Northwind Database Analysis

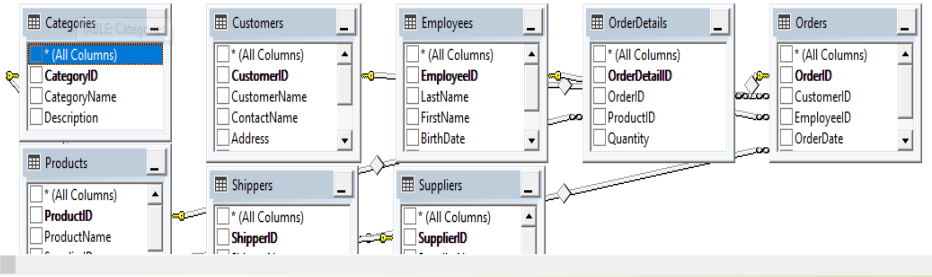
By:, El-Sayed El-Sawaf, Hamdy Khaled, Mahmoud Elsaeed, Adel Wasel

- analyze Northwind Database using SQL, Excel, Power BI, and Python.

Overview

Northwind Database consists of 8 Tables:

- **Categories:** Category ID, Category Name, Description
- **Customers:** Customer ID, Customer name, Contact name, Address, City, Postal code, Country
- **Employees:** Employee ID, Last name, First name, Birth date, Photo, Notes
- **Order Details:** OrderDetails ID, Order ID, Product Quantity
- **Order:** Order ID, Customer ID, Employee ID, Orde date, Shipper ID
- **Products:** Product ID, Product name, Supplier ID, Category ID, Unit, Price
- **Shippers:** Shipper ID, Shipper name, Phone
- **Suppliers:** Supplier ID, Supplier name, Contact name, Address, City, Postal code, Country, Phone



Our Analytical Approach

Step 1: Data Exploration

Step 2: Data Cleaning

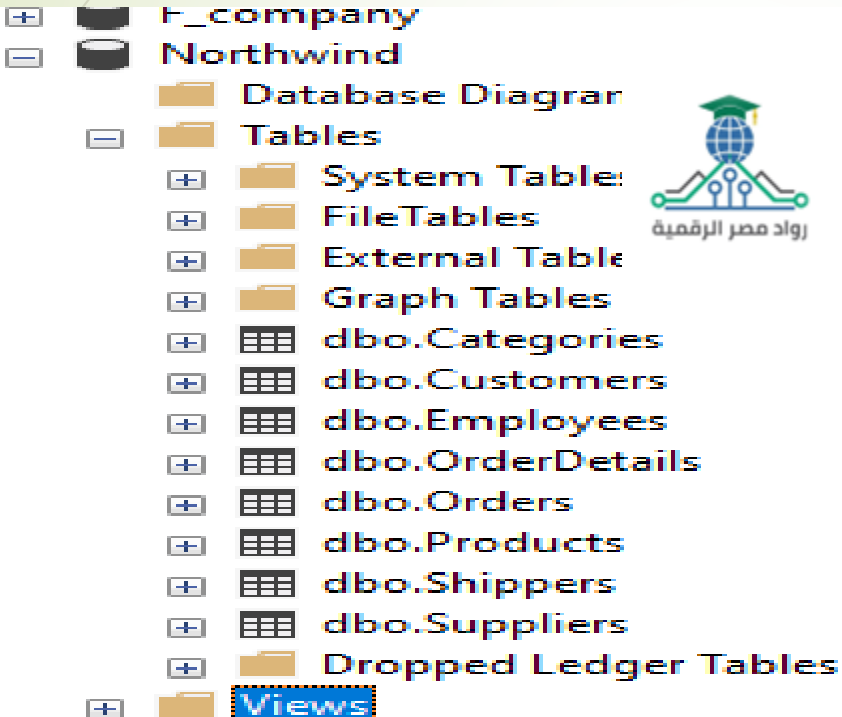
Step 3: Define the Question

Step 4: Data Analysis

Step 5: Visualization

Our Analytical Approach

Step 1: Data Exploration → Step 2: Data Cleaning → Step 3: Define the Question → Step 4: Data Analysis → Step 5: Visualization



Our Analytical Approach

Step 1: Data Exploration → Step 2: Data Cleaning → Step 3: Define the Question → Step 4: Data Analysis → Step 5: Visualization

```
-- Creating Realtion Betwwen Tables --

SELECT      Orders.OrderID, Customers.CustomerID, Employees.EmployeeID, Products.ProductID, OrderDetails.OrderDetailID, Suppliers.SupplierID, Categories.CategoryID, Shippers.S
            Employees.LastName, Employees.FirstName, OrderDetails.Quantity, Products.Unit, Products.Price, Suppliers.SupplierName, Shippers.ShipperName, Suppliers.(
            Categories.CategoryName
FROM         Orders INNER JOIN
            Customers ON Orders.CustomerID = Customers.CustomerID INNER JOIN
            Employees ON Orders.EmployeeID = Employees.EmployeeID INNER JOIN
            OrderDetails ON Orders.OrderID = OrderDetails.OrderID INNER JOIN
            Products ON OrderDetails.ProductID = Products.ProductID INNER JOIN
            Categories ON Products.CategoryID = Categories.CategoryID INNER JOIN
            Shippers ON Orders.ShipperID = Shippers.ShipperID INNER JOIN
            Suppliers ON Products.SupplierID = Suppliers.SupplierID
```

Our Analytical Approach

Step 1: Data Exploration → Step 2: Data Cleaning → **Step 3: Define the Question** → Step 4: Data Analysis → Step 5: Visualization

SQLQuery Final Project Team work.sql - DESKTOP-FN6QI35.Northwind (DESKTOP-FN6QI35\YB Store (72)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

New Query | Northwind | Execute

DESKTOP-FN6QI35....dbo.Northwindview* | DESKTOP-FN6QI35....wind - dbo.View_1 | Object Explorer | SQLQuery Final Proj...QI35\YB Sto

```
use Northwind
-- cleaning---

--Quations--

select * from Customers
--1.     Show the list of all customers from 'Germany?
select * from Customers
where Country = 'Germany'

--2.     What are the names of the products supplied by 'Exotic Liquid' ?
select productname,suppliername
from Products as pro
inner join Suppliers as sup
on pro.SupplierID=sup.SupplierID
where SupplierName='Exotic Liquid'
select * from Customers
select * from orders
select * from OrderDetails
-- 3     Which employee has the most orders?
select top 1 emp.EmployeeID, concat(emp.FirstName,' ',emp.LastName )As Full_name,count(*)AS Order_count
from Employees as emp
join Orders as o
on emp.EmployeeID=o.EmployeeID

group by emp.EmployeeID,emp.FirstName,emp.LastName
order by Order_count desc

--4-     List the products that have never been ordered?

```

Our Analytical Approach

Step 1: Data Exploration → Step 2: Data Cleaning → **Step 3: Define the Question** → Step 4: Data Analysis → Step 5: Visualization

Import libraries and read in the NorthWind xlsx file

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

```
## Read in the NorthWind xlsx file
df = pd.read_excel('/content/Northwind.python view.xlsx')
```

2 Explore Data

```
#1- read data
df
```

	OrderID	CustomerID	EmployeeID	ProductID	OrderDetailID	SupplierID	CategoryID	ShipperID	CustomersCity	CustomersCountry	...	Quantity	Unit	Price	SupplierName
0	10248	90	5	11	1	5	4	3	Helsinki	Finland	...	12	1 kg pkg.	21	Cooperativa de Quesos 'Las Cabras'
1	10248	90	5	42	2	20	5	3	Helsinki	Finland	...	10	32 - 1 kg pkgs.	14	Leka Trading
2	10248	90	5	72	3	14	4	3	Helsinki	Finland	...	5	24 - 200 g pkgs.	35	Formaggi Fortini s.r.l.
3	10249	81	6	14	4	6	7	1	Sao Paulo	Brazil	...	9	40 - 100 g pkgs.	23	Mayumi's
4	10249	81	6	51	5	24	7	1	Sao Paulo	Brazil	...	40	50 - 300 g pkgs.	53	G'day, Mate

```
#2- info about data
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 518 entries, 0 to 517
Data columns (total 22 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   OrderID          518 non-null    int64
 1   CustomerID       518 non-null    int64
 2   EmployeeID       518 non-null    int64
 3   ProductID        518 non-null    int64
 4   OrderDetailID    518 non-null    int64
 5   SupplierID       518 non-null    int64
 6   CategoryID       518 non-null    int64
 7   ShipperID        518 non-null    int64
 8   CustomersCity    518 non-null    object
 9   CustomersCountry 518 non-null    object
 10  LastName         518 non-null    object
 11  FirstName        518 non-null    object
 12  Quantity         518 non-null    int64
 13  Unit             518 non-null    object
 14  Price            518 non-null    int64
 15  SupplierName     518 non-null    object
 16  ShipperName      518 non-null    object
 17  SupplierCity     518 non-null    object
 18  SuppliersCountry 518 non-null    object
 19  CategoryName     518 non-null    object
 20  ProductName      518 non-null    object
 21  OrderDate        518 non-null    datetime64[ns]
dtypes: datetime64[ns](1), int64(10), object(11)
memory usage: 89.2+ KB
```

For Number

For Text

Columns

Rows

```
[5] #3- shape of data
    df.shape
```

```
(518, 22
```

```
[6] #4- Name of columns
    df.columns
```

```
Index(['OrderID', 'CustomerID', 'EmployeeID', 'ProductID', 'OrderDetailID',
       'SupplierID', 'CategoryID', 'ShipperID', 'CustomersCity',
       'CustomersCountry', 'LastName', 'FirstName', 'Quantity', 'Unit',
       'Price', 'SupplierName', 'ShipperName', 'SupplierCity',
       'SuppliersCountry', 'CategoryName', 'ProductName', 'OrderDate'],
      dtype='object')
```

```
#5- Instical Summary
df.describe().T #Transpose
```

	count	mean	min	25%	50%	75%	max	std
OrderID	518.0	10344.791506	10248.0	10296.0	10344.0	10393.0	10443.0	56.461272
CustomerID	518.0	47.685328	2.0	24.0	49.0	69.0	91.0	25.544277
EmployeeID	518.0	4.351351	1.0	3.0	4.0	6.0	9.0	2.40408
ProductID	518.0	42.055985	1.0	24.0	41.0	61.0	77.0	21.992322
OrderDetailID	518.0	259.5	1.0	130.25	259.5	388.75	518.0	149.677988
SupplierID	518.0	14.519305	1.0	8.0	14.0	21.0	29.0	8.096529
CategoryID	518.0	4.092664	1.0	2.0	4.0	6.0	8.0	2.295211
ShipperID	518.0	2.061776	1.0	1.0	2.0	3.0	3.0	0.796539
Quantity	518.0	24.600386	1.0	10.0	20.0	32.75	120.0	18.376899
Price	518.0	29.26834	3.0	13.0	20.0	35.0	264.0	33.937982
OrderDate	518	1996-10-29 03:39:36.833976832	1996-07-04 00:00:00	1996-09-03 00:00:00	1996-11-01 00:00:00	1996-12-25 00:00:00	1997-02-12 00:00:00	NaN

```
#6- Check missing values
df.isnull().sum()
```

	0
OrderID	0
CustomerID	0
EmployeeID	0
ProductID	0
OrderDetailID	0
SupplierID	0
CategoryID	0
ShipperID	0
CustomersCity	0
CustomersCountry	0
LastName	0
FirstName	0
Quantity	0
Unit	0
Price	0
SupplierName	0
ShipperName	0
SupplierCity	0
SuppliersCountry	0
CategoryName	0
ProductName	0
OrderDate	0

dtype: int64

Check Missing Value

```
#8- data types
df.dtypes
```

	0
OrderID	int64
CustomerID	int64
EmployeeID	int64
ProductID	int64
OrderDetailID	int64
SupplierID	int64
CategoryID	int64
ShipperID	int64
CustomersCity	object
CustomersCountry	object
LastName	object
FirstName	object
Quantity	int64
Unit	object
Price	int64
SupplierName	object
ShipperName	object
SupplierCity	object
SuppliersCountry	object
CategoryName	object
ProductName	object
OrderDate	datetime64[ns]

dtype: object

Check Data Type

```
#9- unique values and value count for each column
df.nunique()
```

	0
OrderID	196
CustomerID	74
EmployeeID	9
ProductID	77
OrderDetailID	518
SupplierID	29
CategoryID	8
ShipperID	3
CustomersCity	58
CustomersCountry	21
LastName	9
FirstName	9
Quantity	43
Unit	70
Price	43
SupplierName	29
ShipperName	3
SupplierCity	29
SuppliersCountry	16
CategoryName	8
ProductName	77
OrderDate	160

dtype: int64

Check unique Value & Count

[8]
```
#5- Instical Summary
df.describe().T #Transpose
```

For Number

	count	mean	min	25%	50%	75%	max	std
OrderID	518.0	10344.791506	10248.0	10296.0	10344.0	10393.0	10443.0	56.461272
CustomerID	518.0	47.685328	2.0	24.0	49.0	69.0	91.0	25.544277
EmployeeID	518.0	4.351351	1.0	3.0	4.0	6.0	9.0	2.40408
ProductID	518.0	42.055985	1.0	24.0	41.0	61.0	77.0	21.992322
OrderDetailID	518.0	259.5	1.0	130.25	259.5	388.75	518.0	149.677988
SupplierID	518.0	14.519305	1.0	8.0	14.0	21.0	29.0	8.096529
CategoryID	518.0	4.092664	1.0	2.0	4.0	6.0	8.0	2.295211
ShipperID	518.0	2.061776	1.0	1.0	2.0	3.0	3.0	0.796539
Quantity	518.0	24.600386	1.0	10.0	20.0	32.75	120.0	18.376899
Price	518.0	29.26834	3.0	13.0	20.0	35.0	264.0	33.937982
OrderDate	518	1996-10-29 03:39:36.833976832	1996-07-04 00:00:00	1996-09-03 00:00:00	1996-11-01 00:00:00	1996-12-25 00:00:00	1997-02-12 00:00:00	NaN

```
#6- Instical Summary
df.describe(include = 'object').T
```

For Text

	count	unique	top	freq
CustomersCity	518	58	Graz	35
CustomersCountry	518	21	USA	76
LastName	518	9	Peacock	123
FirstName	518	9	Margaret	123
Unit	518	70	24 - 12 oz bottles	27
SupplierName	518	29	Plutzer Lebensmittelgroßmärkte AG	42
ShipperName	518	3	United Package	188
SupplierCity	518	29	Frankfurt	42
SuppliersCountry	518	16	Germany	65

Ask Question

Which employee has the most orders?

```
import pandas as pd

# Load the data

file_path = 'Northwind.python view.xlsx'
data = pd.read_excel(file_path, sheet_name='Query1')

# Count the number of orders per employee

Employee_O_Count = data.groupby('EmployeeID')['OrderID'].nunique()

# Find the employee with the most orders

most_orders_employee = Employee_O_Count.idxmax()
most_orders_count = Employee_O_Count.max()

# Display The results

print(f"Employee ID {most_orders_employee} has the most orders with {most_orders_count} orders.")
```

```
Employee ID 4 has the most orders with 40 orders.
```

Which orders contain more than 5 different products?

```
import pandas as pd

# Load the data

file_path = 'Northwind.python view.xlsx'
data = pd.read_excel(file_path, sheet_name='Query1')

# Group by of unique products

Orders_p_count = data.groupby('OrderID')['ProductID'].nunique()

# Filter orders

Five_products = Orders_p_count[Orders_p_count > 5]

# Display the results

print(Five_products)
```

```
Series([], Name: ProductID, dtype: int64)
```

Show the list of all customers from 'Germany'?

```
[16] import pandas as pd

# Load the data

file_path = 'Northwind.python view.xlsx'
data = pd.read_excel(file_path, sheet_name='Query1')

# Filter for customers from Germany and list unique

customers_germany = data[data['CustomersCountry'] == 'Germany']['CustomerID'].unique()

# Display The results

print(customers_germany)
```

```
[25 63 52 44 86 39 17 56 79]
```

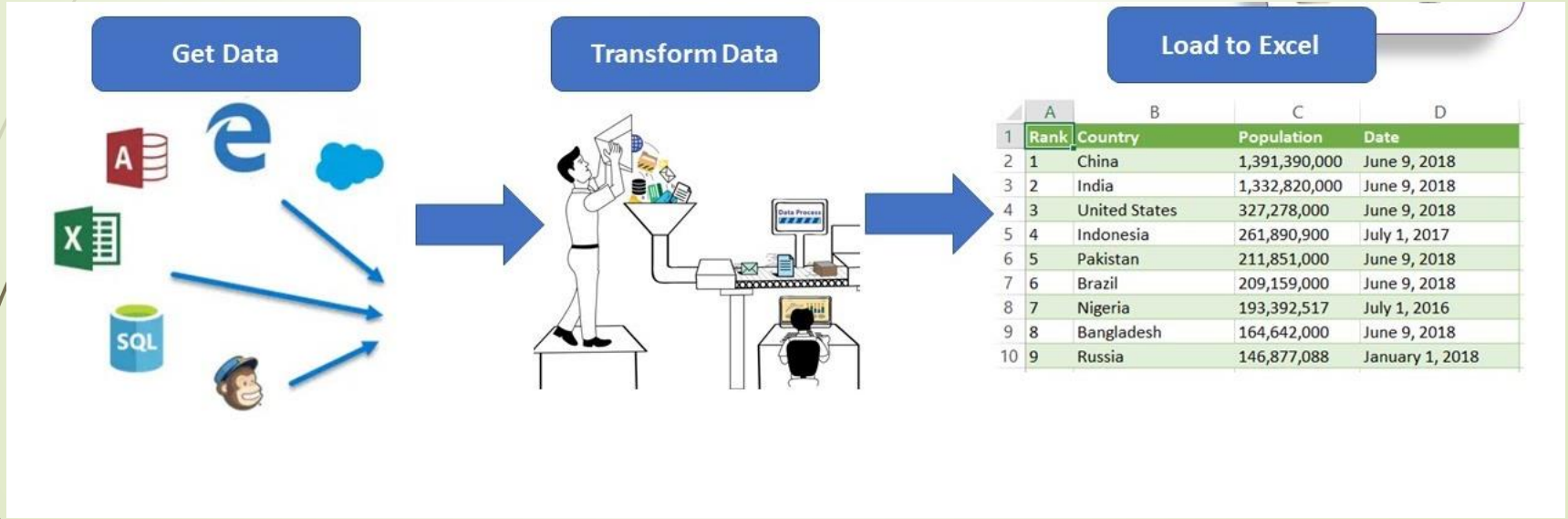
What are the names of the products supplied by 'Exotic Liquid?

```
import pandas as pd

# Load the data

file_path = 'Northwind.python view.xlsx'
data = pd.read_excel(file_path, sheet_name='Query1')

# Filter products supplied by 'Exotic Liquid'

products_exotic_liquid = data[data['SupplierName'] == 'Exotic Liquid']['ProductName'].unique()

# Display The results

print(products_exotic_liquid)
```

```
['Chang' 'Chais' 'Aniseed Syrup']
```

Our Analytical Approach

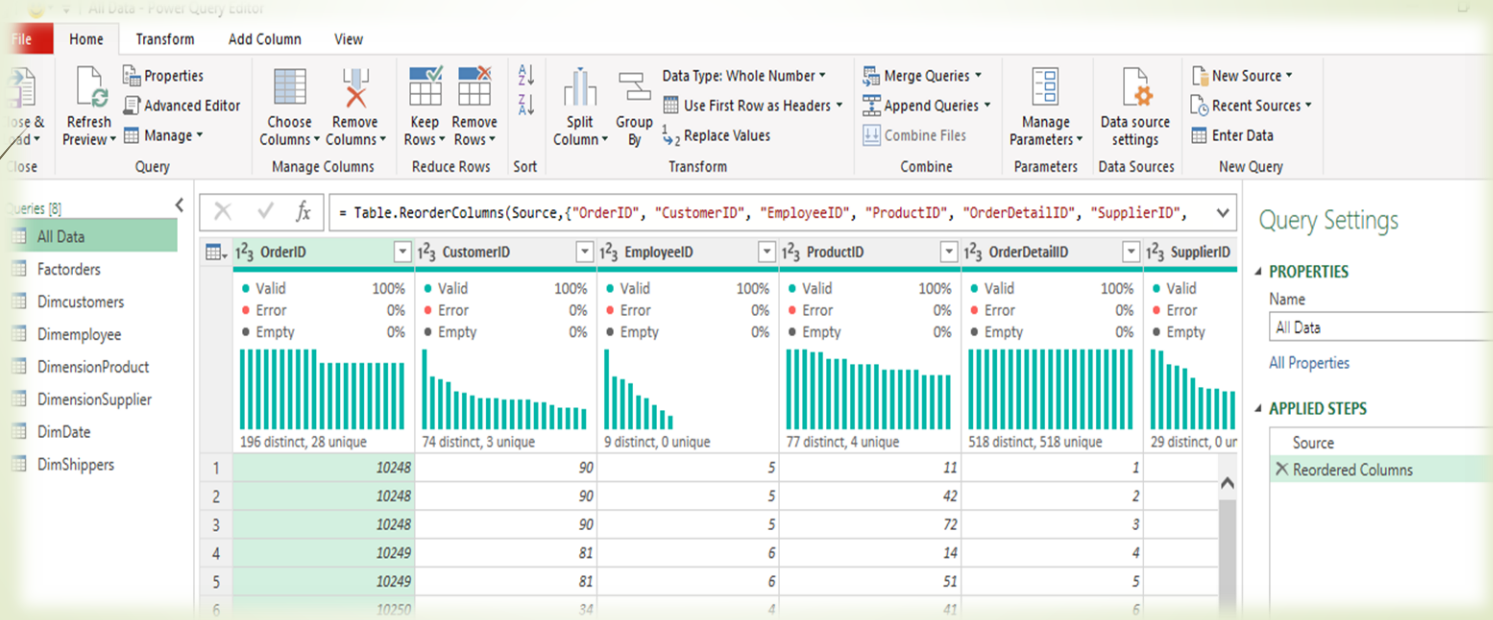
Step 1: Data Exploration → Step 2: Data Cleaning → Step 3: Define the Question → Step 4: Data Analysis → Step 5: Visualization



	A	B	C	D
1	Rank	Country	Population	Date
2	1	China	1,391,390,000	June 9, 2018
3	2	India	1,332,820,000	June 9, 2018
4	3	United States	327,278,000	June 9, 2018
5	4	Indonesia	261,890,900	July 1, 2017
6	5	Pakistan	211,851,000	June 9, 2018
7	6	Brazil	209,159,000	June 9, 2018
8	7	Nigeria	193,392,517	July 1, 2016
9	8	Bangladesh	164,642,000	June 9, 2018
10	9	Russia	146,877,088	January 1, 2018

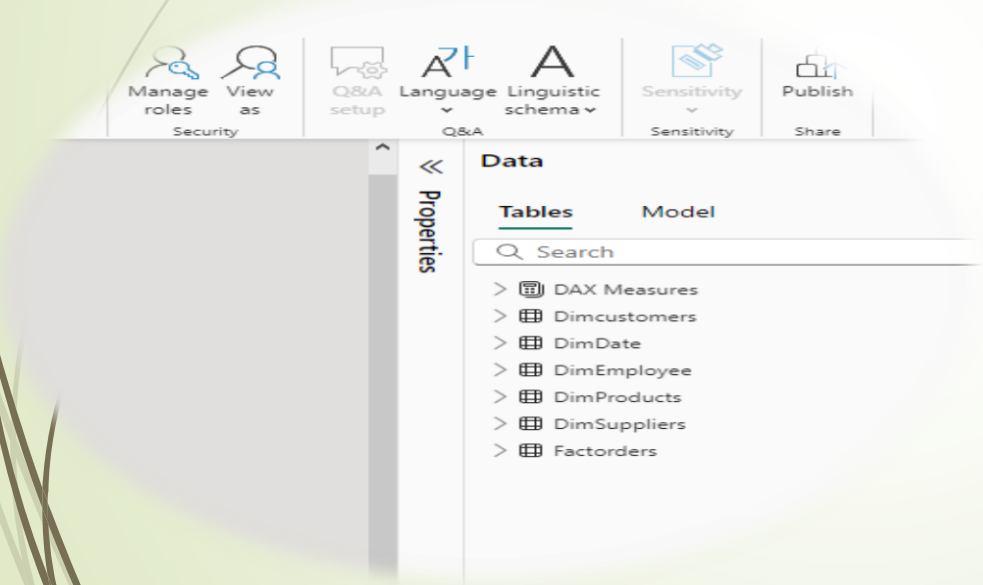
Our Analytical Approach

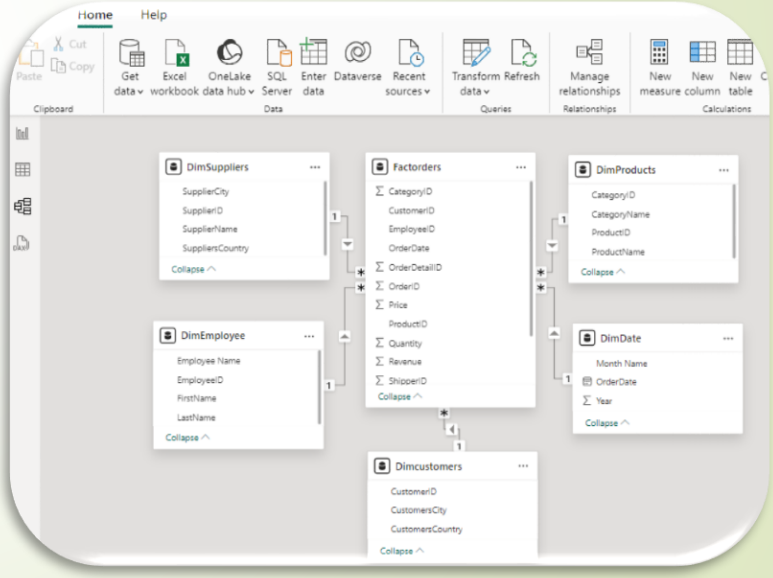
Step 1: Data Exploration → Step 2: Data Cleaning → Step 3: Define the Question → Step 4: Data Analysis → Step 5: Visualization



Our Analytical Approach

Step 1: Data Exploration → **Step 2:** Data Cleaning → **Step 3:** Define the Question → **Step 4:** Data Analysis → **Step 5:** Visualization





Our Analytical Approach

Step 1: Data Exploration → **Step 2: Data Cleaning** → **Step 3: Define the Question** → **Step 4: Data Analysis** → **Step 5: Visualization**

Country	Total Sales
USA	69722
Austria	51687
Germany	47316
Brazil	40272
Canada	31402
France	29559
Denmark	17954
UK	16764
Ireland	15405
Venezuela	13559
Sweden	9736
Switzerland	8109
Belgium	8075
Finland	6478
Mexico	5882
Italy	4339
Spain	4316
Portugal	4169
Norway	1341
Poland	585
Argentina	396
Grand Total	387066

City	Total Sales
Graz	35630
Sao Paulo	26269
Montréal	23434
Boise	22516
Albuquerque	18442
Cunewalde	18171
Salzburg	16057
London	15630
Cork	15405
Strasbourg	15268
Grand Total	206822

Supplier	Total Sales
Exotic Liquid	95843
Grandma Kelly	135876
New Orleans C	155347
Grand Total	387066

Category Name	Sum of Quantity	Total Sales
Beverages	2289	99699
Dairy Products	2601	70530
Confections	2110	54729
Meat/Poultry	1288	51636
Condiments	1383	35024
Seafood	1445	29639
Produce	715	23439
Grains/Cereals	912	22370
Grand Total	12743	387066

Employee Name	Sum of Quantity	Total Sales
Margaret Peacock	3232	105926
Nancy Davolio	1924	57765
Janet Leverling	1725	42823
Robert King	733	39843
Laura Callahan	1293	39341
Andrew Fuller	1315	32559
Steven Buchanan	778	27606
Michael Suyama	1094	25501
Anne Dodsworth	649	15702

Top 10Product Name	Total Quantity
Côte de Blaye	63096
Thüringer Rostbratwurst	20832
Raclette Courdavault	19030
Tarte au sucre	15925
Camembert Pierrot	14620
Alice Mutton	12909
Gnocchi di nonna Alice	10222
Mozzarella di Giovanni	9450
Vegie-spread	9196
Rssle Sauerkraut	8694
Grand Total	183974

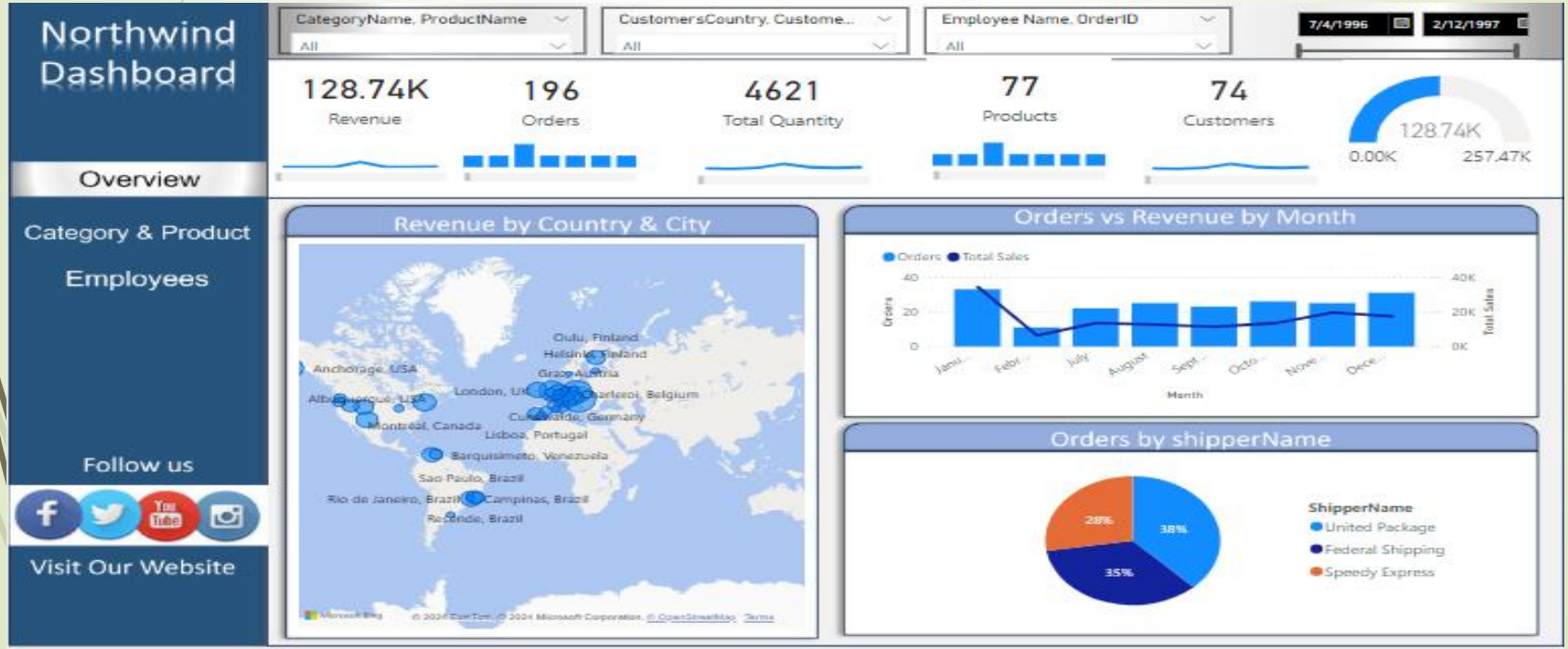
Product Name	Sum of Multiplication
Federal Shipping	387066
Speedy Express	387066
United Package	387066
Grand Total	387066

Product Name	Sum of Multiplication

s

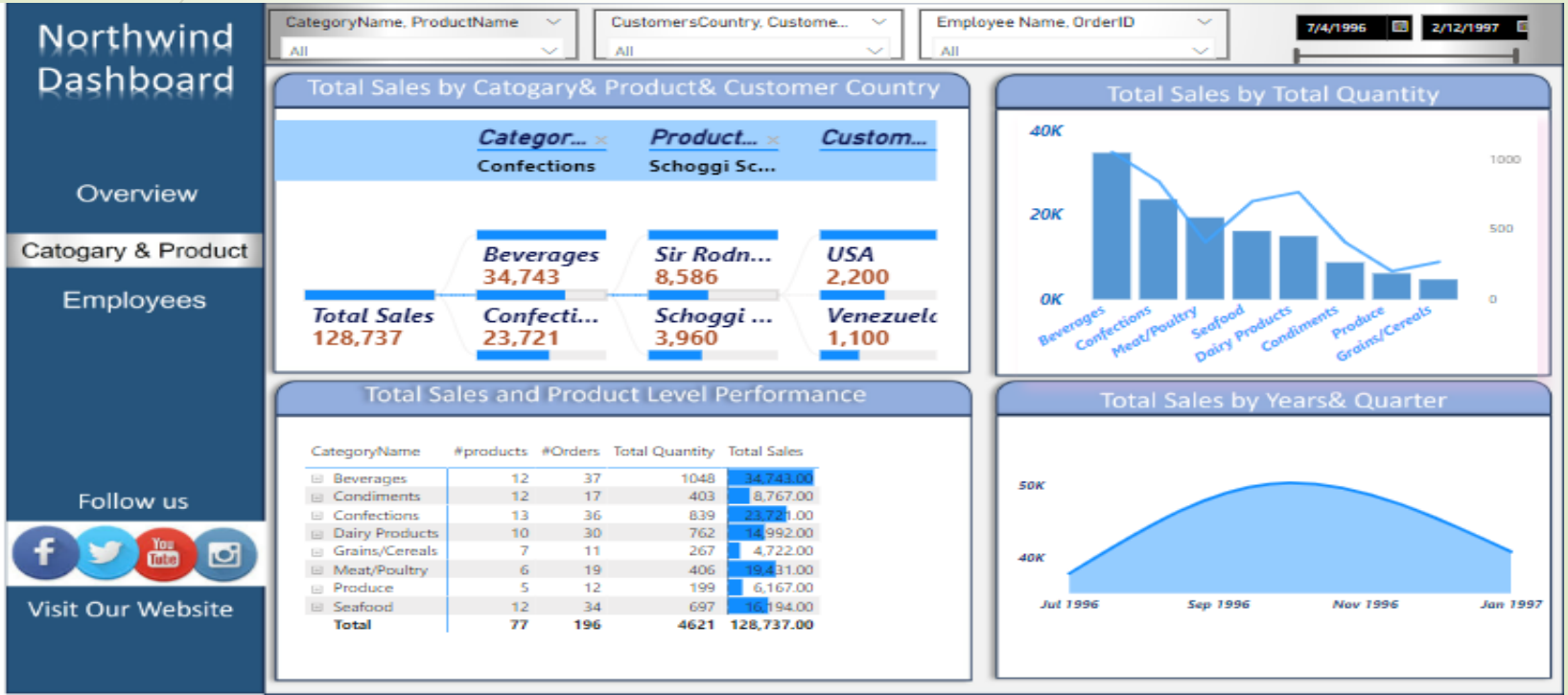
Our Analytical Approach

Step 1: Data Exploration → **Step 2:** Data Cleaning → **Step 3:** Define the Question → **Step 4:** Data Analysis → **Step 5:** Visualization



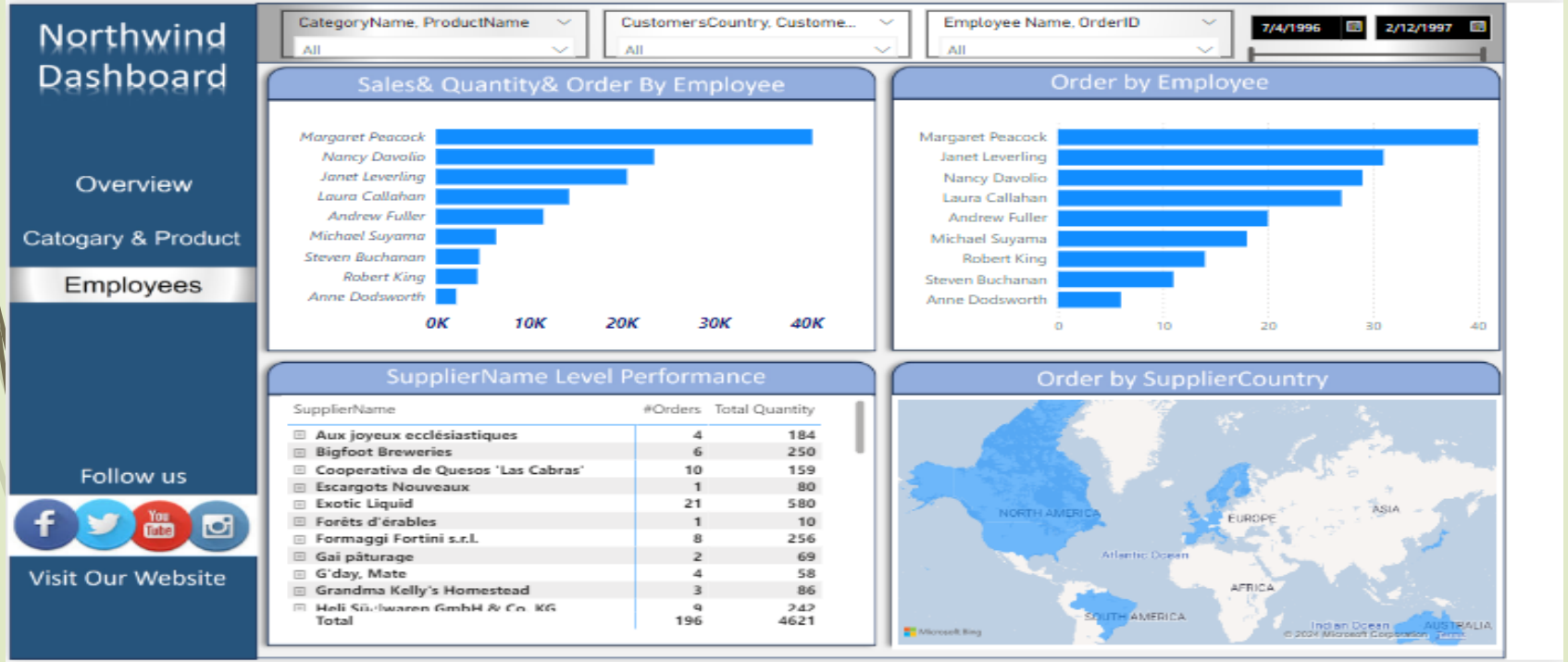
Our Analytical Approach

Step 1: Data Exploration → **Step 2:** Data Cleaning → **Step 3:** Define the Question → **Step 4:** Data Analysis → **Step 5:** Visualization



Our Analytical Approach

Step 1: Data Exploration → **Step 2:** Data Cleaning → **Step 3:** Define the Question → **Step 4:** Data Analysis → **Step 5:** Visualization



Our Analytical Approach

Step 1: Data Exploration → Step 2: Data Cleaning → Step 3: Define the Question → Step 4: Data Analysis → Step 5: Visualization

ShipperName	Federal Shipping		Speedy Express		United Package		Total	
CustomersCountry	Total Quantity	Total Sales	Total Quantity	Total Sales	Total Quantity	Total Sales	Total Quantity	Total Sales
Argentina			12	276.00			12	276.00
Austria	143	3,429.00	207	9,437.00	130	2,025.00	480	14,891.00
Belgium					80	4,800.00	80	4,800.00
Brazil	97	1,239.00	71	2,863.00	132	3,902.00	300	8,004.00
Canada	89	967.00			218	6,812.00	307	7,779.00
Denmark	154	14,508.00					154	14,508.00
Finland	84	1,778.00	45	1,065.00	20	460.00	149	3,303.00
France	56	2,724.00	147	2,965.00	126	2,260.00	329	7,949.00
Germany	268	4,702.00	180	4,430.00	324	7,640.00	772	16,772.00
Ireland	98	1,508.00	20	440.00	87	1,533.00	205	3,481.00
Italy			80	1,488.00	30	510.00	110	1,998.00
Mexico	78	2,025.00			54	1,104.00	132	3,129.00
Norway					15	75.00	15	75.00
Poland	30	390.00					30	390.00
Portugal	47	1,609.00	10	100.00	18	396.00	75	2,105.00
Spain	10	260.00	12	355.00	69	1,353.00	91	1,968.00
Sweden	69	3,065.00	37	535.00	22	430.00	128	4,030.00
Switzerland	20	380.00			90	2,385.00	110	2,765.00
UK	147	4,896.00	70	1,355.00	54	766.00	271	7,017.00
USA	168	5,186.00	161	1,771.00	303	10,172.00	632	17,129.00
Venezuela	99	3,730.00	140	2,638.00			239	6,368.00

Conclusion

Key Findings:

- **USA** is the top-selling country for all three companies.
- **Federal Shipping** has the highest total sales and total quantity sold
- **Denmark** has the highest total sales and quantity sold
- **Beverages** have the highest total sales
- **United Package** has the lowest total sales and total quantity sold.
- **Produce** has the lowest total sales
- **Argentina** has the lowest total sales and quantity sold among all countries.
- **Product-Country Relationships:** Some countries might prefer specific product categories. For example, **Austria** might have a high demand for **Beverages**.

Recommendations:

- **Federal Shipping:** Continue to leverage its market leadership position by focusing on customer satisfaction, product innovation, and efficient operations.
- **Target High-Performing Countries and Cities:** Focus marketing efforts and product offerings in regions with high sales potential.
- **Leverage Employee Expertise:** Assign employees to countries or regions where they have a proven track record of success

Our Team

El Sayed El Sawaf

Hamdy Khaled

Mahmoud El Saeed

Adel Wasel

Thanks