TERPRESSO: 0502_05

Terpresso DATABASE

BY GROUP 0502-05

Anwesha Gupta (0502)
Janhavi Namjoshi (0502)
Madhulika Nambi (0502)
Udit Singh (0501)

WELCOME TO Terpresso

We are a rising chain of coffee houses, trying to build data management processes that would hold true as we expand

We want to monitor our product inventory, customer database, transactions, and supplier database and use it to improve our menu and profits.
We want to know how our products perform and improve our relationship with our suppliers.

Data Source: Circulated google form link among peers and collected data for the database

SPECIALISATION:

COFFEE BEANS AND TEA LEAVES

PRIMARY PRODUCTS:

BEVERAGES

USERS:

ANALYSTS

STORE MANAGERS

MARKETING TEAM

Terpresso

MISSION STATEMENTS

RANKED

MOST AND LEAST SELLING PRODUCTS

TOP 3

MOST AND LEAST PROFITABLE PRODUCTS

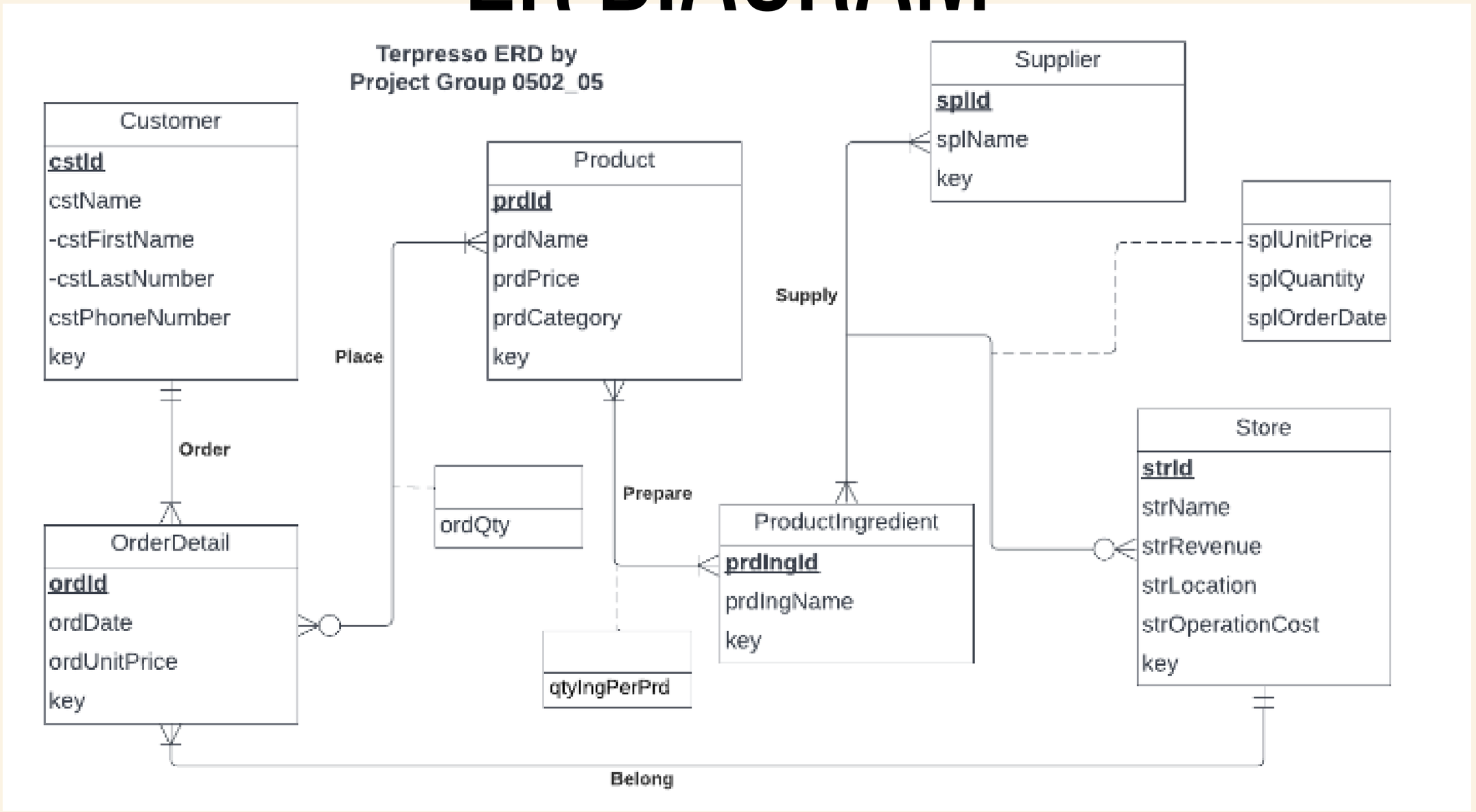
CUSTOMER LOYALTY

IDENTIFY RETURNING CUSTOMERS AND PUSH PROMOTIONAL MATERIAL

TOP STORE

IDENTIFY BEST PERFORMING STORE TO REDUCE OVERHEAD, INVEST BETTER TO EXPAND MORE

ER DIAGRAM


Terpresso ERD by
Project Group 0502_05
Customer
cstId
cstName
-cstFirstName
-cstLastNumber
cstPhoneNumber
key
Product
prdId
prdName
prdPrice
prdCategory
key
Supplier
splId
splName
key
splUnitPrice
splQuantity
splOrderDate
Place
Supply
Order
ordQty
Prepare
OrderDetail
ordId
ordDate
ordUnitPrice
key
ProductIngredient
prdIngId
prdIngName
key
qtyIngPerPrd
Store
strId
strName
strRevenue
strLocation
strOperationCost
key
Belong

RELATIONAL SCHEMA

Relational Schema:

Customer (**cstId**, key, cstFirstName, cstLastName, cstPhoneNumber)

OrderDetail (**ordId**, key, *cstId, strId*, ordDate, ordUnitPrice)

Product (**prdId**, key, prdName, prdPrice, prdCategory)

ProductIngredient (**prdingId**, key, prdIngName)

Prepare (***prdIngId***, ***prdId*** , qtyIngPerPrd)

Store (**strId**, key, strName, strLocation, strRevenue, strOperationCost).

Supplier (**splId**, key, splName)

Place (***ordId***, ***prdId***, ordQty)

Supply (***splId***, ***prdIngId***, ***strId***, splUnitPrice, splQuantity, splOrderDate)

PHYSICAL DATABASE DESIGN

```
CREATE TABLE Customer(
    cstKey INT IDENTITY(10,1),
   cstId AS 'CST' + RIGHT('00' + CAST(cstKey AS VARCHAR(10)),7)PERSISTED NOT NULL,
   cstFirstName VARCHAR(20),
   cstLastName VARCHAR(20),
   cstPhoneNumber CHAR(10),
 CONSTRAINT pk_Customer_cstId  PRIMARY KEY (cstId))
```

PHYSICAL DATABASE DESIGN

```
CREATE TABLE Store(
    strKey INT IDENTITY(10,1),
    strId AS 'STR' + RIGHT('00' + CAST(strKey AS VARCHAR(10)),7)PERSISTED NOT NULL,
    strName VARCHAR(50),
    strLocation VARCHAR(50),
    strRevenue DECIMAL(7,2),
    strOperationCost DECIMAL(7,2),
CONSTRAINT pk_Store_strId  PRIMARY KEY (strId))
```

PHYSICAL DATABASE DESIGN

```
CREATE TABLE OrderDetail(
    ordKey INT IDENTITY(10,1),
    ordId AS 'ORD' + RIGHT('00' + CAST(ordKey AS VARCHAR(10)),7)PERSISTED NOT NULL,
    cstId VARCHAR(10) NOT NULL,
    strId VARCHAR(10) NOT NULL,
    ordDate DATE,
    ordUnitPrice DECIMAL(5,2),
    CONSTRAINT pk_OrderDetail_ordId PRIMARY KEY (ordId),
    CONSTRAINT fk_OrderDetail_cstId FOREIGN KEY (cstId)
                    REFERENCES Customer(cstId)
                    ON DELETE NO ACTION ON UPDATE CASCADE,
    CONSTRAINT fk_OrderDetail_strId FOREIGN KEY (strId)
                    REFERENCES Store (strId)
                    ON DELETE CASCADE ON UPDATE CASCADE)
```

PHYSICAL DATABASE DESIGN

```
CREATE TABLE Product (
    prdKey INT IDENTITY(10,1),
    prdId AS 'PRD' + RIGHT('00' + CAST(prdKey AS VARCHAR(10)),7)PERSISTED NOT NULL,
    prdName VARCHAR(50),
    prdPrice DECIMAL(5,2),
    prdCategory VARCHAR(20),
    CONSTRAINT pk_Product_prdId  PRIMARY KEY (prdId))
```

PHYSICAL DATABASE DESIGN

```
CREATE TABLE Supplier(
    splKey INT IDENTITY(10,1),
    splId AS 'SPL' + RIGHT('00' + CAST(splKey AS VARCHAR(10)),7)PERSISTED NOT NULL,
    splName VARCHAR(50),
    CONSTRAINT pk_Supplier_splId  PRIMARY KEY (splId))
```

TERPRESSO: 0502_05

PHYSICAL DATABASE DESIGN

```
CREATE TABLE Place (
   ordId VARCHAR (10) NOT NULL,
prdId VARCHAR (10) NOT NULL,
ordQty INT,
CONSTRAINT pk_Place_ordId_prdId
 PRIMARY KEY (ordId ,prdId),
CONSTRAINT fk_Place_ordId FOREIGN KEY (ordId)
 REFERENCES OrderDetail(ordId)
 ON DELETE CASCADE ON UPDATE CASCADE,
CONSTRAINT fk_Place_prdId FOREIGN KEY (prdId )
 REFERENCES Product(prdId )
 ON DELETE CASCADE ON UPDATE CASCADE )
```

PHYSICAL DATABASE DESIGN

```
CREATE TABLE ProductIngredient(
   prdIngKey INT IDENTITY(10,1),
 prdIngId AS 'PIN' + RIGHT('00' + CAST(prdIngKey AS VARCHAR(10)),7)PERSISTED NOT NULL,
 prdIngName VARCHAR(50),
 CONSTRAINT pk_ProductIngredient_prdIngId  PRIMARY KEY (prdIngId))

CREATE TABLE Prepare (
   prdIngId VARCHAR(10) NOT NULL,
   prdId VARCHAR(10) NOT NULL,
 qtyIngPerPrd DECIMAL(5,2),
 CONSTRAINT pk_Prepare_prdIngId_prdId
  PRIMARY KEY (prdIngId ,prdId),
 CONSTRAINT fk_Prepare_prdIngId  FOREIGN KEY (prdIngId )
  REFERENCES ProductIngredient(prdIngId)
  ON DELETE CASCADE ON UPDATE CASCADE,
 CONSTRAINT fk_Prepare_prdId FOREIGN KEY (prdId )
  REFERENCES Product(prdId )
  ON DELETE CASCADE ON UPDATE CASCADE )
```

PHYSICAL DATABASE DESIGN

```
CREATE TABLE Supply (
   splId VARCHAR (10) NOT NULL,
   prdIngId VARCHAR (10) NOT NULL,
   strId VARCHAR (10) NOT NULL,
splUnitPrice DECIMAL(6,2),
   splQuantity INT,
   splOrderDate DATE,
   CONSTRAINT pk_Supply__splId_prdId_strId
PRIMARY KEY (splId , prdIngId, strId),
   CONSTRAINT fk_Supply_splId FOREIGN KEY (splId)
REFERENCES Supplier(splId )
ON DELETE CASCADE ON UPDATE CASCADE,
   CONSTRAINT fk_Supply_prdIngId FOREIGN KEY (prdIngId)
REFERENCES ProductIngredient(prdIngId )
ON DELETE CASCADE ON UPDATE CASCADE,
   CONSTRAINT fk_Supply_strId FOREIGN KEY (strId)
REFERENCES Store(strId)
ON DELETE CASCADE ON UPDATE CASCADE)
```

Terpresso DATABASE
BY GROUP 0502-05

WHAT ARE THE RANKED MOST AND LEAST SELLING PRODUCTS?

```sql
CREATE VIEW BestAndLeastProduct
AS
    SELECT pl.prdId, pr.prdName
    FROM Place pl, Product pr
    WHERE pr.prdId = pl.prdId

WITH CTE_RankedProducts AS
(
SELECT b.prdId AS [Product ID], b.prdName AS [Product Name],
       COUNT(b.prdId) AS [Frequency Ordered Product],
       RANK() OVER (ORDER BY COUNT(b.prdId) DESC) AS [Rank]
FROM BestAndLeastProduct b
GROUP BY b.prdId, b.prdName, b.prdId
)
SELECT * FROM CTE_RankedProducts a,
      (select TOP 1 RANK
        from CTE_RankedProducts W
        order by W.Rank desc) v
WHERE a.Rank <= 3
OR a.Rank = v.Rank
```

	Product ID	Product Name	Frequency Ordered Product	Rank
1	PRD0021	Classic Chai Tea Latte	15	1
2	PRD0031	Everything Breakfast Sandwich	14	2
3	PRD0028	Butter Croissant	11	3
4	PRD0034	Sausage & Cheddar	3	22
5	PRD0032	Spinach & Feta Wrap	3	22
6	PRD0016	Caramel Macchiato	3	22
7	PRD0019	White Chocolate Mocha	3	22

Terpresso DATABASE

BY GROUP 0502-05

WHAT ARE THE TOP 3 MOST AND LEAST PROFITABLE PRODUCTS?

```sql
--Displaying the top 3 profitable and loss making products
SELECT * FROM (
     SELECT TOP 3  p.prdName AS 'Product Name',  pt.prdId AS 'Product ID',
            CAST(pt.CostPrice AS DECIMAL(5, 2)) AS 'Cost Price Of Product',
            pt.SellingPrice  AS 'Product SellingPrice',CAST(pt.Profit AS DECIMAL(5, 2)) AS 'Profit Per Product'
    FROM ProfitGenerated pt, Product p
    WHERE pt.prdId = p.prdId
    ORDER BY Profit DESC
                ) TopProfitableProducts
UNION
SELECT * FROM(
    SELECT TOP 3  p.prdName AS 'Product Name',  pt.prdId AS 'Product ID',
            CAST(pt.CostPrice AS DECIMAL(5, 2)) AS 'Cost Price Of Product',
            pt.SellingPrice  AS 'Product SellingPrice',CAST(pt.Profit AS DECIMAL(5, 2)) AS 'Profit Per Product'
    FROM ProfitGenerated pt, Product p
    WHERE pt.prdId = p.prdId
    ORDER BY Profit ASC
            ) LeastProfitableProducts
ORDER BY [Profit Per Product] DESC
```

	Product Name	Product ID	Cost Price Of Product	Product SellingPrice	Profit Per Product
1	Everything Breakfast Sandwich	PRD0031	26.60	133.70	107.10
2	Butter Croissant	PRD0028	18.70	60.50	41.80
3	Freshly Brewed - Medium	PRD0013	28.00	60.00	32.00
4	Café Mocha	PRD0020	82.50	36.75	-45.75
5	Cinnamon Chai Latte	PRD0025	265.50	60.75	-204.75
6	Classic Chai Tea Latte	PRD0021	435.00	101.25	-333.75

Terpresso DATABASE
BY GROUP 0502-05

WHO ARE TERPRESSO'S MOST FREQUENTLY RETURNING CUSTOMERS AND THEIR CONTACT DETAILS?

```sql
CREATE VIEW RegularCustomers
AS
    SELECT c.cstId AS 'Customer Id', c.cstFirstName + ' ' + c.cstLastName AS 'Customer Full Name',
           o.cstId AS 'Order Details Customer Id', o.ordId AS 'Order Id',
           o.ordDate AS 'Order Date', c.cstPhoneNumber AS 'Customer Phone Number'
    FROM Customer c, OrderDetail o WHERE c.cstId = o.cstId

-- Who are Terpresso's most frequently returning customers and their contact details?
WITH CTE_TopCustomer AS
(
    SELECT [Customer Full Name], [Customer Phone Number],
           COUNT([Customer Id]) AS [Ordered Frequency],
           RANK() OVER (ORDER BY COUNT([Customer Id]) DESC)
           As [Rank]
    FROM RegularCustomers
    GROUP BY [Customer Full Name], [Customer Phone Number], [Customer Id]
)
SELECT [Customer Full Name] AS 'Top Returning Customers',
       [Customer Phone Number] AS 'Customer Contact Detail',
       [Ordered Frequency] AS 'Number of times ordered from Terpresso',[Rank]
FROM CTE_TopCustomer
WHERE [Rank] = 1
```

	Top Returning Customers	Customer Contact Detail	Number of times ordered from Terpresso	Rank
1	Madhulika Nambi	9764568890	7	1
2	Taylor McKessie	4568762345	7	1

Terpresso DATABASE
BY GROUP 0502-05

WHAT IS THE STORE INFORMATION AND THE TOTAL SALES PROFIT PER STORE?

```sql
CREATE VIEW TopPerformingStore
AS
    SELECT pl.prdId AS 'Product ID', p.prdName AS 'Product Name', p.prdCategory AS 'Product Category',
           ( p.prdPrice*pl.ordQty) AS 'Total Product Price ', s.strId AS 'Store ID' ,s.strName AS 'Store Name' ,
           s.strLocation AS 'Store Location', s.strOperationCost AS 'Store Operation Cost',
           s.strRevenue AS'Store Revenue(Other Services)'
    FROM Place pl, Product p, OrderDetail o, Store s
    WHERE pl.ordId = o.ordId
           AND o.strId = s.strId
           AND pl.prdId = p.prdId

-- What  is the store information and the total sales profit per store?
SELECT  [Store Name],[Store Location],
        (([Store Revenue(Other Services)] + b.PrdRevenue) - [Store Operation Cost] )AS 'Total Store Sales Profit'
FROM TopPerformingStore tp
LEFT JOIN
(
SELECT [Store ID], SUM([Total Product Price ]) AS PrdRevenue
FROM TopPerformingStore
GROUP BY [Store ID]
) b
ON tp.[Store ID] = b.[Store ID]
GROUP BY tp.[Store ID], [Store Name],
        [Store Location],
        (([Store Revenue(Other Services)] + b.PrdRevenue) - [Store Operation Cost] )
```

	Store Name	Store Location	Total Store Sales Profit
1	Terpresso Local	College Park, MD	8672.05
2	Terpresso Express	Hyattsville, MD	2471.35

TERPRESSO: 0502_05
Anwesha Gupta (0502)
Janhavi Namjoshi (0502)
Madhulika Nambi (0502)
Udit Singh (0501)

Terpresso

THANK YOU