**Query 1**

i)start before aug end at aug,

ii)start at aug end after aug,

iii)start and end at aug,

iv)start before aug end after aug

SELECT p1.SName,AVG(p1.Price) as AvgPrice\_IPhoneX

FROM PRICE\_HISTORY p1

WHERE ((p1.EndDate>='2021-08-01' AND p1.EndDate<='2021-08-31')OR(p1.StartDate>='2021-08-01' AND p1.StartDate<='2021-08-31')

OR(p1.StartDate<'2021-08-01'AND p1.EndDate>'2021-08-31'))

AND p1.PName LIKE 'Apple iPhone X'

GROUP BY p1.SName;

**Query 2**

SELECT FEEDBACK\_R1.PName,ROUND(AVG(CAST(Rating AS FLOAT)),2) AS AvgRating

FROM FEEDBACK\_R1

INNER JOIN PRODUCTS

ON FEEDBACK\_R1.PName = PRODUCTS.PName

WHERE [FDate\_time] >= '2021-08-01'

AND [FDate\_time] < '2021-09-01'

AND FEEDBACK\_R1.PName IN (SELECT FEEDBACK\_R1.PName

FROM FEEDBACK\_R1

INNER JOIN PRODUCTS

ON FEEDBACK\_R1.PName = PRODUCTS.PName

AND [FDate\_time] >= '2021-08-01'

AND [FDate\_time] < '2021-09-01'

GROUP BY FEEDBACK\_R1.PName, Rating

HAVING Rating = 5 AND COUNT(\*) >= 100)

GROUP BY FEEDBACK\_R1.PName

**Query 3:**

For all products purchased in June 2021 that have been delivered, find the average time from the ordering date to the delivery date.

* Products, Products-in-Orders, Orders (join orders and products-in-orders)

# PRODUCTS (PName, Maker, Category)

# PRODUCTS-IN-ORDERS (PName, SName, OID, OPID, OPrice, OQuantity, Delivery-date, Status)

# ORDERS (OID, UID, Date-time, Shipping-address)

SELECT PO.PName, AVG(DATEDIFF(day, O.ODate\_time, PO.Delivery\_date)) as average\_time

FROM PRODUCTS\_IN\_ORDERS AS PO, ORDERS AS O

WHERE PO.Ostatus = 'delivered'

AND (O.ODate\_time >= '2021-06-01' AND O.ODate\_time <= '2021-06-30')

AND PO.OID = O.OID

GROUP BY PO.PName;

**Query 4:**

WITH A AS (SELECT E.EmployeeID, AVG(DATEDIFF(day, C.Filed\_DATE, H.handled\_DATE)) AS Latency, E.EName

FROM dbo.COMPLAINTS AS C, dbo.EMPLOYEE AS E, dbo.HANDLED AS H

WHERE C.CID = H.CID AND E.EmployeeID = H.EmployeeID

GROUP BY E.EmployeeID, E.EName)

SELECT A.EmployeeID, A.EName

FROM A

WHERE Latency = (SELECT MIN(Latency) FROM A)

**//Alternative**

SELECT TOP 1 EMPLOYEE.EmployeeID,EMPLOYEE.EName

FROM EMPLOYEE

INNER JOIN HANDLED ON EMPLOYEE.EmployeeID=HANDLED.EmployeeID

INNER JOIN COMPLAINTS ON COMPLAINTS.CID=HANDLED.CID

GROUP BY EMPLOYEE.EmployeeID,EMPLOYEE.EName

ORDER BY AVG(DATEDIFF(day, COMPLAINTS.Filed\_DATE, HANDLED.handled\_DATE)) ASC;

**Query 5:**

SELECT PRODUCTS.PName AS Product, COUNT(PRODUCTS\_IN\_SHOPS\_R2.PName) AS No\_of\_Shops

FROM PRODUCTS

LEFT JOIN PRODUCTS\_IN\_SHOPS\_R2

ON PRODUCTS.PName = PRODUCTS\_IN\_SHOPS\_R2.PName

WHERE maker = 'Samsung'

GROUP BY PRODUCTS.PName;

**Query 6:**

SELECT TOP 1 SName

FROM PRODUCTS\_IN\_ORDERS AS PO JOIN ORDERS AS O ON O.OID = PO.OID

WHERE MONTH(ODate\_time) = 8 AND YEAR (ODate\_time) = 2021

GROUP BY PO.SName

ORDER BY SUM(OPrice \* OQuantity) DESC

**Query 7:**

For users that made the most amount of complaints, find the most expensive products he/she has

ever purchased.

* Complaints,Users,Orders,Products\_in\_Orders
* COMPLAINTS(CID(auto),UID, CText, CStatus, Filed\_DATE)
* USERS(UID, UNAME)
* ORDERS(OID, UID, ODate\_time, Shipping\_address)
* PRODUCTS\_IN\_ORDERS(PName,SName,OID,OPID,OPrice,OQuantity,Delivery\_date,OStatus)

**//Alternative (Apple iPhone X)**

SELECT TOP 1 PName

FROM PRODUCTS\_IN\_ORDERS

INNER JOIN ORDERS ON PRODUCTS\_IN\_ORDERS.OID=ORDERS.OID

WHERE UID = (SELECT TOP 1 UID

FROM COMPLAINTS

GROUP BY UID

ORDER BY COUNT(CID) DESC)

ORDER BY OPrice DESC

JK:

SELECT PNAME

FROM

SELECT PName, MAX(Oprice/OQuantity) AS UnitPrice)

FROM R2, PRODUCTS\_IN\_ORDERS AS P

WHERE R2.OID = P.OID

GROUP BY PName

R3[PName, maxprice]

//find UID with most number of complaints

SELECT UID, COUNT(CID) AS num\_complaints

FROM COMPLAINTS

GROUP BY UID

ORDER BY num\_complaints DESC

LIMIT 1

R1 = [UID, num\_complaints]

//return OIDs of selected user

SELECT OID

FROM ORDERS

WHERE R1.UID = ORDERS.UID

R2 = [OID]

//join [OID, PName, UnitPrice], Might have duplicates because of removal of OPID

SELECT PName, MAX(Oprice/OQuantity) AS UnitPrice)

FROM R2, PRODUCTS\_IN\_ORDERS AS P

WHERE R2.OID = P.OID

GROUP BY PName

R3[PName, maxprice]

**Query 8**

SELECT TOP 5 PName

FROM PRODUCTS\_IN\_ORDERS

WHERE PName IN (

SELECT PName

FROM PRODUCTS\_IN\_ORDERS JOIN dbo.ORDERS ON PRODUCTS\_IN\_ORDERS.OID = ORDERS.OID

GROUP BY PName

HAVING COUNT(DISTINCT UID) < (

SELECT COUNT(UID) FROM dbo.USERS

)

)

GROUP BY PName

ORDER BY SUM(OQuantity) DESC

**Query 9(adding new products)**

WITH A AS (

SELECT PName, DATEFROMPARTS(YEAR(ODate\_time), MONTH(ODate\_time), 1) AS yearmonth, SUM(OQuantity) AS totalQuantity

FROM dbo.PRODUCTS\_IN\_ORDERS JOIN dbo.ORDERS ON PRODUCTS\_IN\_ORDERS.OID = ORDERS.OID

GROUP BY PName, DATEFROMPARTS(YEAR(ODate\_time), MONTH(ODate\_time), 1)

)

SELECT A.PName FROM A, A AS B, A AS C

WHERE (

A.PName = B.PName AND

B.PName = C.PName AND

DATEDIFF(month, A.yearmonth, B.yearmonth) = 1 AND

DATEDIFF(month, B.yearmonth, C.yearmonth) = 1 AND

A.totalQuantity <= B.totalQuantity AND

B.totalQuantity <= C.totalQuantity

)