My perspective on how it works.

Before you can start with the ETL process you need to have a operational database and the data warehouse which you made from the star scheme based on the 4-step dimensional design process. Then you follow the steps **E T L**

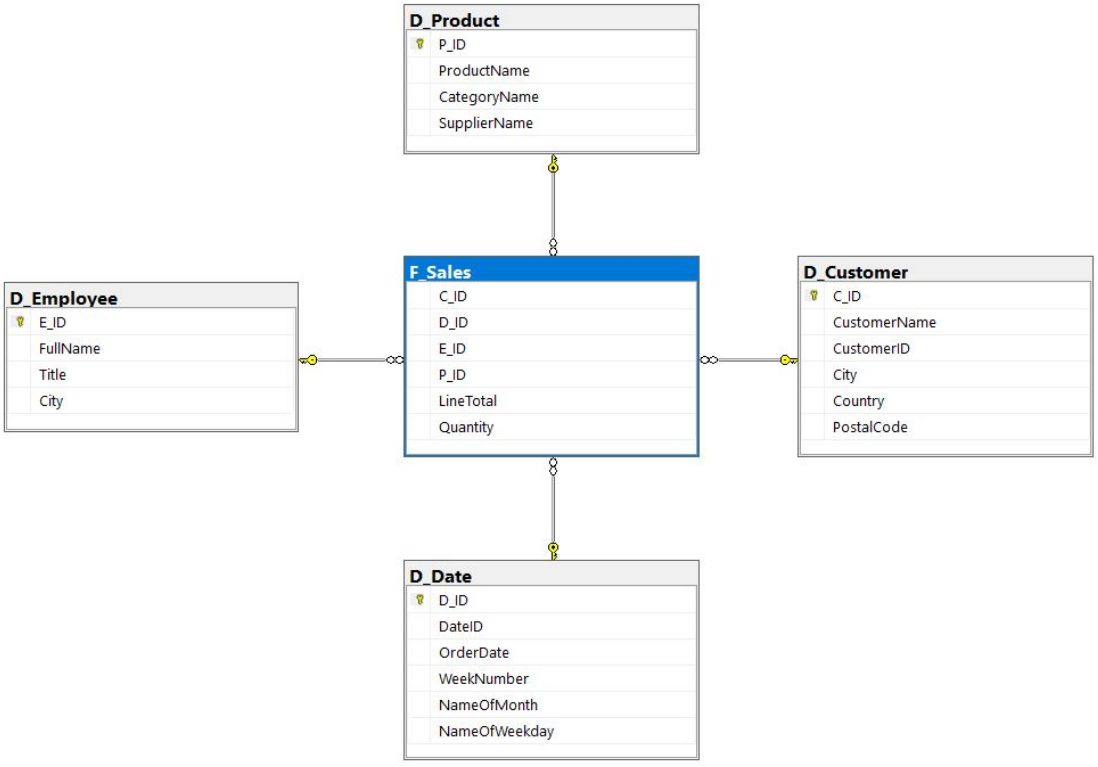
Extract:
Start by creating the stage tables in the stage database. The stage tables has somewhat the same structure as the data warehouse.
You extract the data from the operational database into the stage database, check if the data has errors or missing data - this is why you allow Null values in the staging table, so you can repair it in the next phase.
You start with the stage fact table and fill it up with values from the operational database, except the surrogate fields (usually an ID of some sort), this is also here you transform some of them, if needed. Then you update the surrogate keys from the data warehouse dimensions which makes you ready to fill the data warehouse fact table.

Transform:
During the transform process, you cleanse and repair the broken data if needed. etc. you could have some null values that maybe was extracted bad. You could then run some sort of an update that checks for nulls and sets them to ‘No value’ such that you are aware of the broken data - without breaking the data warehouse.
This is also here where you actually transform the data - etc merging columns first_name + last_name = full_name.

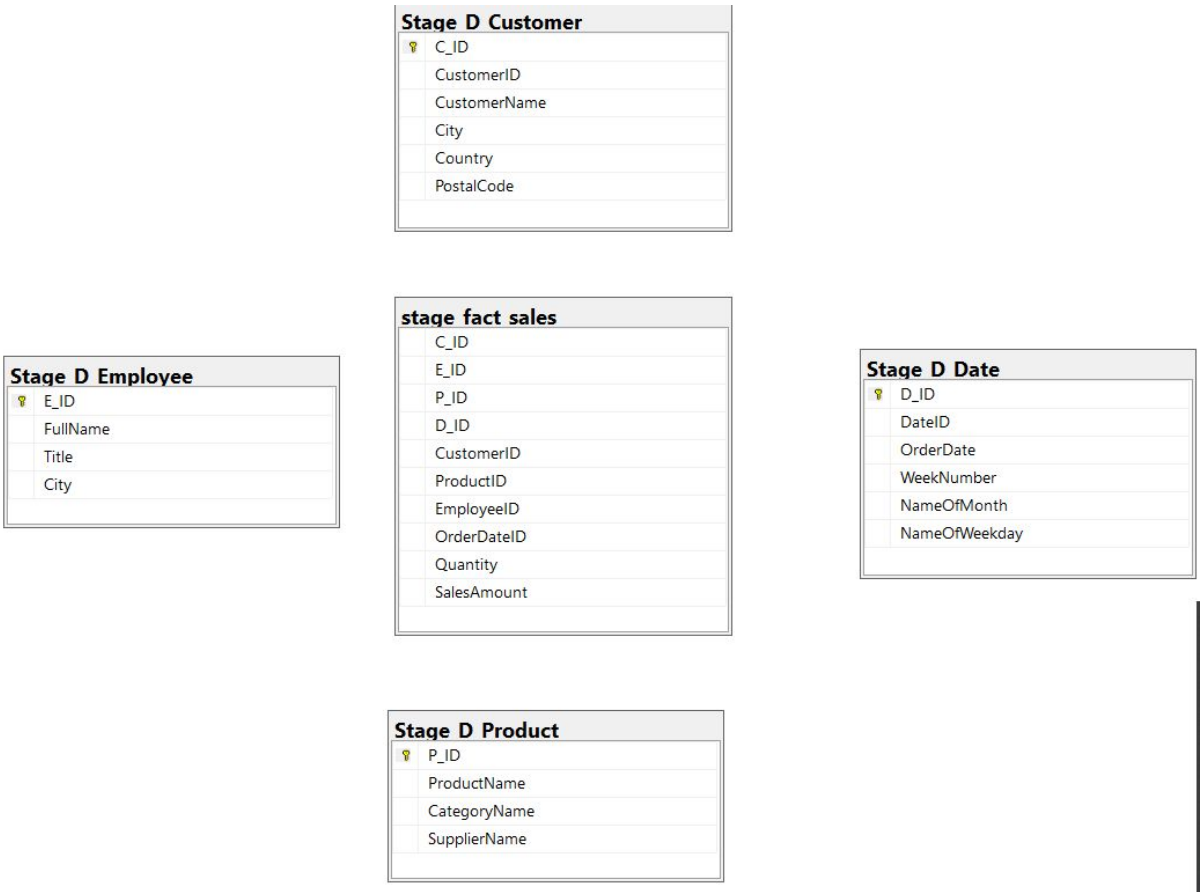
Load:
During the load process, you’re inserting the data, which you have extracted and possibly transformed, from the stage database into the data warehouse.

My process of the ETL:

1. Create data warehouse dimensional tables
   After doing the 4-step dimensional design process, i made a star scheme of which i used to create the data warehouse dimensional table. I've revised the scheme by removing the UnitPrice column on the Product table, because it isn't needed.
   My star scheme looks like this:



2. Create stage dimensional tables
   My stage dimensional tables is almost the same as the star scheme above. It contains addional surrogate keys (CustomerID, EmployeeID,OrderDateID and ProductID) which is used when populating the actual fact table in the data warehouse to make sure that they are connected in the correct way etc: In that way its easy to see that there exists 4 sales on the DateID (D_ID) 427



3. Fill stage dimensionals
   I extract the columns from the operational database into the columns i want in the stage database. Some irrelevant data isn't being extracted and some other data is getting merged (etc. first name + last name in employee)

4. Fill data warehouse dimensional tables from stage
   After doing so, i can load the data onwards to the data warehouse without putting the extraction and transformation pressure on the data warehouse database and thuse filling the dw dimensional tables.

5. Create stage Fact table
   Creating the stage table, where i allow nulls, for spotting broken data, so i know where to repair. Also because the surrogate keys aren't being set at this moment.

6. Fill stage fact table
   Now i extract from the operational database into the stage fact table with a large join, where i also transform some columns (SalesAmount, also known as LineTotal) is UnitPrice multiplied with Quantity.

7. Update stage fact table surrogate keys
   Now its ready to get the surrogate keys from the data warehouse, and here i added some addional surrogate keys to make sure that the data is going the right places.

8. Populate data warehouse fact table from the stage fact table
   Now the stage fact table is complete and ready to be loaded into the data warehouse almost without any pressure being applied onto the data warehouse.

9. See the result
   When i do a select all from Data warehouse fact table (and scroll down) this is what the result looks like. 450-453 was sold to the same customer, by the same employee at the same date with 4 different products.

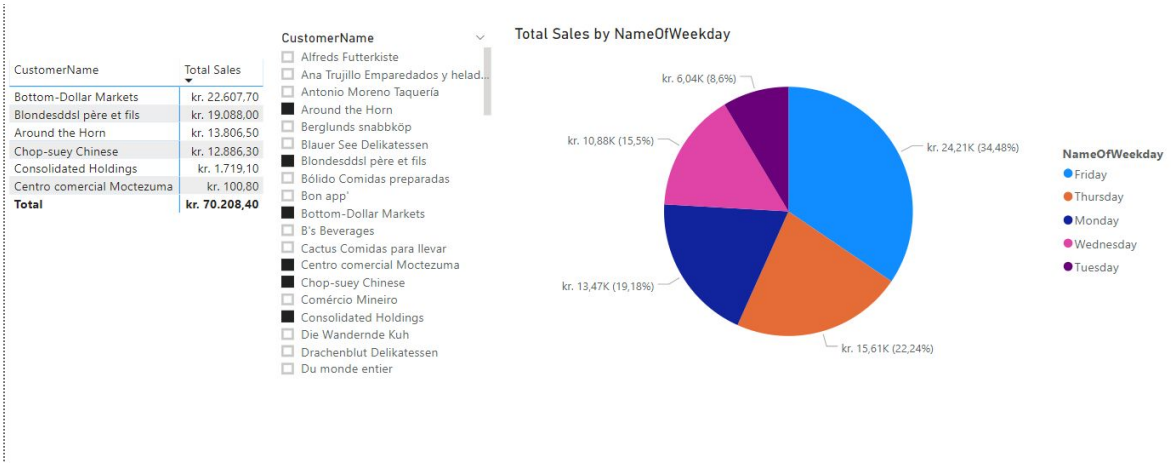
	C_ID	D_ID	E_ID	P_ID	LineTotal	Quantity
450	73	169	4	38	10540	50
451	73	169	4	46	19,2	2
452	73	169	4	68	360	36
453	73	169	4	77	364	35
454	63	171	4	2	912	60
455	63	171	4	47	418	55
456	63	171	4	61	364,8	16
457	63	171	4	74	120	15
458	68	172	4	60	1632	60
459	68	172	4	69	576	20
460	88	173	3	9	1552	20
461	88	173	3	13	9,6	2
462	88	173	3	70	96	8
463	88	173	3	73	240	20
464	61	173	8	19	29,2	4
465	61	173	8	26	747	30

My Date dimension table

My Date table contains the full date, if that was needed sometimes later on, other than that it contains week number, Name of month and name of day, which we can use for sales statistics.

PowerBI

For my power bi, I've chosen to go with a table that contains the measure Total Sales, which is written as Total Sales = SUM(F_Sales[LineTotal]).
Then im using a slicer and a pie chart. With this its so easy to see how much certain customers are buying and on which days in the week.



SQL CODE:

```
---Create data warehouse dimensional tables
USE myNorthWindDB
go
DROP TABLE F_Sales;
DROP TABLE D_Date;
DROP TABLE D_Customer;
DROP TABLE D_Employee;
DROP TABLE D_Product;

CREATE TABLE D_Product(
P_ID INT PRIMARY KEY IDENTITY (1, 1),
ProductName NVARCHAR(50),
CategoryName NVARCHAR(50),
SupplierName NVARCHAR(50),
);

CREATE TABLE D_Employee(
E_ID INT PRIMARY KEY IDENTITY (1, 1),
FullName NVARCHAR(50),
Title NVARCHAR(50),
City NVARCHAR(50)
);

CREATE TABLE D_Customer(
C_ID INT PRIMARY KEY IDENTITY (1, 1),
CustomerName NVARCHAR(50),
CustomerID NVARCHAR(5),
City  NVARCHAR(50),
Country NVARCHAR(50),
PostalCode NVARCHAR(10)
);

CREATE TABLE D_Date(
D_ID INT PRIMARY KEY IDENTITY (1, 1),
DateID NVARCHAR(50),
OrderDate DATE,
WeekNumber INT,
NameOfMonth NVARCHAR(50),
NameOfWeekday NVARCHAR(50)
);


CREATE TABLE F_Sales(
```

```
      C_ID INT REFERENCES D_Customer(C_ID),
      D_ID INT REFERENCES D_Date(D_ID),
      E_ID INT REFERENCES D_Employee(E_ID),
      P_ID INT REFERENCES D_Product(P_ID),
      LineTotal FLOAT,
      Quantity INT
);

---Create stage dimensional tables
Use StageNorthwind
go
DROP TABLE Stage_D_Date;
DROP TABLE Stage_D_Customer;
DROP TABLE Stage_D_Employee;
DROP TABLE Stage_D_Product;

CREATE TABLE Stage_D_Product(
P_ID INT PRIMARY KEY IDENTITY (1, 1),
ProductName NVARCHAR(50),
CategoryName NVARCHAR(50),
SupplierName NVARCHAR(50)
);

CREATE TABLE Stage_D_Employee(
E_ID INT PRIMARY KEY IDENTITY (1, 1),
FullName NVARCHAR(50),
Title NVARCHAR(50),
City NVARCHAR(50)
);

CREATE TABLE Stage_D_Customer(
C_ID INT PRIMARY KEY IDENTITY (1, 1),
CustomerID NVARCHAR(5),
CustomerName NVARCHAR(50),
City  NVARCHAR(50),
Country NVARCHAR(50),
PostalCode NVARCHAR(10)
);

CREATE TABLE Stage_D_Date(
D_ID INT PRIMARY KEY IDENTITY (1, 1),
DateID NVARCHAR(50),
OrderDate DATETIME,
WeekNumber INT,
NameOfMonth NVARCHAR(50),
```

```
NameOfWeekday NVARCHAR(50)
);

---Fill stage dimensionals
--Customer
INSERT INTO Stage_D_Customer(CustomerID,CustomerName, City,Country,
PostalCode)
SELECT c.CustomerID,c.CompanyName,c.City,c.Country,c.PostalCode
FROM NorthwindDB.dbo.Customers c

--Employee
INSERT INTO Stage_D_Employee(FullName, Title, City)
SELECT e.FirstName +' '+ e.LastName, e.Title, e.City
FROM NorthwindDB.dbo.Employees e

--Product
INSERT INTO Stage_D_Product(ProductName, CategoryName,SupplierName)
SELECT p.ProductName, c.CategoryName, s.CompanyName
FROM NorthwindDB.dbo.Products p
JOIN NorthwindDB.dbo.Categories c on c.CategoryID = p.CategoryID
JOIN NorthwindDB.dbo.Suppliers s on s.SupplierID = p.SupplierID

--Date
INSERT INTO  Stage_D_Date(OrderDate,DateID,WeekNumber,NameOfMonth,
NameOfWeekday)
Select o.OrderDate,o.OrderDate,DATEPART(week, o.OrderDate),
DATENAME(month, o.OrderDate), DATENAME(WEEKDAY, o.OrderDate)
From NorthwindDB.dbo.Orders o

---Fill dw dimensional tables from stage
Use myNorthWindDB
--Customer
INSERT INTO D_Customer(CustomerID,CustomerName, City,Country,
PostalCode)
Select sc.CustomerID, sc.CustomerName, sc.City, sc.Country,
sc.PostalCode
From StageNorthwind.dbo.Stage_D_Customer sc

--Employee
INSERT INTO D_Employee(FullName, Title, City)
Select se.FullName, se.Title, se.City
From StageNorthwind.dbo.Stage_D_Employee se

--Product
INSERT INTO D_Product(ProductName, CategoryName,SupplierName)
```

```
Select sp.ProductName, sp.CategoryName, sp.SupplierName
From StageNorthwind.dbo.Stage_D_Product sp

--Date
INSERT INTO D_Date(OrderDate,DateID,WeekNumber,NameOfMonth,
NameOfWeekday)
Select sd.OrderDate,sd.DateID, sd.WeekNumber, sd.NameOfMonth,
sd.NameOfWeekday
From StageNorthwind.dbo.Stage_D_Date sd

---Create stage Fact table
USE StageNorthwind
GO
DROP TABLE stage_fact_sales
CREATE TABLE stage_fact_sales(
[C_ID] [int] NULL,
[E_ID] [int] NULL,
[P_ID] [int] NULL,
[D_ID] [int] NULL,
[CustomerID] [nchar](5) NULL,
[ProductID] [int] NULL,
[EmployeeID] [int] NULL,
[OrderDateID] [nvarchar](50) NULL,
[Quantity] [bigint] NULL,
[SalesAmount] [decimal](18, 2) NULL
)

---Fill stage fact table
USE NorthWindDB
GO
INSERT INTO StageNorthwind.dbo.stage_fact_sales
(Quantity,SalesAmount,ProductID,CustomerID,EmployeeID,OrderDateID)
SELECT
Quantity,
[Order Details].UnitPrice* Quantity,
Products.ProductID,
Customers.CustomerID,
Employees.EmployeeID,
Orders.OrderDate
FROM [Order Details]
JOIN Orders ON  Orders.OrderID= [Order Details].OrderID
JOIN Customers on Orders.CustomerID= Customers.CustomerID
JOIN Employees on Employees.EmployeeID= Orders.EmployeeID
JOIN Products on Products.ProductID= [Order Details].ProductID
```

```
---Update stage fact table surrogate keys
Use StageNorthwind
--Customer
UPDATE stage_fact_sales
SET C_ID = (
      SELECT C_ID
      FROM myNorthWindDB.dbo.D_Customer c
      WHERE c.CustomerID = stage_fact_sales.CustomerID
)

--Employee
UPDATE stage_fact_sales
SET E_ID = (
      SELECT E_ID
      FROM myNorthWindDB.dbo.D_Employee e
      WHERE e.E_ID = stage_fact_sales.EmployeeID
)

--Product
UPDATE stage_fact_sales
SET P_ID = (
      SELECT P_ID
      FROM myNorthWindDB.dbo.D_Product p
      WHERE p.P_ID = stage_fact_sales.ProductID
)

--Date
UPDATE stage_fact_sales
SET D_ID = (
      SELECT top 1 (d.D_ID) --should only return one
      FROM myNorthWindDB.dbo.D_Date d
      WHERE d.DateID = stage_fact_sales.OrderDateID
)

---Populate data warehouse fact table from the stage fact table
USE StageNorthwind
GO
INSERT INTO myNorthWindDB.dbo.F_Sales(D_ID,C_ID,P_ID,E_ID,Quantity,
LineTotal)
SELECT s.D_ID, s.C_ID, s.P_ID, s.E_ID, s.Quantity, s.SalesAmount FROM
stage_fact_sales s

Select * from myNorthWindDB.dbo.F_Sales
```