

BARBER HAIR SALON

CUT & SHAVE



Agenda

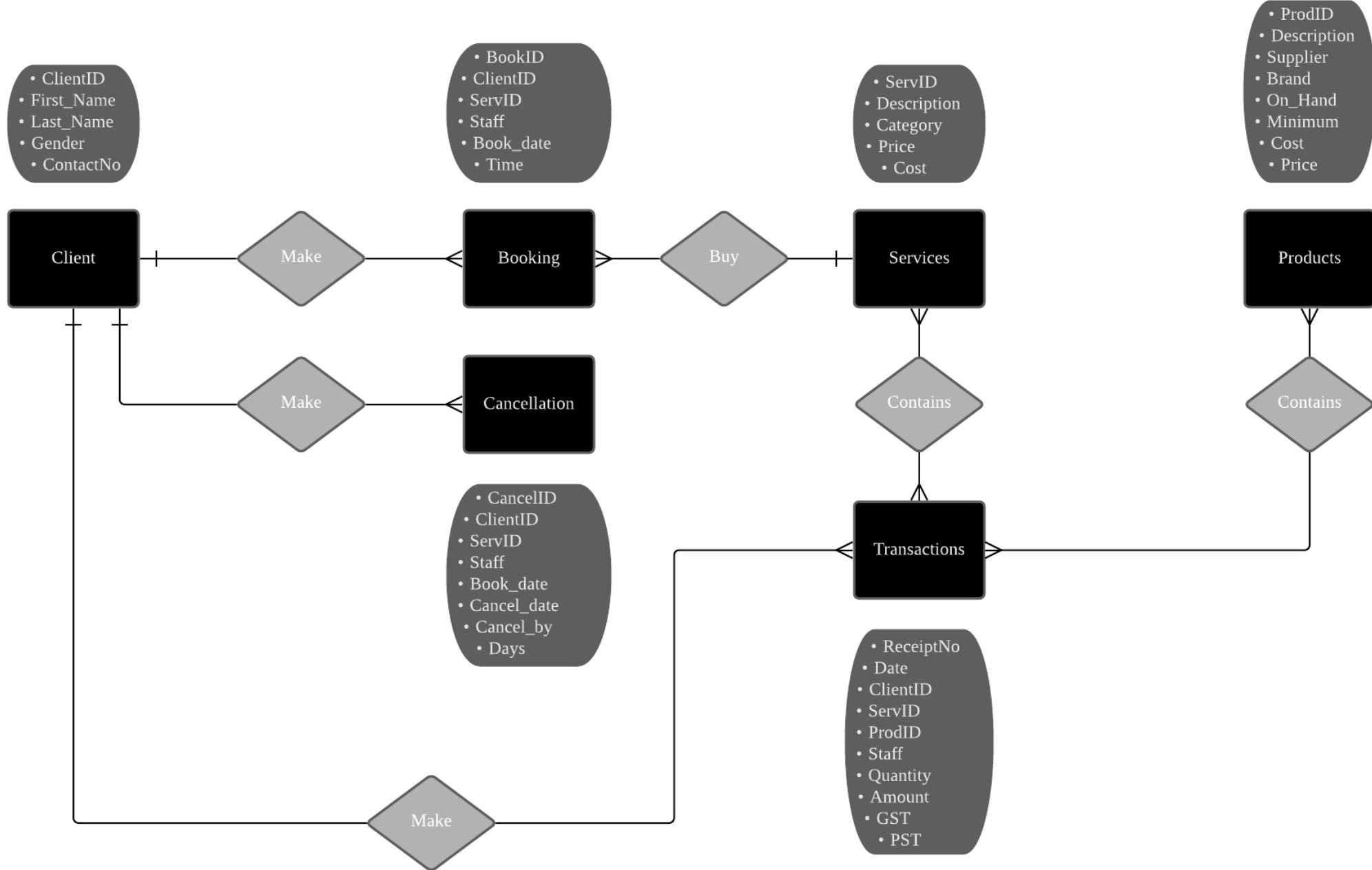
- Data source
- ER Diagram
- Data Modifications
- Business Needs
- Insights



Data Source

- Kaggle
- The data was based on an actual hair salon in Toronto, Canada
- The data consists primarily of attributes of the appointment booking and excludes client or staff member details
- Bookings table time series between 2018-03-14 to 2019-02-15
- Transactions table time series between 2018-03-15 to 2018-07-29







Data Modifications

1 →

```
--Category for all products is Retail.  
ALTER TABLE Products  
DROP COLUMN Category;  
  
--Maximum amount same as Minimum  
ALTER TABLE Products  
DROP COLUMN Maximum;
```

3 →

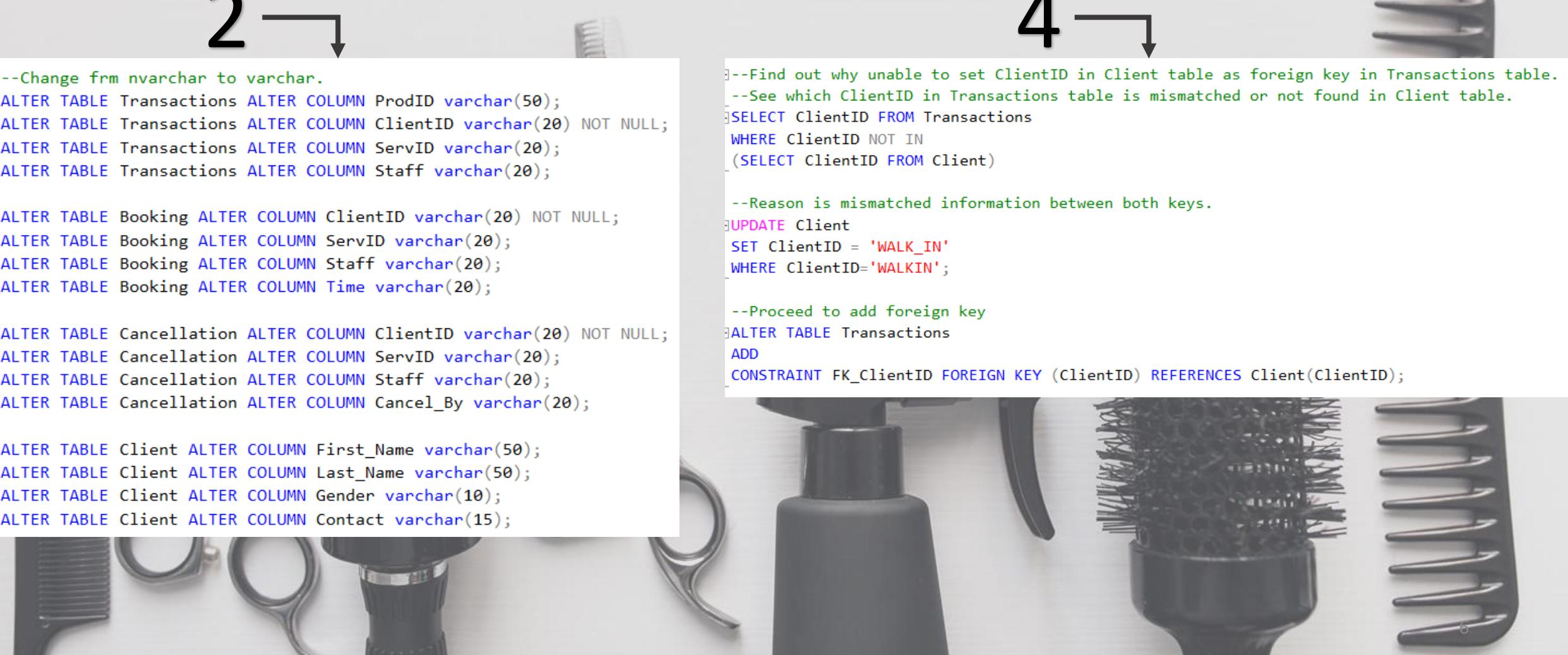
```
--Set foreign keys  
ALTER TABLE Transactions  
ADD  
CONSTRAINT FK_ServID FOREIGN KEY (ServID) REFERENCES Services(ServID),  
CONSTRAINT FK_ProdID FOREIGN KEY (ProdID) REFERENCES Products(ProdID),  
CONSTRAINT FK_ClientID FOREIGN KEY (ClientID) REFERENCES Client(ClientID);
```

2 →

```
--Change frm nvarchar to varchar.  
ALTER TABLE Transactions ALTER COLUMN ProdID varchar(50);  
ALTER TABLE Transactions ALTER COLUMN ClientID varchar(20) NOT NULL;  
ALTER TABLE Transactions ALTER COLUMN ServID varchar(20);  
ALTER TABLE Transactions ALTER COLUMN Staff varchar(20);  
  
ALTER TABLE Booking ALTER COLUMN ClientID varchar(20) NOT NULL;  
ALTER TABLE Booking ALTER COLUMN ServID varchar(20);  
ALTER TABLE Booking ALTER COLUMN Staff varchar(20);  
ALTER TABLE Booking ALTER COLUMN Time varchar(20);  
  
ALTER TABLE Cancellation ALTER COLUMN ClientID varchar(20) NOT NULL;  
ALTER TABLE Cancellation ALTER COLUMN ServID varchar(20);  
ALTER TABLE Cancellation ALTER COLUMN Staff varchar(20);  
ALTER TABLE Cancellation ALTER COLUMN Cancel_By varchar(20);  
  
ALTER TABLE Client ALTER COLUMN First_Name varchar(50);  
ALTER TABLE Client ALTER COLUMN Last_Name varchar(50);  
ALTER TABLE Client ALTER COLUMN Gender varchar(10);  
ALTER TABLE Client ALTER COLUMN Contact varchar(15);
```

4 →

```
--Find out why unable to set ClientID in Client table as foreign key in Transactions table.  
--See which ClientID in Transactions table is mismatched or not found in Client table.  
SELECT ClientID FROM Transactions  
WHERE ClientID NOT IN  
(SELECT ClientID FROM Client)  
  
--Reason is mismatched information between both keys.  
UPDATE Client  
SET ClientID = 'WALK_IN'  
WHERE ClientID='WALKIN';  
  
--Proceed to add foreign key  
ALTER TABLE Transactions  
ADD  
CONSTRAINT FK_ClientID FOREIGN KEY (ClientID) REFERENCES Client(ClientID);
```



5. Create Client Table

- Columns:
 1. ClientID (based on Booking and Cancellation tables)
 2. Gender (VLOOKUP the services in Booking and Cancellation table)
 3. First_Name
`[IF(D2="F",INDEX(Sheet1!$A:$A,RANDBETWEEN(1,COUNTA(Sheet1!$A:$A)),1),INDEX(Sheet1!$D:$D,RANDBETWEEN(1,COUNTA(Sheet1!$D:$D)),1))]`
 4. Last_Name
`[INDEX($A:$A,RANDBETWEEN(1,COUNTA($A:$A)),1)]`
 5. Contact `[RAND()*(999999999-1000000000)+1000000000]`



Business Needs

1. What are the top 10 popular products?
2. What are the top and bottom 10 services?
3. Which is the busiest day?
4. Who is the top client who spend more than \$600 and has more than 10 bookings and transactions?
5. Which staff has the best performance in sales?
6. Is the no show rate high?



Load SQL syntax into Excel



SQL Server database

Server ⓘ
ADHL\SQLEXPRESS

Database (optional)
HairSalon

▲ Advanced options

Command timeout in minutes (optional)

SQL statement (optional, requires database)

```
INNER JOIN (SELECT ClientID, COUNT(ClientID) AS Total_Transaction, SUM(Money) AS Total_Spend  
ON B.ClientID=T.ClientID  
INNER JOIN Client AS C  
ON B.ClientID=C.ClientID  
WHERE B.Total_Book>10 AND T.Total_Transaction>10 AND T.Total_Spend>600  
ORDER BY T.Total_Spend DESC;
```

Include relationship columns

Navigate using full hierarchy

Enable SQL Server Failover support

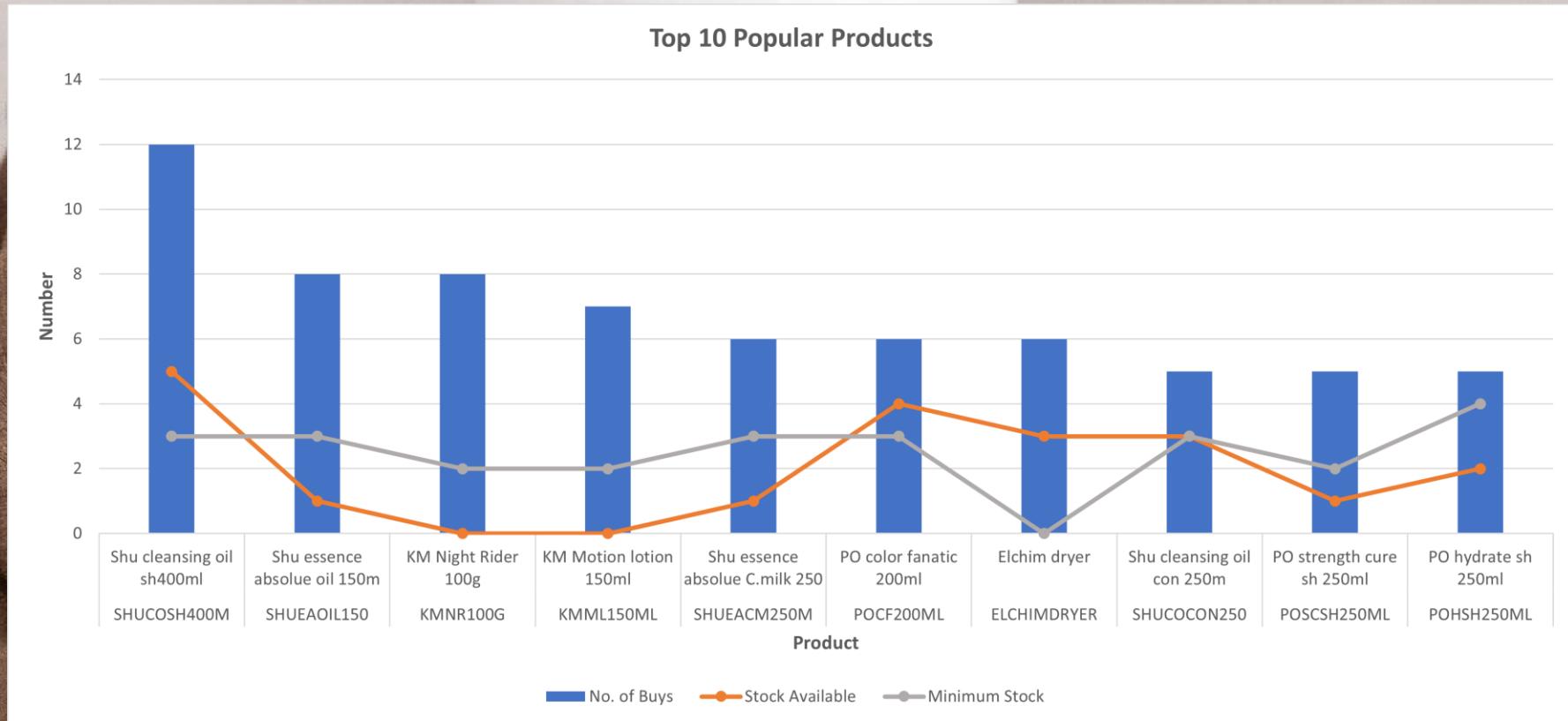
OK Cancel

Insights



Top 10 Products

```
--TOP 10 popular products based on No. of transactions(buys)
SELECT TOP 10 P.ProdID, P.Description, COUNT(T.ProdID) AS No_of_Buys, P.On_Hand, P.Minimum
FROM Products AS P
INNER JOIN Transactions AS T
ON P.ProdID=T.ProdID
GROUP BY P.ProdID, P.Description, P.On_Hand, P.Minimum
ORDER BY No_of_Buys DESC;
```



Products Stock-take View

```
--Stock-take
CREATE VIEW vStocktake
AS
SELECT ProdID, Description, Supplier, On_Hand, Minimum, Minimum-On_Hand AS Stock_to_add
FROM Products
WHERE On_Hand<Minimum
GO

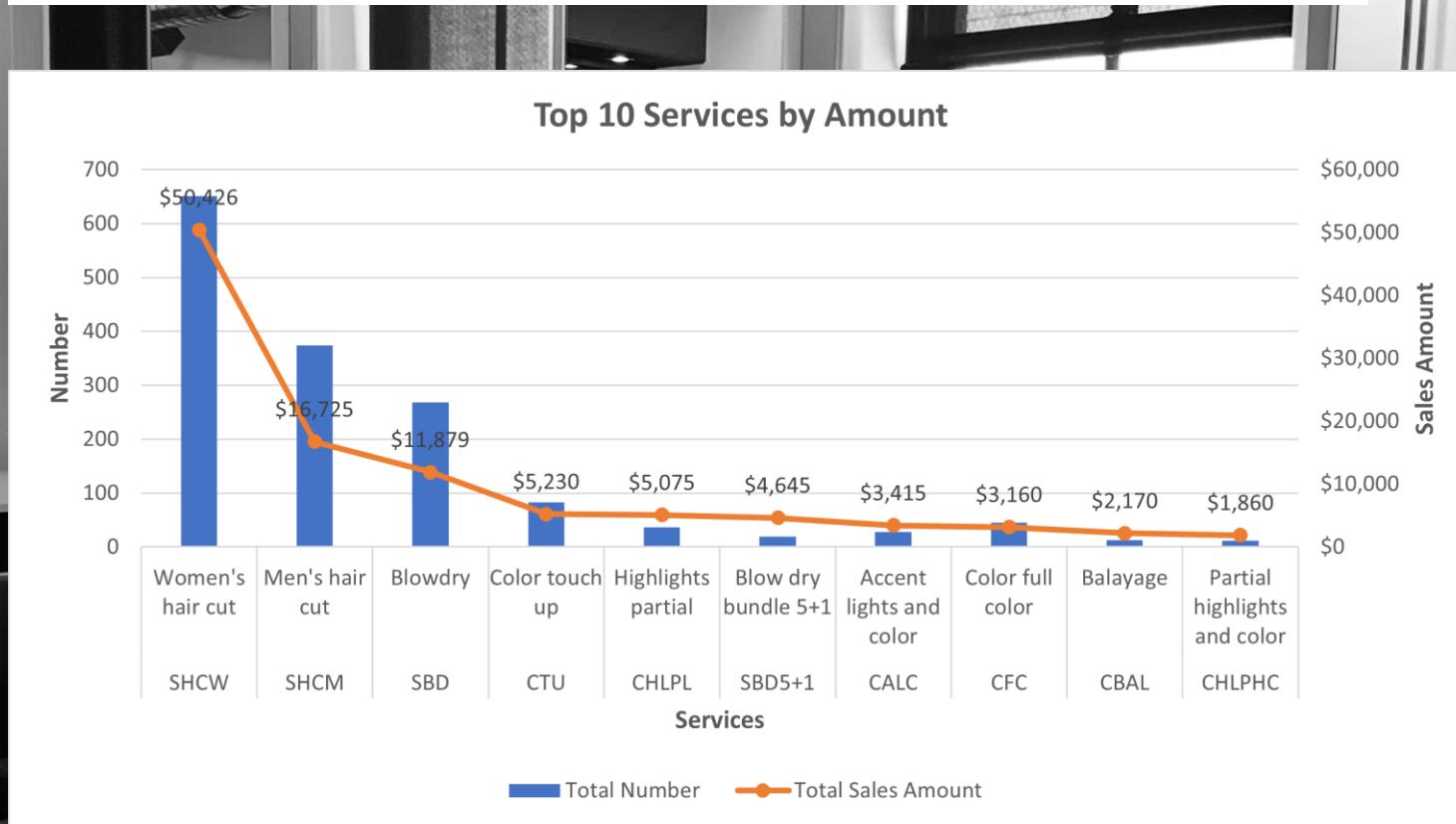
SELECT * FROM vStocktake;
```

100 % ▾

	ProdID	Description	Supplier	On_Hand	Minimum	Stock_to_add
1	AGDRYLIFT44ML	AG Dry Lift 44ml	VENUS	1	2	1
2	KMBB350M	KM Body builder 350m	TBBS	0	1	1
3	KMDO250ML	KM Doo over 250ml	TBBS	1	2	1
4	KMML150ML	KM Motion lotion 150ml	TBBS	0	2	2
5	KMNR100G	KM Night Rider 100g	TBBS	0	2	2
6	KMRRG100G	KM Rough Rider 100g	TBBS	0	1	1
7	KMSA200M	KM Smooth Again 200ml	TBBS	0	1	1
8	LPHOMST150M	LP homme strong 150m	LOREAL	1	2	1
9	LPSILVESH300	LP Silver Shampoo 300ml	LOREAL	1	2	1
10	LPWSSCRUFF150M	LP wild stylers scruff me 150m	LOREAL	1	3	2
11	NEUAO75M	NeuMa Argan Oil 75ml	TBBS	0	1	1
12	NEUBDL250ML	NeuMa Blowdry Lotion 250ml	TBBS	0	1	1

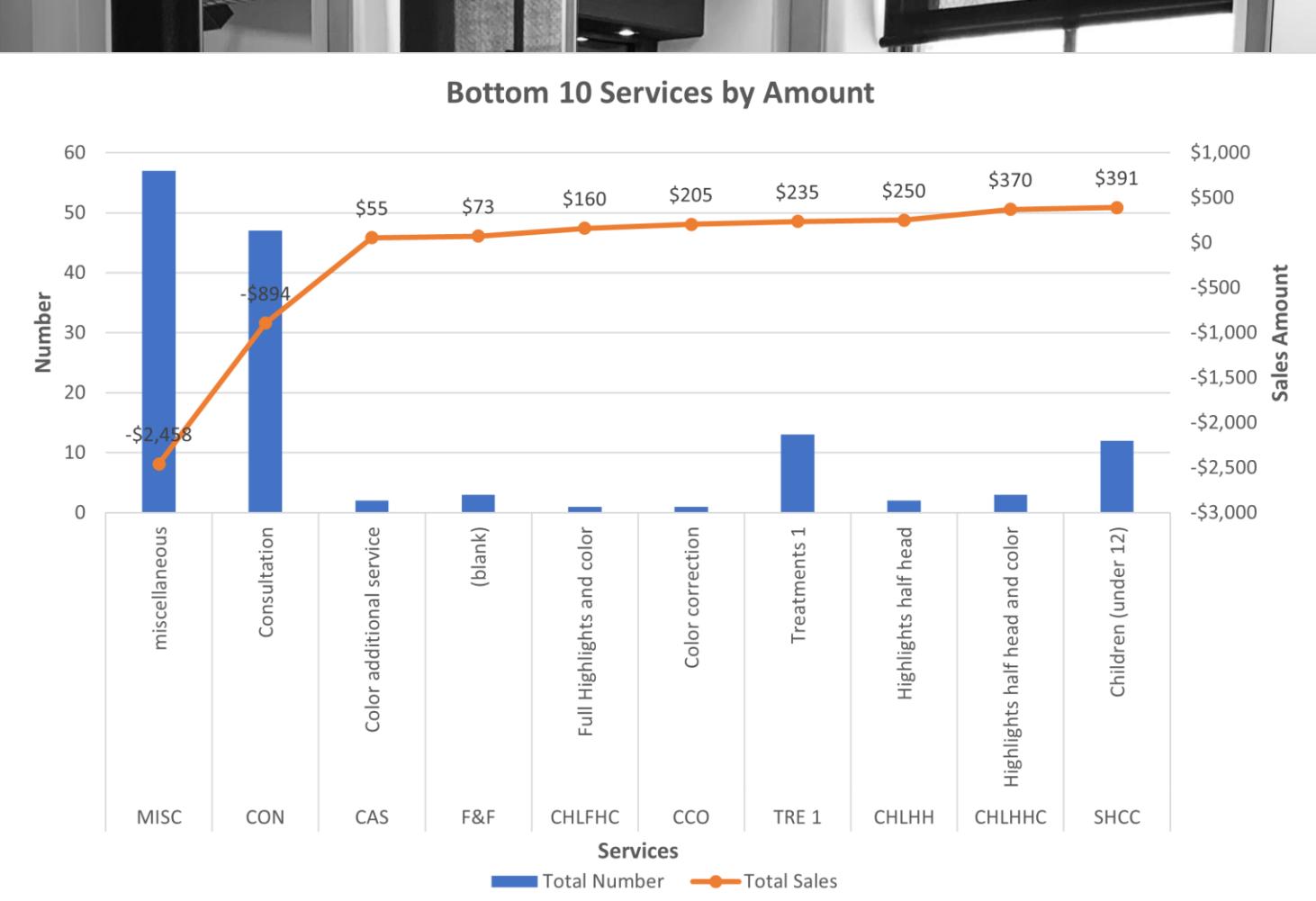
Top 10 Services

```
-- Top 10 services by amount
SELECT TOP 10 T.ServID, Description, COUNT(T.ServID) AS Total_Num, SUM(Amount) AS Total_Sales_Amt_ExTax
FROM Transactions AS T
INNER JOIN Services AS S
ON T.ServID=S.ServID
GROUP BY T.ServID, S.Description
ORDER BY Total_Sales_Amt_ExTax DESC;
```



Bottom 10 Services

```
--Bottom 10 services by amount  
SELECT TOP 10 T.ServID, S.Description, COUNT(T.ServID) AS Total_Num, SUM(Amount) AS Total_Sales_Amt_ExTax  
FROM Transactions AS T  
INNER JOIN Services AS S  
ON T.ServID=S.ServID  
GROUP BY T.ServID, S.Description  
ORDER BY Total_Sales_Amt_ExTax;
```



Bottom 10 Services (Check)

```
--1.Find out which receipt no. contains negative amount for Misc and CON  
SELECT Receipt_No, ServID, Amount FROM Transactions  
WHERE ServID IN ('Misc', 'CON') AND Amount < 0;
```

100 %

Results Messages

Receipt_No	ServID	Amount
1	203	CON -50.00
2	238	MISC -50.00
3	240	MISC -50.00
4	276	CON -50.00
5	291	CON -50.00
6	313	CON -50.00
7	314	CON -50.00
8	343	CON -52.00
9	364	CON -50.00
10	387	CON -50.00
11	388	CON -45.00
12	405	MISC -50.00
13	429	CON -52.00
14	475	CON -50.00

```
--2.Sample size of receipts for investigation. Misc and CON with negative value are used to offset SBD.
```

```
SELECT Receipt_No, T.ServID, Description, Amount  
FROM Transactions AS T  
INNER JOIN Services AS S  
ON T.ServID = S.ServID  
WHERE Receipt_No IN (203,238,240,276,291,343,388,429,475,581,636,1138,1175,1197,1420,1470)
```

100 %

Results Messages

Receipt_No	ServID	Description	Amount
1	203	SBD Blowdry	50.00
2	203	CON Consultation	-50.00
3	238	CHLPL Highlights partial	0.00
4	238	SBD Blowdry	50.00
5	238	MISC miscellaneous	-50.00
6	240	SBD Blowdry	50.00
7	240	MISC miscellaneous	-50.00
8	276	SBD Blowdry	50.00
9	276	CON Consultation	-50.00
10	291	SBD Blowdry	50.00
11	291	CON Consultation	-50.00
12	343	SBD Blowdry	52.00
13	343	CON Consultation	-52.00

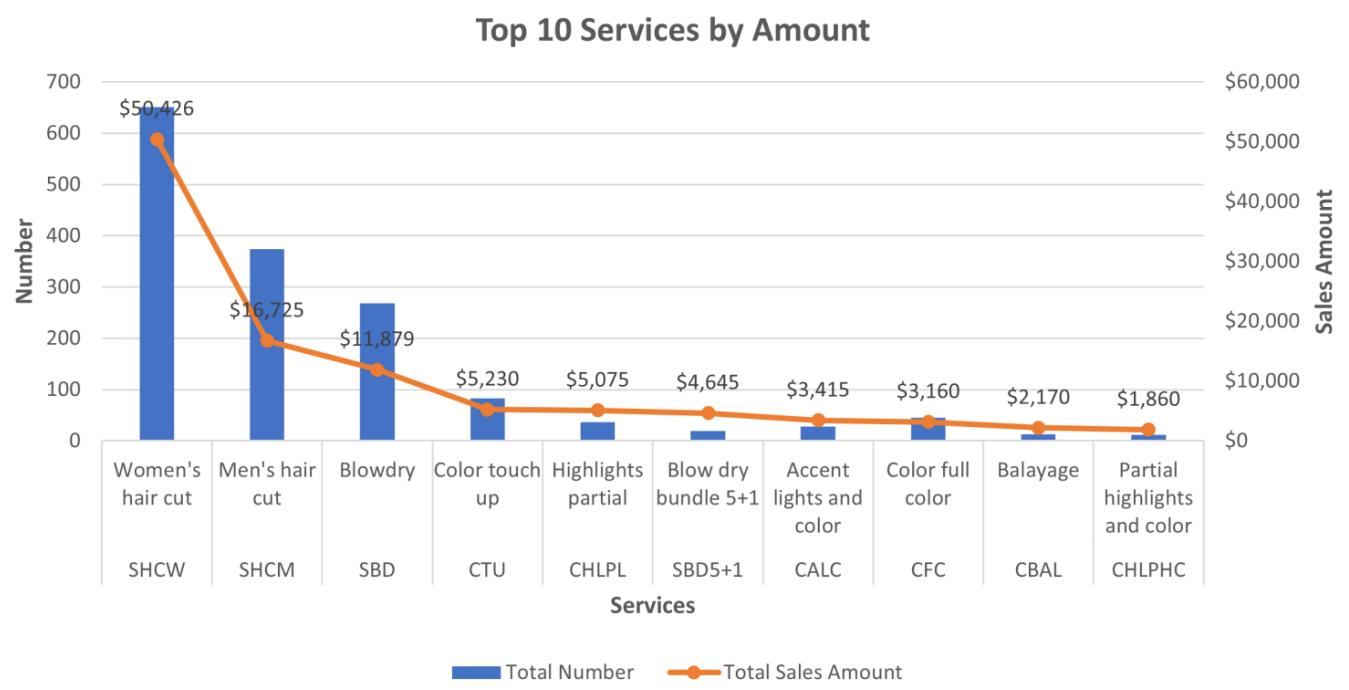
Bottom 10 Services (Check)

Is Blowdry still Top 3 Services?

```
--3.Total amount received for SBD after Misc and CON adjustments.  
SELECT T.ServID, S.Description, SUM(T.Amount) AS Net_Amt  
FROM Transactions AS T  
INNER JOIN Services AS S  
ON T.ServID=S.ServID  
WHERE T.ServID = 'SBD'  
AND  
--To restrict the list of receipt number with SBD not offset by Misc or CON.  
Receipt_No IN  
(SELECT DISTINCT Receipt_No FROM Transactions WHERE ServID IN ('SBD','Misc','CON'))  
GROUP BY Receipt_No  
HAVING SUM(Amount) > 0  
GROUP BY T.ServID, S.Description;
```

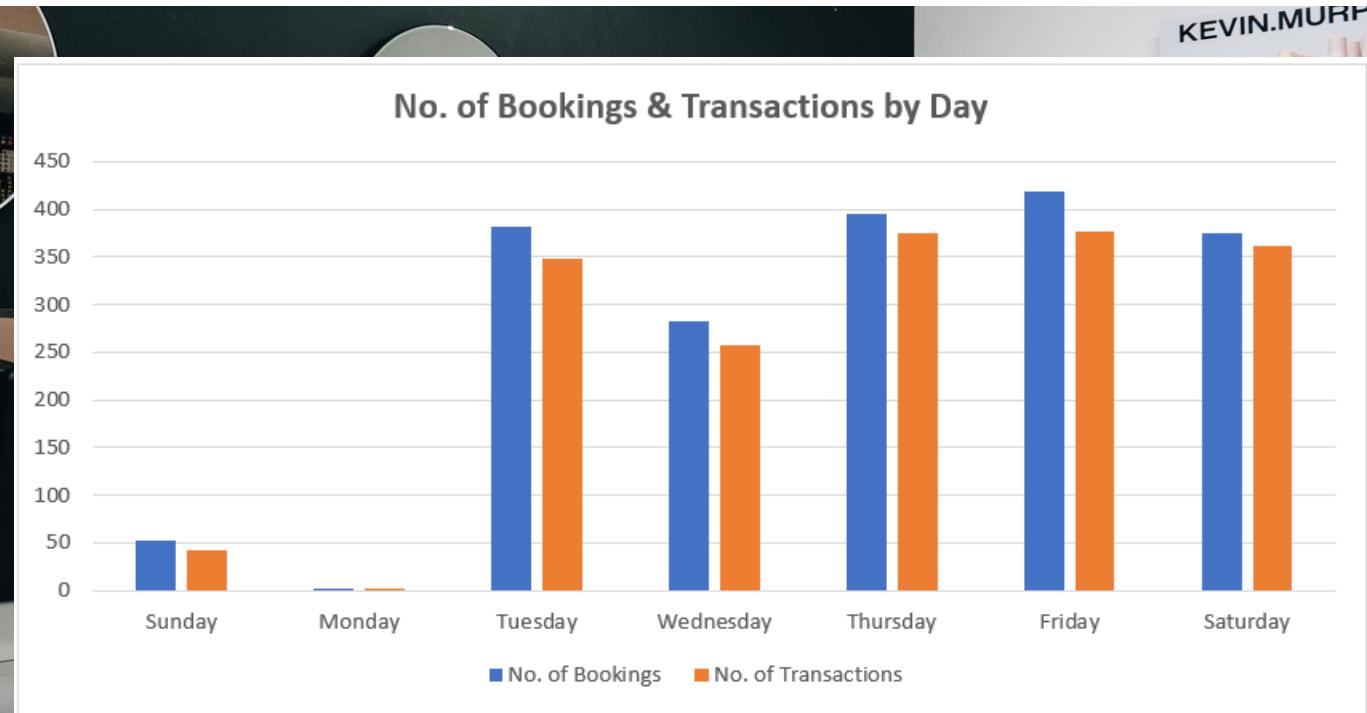
100 %

ServID	Description	Net_Amt	
1	SBD	Blowdry	8564.00



```
--The number of bookings and transactions by day.
SELECT B.Day, B.No_of_Bookings AS 'No. of Bookings', T.No_of_Transactions AS 'No. of Transactions'
FROM
(SELECT DATENAME(WEEKDAY,Date) AS Day, COUNT(DATENAME(WEEKDAY,Date)) AS No_of_Bookings FROM Booking GROUP BY DATENAME(WEEKDAY,Date)) AS B
INNER JOIN
(SELECT DATENAME(WEEKDAY,Date) AS Day, COUNT(DATENAME(WEEKDAY,Date)) AS No_of_Transactions FROM Transactions WHERE ServID IS NOT NULL GROUP BY DATENAME(WEEKDAY,Date)) AS T
ON B.Day=T.Day
ORDER BY
CASE
WHEN B.Day = 'Sunday' THEN 1
WHEN B.Day = 'Monday' THEN 2
WHEN B.Day = 'Tuesday' THEN 3
WHEN B.Day = 'Wednesday' THEN 4
WHEN B.Day = 'Thursday' THEN 5
WHEN B.Day = 'Friday' THEN 6
WHEN B.Day = 'Saturday' THEN 7
END ASC;
```

Busiest Day



Top Clients

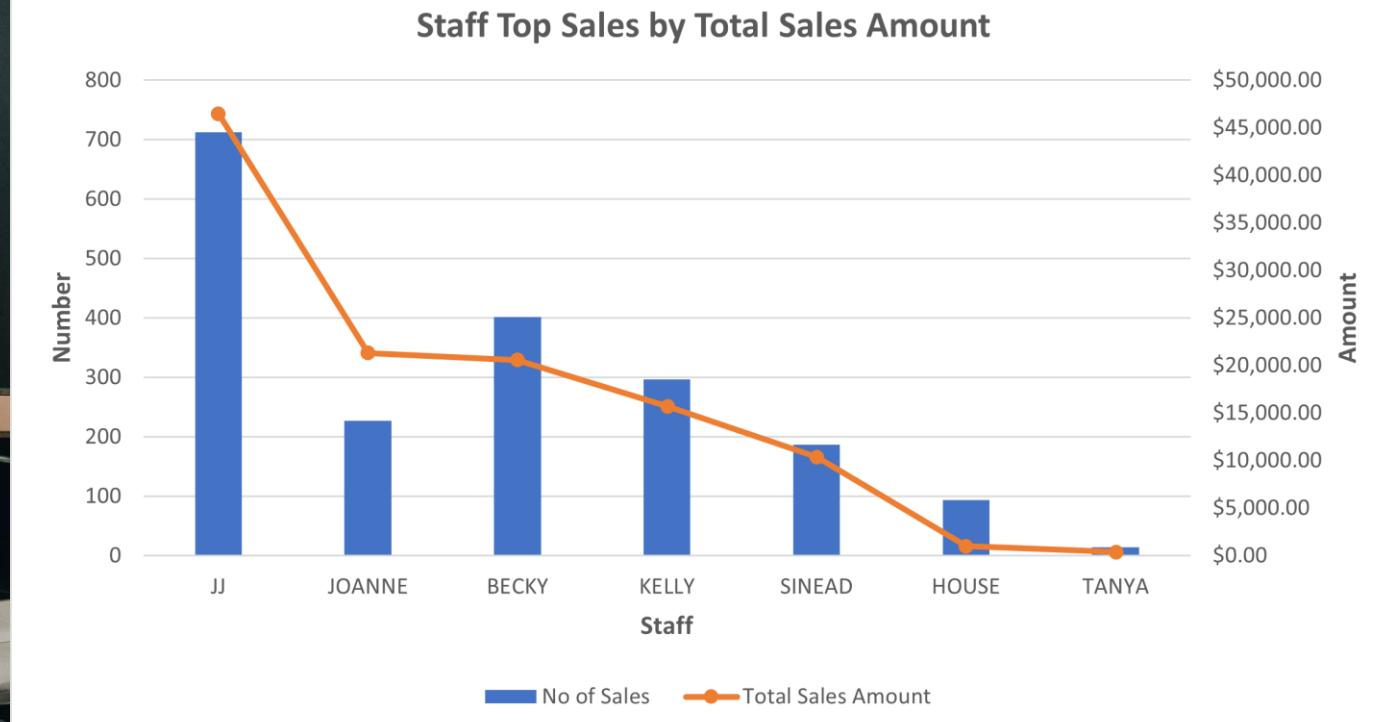
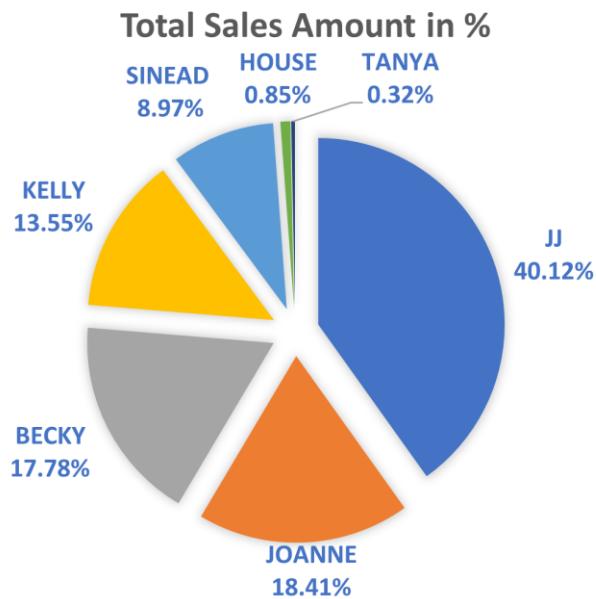


```
--Total bookings, total transaction and total spend by clients
|SELECT B.ClientID, C.First_Name, C.Last_Name, B.Total_Book, T.Total_Transaction, Total_Spend
FROM
(SELECT ClientID, COUNT(ClientID) AS Total_Book FROM Booking WHERE Date BETWEEN '2018-03-15' AND '2018-07-29' GROUP BY ClientID) AS B
INNER JOIN
(SELECT ClientID, COUNT(ClientID) AS Total_Transaction, SUM(Amount) AS Total_Spend FROM Transactions GROUP BY ClientID) AS T
ON B.ClientID=T.ClientID
INNER JOIN Client AS C
ON T.ClientID=C.ClientID
WHERE
B.Total_Book>10
AND T.Total_Transaction>10
AND T.Total_Spend>600
ORDER BY T.Total_Spend DESC;
```



Staff Top Sales

```
--Staff total number of sales and total sales amount  
SELECT Staff, COUNT(Staff) AS No_of_Sales, SUM(Amount) AS Total_Sales_Amt  
FROM Transactions  
GROUP BY Staff  
ORDER BY Total_Sales_Amt DESC;
```



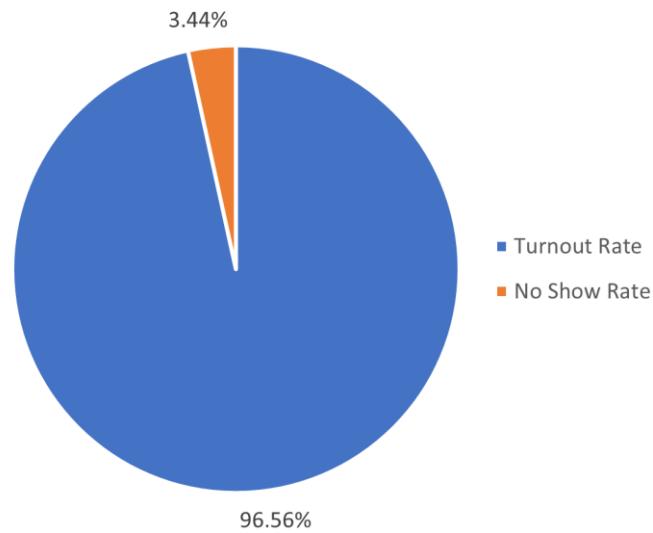
Staff Top Sales (Check)

```
SELECT Staff, T.ServID, S.Description, COUNT(T.ServID) AS Total_Num, SUM(Amount) AS 'Sales Amount'  
FROM Transactions AS T  
INNER JOIN Services AS S  
ON T.ServID=S.ServID  
WHERE Staff IN ('TANYA', 'HOUSE')  
GROUP BY Staff, T.ServID, S.Description  
ORDER BY Total_Num DESC;
```

Staff	ServID	Description	Total_Num	Sales Amount
1 HOUSE	MISC	miscellaneous	49	-2376.00
2 HOUSE	CON	Consultation	21	-949.00
3 HOUSE	SBD5+1	Blow dry bundle 5+1	17	4185.00
4 TANYA	SBD	Blowdry	9	205.00
5 TANYA	CTU	Color touch up	3	115.00
6 TANYA	TRE 1	Treatments 1	1	20.00
7 TANYA	CFC	Color full color	1	35.00

No Show Rate

Turnout & No Show Rate



```
--Turnout rate and no show rate
SELECT 'Turnout Rate' AS Type,(1.0*(
(SELECT 1.0*COUNT(T.ClientID)
FROM Booking AS B
INNER JOIN Transactions AS T
ON
(B.Date=T.Date) AND
(T.ClientID=B.ClientID) AND
(T.ServID=B.ServID) AND
(T.Staff=B.Staff)))/(SELECT COUNT(ClientID) FROM Booking WHERE Date BETWEEN '2018-03-15' AND '2018-07-29')
)) AS Rate
UNION
SELECT 'No Show Rate' AS Type, 1.0-(1.0*(
(SELECT 1.0*COUNT(T.ClientID)
FROM Booking AS B
INNER JOIN Transactions AS T
ON
(B.Date=T.Date) AND
(T.ClientID=B.ClientID) AND
(T.ServID=B.ServID) AND
(T.Staff=B.Staff)))/(SELECT COUNT(ClientID) FROM Booking WHERE Date BETWEEN '2018-03-15' AND '2018-07-29')
)) AS Rate
```

Add-ons



Stored Procedures

```
-- Stored procedure to fetch client details from searching client ID.  
CREATE PROCEDURE  
Client_ID  
(@ClientID varchar(20))  
AS  
BEGIN  
SELECT ClientID, First_Name, Last_Name, Gender, Contact  
FROM Client  
WHERE ClientID=@ClientID  
End;  
  
EXEC Client_ID 'COLG01';
```

100 %

Results Messages

	ClientID	First_Name	Last_Name	Gender	Contact
1	COLG01	Jesse	Chapman	F	5111926580

```
--Stored procedure to fetch client details from searching first or last name.  
CREATE PROCEDURE  
Client_Name  
(@Clientname varchar(50))  
AS  
BEGIN  
SELECT ClientID, First_Name, Last_Name, Gender, Contact  
FROM Client  
WHERE First_Name LIKE CONCAT('%',@Clientname,'%') OR Last_Name LIKE CONCAT('%',@Clientname,'%')  
End;  
  
EXEC Client_Name jo;
```

100 %

Results Messages

	ClientID	First_Name	Last_Name	Gender	Contact
1	AINM01	Jordyn	Ramsey	F	9239617733
2	BOTC01	Adelynn	Johnson	F	7330432917
3	BROS02	Jordan	Peterson	M	2255982476
4	BUDG02	Joshua	Owen	M	2144923312
5	BYRP01	Joel	Everett	M	5793728092
6	DRAA01	Journee	West	F	7987934834
7	ERVA01	Jocelyn	Eaton	F	6450305311
8	GA01	Poppy	Johnston	F	9699150433
9	GREB01	Jonah	Gardner	M	6749543725
10	GRET01	Jonathan	Weeks	M	5089020025

Function

```
-- Function to calculate price after taxes
CREATE FUNCTION
Cal_Total
(@Total money)
RETURNS money
AS
BEGIN
    RETURN (@Total*(1+0.05+0.08));
END;

SELECT dbo.Cal_Total(82) AS Total_Amount
```

Total_Amount
92.66



Questions?



References

- Kaggle,
<https://www.kaggle.com/frederickferguson/hair-salon-no-show-data-set?select=Product+Listing+%28Retail%290.csv>
- Data.gov,
<https://catalog.data.gov/dataset/baby-names-from-social-security-card-applications-national-level-data>
- Namecensus,
<https://namecensus.com/data/100.html>