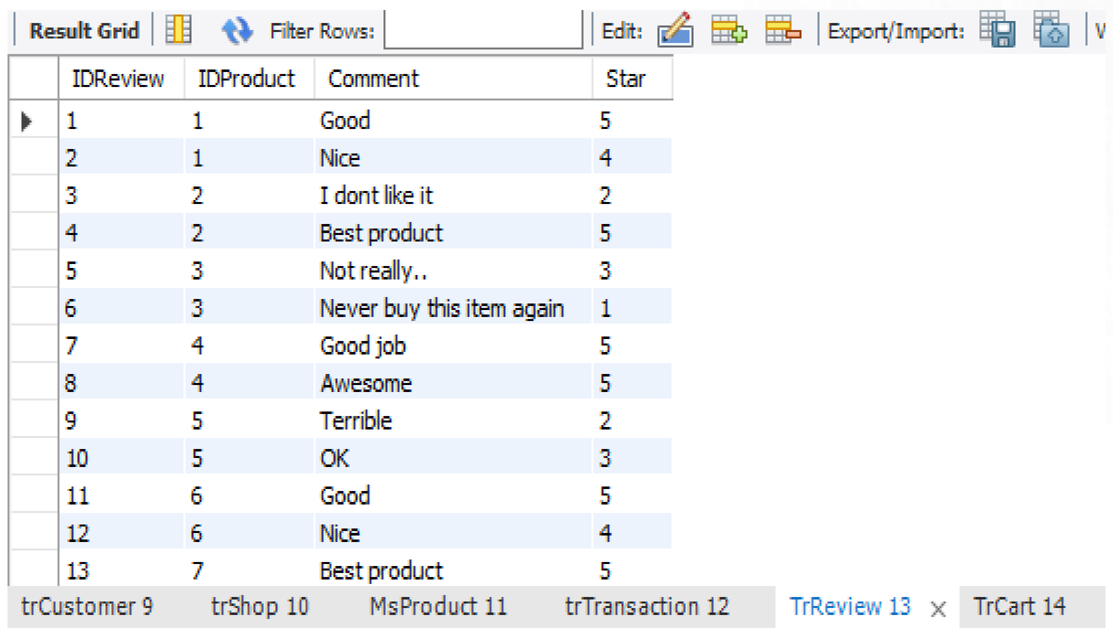
IDShop	Name	Owner	isOfficial	Address
SH102Y	Fushion Shop	Alex Fish	1	89068 Fir Butte Rd, Eugene, OR, 97402
SH145N	Fortune Shop	Clarke Platt	0	204 Peed Smith Rd, Hamilton, GA, 31811
SH223Y	Jaya Shop	Fred Wilson	1	4932 Reuter St, Dearborn, MI, 48126
SH359Y	Surya Shop	Naomi Rockatansky	1	4971 Good Luck Rd, Aynor, SC, 29511
SH483N	Sinar Shop	Jenna Vader	0	5401 A Tech Cir, Moorpark, CA, 93021
SH592Y	Terang Shop	Mary Parkes	1	7120 Crestwood Ave, Jenison, MI, 49428
SH673N	Parlor Shop	Sophia Willis	0	185 Red Maple Dr, Hampton, GA, 30228
SH778N	Inn Shop	Suzanne Ball	0	106 Southwind Dr, Pleasant Hill, CA, 94523
SH832N	Deli Shop	Alex Barker	0	2337 School House Rd, Fairmont, WV, 26554
SH912Y	Buzz Shop	Sandie Doop	1	5544 East Torino, Port Saint Lucie, FL, 34986
NULL	NULL	NULL	NULL	NULL



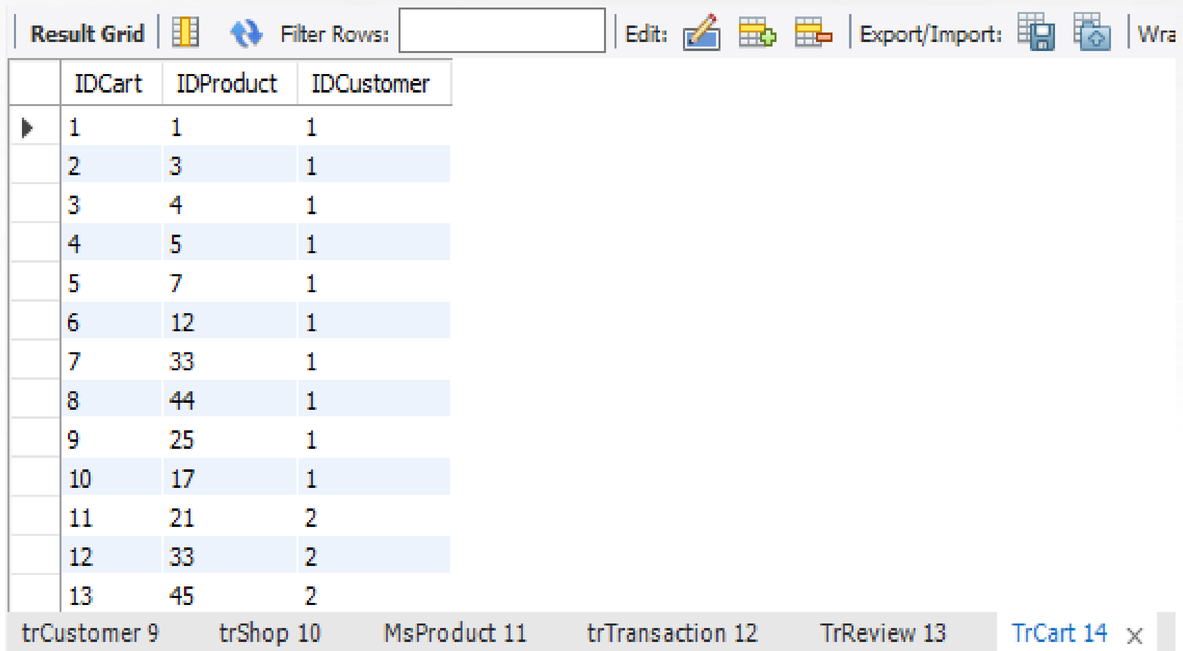
IDProduct	IDShop	Name	Stock	Price
1	SH145N	Fidget Spinner	110	49000
2	SH145N	Fidget Box	78	39000
3	SH145N	Slime	40	12000
4	SH145N	Lego	103	56000
5	SH145N	Gundam Master Grade	5	405000
6	SH223Y	Computer	5	5000000
7	SH223Y	VGA	26	1000000
8	SH223Y	Mouse	98	340000
9	SH223Y	Keyboard	63	760000
10	SH223Y	Earphone	120	120000
11	SH359Y	Soap	50	5000
12	SH359Y	Shampoo	45	21000
13	SH359Y	Tooth Brush	81	140000



IDTransaction	IDProduct	IDCustomer	TransactionDate	Qty	TotalPrice	Done	PaymentMethod
TR001	1	1	2018-03-12 12:23:01	2	98000	0	Credit Card
TR002	2	2	2018-05-01 07:21:01	1	39000	1	Debit
TR003	3	3	2018-02-23 20:45:56	1	12000	1	Credit Card
TR004	4	4	2018-09-15 17:38:59	1	56000	1	Credit Card
TR005	5	5	2018-08-05 10:11:01	2	105000	0	Debit
TR006	6	1	2018-01-23 12:23:31	1	5000000	0	Debit
TR007	7	2	2018-02-10 17:38:41	2	2000000	1	Debit
TR008	8	3	2018-03-22 20:23:17	3	1020000	0	Debit
TR009	9	4	2018-08-27 01:38:12	1	760000	1	Credit Card
TR010	10	5	2018-10-01 10:01:51	2	24000	0	Debit
TR011	31	6	2018-12-01 18:43:56	2	1080000	1	Credit Card
TR012	32	7	2018-11-25 07:26:41	5	2250000	0	Debit
TR013	33	8	2018-10-17 23:25:26	2	70000	0	Debit



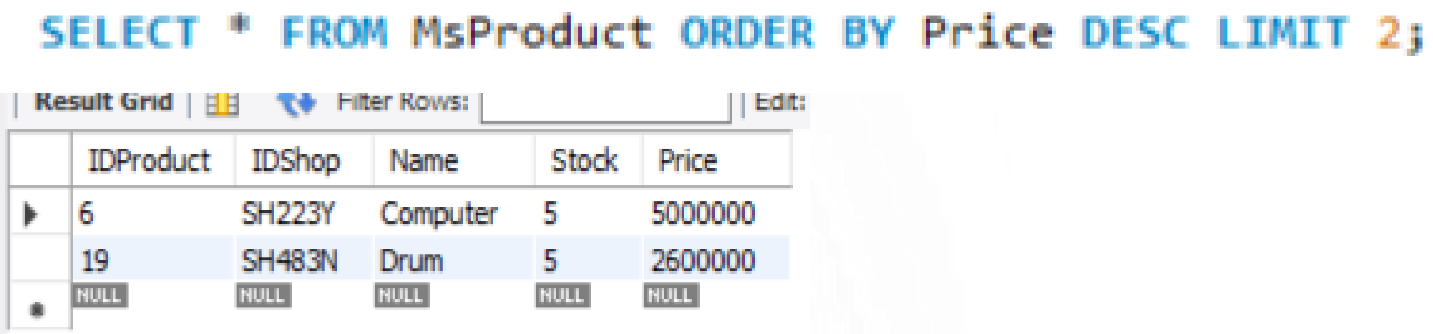
IDReview	IDProduct	Comment	Star
1	1	Good	5
2	1	Nice	4
3	2	I dont like it	2
4	2	Best product	5
5	3	Not really..	3
6	3	Never buy this item again	1
7	4	Good job	5
8	4	Awesome	5
9	5	Terrible	2
10	5	OK	3
11	6	Good	5
12	6	Nice	4
13	7	Best product	5



IDCart	IDProduct	IDCustomer
1	1	1
2	3	1
3	4	1
4	5	1
5	7	1
6	12	1
7	33	1
8	44	1
9	25	1
10	17	1
11	21	2
12	33	2
13	45	2

Using queries to display data

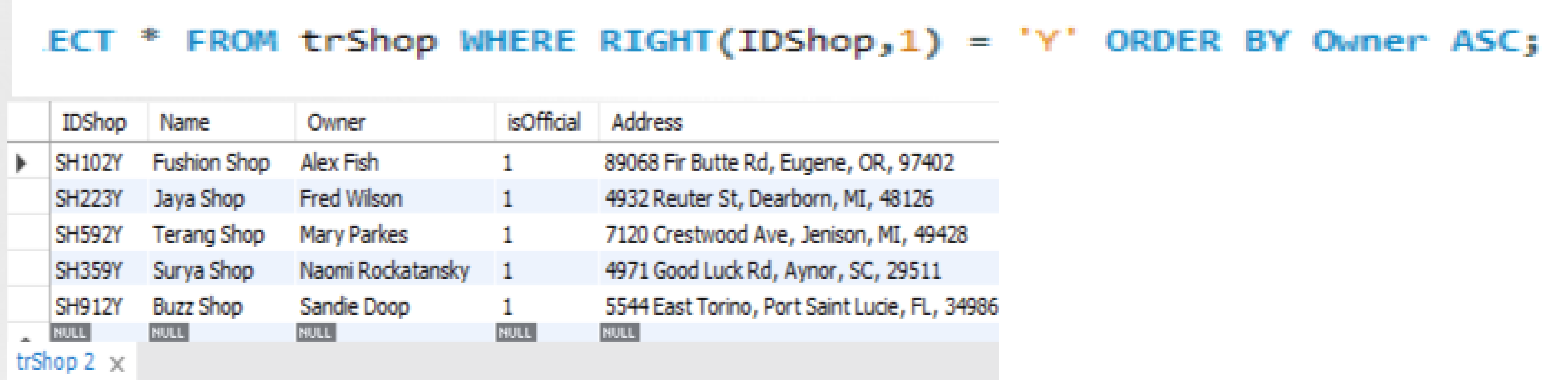
1. Showing the 2 most expensive items.



```
SELECT * FROM MsProduct ORDER BY Price DESC LIMIT 2;
```

IDProduct	IDShop	Name	Stock	Price
6	SH223Y	Computer	5	5000000
19	SH483N	Drum	5	2600000
NULL	NULL	NULL	NULL	NULL

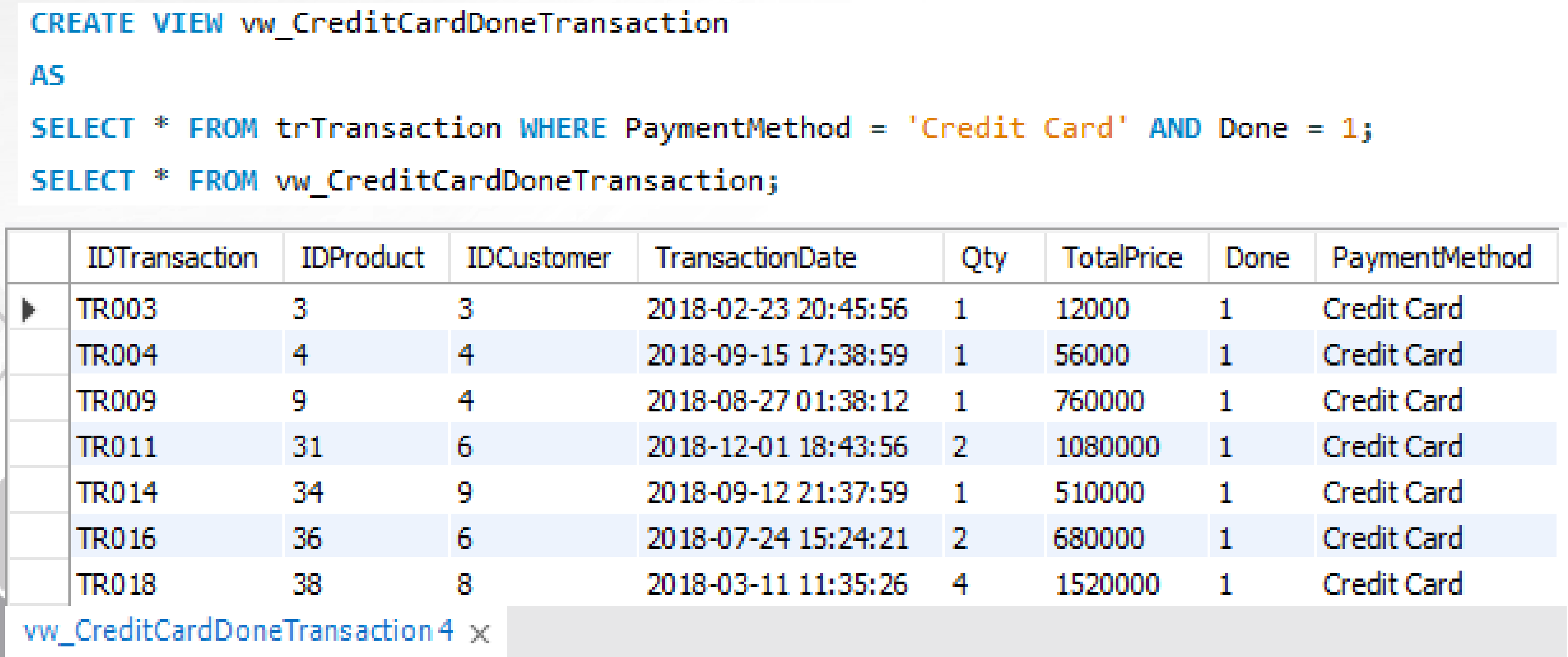
2. Show official shop details sorted by the name of the largest shop owner [A - Z]



```
ECT * FROM trShop WHERE RIGHT(IDShop,1) = 'Y' ORDER BY Owner ASC;
```

IDShop	Name	Owner	isOfficial	Address
SH102Y	Fushion Shop	Alex Fish	1	89068 Fir Butte Rd, Eugene, OR, 97402
SH223Y	Jaya Shop	Fred Wilson	1	4932 Reuter St, Dearborn, MI, 48126
SH592Y	Terang Shop	Mary Parkes	1	7120 Crestwood Ave, Jenison, MI, 49428
SH359Y	Surya Shop	Naomi Rockatansky	1	4971 Good Luck Rd, Aynor, SC, 29511
SH912Y	Buzz Shop	Sandie Doop	1	5544 East Torino, Port Saint Lucie, FL, 34986
NULL	NULL	NULL	NULL	NULL

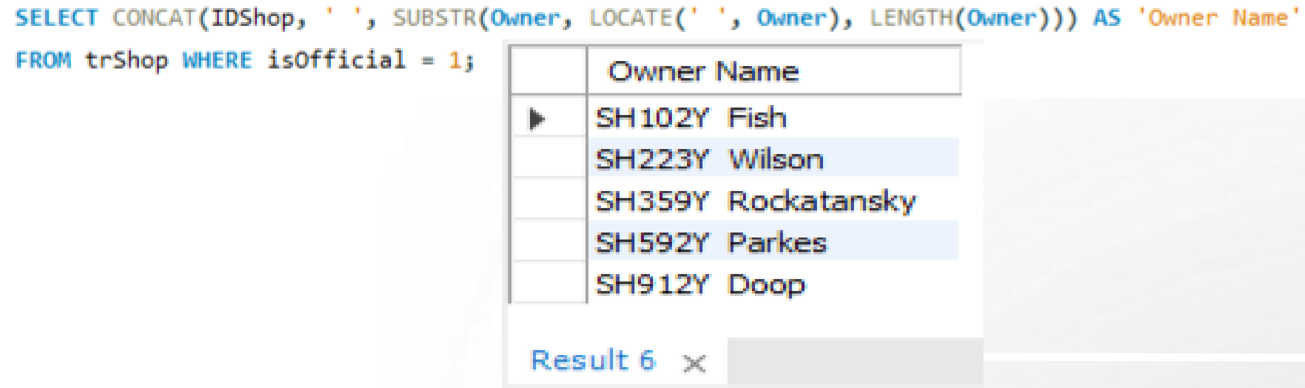
3. Create a view named 'vw_CreditCardDoneTransaction', displaying details of transactions that have been completed and used a Credit Card



```
CREATE VIEW vw_CreditCardDoneTransaction
AS
SELECT * FROM trTransaction WHERE PaymentMethod = 'Credit Card' AND Done = 1;
SELECT * FROM vw_CreditCardDoneTransaction;
```

IDTransaction	IDProduct	IDCustomer	TransactionDate	Qty	TotalPrice	Done	PaymentMethod
TR003	3	3	2018-02-23 20:45:56	1	12000	1	Credit Card
TR004	4	4	2018-09-15 17:38:59	1	56000	1	Credit Card
TR009	9	4	2018-08-27 01:38:12	1	760000	1	Credit Card
TR011	31	6	2018-12-01 18:43:56	2	1080000	1	Credit Card
TR014	34	9	2018-09-12 21:37:59	1	510000	1	Credit Card
TR016	36	6	2018-07-24 15:24:21	2	680000	1	Credit Card
TR018	38	8	2018-03-11 11:35:26	4	1520000	1	Credit Card

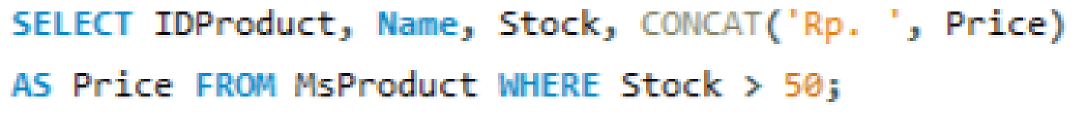
4. Displays the name of the OFFICIAL shop owner in the format: [shop code + shop owner's last name]



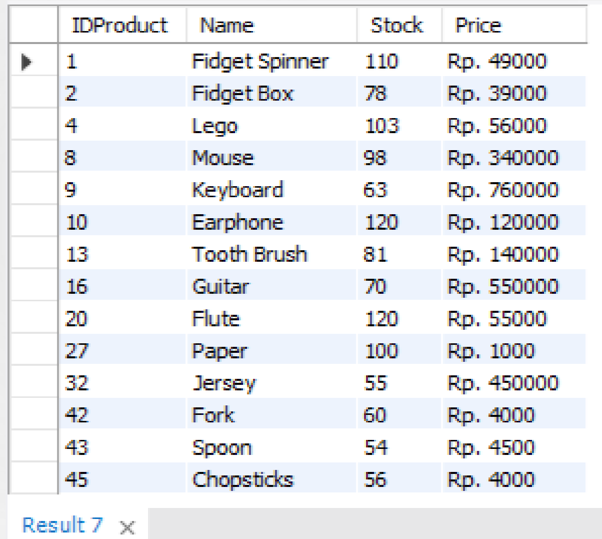
```
SELECT CONCAT(IDShop, ' ', SUBSTR(Owner, LOCATE(' ', Owner), LENGTH(Owner))) AS 'Owner Name'
FROM trShop WHERE isOfficial = 1;
```

Owner Name
SH102Y Fish
SH223Y Wilson
SH359Y Rockatansky
SH592Y Parkes
SH912Y Doop

5. Displays product code, product name, product stock, price in format ['Rp. ' + Price] of products that have more than 50 stocks

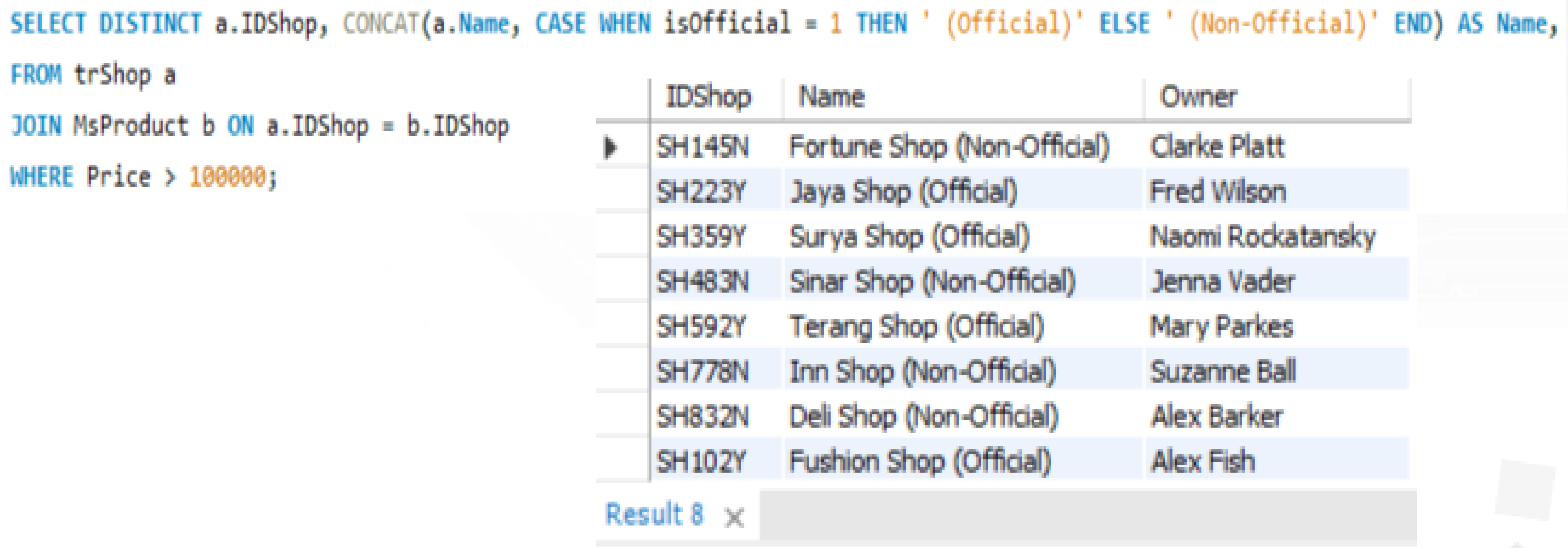


```
SELECT IDProduct, Name, Stock, CONCAT('Rp. ', Price)
AS Price FROM MsProduct WHERE Stock > 50;
```



IDProduct	Name	Stock	Price
1	Fidget Spinner	110	Rp. 49000
2	Fidget Box	78	Rp. 39000
4	Lego	103	Rp. 56000
8	Mouse	98	Rp. 340000
9	Keyboard	63	Rp. 760000
10	Earphone	120	Rp. 120000
13	Tooth Brush	81	Rp. 140000
16	Guitar	70	Rp. 550000
20	Flute	120	Rp. 55000
27	Paper	100	Rp. 1000
32	Jersey	55	Rp. 450000
42	Fork	60	Rp. 4000
43	Spoon	54	Rp. 4500
45	Chopsticks	56	Rp. 4000

6. Displays the shop code, shop name in the format shop_name + official/non-official, owner, address that has a price of more than 100,000

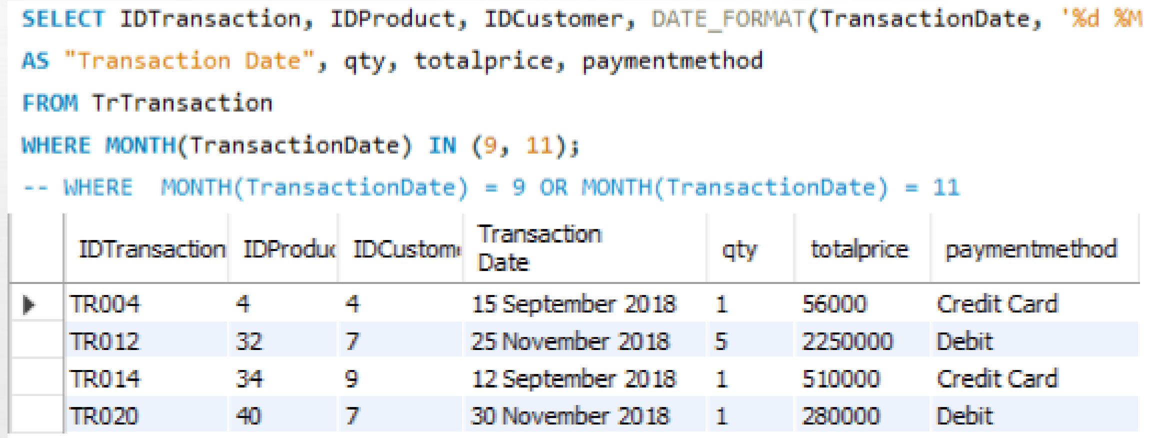


```
SELECT DISTINCT a.IDShop, CONCAT(a.Name, CASE WHEN isOfficial = 1 THEN ' (Official)' ELSE ' (Non-Official)' END) AS Name,
FROM trShop a
JOIN MsProduct b ON a.IDShop = b.IDShop
WHERE Price > 100000;
```

IDShop	Name	Owner
SH145N	Fortune Shop (Non-Official)	Clarke Platt
SH223Y	Jaya Shop (Official)	Fred Wilson
SH359Y	Surya Shop (Official)	Naomi Rockatansky
SH483N	Sinar Shop (Non-Official)	Jenna Vader
SH592Y	Terang Shop (Official)	Mary Parkes
SH778N	Inn Shop (Non-Official)	Suzanne Ball
SH832N	Deli Shop (Non-Official)	Alex Barker
SH102Y	Fushion Shop (Official)	Alex Fish

Using queries to display data

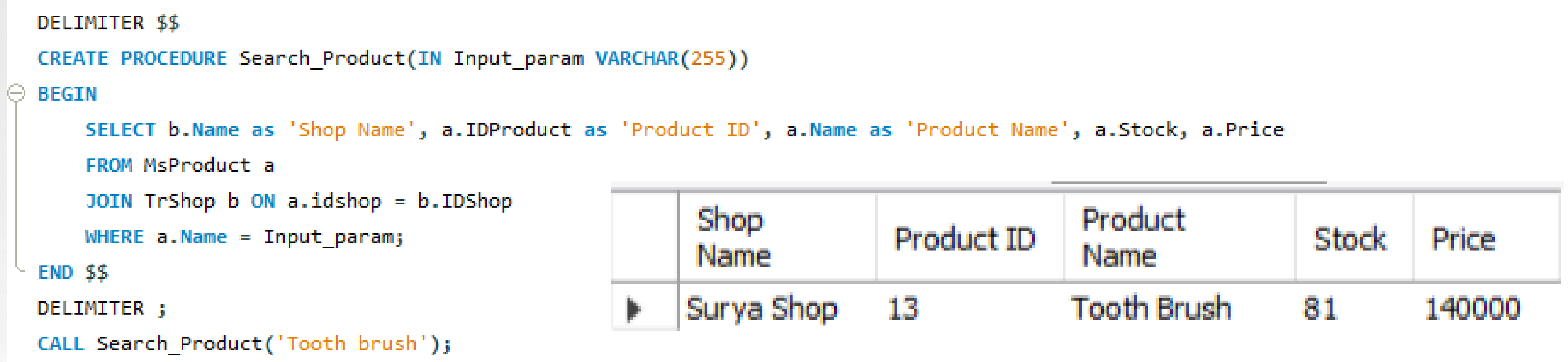
7. Displays transaction code, product code, customer code, transaction date in the format dd mm yyyy, qty, total price, payment method from transactions that occurred in September and November.



```
SELECT IDTransaction, IDProduct, IDCustomer, DATE_FORMAT(TransactionDate, '%d %M %Y')
AS "Transaction Date", qty, totalprice, paymentmethod
FROM TrTransaction
WHERE MONTH(TransactionDate) IN (9, 11);
-- WHERE  MONTH(TransactionDate) = 9 OR MONTH(TransactionDate) = 11
```

IDTransaction	IDProduc	IDCustom	Transaction Date	qty	totalprice	paymentmethod
TR004	4	4	15 September 2018	1	56000	Credit Card
TR012	32	7	25 November 2018	5	2250000	Debit
TR014	34	9	12 September 2018	1	510000	Credit Card
TR020	40	7	30 November 2018	1	280000	Debit

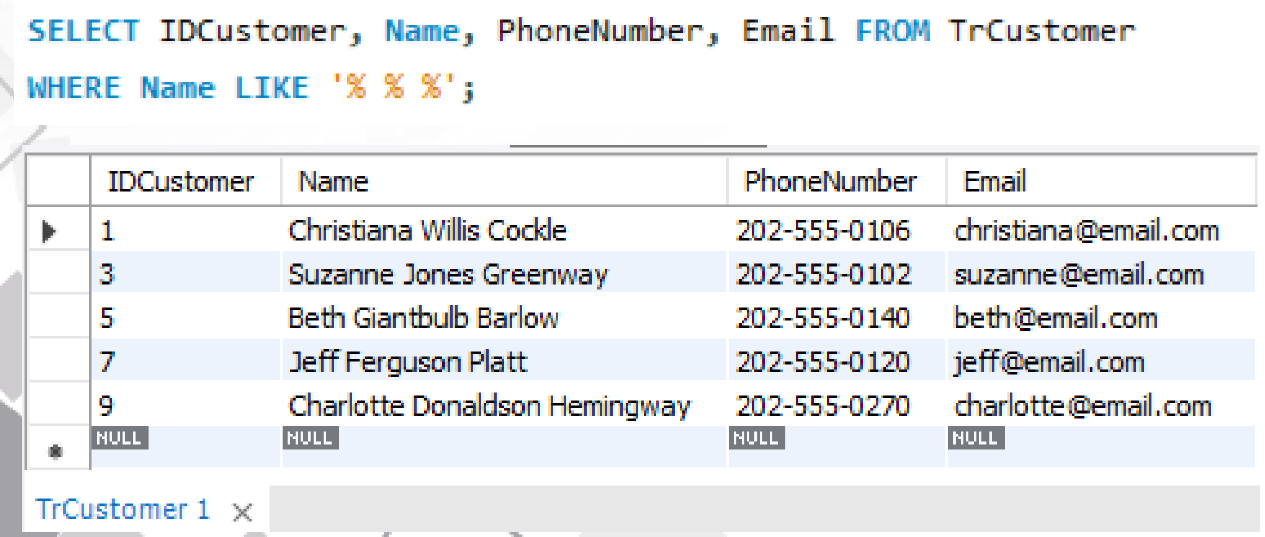
8. Create a Stored Procedure called 'Search_Product' which accepts input/parameters of item names, and displays the name of the shop that sells the item, item code, item name, stock, price



```
DELIMITER $$
CREATE PROCEDURE Search_Product(IN Input_param VARCHAR(255))
BEGIN
    SELECT b.Name as 'Shop Name', a.IDProduct as 'Product ID', a.Name as 'Product Name', a.Stock, a.Price
    FROM MsProduct a
    JOIN TrShop b ON a.idshop = b.IDShop
    WHERE a.Name = Input_param;
END $$
DELIMITER ;
CALL Search_Product('Tooth brush');
```

Shop Name	Product ID	Product Name	Stock	Price
Surya Shop	13	Tooth Brush	81	140000

9. Displays customer code, customer name, PhoneNumber, and email that has a name with at least 3 words

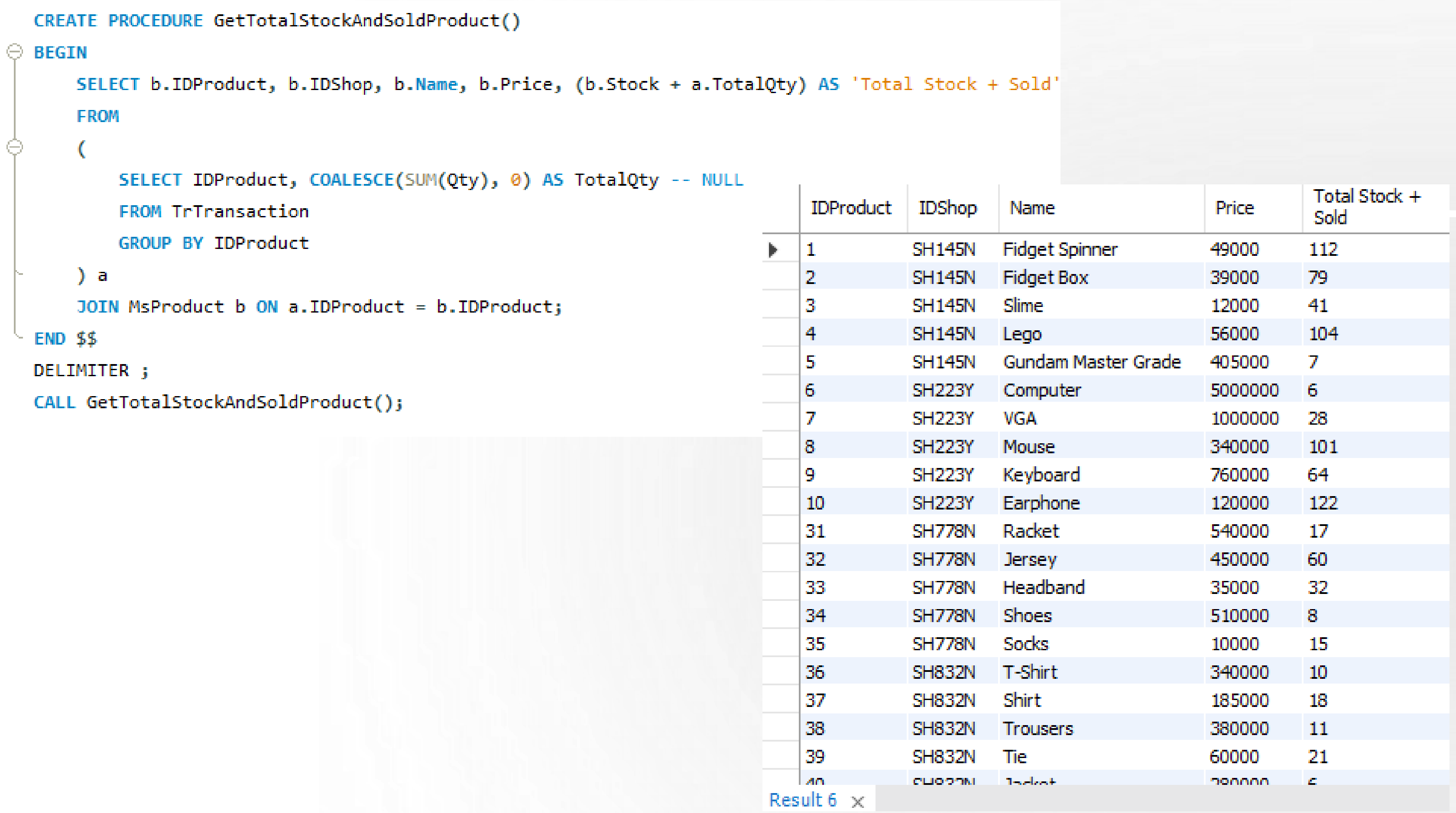


```
SELECT IDCustomer, Name, PhoneNumber, Email FROM TrCustomer
WHERE Name LIKE '% % %';
```

IDCustomer	Name	PhoneNumber	Email
1	Christiana Willis Cockle	202-555-0106	christiana@email.com
3	Suzanne Jones Greenway	202-555-0102	suzanne@email.com
5	Beth Giantbulb Barlow	202-555-0140	beth@email.com
7	Jeff Ferguson Platt	202-555-0120	jeff@email.com
9	Charlotte Donaldson Hemingway	202-555-0270	charlotte@email.com
NULL	NULL	NULL	NULL

TrCustomer 1 ×

10. Create a Stored Procedure called 'GetTotalStockAndSoldProduct', does not accept input/parameters which function to display all product details and [Total Stock + Sold] = total stock product + number of products that have been sold

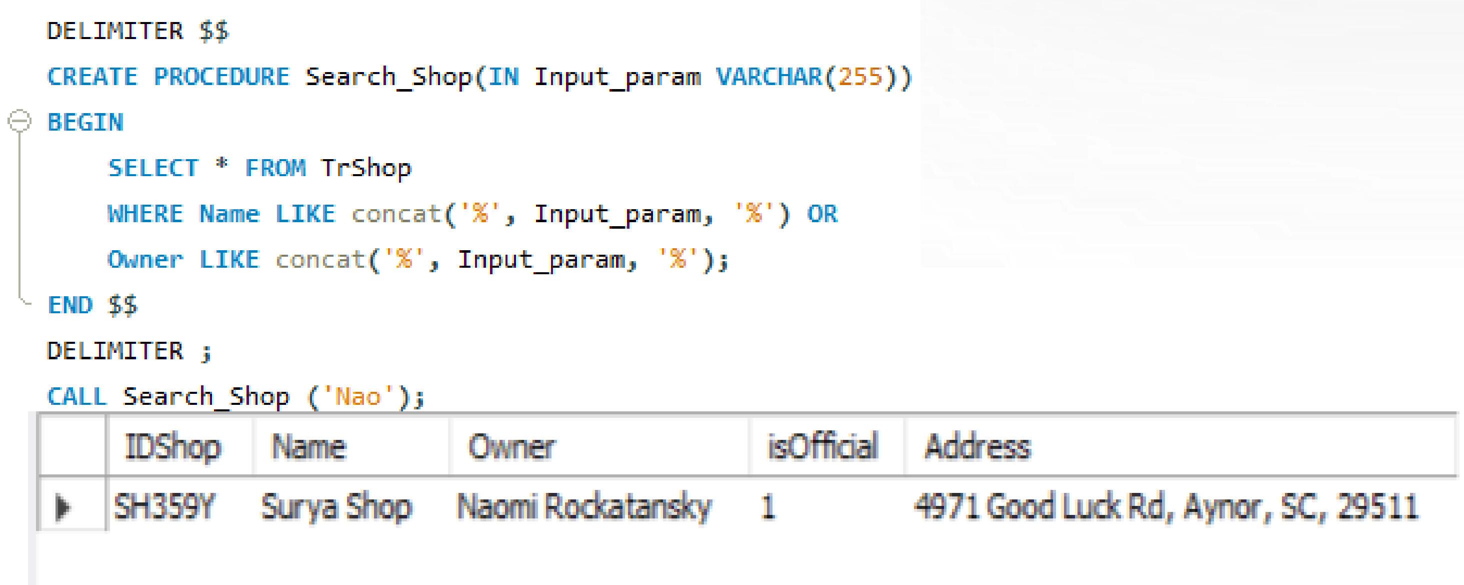


```
CREATE PROCEDURE GetTotalStockAndSoldProduct()
BEGIN
    SELECT b.IDProduct, b.IDShop, b.Name, b.Price, (b.Stock + a.TotalQty) AS 'Total Stock + Sold'
    FROM
    (
        SELECT IDProduct, COALESCE(SUM(Qty), 0) AS TotalQty -- NULL
        FROM TrTransaction
        GROUP BY IDProduct
    ) a
    JOIN MsProduct b ON a.IDProduct = b.IDProduct;
END $$
DELIMITER ;
CALL GetTotalStockAndSoldProduct();
```

IDProduct	IDShop	Name	Price	Total Stock + Sold
1	SH145N	Fidget Spinner	49000	112
2	SH145N	Fidget Box	39000	79
3	SH145N	Slime	12000	41
4	SH145N	Lego	56000	104
5	SH145N	Gundam Master Grade	405000	7
6	SH223Y	Computer	5000000	6
7	SH223Y	VGA	1000000	28
8	SH223Y	Mouse	340000	101
9	SH223Y	Keyboard	760000	64
10	SH223Y	Earphone	120000	122
31	SH778N	Racket	540000	17
32	SH778N	Jersey	450000	60
33	SH778N	Headband	35000	32
34	SH778N	Shoes	510000	8
35	SH778N	Socks	10000	15
36	SH832N	T-Shirt	340000	10
37	SH832N	Shirt	185000	18
38	SH832N	Trousers	380000	11
39	SH832N	Tie	60000	21
40	SH832N	Jacket	280000	6

Result 6 ×

11. Create a Stored Procedure called 'Search_Shop', accepts as input/parameter the name of the shop OR the name of the owner which functions to display shop data according to the shop/owner entered



```
DELIMITER $$
CREATE PROCEDURE Search_Shop(IN Input_param VARCHAR(255))
BEGIN
    SELECT * FROM TrShop
    WHERE Name LIKE concat('%', Input_param, '%') OR
    Owner LIKE concat('%', Input_param, '%');
END $$
DELIMITER ;
CALL Search_Shop ('Nao');
```

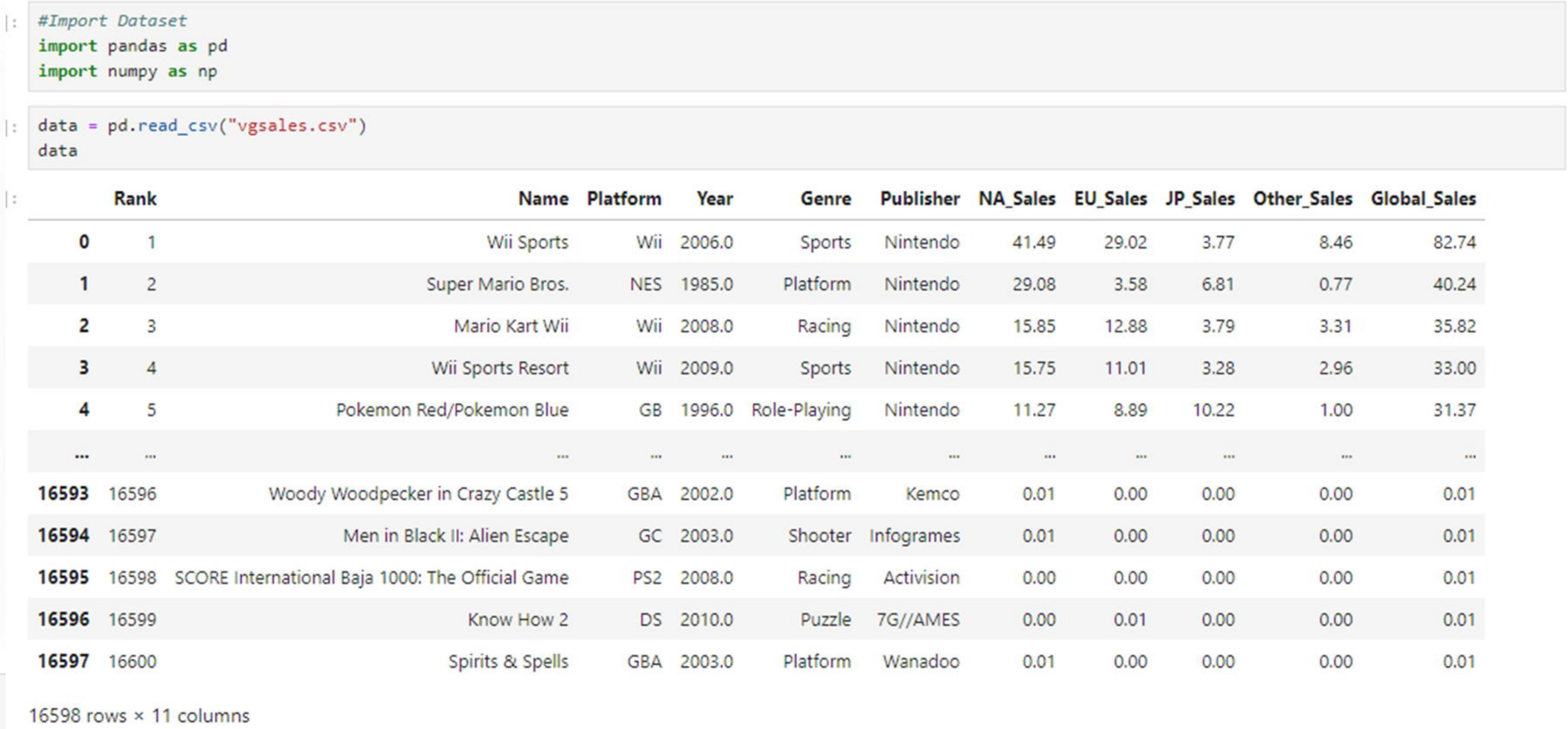
IDShop	Name	Owner	isOfficial	Address
SH359Y	Surya Shop	Naomi Rockatansky	1	4971 Good Luck Rd, Aynor, SC, 29511

PROJECT 02

Python Programming.

Perform data preparation and cleaning

Import Data using Pandas and Numpy



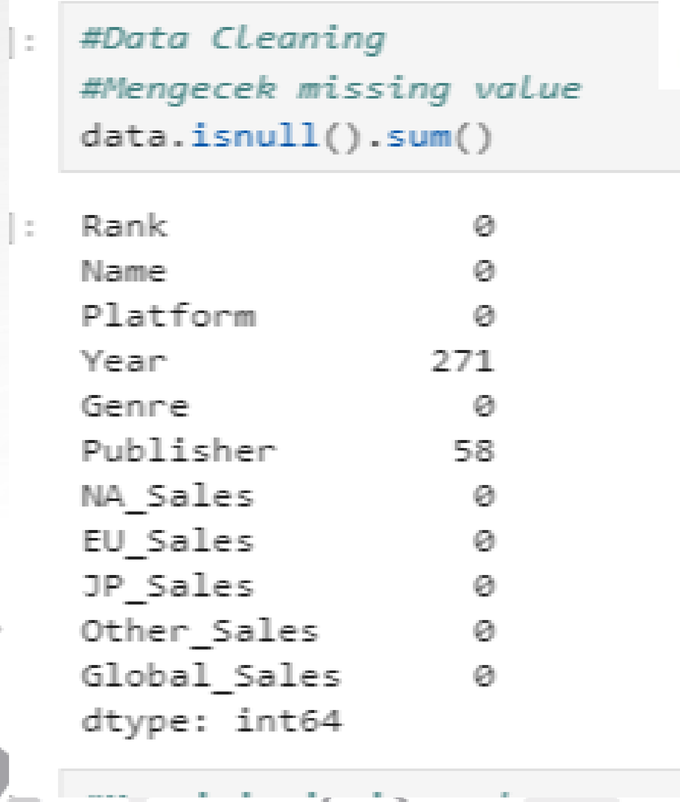
```
#Import Dataset
import pandas as pd
import numpy as np
```

```
data = pd.read_csv("vgsales.csv")
data
```

	Rank	Name	Platform	Year	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales
0	1	Wii Sports	Wii	2006.0	Sports	Nintendo	41.49	29.02	3.77	8.46	82.74
1	2	Super Mario Bros.	NES	1985.0	Platform	Nintendo	29.08	3.58	6.81	0.77	40.24
2	3	Mario Kart Wii	Wii	2008.0	Racing	Nintendo	15.85	12.88	3.79	3.31	35.82
3	4	Wii Sports Resort	Wii	2009.0	Sports	Nintendo	15.75	11.01	3.28	2.96	33.00
4	5	Pokemon Red/Pokemon Blue	GB	1996.0	Role-Playing	Nintendo	11.27	8.89	10.22	1.00	31.37
...	...	...	...	...	...	...	...	...	...	...	...
16593	16596	Woody Woodpecker in Crazy Castle 5	GBA	2002.0	Platform	Kemco	0.01	0.00	0.00	0.00	0.01
16594	16597	Men in Black II: Alien Escape	GC	2003.0	Shooter	Infogrames	0.01	0.00	0.00	0.00	0.01
16595	16598	SCORE International Baja 1000: The Official Game	PS2	2008.0	Racing	Activision	0.00	0.00	0.00	0.00	0.01
16596	16599	Know How 2	DS	2010.0	Puzzle	7G//AMES	0.00	0.01	0.00	0.00	0.01
16597	16600	Spirits & Spells	GBA	2003.0	Platform	Wanadoo	0.01	0.00	0.00	0.00	0.01

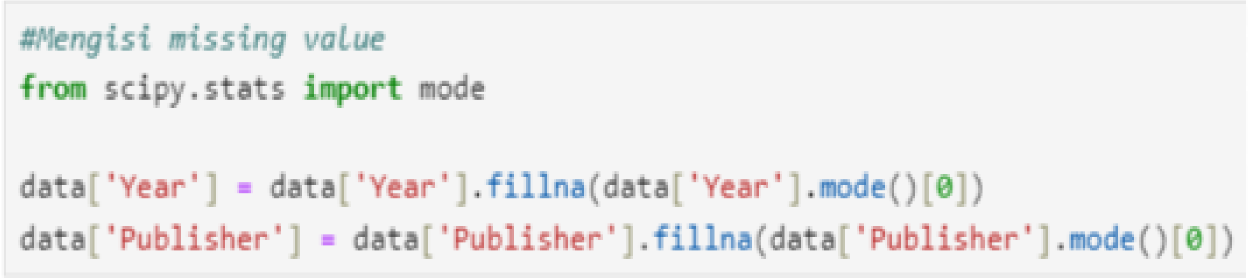
16598 rows × 11 columns

The year and publisher variables have missing values, then fill in the missing values using the mode of each variable



```
#Data Cleaning
#Mengecek missing value
data.isnull().sum()
```

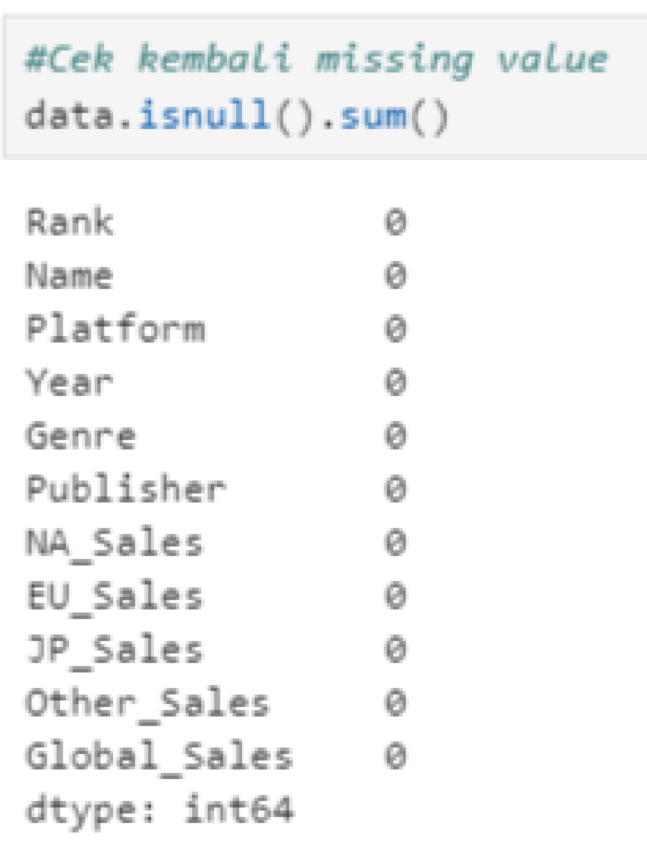
```
Rank              0
Name              0
Platform          0
Year            271
Genre             0
Publisher        58
NA_Sales          0
EU_Sales          0
JP_Sales          0
Other_Sales       0
Global_Sales      0
dtype: int64
```



```
#Mengisi missing value
from scipy.stats import mode

data['Year'] = data['Year'].fillna(data['Year'].mode()[0])
data['Publisher'] = data['Publisher'].fillna(data['Publisher'].mode()[0])
```

Handling missing values uses the following syntax

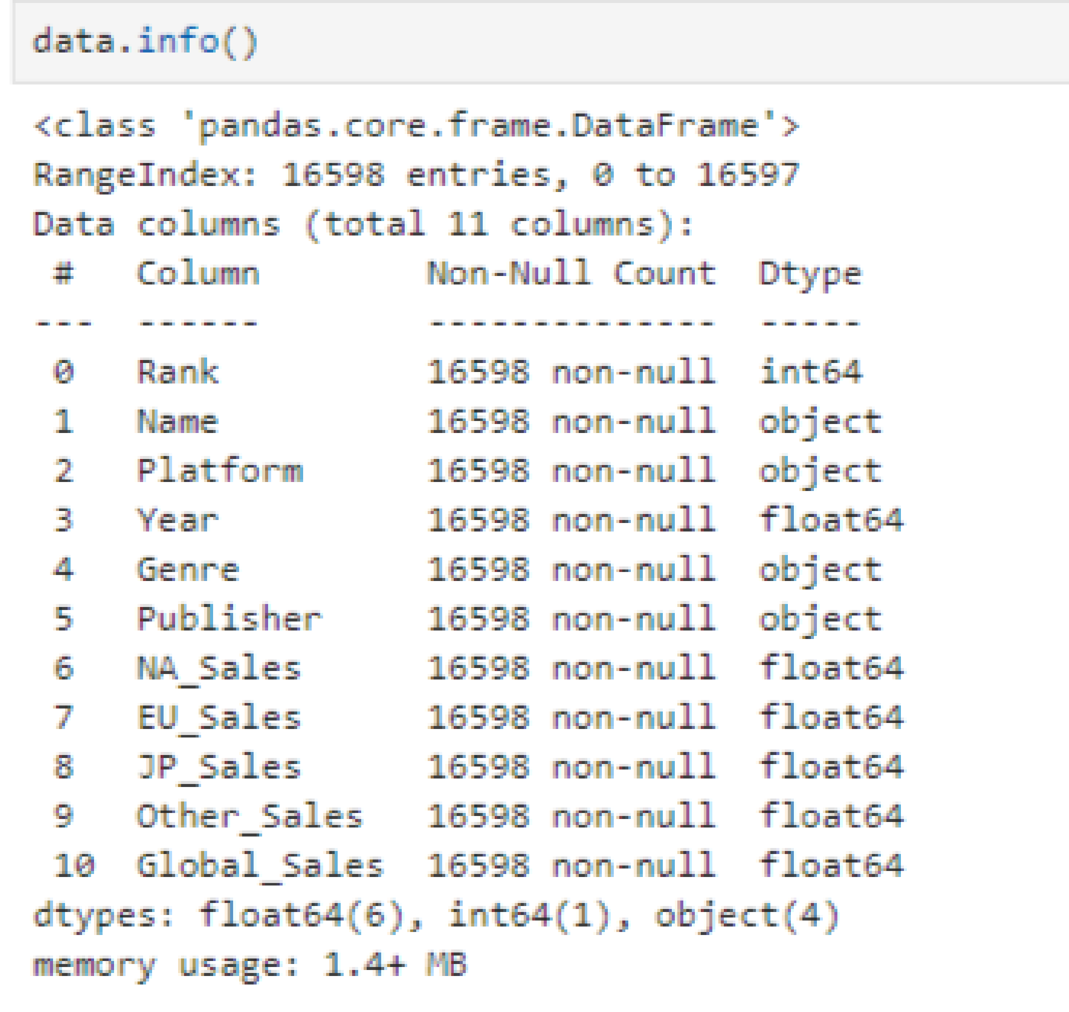
There are no missing values in the data variables that will be used



```
#Cek kembali missing value
data.isnull().sum()
```

```
Rank            0
Name            0
Platform        0
Year            0
Genre           0
Publisher       0
NA_Sales        0
EU_Sales        0
JP_Sales        0
Other_Sales     0
Global_Sales    0
dtype: int64
```

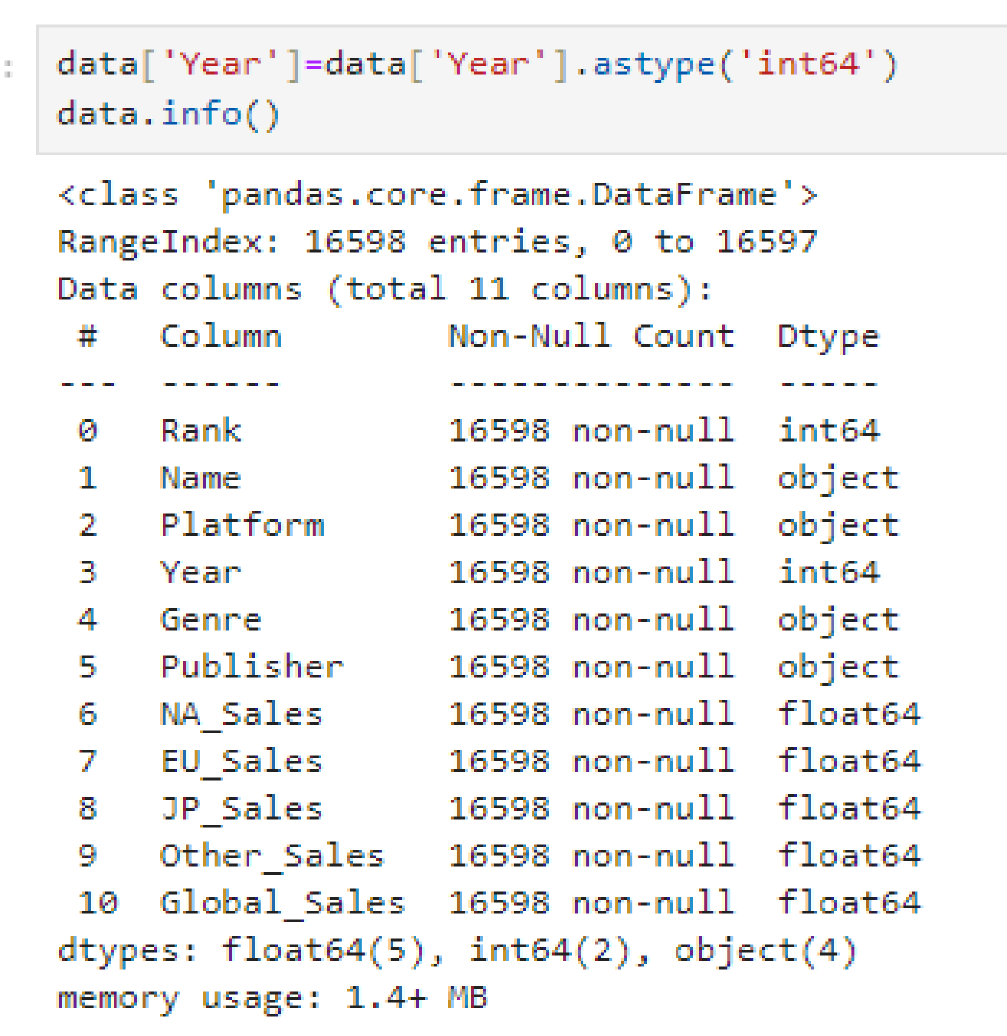
Carry out data quality testing



```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 16598 entries, 0 to 16597
Data columns (total 11 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   Rank          16598 non-null  int64  
 1   Name          16598 non-null  object 
 2   Platform      16598 non-null  object 
 3   Year          16598 non-null  float64
 4   Genre         16598 non-null  object 
 5   Publisher     16598 non-null  object 
 6   NA_Sales      16598 non-null  float64
 7   EU_Sales      16598 non-null  float64
 8   JP_Sales      16598 non-null  float64
 9   Other_Sales   16598 non-null  float64
 10  Global_Sales  16598 non-null  float64
dtypes: float64(6), int64(1), object(4)
memory usage: 1.4+ MB
```

The data type for year is float so it is converted to integer so that visualization can be carried out

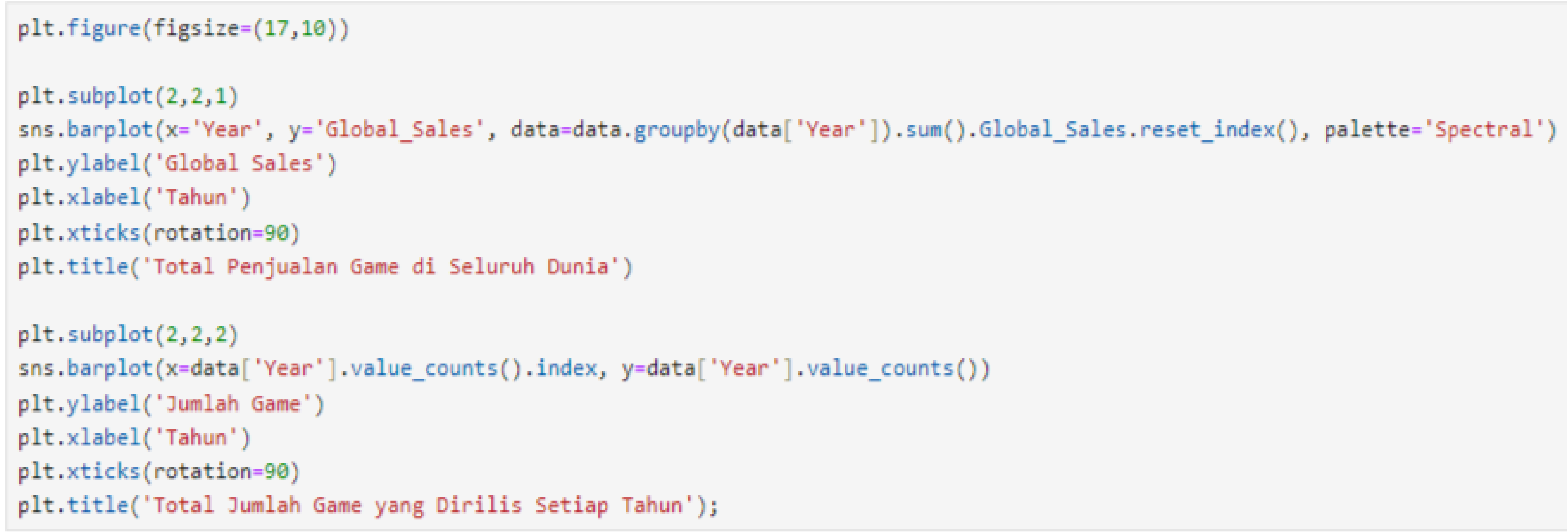


```
data['Year']=data['Year'].astype('int64')
data.info()
```

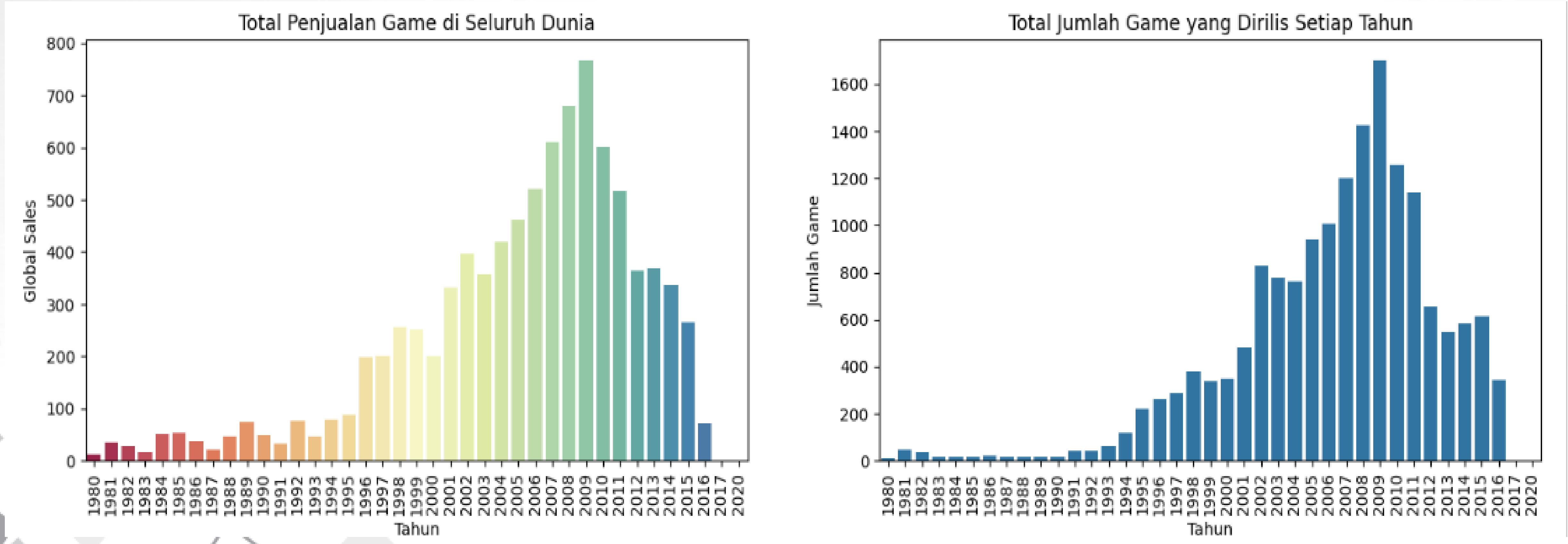
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 16598 entries, 0 to 16597
Data columns (total 11 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   Rank          16598 non-null  int64  
 1   Name          16598 non-null  object 
 2   Platform      16598 non-null  object 
 3   Year          16598 non-null  int64  
 4   Genre         16598 non-null  object 
 5   Publisher     16598 non-null  object 
 6   NA_Sales      16598 non-null  float64
 7   EU_Sales      16598 non-null  float64
 8   JP_Sales      16598 non-null  float64
 9   Other_Sales   16598 non-null  float64
 10  Global_Sales  16598 non-null  float64
dtypes: float64(5), int64(2), object(4)
memory usage: 1.4+ MB
```

Perform data visualization

Create visualizations using histograms



```
plt.figure(figsize=(17,10))

plt.subplot(2,2,1)
sns.barplot(x='Year', y='Global_Sales', data=data.groupby(data['Year']).sum().Global_Sales.reset_index(), palette='Spectral')
plt.ylabel('Global Sales')
plt.xlabel('Tahun')
plt.xticks(rotation=90)
plt.title('Total Penjualan Game di Seluruh Dunia')

plt.subplot(2,2,2)
sns.barplot(x=data['Year'].value_counts().index, y=data['Year'].value_counts())
plt.ylabel('Jumlah Game')
plt.xlabel('Tahun')
plt.xticks(rotation=90)
plt.title('Total Jumlah Game yang Dirilis Setiap Tahun');
```



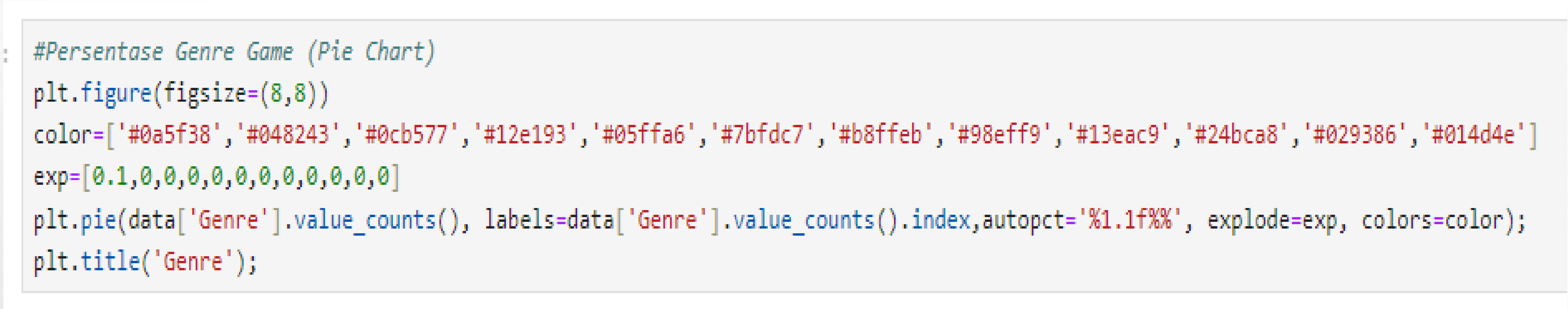
Total Penjualan Game di Seluruh Dunia

Total Jumlah Game yang Dirilis Setiap Tahun

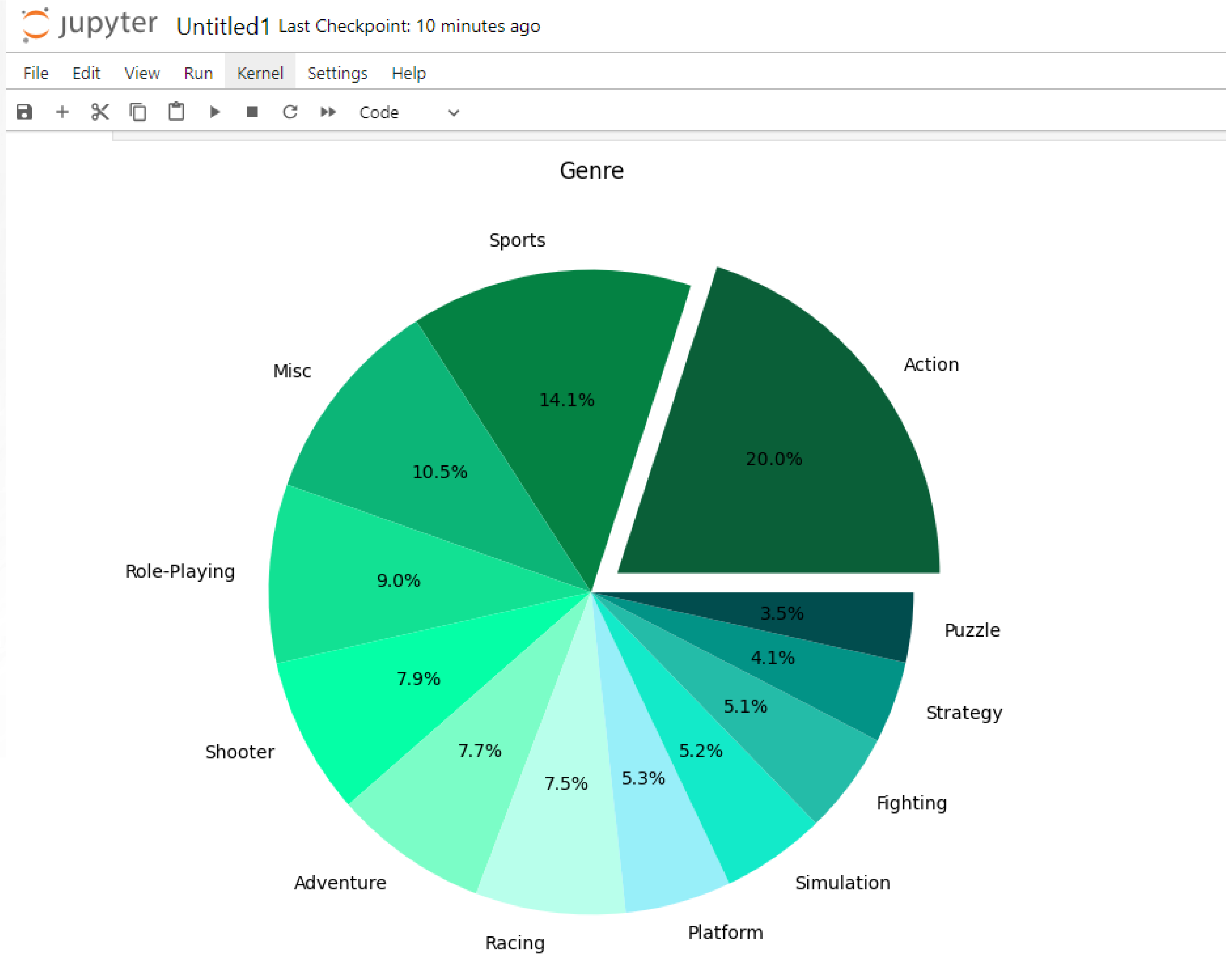
Visualize based on total game sales worldwide and total games released each year. From the results of the visualization, it can be seen that 2009 was the year with the most game sales because in the same year, 2009 was the year with the highest number of games released compared to the previous year.

Perform data visualization

Create visualizations using pie charts



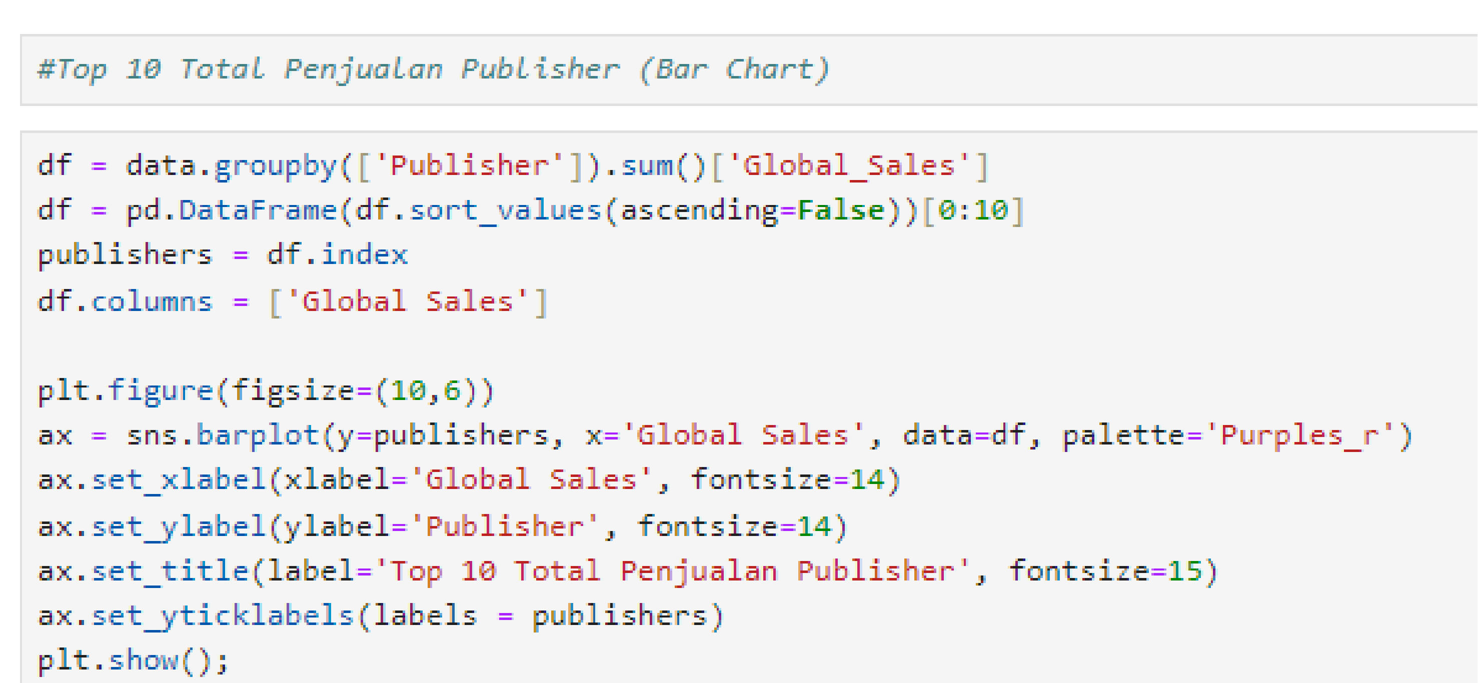
```
#Persentase Genre Game (Pie Chart)
plt.figure(figsize=(8,8))
color=['#0a5f38','#048243','#0cb577','#12e193','#05ffa6','#7bfdc7','#b8ffeb','#98eff9','#13eac9','#24bca8','#029386','#014d4e']
exp=[0.1,0,0,0,0,0,0,0,0,0,0,0]
plt.pie(data['Genre'].value_counts(), labels=data['Genre'].value_counts().index,autopct='%1.1f%%', explode=exp, colors=color);
plt.title('Genre');
```



Visualize based on the genre of games sold. From the results of the visualization, it can be seen that games with the action genre are most popular with game lovers from all over the world at 20%, followed by games with the sports genre at 14.1%.

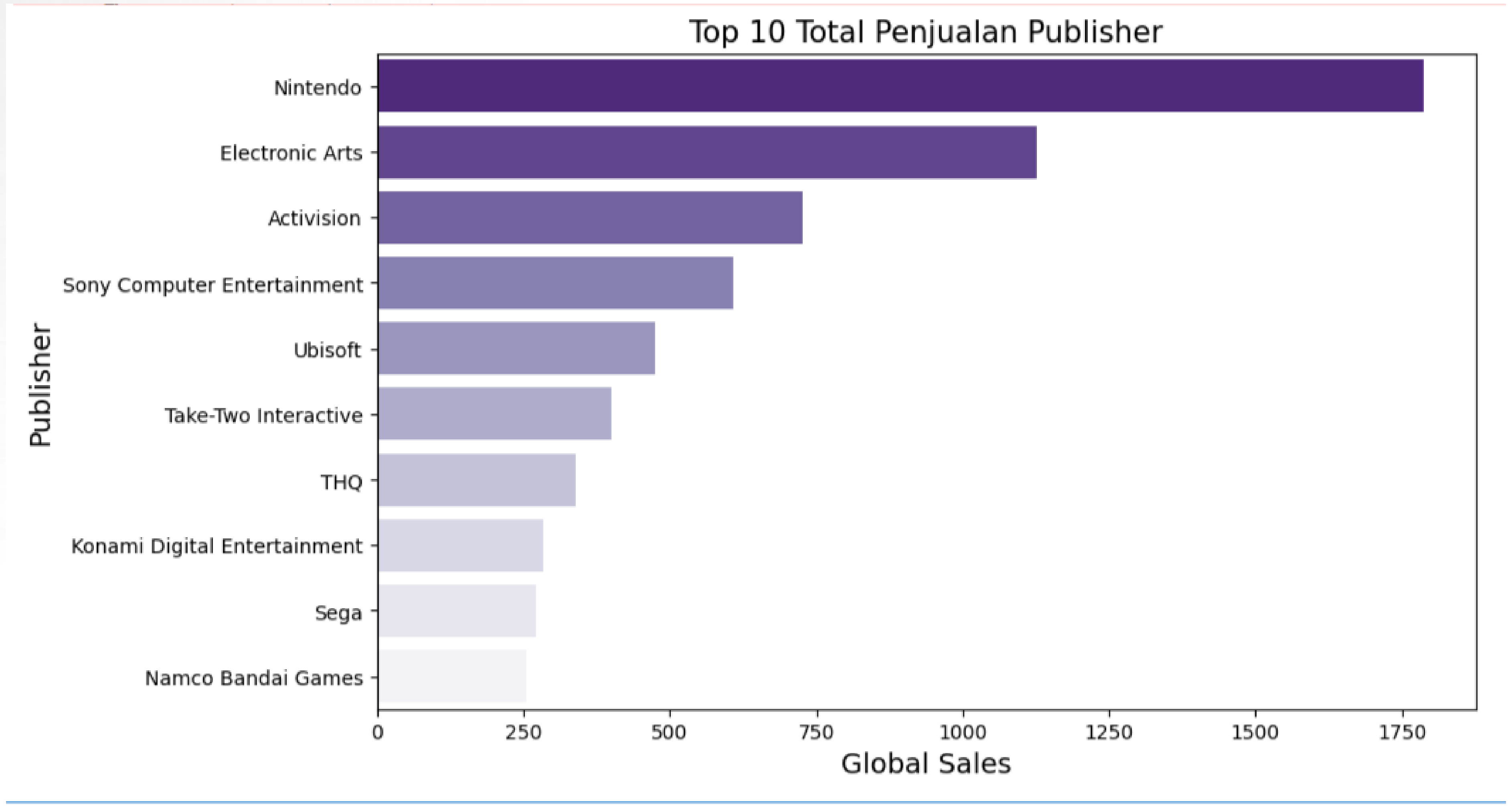
Perform data visualization

Create visualizations using bar charts



```
#Top 10 Total Penjualan Publisher (Bar Chart)
```

```
df = data.groupby(['Publisher']).sum()['Global_Sales']
df = pd.DataFrame(df.sort_values(ascending=False))[0:10]
publishers = df.index
df.columns = ['Global Sales']

plt.figure(figsize=(10,6))
ax = sns.barplot(y=publishers, x='Global Sales', data=df, palette='Purples_r')
ax.set_xlabel(xlabel='Global Sales', fontsize=14)
ax.set_ylabel(ylabel='Publisher', fontsize=14)
ax.set_title(label='Top 10 Total Penjualan Publisher', fontsize=15)
ax.set_yticklabels(labels = publishers)
plt.show();
```



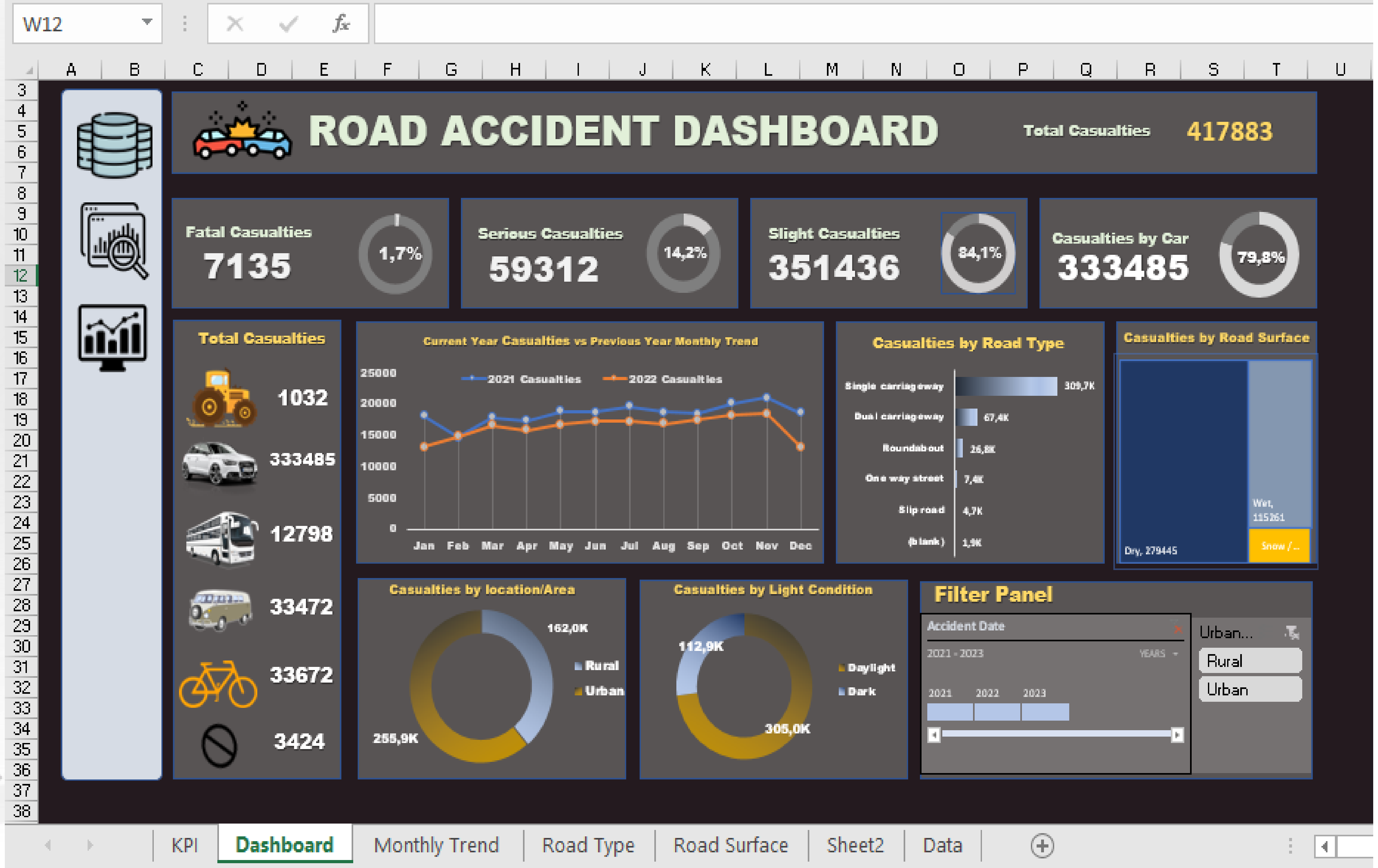
Create a visualization based on the top 10 total sales from publishers/game makers. From the results of the visualization it can be seen that Nintendo games are the best-selling games on the market

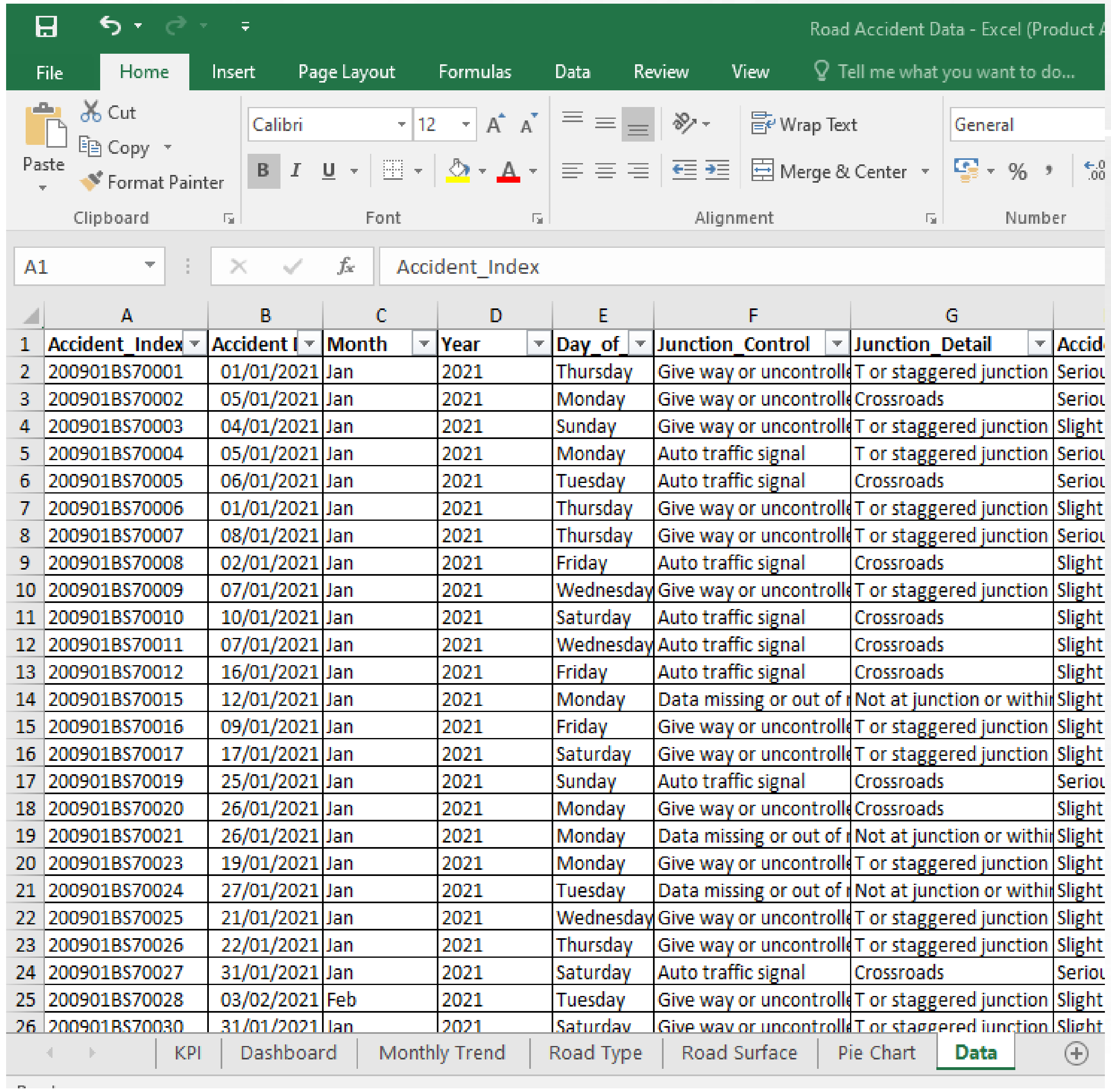
PROJECT 03

Microsoft

Excel.

Road Accident Dashboard





A1 Accident_Index

	A	B	C	D	E	F	G	
1	Accident_Index	Accident I	Month	Year	Day_of_	Junction_Control	Junction_Detail	Accid
2	200901BS70001	01/01/2021	Jan	2021	Thursday	Give way or uncontrolle	T or staggered junction	Seriou
3	200901BS70002	05/01/2021	Jan	2021	Monday	Give way or uncontrolle	Crossroads	Seriou
4	200901BS70003	04/01/2021	Jan	2021	Sunday	Give way or uncontrolle	T or staggered junction	Slight
5	200901BS70004	05/01/2021	Jan	2021	Monday	Auto traffic signal	T or staggered junction	Seriou
6	200901BS70005	06/01/2021	Jan	2021	Tuesday	Auto traffic signal	Crossroads	Seriou
7	200901BS70006	01/01/2021	Jan	2021	Thursday	Give way or uncontrolle	T or staggered junction	Slight
8	200901BS70007	08/01/2021	Jan	2021	Thursday	Give way or uncontrolle	T or staggered junction	Seriou
9	200901BS70008	02/01/2021	Jan	2021	Friday	Auto traffic signal	Crossroads	Slight
10	200901BS70009	07/01/2021	Jan	2021	Wednesday	Give way or uncontrolle	T or staggered junction	Slight
11	200901BS70010	10/01/2021	Jan	2021	Saturday	Auto traffic signal	Crossroads	Slight
12	200901BS70011	07/01/2021	Jan	2021	Wednesday	Auto traffic signal	Crossroads	Slight
13	200901BS70012	16/01/2021	Jan	2021	Friday	Auto traffic signal	Crossroads	Slight
14	200901BS70015	12/01/2021	Jan	2021	Monday	Data missing or out of r	Not at junction or withi	Slight
15	200901BS70016	09/01/2021	Jan	2021	Friday	Give way or uncontrolle	T or staggered junction	Slight
16	200901BS70017	17/01/2021	Jan	2021	Saturday	Give way or uncontrolle	T or staggered junction	Slight
17	200901BS70019	25/01/2021	Jan	2021	Sunday	Auto traffic signal	Crossroads	Seriou
18	200901BS70020	26/01/2021	Jan	2021	Monday	Give way or uncontrolle	Crossroads	Slight
19	200901BS70021	26/01/2021	Jan	2021	Monday	Data missing or out of r	Not at junction or withi	Slight
20	200901BS70023	19/01/2021	Jan	2021	Monday	Give way or uncontrolle	T or staggered junction	Slight
21	200901BS70024	27/01/2021	Jan	2021	Tuesday	Data missing or out of r	Not at junction or withi	Slight
22	200901BS70025	21/01/2021	Jan	2021	Wednesday	Give way or uncontrolle	T or staggered junction	Slight
23	200901BS70026	22/01/2021	Jan	2021	Thursday	Give way or uncontrolle	T or staggered junction	Slight
24	200901BS70027	31/01/2021	Jan	2021	Saturday	Auto traffic signal	Crossroads	Seriou
25	200901BS70028	03/02/2021	Feb	2021	Tuesday	Give way or uncontrolle	T or staggered junction	Slight
26	200901BS70030	31/01/2021	Jan	2021	Saturday	Give way or uncontrolle	T or staggered junction	Slight

Summary

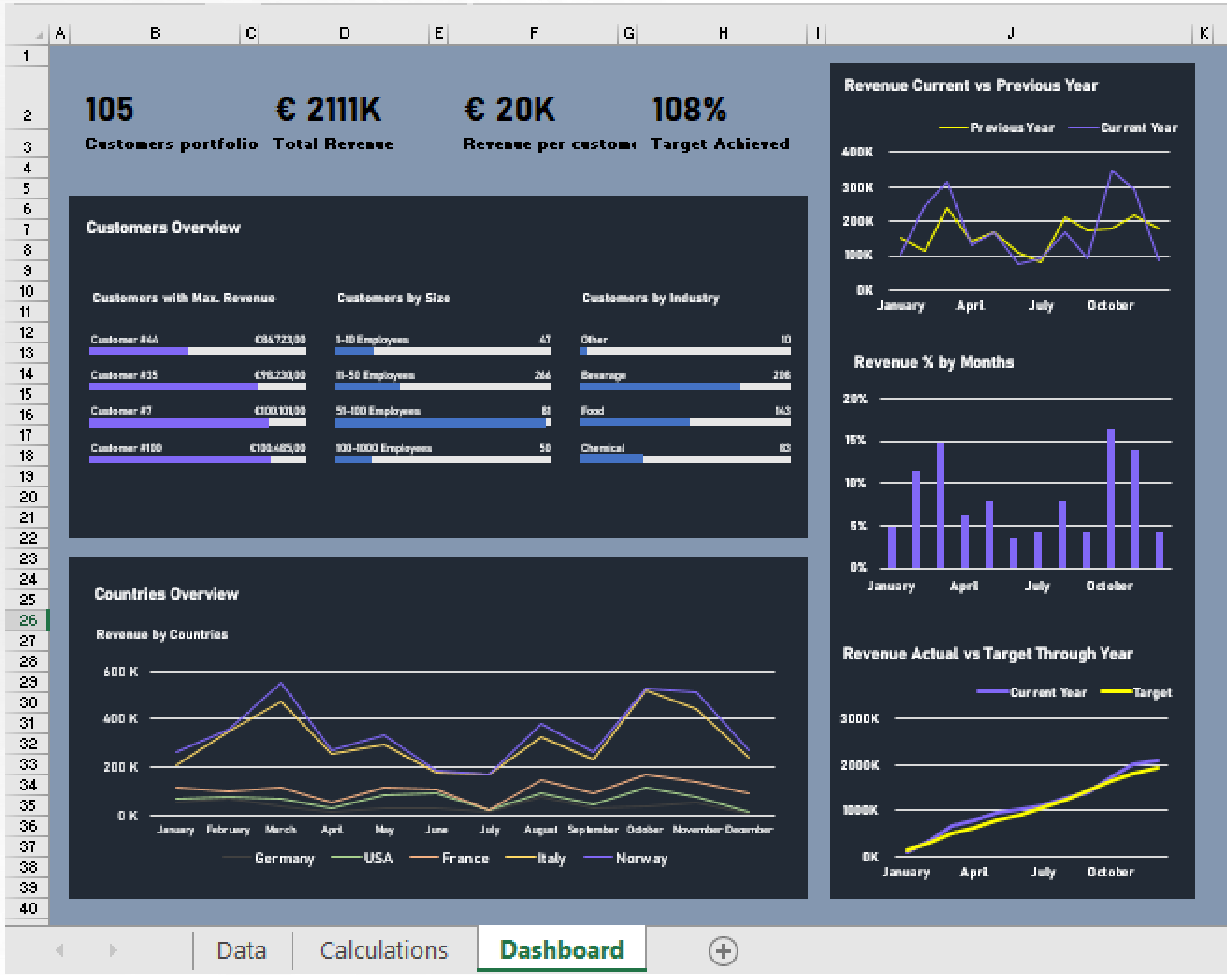
What did I do

- Create KPIs (Key Performance Indicators) such as total fatalities occurring after the accident, total fatalities and percentage of the total based on the severity of the accident and maximum number of casualties by vehicle type, and total casualties by vehicle type
- Create a visualization model of monthly trends that shows the comparison of the number of victims in the current year and the previous year, the maximum trend that shows the comparison of the number of victims in the current year and the previous year, the maximum number of victims by road type, the distribution of the number of victims based on road surface, and the relationship between victims by region/location and by day or night

Result

From the KPI (Key Performance Indicator) it can be seen the total number of fatalities that occurred after the accident, the total fatalities and the total percentage based on the severity of the accident and the maximum number of victims by vehicle type, and the total casualties based on the type of vehicle and from the visualization model provided. data can be read based on the factors that cause accidents and can be used as an evaluation for the authorities and the public to be more alert in the future

Company Financial Dashboard

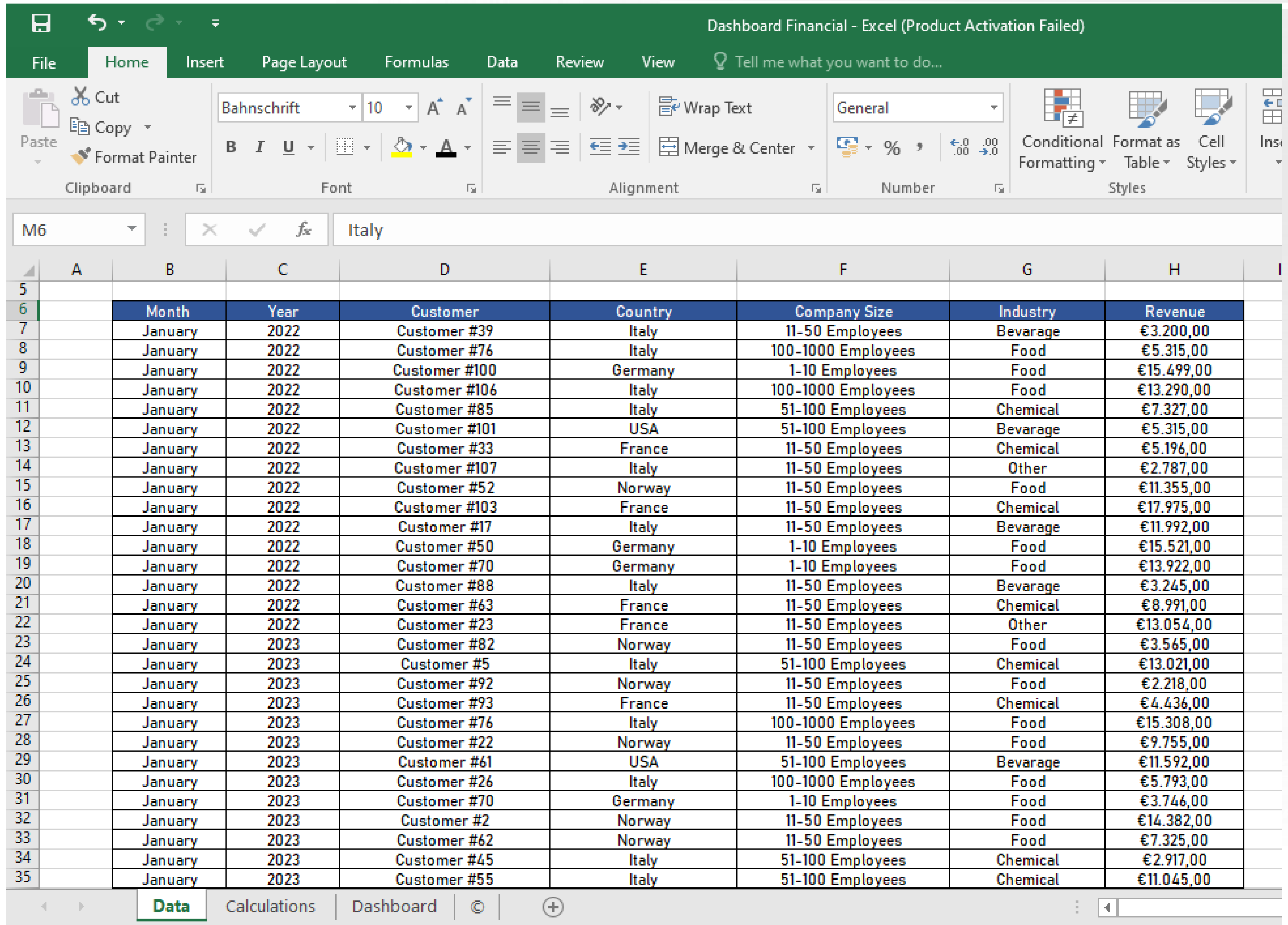


105 Customers portfolio

€ 2111K Total Revenue

€ 20K Revenue per custome

108% Target Achieved



Month	Year	Customer	Country	Company Size	Industry	Revenue
January	2022	Customer #39	Italy	11-50 Employees	Bevarage	€3.200,00
January	2022	Customer #76	Italy	100-1000 Employees	Food	€5.315,00
January	2022	Customer #100	Germany	1-10 Employees	Food	€15.499,00
January	2022	Customer #106	Italy	100-1000 Employees	Food	€13.290,00
January	2022	Customer #85	Italy	51-100 Employees	Chemical	€7.327,00
January	2022	Customer #101	USA	51-100 Employees	Bevarage	€5.315,00
January	2022	Customer #33	France	11-50 Employees	Chemical	€5.196,00
January	2022	Customer #107	Italy	11-50 Employees	Other	€2.787,00
January	2022	Customer #52	Norway	11-50 Employees	Food	€11.355,00
January	2022	Customer #103	France	11-50 Employees	Chemical	€17.975,00
January	2022	Customer #17	Italy	11-50 Employees	Bevarage	€11.992,00
January	2022	Customer #50	Germany	1-10 Employees	Food	€15.521,00
January	2022	Customer #70	Germany	1-10 Employees	Food	€13.922,00
January	2022	Customer #88	Italy	11-50 Employees	Bevarage	€3.245,00
January	2022	Customer #63	France	11-50 Employees	Chemical	€8.991,00
January	2022	Customer #23	France	11-50 Employees	Other	€13.054,00
January	2023	Customer #82	Norway	11-50 Employees	Food	€3.565,00
January	2023	Customer #5	Italy	51-100 Employees	Chemical	€13.021,00
January	2023	Customer #92	Norway	11-50 Employees	Food	€2.218,00
January	2023	Customer #93	France	11-50 Employees	Chemical	€4.436,00
January	2023	Customer #76	Italy	100-1000 Employees	Food	€15.308,00
January	2023	Customer #22	Norway	11-50 Employees	Food	€9.755,00
January	2023	Customer #61	USA	51-100 Employees	Bevarage	€11.592,00
January	2023	Customer #26	Italy	100-1000 Employees	Food	€5.793,00
January	2023	Customer #70	Germany	1-10 Employees	Food	€3.746,00
January	2023	Customer #2	Norway	11-50 Employees	Food	€14.382,00
January	2023	Customer #62	Norway	11-50 Employees	Food	€7.325,00
January	2023	Customer #45	Italy	51-100 Employees	Chemical	€2.917,00
January	2023	Customer #55	Italy	51-100 Employees	Chemical	€11.045,00

Summary

What did I do?

- Create KPIs (Key Performance Indicators) such as number of customer portfolios, total revenue, revenue per customer, and target achievements
- Create a visualization in the form of a bar chart that shows an overview of customers divided into three views, namely based on customers with maximum income, customers by size, and customers by industry origin
- Create visualizations in the form of graphs or line charts to show a general overview of income according to country of origin
- Create visualizations in the form of graphs or line charts to show current revenue vs previous year
- Create a visualization in the form of a histogram to show revenue % by month
- Create visualizations in the form of graphs or line charts to show actual vs target income throughout the year

Result

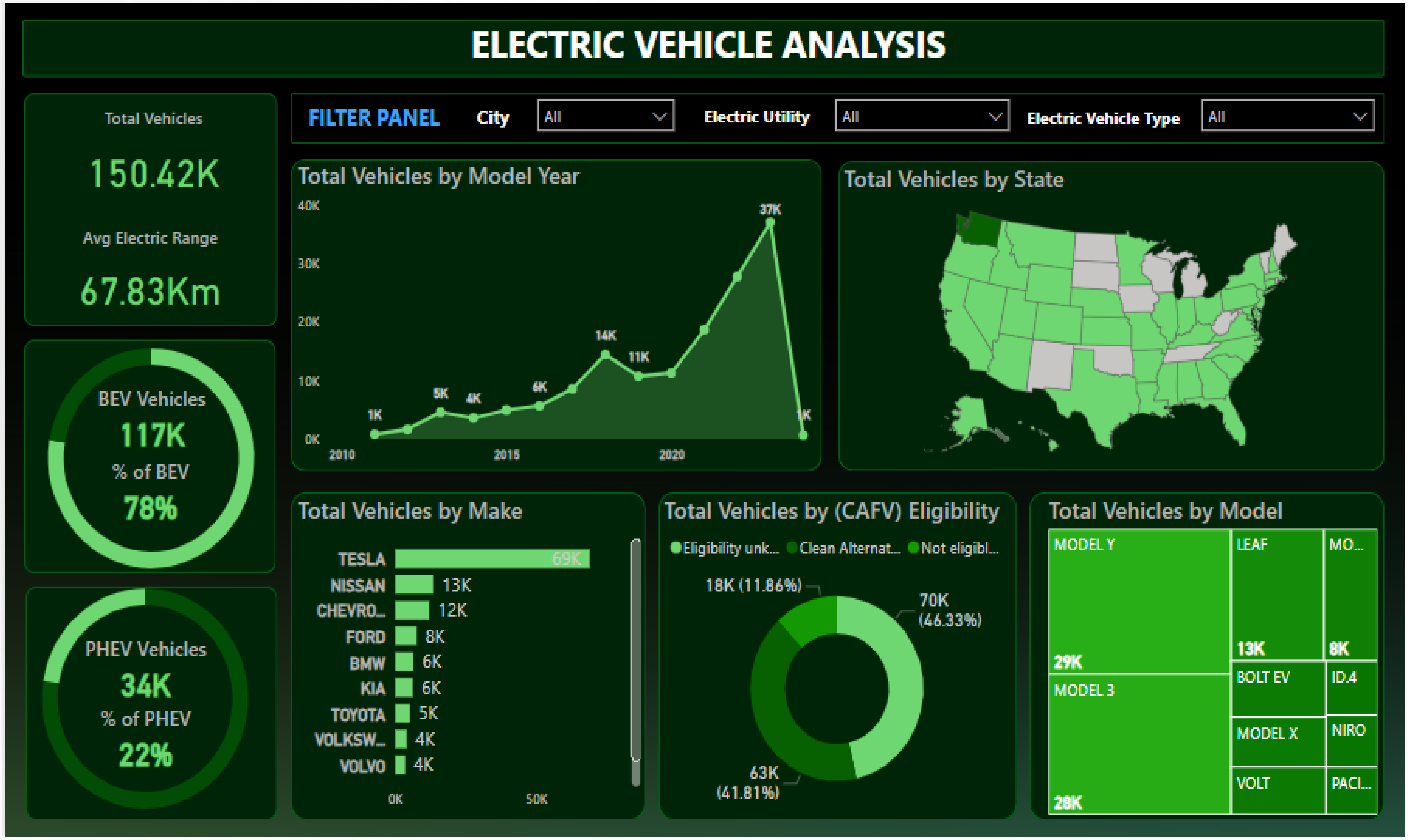
From the KPI (Key Performance Indicator) you can find out the total number of customer portfolios, total revenue, income for each customer, and target achievements. Then, from the results of the visualization, you can find out an overview of customers, an overview of revenue by country, current revenue vs previous year, revenue % by month, and actual revenue vs target throughout the year.

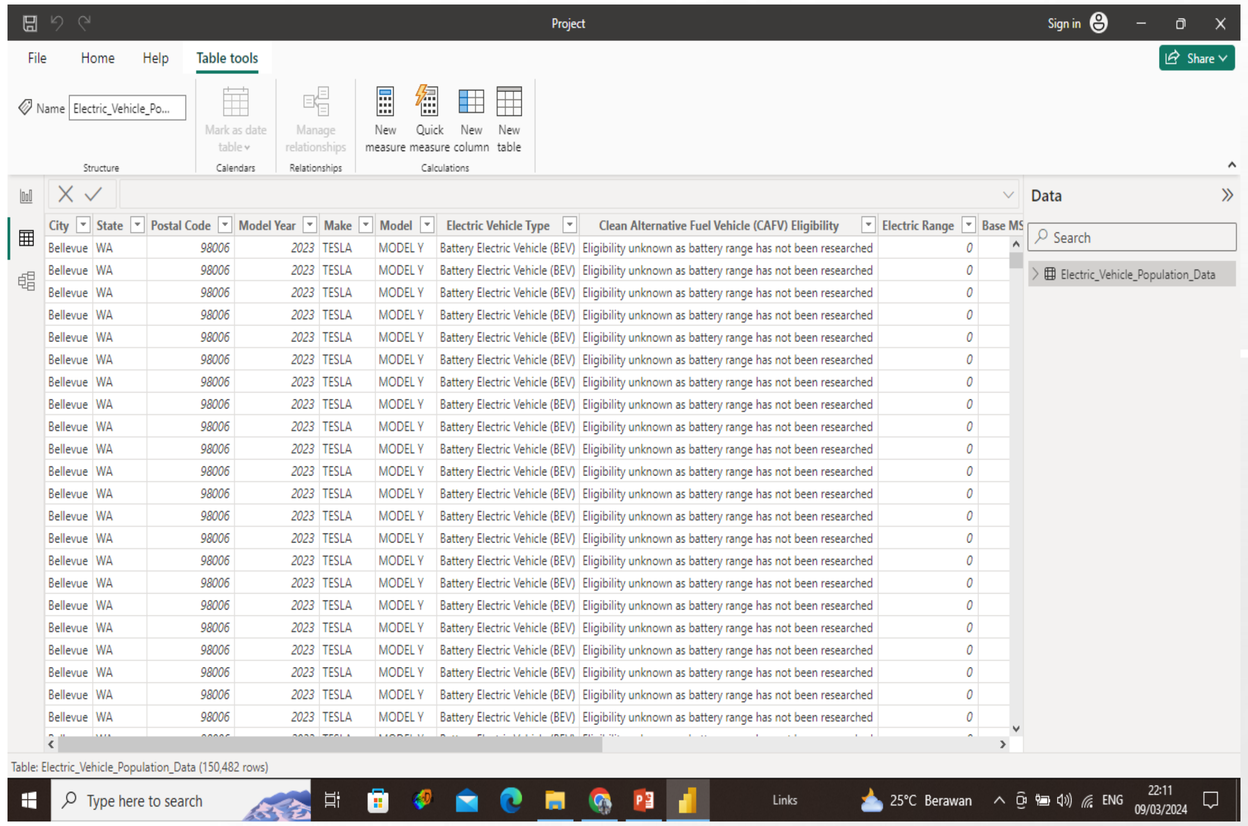
PROJECT 04

Microsoft

Power BI.

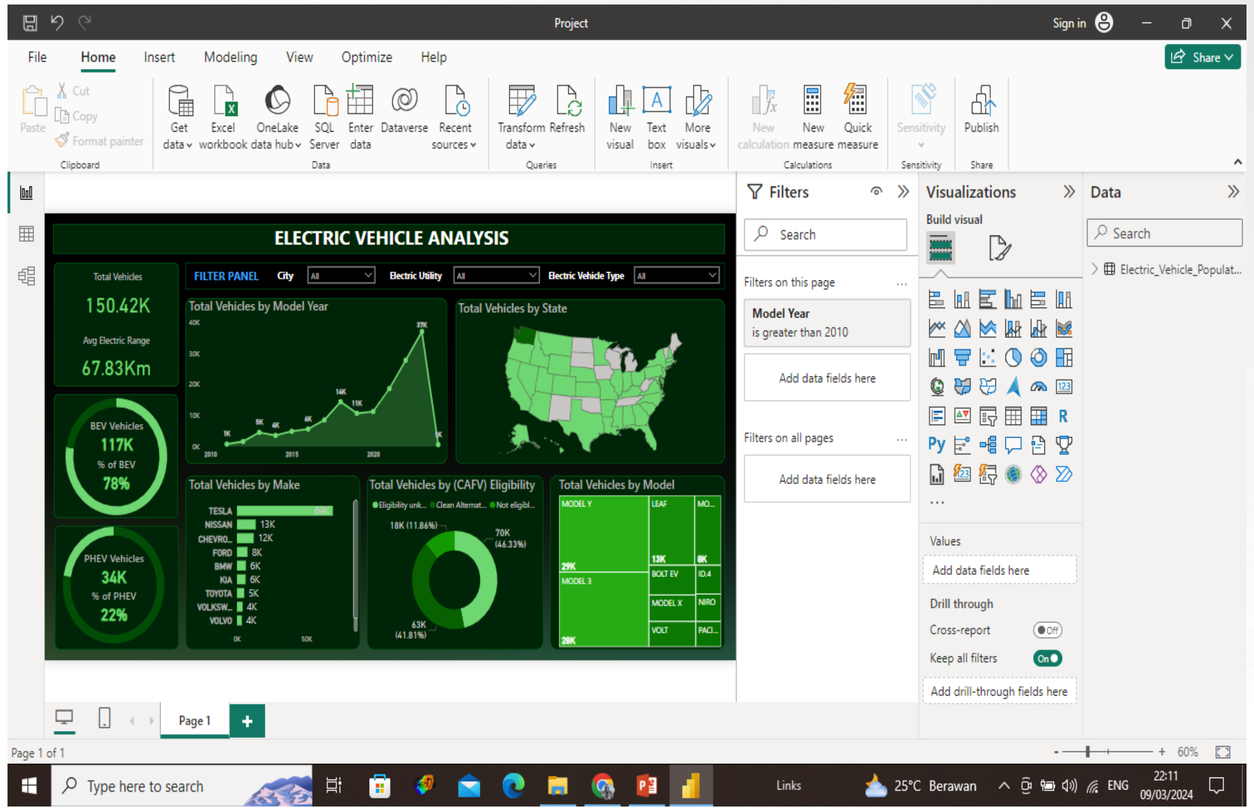
Electric Vehicle Analysis Dashboard





City	State	Postal Code	Model Year	Make	Model	Electric Vehicle Type	Clean Alternative Fuel Vehicle (CAFV) Eligibility	Electric Range	Base MS
Bellevue	WA	98006	2023	TESLA	MODEL Y	Battery Electric Vehicle (BEV)	Eligibility unknown as battery range has not been researched	0	
Bellevue	WA	98006	2023	TESLA	MODEL Y	Battery Electric Vehicle (BEV)	Eligibility unknown as battery range has not been researched	0	
Bellevue	WA	98006	2023	TESLA	MODEL Y	Battery Electric Vehicle (BEV)	Eligibility unknown as battery range has not been researched	0	
Bellevue	WA	98006	2023	TESLA	MODEL Y	Battery Electric Vehicle (BEV)	Eligibility unknown as battery range has not been researched	0	
Bellevue	WA	98006	2023	TESLA	MODEL Y	Battery Electric Vehicle (BEV)	Eligibility unknown as battery range has not been researched	0	
Bellevue	WA	98006	2023	TESLA	MODEL Y	Battery Electric Vehicle (BEV)	Eligibility unknown as battery range has not been researched	0	
Bellevue	WA	98006	2023	TESLA	MODEL Y	Battery Electric Vehicle (BEV)	Eligibility unknown as battery range has not been researched	0	
Bellevue	WA	98006	2023	TESLA	MODEL Y	Battery Electric Vehicle (BEV)	Eligibility unknown as battery range has not been researched	0	
Bellevue	WA	98006	2023	TESLA	MODEL Y	Battery Electric Vehicle (BEV)	Eligibility unknown as battery range has not been researched	0	
Bellevue	WA	98006	2023	TESLA	MODEL Y	Battery Electric Vehicle (BEV)	Eligibility unknown as battery range has not been researched	0	
Bellevue	WA	98006	2023	TESLA	MODEL Y	Battery Electric Vehicle (BEV)	Eligibility unknown as battery range has not been researched	0	
Bellevue	WA	98006	2023	TESLA	MODEL Y	Battery Electric Vehicle (BEV)	Eligibility unknown as battery range has not been researched	0	
Bellevue	WA	98006	2023	TESLA	MODEL Y	Battery Electric Vehicle (BEV)	Eligibility unknown as battery range has not been researched	0	
Bellevue	WA	98006	2023	TESLA	MODEL Y	Battery Electric Vehicle (BEV)	Eligibility unknown as battery range has not been researched	0	
Bellevue	WA	98006	2023	TESLA	MODEL Y	Battery Electric Vehicle (BEV)	Eligibility unknown as battery range has not been researched	0	
Bellevue	WA	98006	2023	TESLA	MODEL Y	Battery Electric Vehicle (BEV)	Eligibility unknown as battery range has not been researched	0	
Bellevue	WA	98006	2023	TESLA	MODEL Y	Battery Electric Vehicle (BEV)	Eligibility unknown as battery range has not been researched	0	
Bellevue	WA	98006	2023	TESLA	MODEL Y	Battery Electric Vehicle (BEV)	Eligibility unknown as battery range has not been researched	0	
Bellevue	WA	98006	2023	TESLA	MODEL Y	Battery Electric Vehicle (BEV)	Eligibility unknown as battery range has not been researched	0	
Bellevue	WA	98006	2023	TESLA	MODEL Y	Battery Electric Vehicle (BEV)	Eligibility unknown as battery range has not been researched	0	
Bellevue	WA	98006	2023	TESLA	MODEL Y	Battery Electric Vehicle (BEV)	Eligibility unknown as battery range has not been researched	0	
Bellevue	WA	98006	2023	TESLA	MODEL Y	Battery Electric Vehicle (BEV)	Eligibility unknown as battery range has not been researched	0	

Table: Electric_Vehicle_Population_Data (150,482 rows)



Summary

What did I do?

- Create requirements for key performance indicators as initial information or as an outline of the problem such as total vehicles, average range of electric vehicles, total BEV vehicles and percentage of total BEV vehicles, total PHEV vehicles and percentage of total PHEV vehicles
- Create charts of Total Vehicles by Model Year (From 2010 onwards), Total Vehicles by State, Top 10 Total Vehicles by Brand, Total Vehicles by CAFV (Clean Alternative Fuel Vehicle) Eligibility, and Top 10 Total Vehicles by Model.
- Create visualization models using Power BI that are attractive and easy to read

What is the project about??

From the performance Requirements Indicators, we want to understand the entire electric vehicle landscape, which includes BEVs and PHEVs, to assess market size and growth, determine the average electric range of electric vehicles in a data set to measure the technological progress and efficiency of electric vehicles and identify and analyze the number of total Battery Electric Vehicles (BEV) and Hybrid vehicles (PHEV) in the data set and determine the percentage of total BEV and PHEV vehicles

Result

From the Performance Requirements Indicator we can find out the total number of BEV and PHEV vehicles to see the growth of the electric vehicle market, the average electric range in kilometers to see the efficiency of the technology used, and the total number of each BEV and PHEV vehicle along with the percentage of the total vehicles

My Contact

☎ +6281330173085

mnafis48@gmail.com

www.linkedin.com/in/muhnafis

Malang, Indonesia

thank you.

Let's Work Together