Accounting System

Group ID: 5
Student Details:

Jay Kothari	202012017
Aaryan Gambhir	202012018
Dharmik Dave	202012019
Twesha Satia	202012020
Drumil Shah	202018009
Preet Patel	202018010

Lab: Lab 7

```
SET search_path TO AccountingSystem,public;
SET DATESTYLE TO European;
```

Queries:

1. Given a date each company can retrieve the financial transactions of that particular date.

Relational Algebra:

$\mathcal{F}_{\text{(FT.TransactionNumber, FT.TransactionDate, ABS(F.Amount) -> Amount, FT.TransactionDescription, AccountName, CompanyName)}}(\sigma_{\text{(CompanyName = 'Myntra' AND FT.TransactionDate = '23-10-2020')}}((((\rho_{ft}(\text{FinancialTransactions}) \bowtie_{<ft.AccountID=f.AccountID>} \rho_{f}(\text{FinancialTransactionEntry})) \bowtie_{<f.AccountID=A.AccountID>} \rho_{A}(\text{Account})) \bowtie_{<A.AccountGroupId=AG.AccountGroupId>} \rho_{AG}(\text{AccountGroup})) \bowtie_{<AG.CompanyID=C.CompanyID>} \rho_{C}(\text{Company}))$

SQL Query:

```
Select FT.TransactionNumber, FT.TransactionDate, ABS(F.Amount) as Amount,
FT.TransactionDescription, AccountName, CompanyName
From FinancialTransaction as FT JOIN FinancialTransactionEntry as F ON (FT.TransactionNumber =
F.TransactionNumber)
JOIN Account as A ON (F.AccountId = A.AccountId)
JOIN AccountGroup as AG ON (A.AccountGroupId=AG.AccountGroupId)
JOIN Company as C ON (C.CompanyID = AG.CompanyId)
Where CompanyName = 'Myntra' AND FT.TransactionDate = '31-03-2020';
```

Output:

	transactionnumber integer	transactiondate date	amount numeric	transactiondescription character varying (300)	accountname character varying (50)	companyname character varying (50)
1	25	2020-03-31	33500.00	5% Depreciation on Building a...	Depreciation	Myntra
2	25	2020-03-31	25000.00	5% Depreciation on Building a...	Building	Myntra
3	25	2020-03-31	8500.00	5% Depreciation on Building a...	Furniture and Fixtures	Myntra

2. Each company can retrieve all the personal accounts that are associated with it.

Relational Algebra:

$\pi_{(PersonalAccount.CompanyName,\ PersonalAccount.GSTIN)}(\sigma_{(Company.CompanyName = 'Myntra')}(((PersonalAccount \bowtie_{<PersonalAccount.AccountId = Account.AccountId>} Account) \bowtie_{<Account.AccountGroupId = AccountGroup.AccountGroupId>} AccountGroup) \bowtie_{<Company.CompanyID = AccountGroup.CompanyId>} Company)))$

SQL Query:

```
Select P.CompanyName, P.GSTIN
From PersonalAccount as P JOIN Account as A ON ( P.AccountId = A.AccountId )
JOIN AccountGroup as AG ON (AG.AccountGroupId = A.AccountGroupId)
JOIN Company as C ON (C.CompanyID = AG.CompanyId)
Where C.CompanyName = 'Myntra';
```

Output:

	companyname character varying (50)	gstin character varying (15)
1	Raman	WDF23U9WP3M231J
2	M/SFitnessApparels	WDF23U9WP3M222J

**Query no. 3,4 & 5 will be only possible with procedures.*

3. Create a Company and all its necessary accounts.

4. Create a Sales invoice.

5. Create a Purchase receipt.

6. Retrieve any invoice by their bill number.

Relational Algebra:

$\pi_{(C.CompanyName,A.AccountName,Si.InvoiceId,Si.Date,It.ItemName,I.Amount,I.Qty)}(\sigma_{(C.CompanyName = 'Myntra' \text{ and } Si.Date= '10\text{-}23\text{-}2020'')}((((SalesInvoice \bowtie_{<Si.AccountId = A.AccountId>} Account) \bowtie_{<C.CompanyID = Si.CompanyId>} Company) \bowtie_{<Si.ReceiptID = I.ReceiptID \text{ and } Si.CompanyId=I.CompanyId>} SalesItem) \bowtie_{<I.ItemID = It.ItemID>} Item))$

SQL Query:

```
Select C.CompanyName,A.AccountName ,Si.InvoiceId,Si.Date,It.ItemName,I.Amount,I.Qty
From SalesInvoice as Si
JOIN Account as A ON ( Si.AccountId = A.AccountId )
JOIN Company as C ON (C.CompanyID = Si.CompanyId)
JOIN SalesItem as I ON (Si.InvoiceId = I.InvoiceId and Si. CompanyId=I. CompanyId)
JOIN Item as It ON ( I.ItemID = It.ItemID )
Where C.CompanyName = 'Myntra' AND Si.InvoiceId=1;
```

Output:

	companyname character varying (50)	accountname character varying (50)	invoiceid integer	date date	itemname character varying (100)	amount numeric (15,2)	qty real
1	Myntra	Cash	1	2020-09...	Men Grey Skinny Fit Low-Rise ...	5000.00	2
2	Myntra	Cash	1	2020-09...	Suit Felted Shacket	7000.00	1

7. Retrieve any receipt by their receipt number.

Relational Algebra:

$\pi_{(C.CompanyName,A.AccountName,Pi.ReceiptID,Pi.Date,It.ItemName,I.Amount,I.Qty)}(\sigma_{(C.CompanyName = 'Myntra' \text{ and } Pi.Date= '10\text{-}23\text{-}2020')}$
$((((PurchaseInvoice \bowtie_{<Pi.AccountId = A.AccountId>} Account) \bowtie_{<C.CompanyID = Pi.CompanyId>} Company) \bowtie_{<Pi.ReceiptID = I.ReceiptID \text{ and } Pi.CompanyId=I.CompanyId>} PurchaseItem) \bowtie_{<I.ItemID = It.ItemID>} Item))$

SQL Query:

```
Select C.CompanyName,A.AccountName ,Pi.ReceiptID,Pi.Date,It.ItemName,I.Amount,I.Qty
From PurchaseInvoice as Pi
JOIN Account as A ON ( Pi.AccountId = A.AccountId )
JOIN Company as C ON (C.CompanyID = Pi.CompanyId)
JOIN PurchaseItem as I ON (Pi.ReceiptID = I.ReceiptID and Pi. CompanyId=I. CompanyId)
JOIN Item as It ON ( I.ItemID = It.ItemID )
Where C.CompanyName = 'Myntra' AND Pi.ReceiptID=1;
```

Output:

	companyname character varying (50)	accountname character varying (50)	receiptid integer	date date	itemname character varying (100)	amount numeric (15,2)	qty real
1	Myntra	Cash	1	2020-10...	Men Grey Casual Fit West-Ris...	7100.00	6

8. Can retrieve details of the purchase made on a particular day.

Relational Algebra:

$\pi_{(C.CompanyName,A.AccountName,Pi.ReceiptID,Pi.Date,It.ItemName,I.Amount,I.Qty)}(\sigma_{(C.CompanyName = 'Myntra' \text{ and } Pi.Date= '08\text{-}10\text{-}2020')}((((PurchaseInvoice \bowtie_{<Pi.AccountId = A.AccountId>} Account) \bowtie_{<C.CompanyID = Pi.CompanyId>} Company) \bowtie_{<Pi.ReceiptID = I.ReceiptID \text{ and } Pi.CompanyId=I.CompanyId>} PurchaseItem) \bowtie_{<I.ItemID = It.ItemID>} Item))$

SQL Query:

```
Select C.CompanyName,A.AccountName,Pi.ReceiptID,Pi.Date,It.ItemName,I.Amount,I.Qty
From PurchaseInvoice as Pi
JOIN Account as A ON (Pi.AccountId = A.AccountId )
JOIN Company as C ON (C.CompanyID = Pi.CompanyId)
JOIN PurchaseItem as I ON (Pi.ReceiptID = I.ReceiptID and Pi. CompanyId=I.CompanyId)
JOIN Item as It ON (I.ItemID = It.ItemID )
Where C.CompanyName = 'Myntra' and Pi.Date= '08-10-2020';
```

Output:

	companyname character varying (50)	accountname character varying (50)	receiptid integer	date date	itemname character varying (100)	amount numeric (15,2)	qty real
1	Myntra	Cash	1	2020-10...	Men Grey Casual Fit West-Ris...	7100.00	6

9. Can retrieve details of the sales made on a particular day.

Relational Algebra:

$\pi_{(C.CompanyName,A.AccountName,Si.InvoiceId,Si.Date,It.ItemName,I.Amount,I.Qty)}(\sigma_{(C.CompanyName = 'Myntra' and Si.Date= '10-23-2020")}((((SalesInvoice \bowtie_{<Si.AccountId = A.AccountId>} Account) \bowtie_{<C.CompanyID = Si.CompanyId>} Company) \bowtie_{<Si.ReceiptID = I.ReceiptID and Si.CompanyId=I.CompanyId>} SalesItem) \bowtie_{<I.ItemID = It.ItemID>} Item))$

SQL Query:

```
Select C.CompanyName,A.AccountName ,Si.InvoiceId,Si.Date,It.ItemName,I.Amount,I.Qty
From SalesInvoice as Si
JOIN Account as A ON ( Si.AccountId = A.AccountId )
JOIN Company as C ON (C.CompanyID = Si.CompanyId)
JOIN SalesItem as I ON (Si.InvoiceId = I.InvoiceId and Si. CompanyId=I. CompanyId)
JOIN Item as It ON (I.ItemID = It.ItemID )
Where C.CompanyName = 'Myntra' and Si.Date= '05-9-2020';
```

Output:

	companyname character varying (50)	accountname character varying (50)	invoiceid integer	date date	itemname character varying (100)	amount numeric (15,2)	qty real
1	Myntra	Cash	1	2020-09...	Men Grey Skinny Fit Low-Rise ...	5000.00	2
2	Myntra	Cash	1	2020-09...	Suit Felted Shacket	7000.00	1

10. Can retrieve details of the purchases whose amount is between a given range.

Relational Algebra:

$\pi_{(C.CompanyName,A.AccountName,Pi.ReceiptID,Pi.Date,It.ItemName,I.Amount,I.Qty)}(\sigma_{(C.CompanyName = 'Myntra' and I.Amount BETWEEN (10000 AND 2000000))}$

$((((PurchaseInvoice \bowtie_{<Pi.AccountId = A.AccountId>} Account) \bowtie_{<C.CompanyID = Pi.CompanyId>} Company) \bowtie_{<Pi.ReceiptID = I.ReceiptID AND Pi.CompanyId=I.CompanyId>} PurchaseItem) \bowtie_{<I.ItemID = It.ItemID>} Item))$

SQL Query:

```
Select C.CompanyName,A.AccountName,Pi.ReceiptID,Pi.Date,It.ItemName,I.Amount,I.Qty
From PurchaseInvoice as Pi JOIN Account as A ON (Pi.AccountId = A.AccountId )
JOIN Company as C ON (C.CompanyID = Pi.CompanyId)
JOIN PurchaseItem as I ON (Pi.ReceiptID = I.ReceiptID and Pi.CompanyId=I.CompanyId)
JOIN Item as It ON (I.ItemID = It.ItemID )
Where C.CompanyName = 'Myntra' AND PI.ReceiptID IN (
        Select PI.ReceiptID
        FROM PurchaseInvoice as PI JOIN PurchaseItem as P ON(PI.ReceiptID = P.ReceiptID)
        GROUP BY PI.ReceiptID
        HAVING SUM(P.Amount*P.Qty) BETWEEN 10000 and 2000000);
```

	companyname character varying (50)	accountname character varying (50)	receiptid integer	date date	itemname character varying (100)	amount numeric (15,2)	qty real
1	Myntra	Cash	1	2020-10...	Men Grey Casual Fit West-Ris...	7100.00	6
2	Myntra	Cash	2	2020-08...	Sui Peach-Coloured Solid Ligh...	2000.00	1
3	Myntra	Banks	4	2020-04...	Exercise Machines	75000.00	1
4	Myntra	Creditors	5	2020-05...	Fitness Apparels	25000.00	1

11. Can retrieve details of the sales whose amount is between a given range.
Relational Algebra:

$\pi_{(C.CompanyName,A.AccountName,Si.InvoiceID,Si.Date,It.ItemName,I.Amount,I.Qty)}$

$(\sigma_{(C.CompanyName = 'Myntra' and I.Amount BETWEEN 10000 and 200000)}$

$((((SalesInvoice \bowtie_{<Si.AccountId = A.AccountId>} Account) \bowtie_{<C.CompanyID = Si.CompanyId>} Company) \bowtie_{<Si.InvoiceID = I.InvoiceID AND Si.CompanyId=I.CompanyId>} SalesItem) \bowtie_{<I.ItemID = It.ItemID>} Item))$

SQL Query:

```
Select C.CompanyName,A.AccountName ,Si.InvoiceId,Si.Date,It.ItemName,I.Amount,I.Qty
From SalesInvoice as SI JOIN Account as A ON ( Si.AccountId = A.AccountId )
JOIN Company as C ON (C.CompanyID = Si.CompanyId)
JOIN SalesItem as I ON (Si.InvoiceId = I.InvoiceId and Si. CompanyId=I. CompanyId)
JOIN Item as It ON (I.ItemID = It.ItemID )
Where C.CompanyName = 'Myntra' AND SI.InvoiceId IN (
        Select SI.InvoiceId
        FROM SalesInvoice as SI JOIN SalesItem as S ON(SI.InvoiceId = S.InvoiceId)
        GROUP BY SI.InvoiceId
        HAVING SUM(S.Amount*S.Qty) BETWEEN 10000 and 200000);
```

	companyname character varying (50)	accountname character varying (50)	invoiceid integer	date date	itemname character varying (100)	amount numeric (15,2)	qty real
1	Myntra	Cash	1	2020-09...	Men Grey Skinny Fit Low-Rise ...	5000.00	2
2	Myntra	Cash	1	2020-09...	Suit Felted Shacket	7000.00	1
3	Myntra	Banks	3	2020-05...	Treadmills	23750.00	2
4	Myntra	Debtors	4	2020-05...	Exercise Machines	23750.00	1
5	Myntra	Cash	5	2020-05...	Exercise Cycles	5415.00	5

12. Can show the position of all the accounts of a given account group of a given company. Like showing the value of all the assets.
Relational Algebra:

$\pi_{AccountName,\ F_{sum(f.account)\ as\ total\_accounts}(\sigma_{(C.CompanyName = 'Myntra' and AG.Name= 'Assets')}}$

$((((\rho_{f}(FinancialTransactions) \bowtie_{<f.AccountID=A.AccountID>} \rho_{A}(Account))$

$\bowtie_{<A.AccountGroupId=AG.AccountGroupId>} \rho_{AG}(AccountGroup))$

$\bowtie_{<AG.CompanyID=C.CompanyID>} \rho_{C}(Company))$

SQL Query:

```
Select FT.TransactionDate, A.AccountName, F.Amount, AG.ShowIn
From FinancialTransaction as FT JOIN FinancialTransactionEntry as F ON (FT.TransactionNumber =
F.TransactionNumber)
JOIN Account as A ON(F.AccountId=A.AccountId)
JOIN AccountGroup as AG ON (AG.AccountGroupId = A.AccountGroupId)
JOIN Company as C ON (C.CompanyID = AG.CompanyId)
Where  C.CompanyName = 'Myntra' AND AG.Name = 'Office and Administration Expenses' AND
((AG.ShowIn = 'Balance Sheet' AND FT.TransactionDate <= '31-03-2020') OR (AG.ShowIn <>
```

'Balance Sheet' AND FT.TransactionDate BETWEEN '01-04-2019' AND '31-03-2020'))
Order By A.AccountName;

	transactiondate date	accountname character varying (50)	amount numeric (15,2)	showin accountingsystem.finalaccount
1	2019-04-07	Electricity	5000.00	Profit and Loss Account
2	2020-03-30	Office Electricity	40000.00	Profit and Loss Account
3	2019-04-15	Telephone	2000.00	Profit and Loss Account
4	2020-03-30	Telephone Charges	50000.00	Profit and Loss Account

13. List all the Journal Entries of a given Financial Year.
Relational Algebra:

$\mathcal{F}_{\text{(Ft.TransactionNumber, Ft.TransactionDate,A.AccountName,ABS(F.Amount) -> "Transaction Amount")}}(\sigma_{\text{(C.CompanyName = 'Myntra' AND Ft.TransactionDate BETWEEN '04-01-2019' AND '03-31-2020')}}(((((\rho_{ft}(\text{FinancialTransactions}) \bowtie_{<ft.AccountID=f.AccountID>} \rho_{f}(\text{FinancialTransactionEntry})) \bowtie_{<f.AccountID=A.AccountID>} \rho_{A}(\text{Account})) \bowtie_{<A.AccountGroupId=AG.AccountGroupId>} \rho_{AG}(\text{AccountGroup})) \bowtie_{<AG.CompanyID=C.CompanyID>} \rho_{C}(\text{Company}))$

SQL Query:

```
Select Ft.TransactionNumber, Ft.TransactionDate,A.accountName,ABS(F.Amount) as "Transaction
Amount"
From FinancialTransaction as FT JOIN FinancialTransactionEntry as F ON (FT.TransactionNumber =
F.TransactionNumber)
JOIN Account as A ON (F.accountId = A.accountId)
JOIN AccountGroup as AG ON (AG.AccountGroupId = A.AccountGroupId)
JOIN Company as C ON (C.CompanyId = AG.CompanyId)
Where C.CompanyName = 'Myntra' AND Ft.TransactionDate BETWEEN '04-01-2019' AND '03-31-
2020'
Order By Ft.TransactionNumber;
```

	transactionnumber integer	transactiondate date	accountname character varying (50)	Transaction Amount numeric
1	5	2019-04-01	Banks	300000.00
2	5	2019-04-01	Capital	300000.00
3	6	2019-04-03	Banks	300000.00
4	6	2019-04-03	Bank Loan	300000.00
5	7	2019-04-03	Building	500000.00
6	7	2019-04-03	Banks	500000.00
7	8	2019-04-03	Furniture and Fixtures	75000.00
8	8	2019-04-03	Banks	75000.00
9	9	2019-04-05	Banks	200000.00
10	9	2019-04-05	Capital	200000.00

14. Retrieve any given ledger account of a particular company.

Relational Algebra:

$\pi_{(FT.TransactionDate, A.AccountName, F.Amount, AG.ShowIn)}(\sigma_{(C.CompanyName = 'Myntra' AND A.AccountName = 'Cash' AND ((AG.ShowIn = 'Balance Sheet' AND FT.TransactionDate <= '31-03-2020') OR (AG.ShowIn <> 'Balance Sheet' AND FT.TransactionDate BETWEEN '01-04-2019' AND '31-03-2020'))}$
$(((((\rho_{ft}(FinancialTransactions)$
$\bowtie_{<ft.AccountID=f.AccountID>} \rho_f(FinancialTransactionEntry)) \bowtie_{<f.AccountID=A.AccountID>} \rho_A(Account))$
$\bowtie_{<A.AccountGroupId=AG.AccountGroupId>} \rho_{AG}(AccountGroup)) \bowtie_{<AG.CompanyID=C.CompanyID>} \rho_C(Company))$

SQL Query:

```
Select FT.TransactionDate, A.AccountName, F.Amount, AG.ShowIn
From FinancialTransaction as FT JOIN FinancialTransactionEntry as F ON (FT.TransactionNumber =
F.TransactionNumber)
JOIN Account as A ON(F.AccountId=A.AccountId)
JOIN AccountGroup as AG ON (AG.AccountGroupId = A.AccountGroupId)
JOIN Company as C ON (C.CompanyID = AG.CompanyId)
Where  C.CompanyName = 'Myntra' AND A.AccountName = 'Purchase' AND ((AG.ShowIn =
'Balance Sheet' AND FT.TransactionDate <= '31-03-2020') OR (AG.ShowIn <> 'Balance Sheet' AND
FT.TransactionDate BETWEEN '01-04-2019' AND '31-03-2020'));
```

	transactiondate date	accountname character varying (50)	amount numeric (15,2)	showin accountingsystem.finalaccount
1	2020-02-20	Purchase	1700000.00	Trading Account

15. Display all the Ledger of a company.

Relational Algebra:

$\pi_{(FT.TransactionDate, A.AccountName, F.Amount, AG.ShowIn)}(\sigma_{(C.CompanyName = 'Myntra' AND ((AG.ShowIn = 'Balance Sheet' AND FT.TransactionDate <= '31-03-2020') OR (AG.ShowIn <> 'Balance Sheet' AND FT.TransactionDate BETWEEN '01-04-2019' AND '31-03-2020')))}(((((\rho_{ft}(FinancialTransactions)$
$\bowtie_{<ft.AccountID=f.AccountID>} \rho_f(FinancialTransactionEntry)) \bowtie_{<f.AccountID=A.AccountID>} \rho_A(Account))$
$\bowtie_{<A.AccountGroupId=AG.AccountGroupId>} \rho_{AG}(AccountGroup)) \bowtie_{<AG.CompanyID=C.CompanyID>} \rho_C(Company))$

SQL Query:

```
Select FT.TransactionDate, A.AccountName, F.Amount, AG.ShowIn
From FinancialTransaction as FT JOIN FinancialTransactionEntry as F ON (FT.TransactionNumber =
F.TransactionNumber)
JOIN Account as A ON(F.AccountId=A.AccountId)
JOIN AccountGroup as AG ON (AG.AccountGroupId = A.AccountGroupId)
JOIN Company as C ON (C.CompanyID = AG.CompanyId)
Where  C.CompanyName = 'Myntra' AND ((AG.ShowIn = 'Balance Sheet' AND FT.TransactionDate
<= '31-03-2020') OR (AG.ShowIn <> 'Balance Sheet' AND FT.TransactionDate BETWEEN '01-04-
2019' AND '31-03-2020'))
Order By A.AccountName;
```

	transactiondate date	accountname character varying (50)	amount numeric (15,2)	showin accountingsystem.finalaccount
1	2020-03-30	Advertisement Banners	5000.00	Profit and Loss Account
2	2020-03-30	Bank Loan	-15000.00	Balance Sheet
3	2019-09-30	Bank Loan	-15000.00	Balance Sheet
4	2019-04-03	Bank Loan	300000.00	Balance Sheet
5	2019-12-30	Bank Loan	-15000.00	Balance Sheet
6	2019-06-30	Bank Loan	-15000.00	Balance Sheet
7	2019-04-07	Banks	-5000.00	Balance Sheet
8	2020-03-30	Banks	-5000.00	Balance Sheet
9	2019-04-15	Banks	-2000.00	Balance Sheet
10	2019-04-23	Banks	-10000.00	Balance Sheet

16. Show the turnover of a company for a particular year.
Relational Algebra:

$C.CompanyName\mathcal{F}(C.CompanyName, SUM(F.Amount) \rightarrow "TurnOver")(\sigma(C.CompanyName = 'Myntra' AND FT.TransactionDate BETWEEN '01-04-2019' AND '31-03-2020' AND AG.Name = 'Sales Account')(((((\rho_{ft}(FinancialTransactions) \bowtie_{<ft.AccountID=f.AccountID>} \rho_f(FinancialTransactionEntry)) \bowtie_{<f.AccountID=A.AccountID>} \rho_A(Account)) \bowtie_{<A.AccountGroupId=AG.AccountGroupId>} \rho_{AG}(AccountGroup)) \bowtie_{<AG.CompanyID=C.CompanyID>} \rho_C(Company))$

SQL Query:

```
Select C.CompanyName, SUM(F.Amount) As "TurnOver"
From FinancialTransaction as FT JOIN FinancialTransactionEntry as F ON (FT.TransactionNumber =
F.TransactionNumber)
JOIN Account as A ON(F.AccountId=A.AccountId)
JOIN AccountGroup as AG ON (AG.AccountGroupId = A.AccountGroupId)
JOIN Company as C ON (C.CompanyID = AG.CompanyId)
Where C.CompanyName = 'Myntra' AND FT.TransactionDate BETWEEN '01-04-2019' AND '31-03-
2020' AND AG.Name = 'Sales Account'
Group By C.CompanyName;
```

	companyname character varying (50)	TurnOver numeric
1	Myntra	2000000.00

17. Rank company by its turnover.
Relational Algebra:

$C.CompanyID\mathcal{F}(C.CompanyID, SUM(F.Amount) \rightarrow "TurnOver")(\sigma(FT.TransactionDate BETWEEN '01-04-2019' AND '31-03-2020' AND AG.Name = 'Sales Account')(((((\rho_{ft}(FinancialTransactions) \bowtie_{<ft.AccountID=f.AccountID>} \rho_f(FinancialTransactionEntry)) \bowtie_{<f.AccountID=A.AccountID>} \rho_A(Account)) \bowtie_{<A.AccountGroupId=AG.AccountGroupId>} \rho_{AG}(AccountGroup)) \bowtie_{<AG.CompanyID=C.CompanyID>} \rho_C(Company))$

SQL Query:

Select C.CompanyID, SUM(F.Amount) As "TurnOver"
From FinancialTransaction as FT JOIN FinancialTransactionEntry as F ON (FT.TransactionNumber = F.TransactionNumber)
JOIN Account as A ON(F.AccountId=A.AccountId)
JOIN AccountGroup as AG ON (AG.AccountGroupId = A.AccountGroupId)
JOIN Company as C ON (C.CompanyID = AG.CompanyId)
Where FT.TransactionDate BETWEEN '01-04-2020' AND '31-03-2021' AND AG.Name = 'Sales Account'
Group By C.CompanyID
Order By "TurnOver" DESC;

	companyid [PK] character varying (20)	TurnOver numeric
1	C00001	115325.00
2	C00002	105000.00

18.Item wise turnover of a company.
Relational Algebra:

$I.ItemID\mathcal{F}_{(I.ItemID,\ SUM(\ SI.Amount\ *\ SI.Qty\ )\ ->\ "TurnOver")}(\sigma_{(C.CompanyName\ =\ 'Myntra'\ And\ S.Date\ BETWEEN\ '01-04-2020'\ AND\ '31-03-2021')}(((((\rho_S(SalesInvoice) \bowtie_{<S.InvoiceId\ =\ SI.InvoiceId>} \rho_{SI}(SalesItem)) \bowtie_{<SI.ItemId\ =\ I.ItemId>} \rho_I(Item)) \bowtie_{<IG.ItemGroupID\ =\ I.ItemGroupID>} \rho_{IG}(ItemGroup)) \bowtie_{<IG.CompanyID\ =\ C.CompanyID>} \rho_C(Company))$

SQL Query:
Select I.ItemID, SUM(SI.Amount * SI.Qty) AS "TurnOver"
From SalesInvoice as S JOIN SalesItem as SI ON(S.InvoiceId = SI.InvoiceId)
JOIN Item as I ON (SI.ItemId = I.ItemId)
JOIN ItemGroup as IG ON (IG.ItemGroupID = I.ItemGroupID)
JOIN Company as C On (IG.CompanyID = C.CompanyID)
Where C.CompanyName = 'Myntra' And S.Date BETWEEN '01-04-2020' AND '31-03-2021'
Group By I.ItemID
Order By SUM(SI.Amount * SI.Qty) DESC;

	itemid [PK] character varying (20)	TurnOver double precision
1	I00012	1547500
2	I00013	500000
3	I00014	95000
4	I00015	27075
5	I00010	20000
6	I00006	14000

19. Calculate Gross Profit and Loss of a company of a given year.
Relational Algebra:

$r1 <- _{A.AccountId}\mathcal{F}_{SUM(F.Amount)->DrTradingSum}(\sigma_{(C.CompanyName\ =\ 'Myntra'\ AND\ AG.Name=\ 'Trading\ Account'\ AND\ AG.Header\ =\ True\ AND\ (F.TransactionDate\ >=\ '04-01-2019'\ AND\ F.TransactionDate\ <=\ '03-31-2020'))}$

$((((\rho_f(\text{FinancialTransactions}) \bowtie_{<f.AccountID=A.AccountID>} \rho_A(\text{Account}))$
$\bowtie_{<A.AccountGroupId=AG.AccountGroupId>} \rho_{AG}(\text{AccountGroup}))$
$\bowtie_{<AG.CompanyID=C.CompanyID>} \rho_C(\text{Company})) \cup \rho_{DrTradingSum}(0)$

DrTable<-$\mathcal{F}_{SUM(r1.DrTradingSum)->DrSide}$

r2<-$_{A.AccountId}\mathcal{F}_{SUM(F.Amount)->CrTradingSum}(\sigma_{(C.CompanyName = 'Myntra' AND AG.Name= 'Trading Account' AND AG.Header = False AND (F.TransactionDate >= '04-01-2019' AND F.TransactionDate <= '03-31-2020'))}$
$((((\rho_f(\text{FinancialTransactions}) \bowtie_{<f.AccountID=A.AccountID>} \rho_A(\text{Account}))$
$\bowtie_{<A.AccountGroupId=AG.AccountGroupId>} \rho_{AG}(\text{AccountGroup}))$
$\bowtie_{<AG.CompanyID=C.CompanyID>} \rho_C(\text{Company})) \cup \rho_{CrTradingSum}(0)$

CrTable<-$\mathcal{F}_{SUM(r2.CrTradingSum)->CrSide}$

GrossProfit<-$\sigma_{(CrSide - DrSide)}$(DrTable x CrTable)

SQL Query:

```
Select (CrSide - DrSide) as "Gross Profit"
From
        (
                Select SUM(DrTradingSum) as DrSide
                From(
                        Select SUM(F.Amount) as DrTradingSum
                        From FinancialTransaction as FT JOIN FinancialTransactionEntry as F ON
(FT.TransactionNumber = F.TransactionNumber)
                        JOIN Account as A ON(F.AccountId=A.AccountId)
                        JOIN AccountGroup as AG ON (AG.AccountGroupId = A.AccountGroupId)
                        JOIN Company as C ON (C.CompanyID = AG.CompanyId)
                        Where C.CompanyName = 'Myntra' AND AG.ShowIn = 'Trading Account' AND
AG.Header = True
                        AND FT.TransactionDate BETWEEN '01-04-2019' AND '31-03-2020'
                        GROUP BY A.AccountId
                        UNION
                        Select 0 as DrTradingSum
                ) as r1
        ) as DrTable,
        (
                Select SUM(CrTradingSum) as CrSide
                From(
                        Select SUM(F.Amount) as CrTradingSum
                        From FinancialTransaction as FT JOIN FinancialTransactionEntry as F ON
(FT.TransactionNumber = F.TransactionNumber)
```

```
                JOIN Account as A ON(F.AccountId=A.AccountId)
                JOIN AccountGroup as AG ON (AG.AccountGroupId = A.AccountGroupId)
                JOIN Company as C ON (C.CompanyID = AG.CompanyId)
                Where C.CompanyName = 'Myntra' AND AG.ShowIn = 'Trading Account' AND
AG.Header = False
                AND FT.TransactionDate BETWEEN '01-04-2019' AND '31-03-2020'
                GROUP BY A.AccountId
                UNION
                Select 0 as CrTradingSum
            ) as r2
        ) as CrTable;
```

	Gross Profit numeric
1	300000.00

20. Calculate the Net profit or loss of a company of a given year.

Relational Algebra:

$r1 \leftarrow {}_{A.AccountId}\mathcal{F}_{SUM(F.Amount)->DrPNLSum}(\sigma_{(C.CompanyName = 'Myntra' AND AG.ShowIn = 'Profit and Loss Account' AND AG.Header = True AND (FT.TransactionDate >= '04-01-2019' AND FT.TransactionDate <= '03-31-2020'))}$
$((((\rho_{ft}(FinancialTransactions)$
$\bowtie_{<ft.AccountID=f.AccountID>} \rho_{f}(FinancialTransactionEntry)) \bowtie_{<f.AccountID=A.AccountID>} \rho_{A}(Account))$
$\bowtie_{<A.AccountGroupId=AG.AccountGroupId>} \rho_{AG}(AccountGroup))$
$\bowtie_{<AG.CompanyID=C.CompanyID>} \rho_{C}(Company)) \cup \rho_{DrPNLSum}(0)$

$DrTable \leftarrow {}_{\mathcal{F}SUM(r1.DrPNLSum)->DrSide}$

$r2 \leftarrow {}_{A.AccountId}\mathcal{F}_{SUM(F.Amount)->CrPNLSum}(\sigma_{(C.CompanyName = 'Myntra' AND AG.ShowIn = 'Profit and Loss Account' AND AG.Header = True AND (FT.TransactionDate >= '04-01-2019' AND FT.TransactionDate <= '03-31-2020'))}$
$((((\rho_{ft}(FinancialTransactions)$
$\bowtie_{<ft.AccountID=f.AccountID>} \rho_{f}(FinancialTransactionEntry)) \bowtie_{<f.AccountID=A.AccountID>} \rho_{A}(Account))$
$\bowtie_{<A.AccountGroupId=AG.AccountGroupId>} \rho_{AG}(AccountGroup))$
$\bowtie_{<AG.CompanyID=C.CompanyID>} \rho_{C}(Company)) \cup \rho_{CrPNLSum}(0)$

$CrTable \leftarrow {}_{\mathcal{F}SUM(r1.CrPNLSum)->CrSide}$

$r3 \leftarrow {}_{A.AccountId}\mathcal{F}_{SUM(F.Amount)->DrTradingSum}(\sigma_{(C.CompanyName = 'Myntra' AND AG.Name= 'Trading Account' AND AG.Header = True AND (FT.TransactionDate >= '04-01-2019' AND FT.TransactionDate <= '03-31-2020'))}$
$((((\rho_{ft}(FinancialTransactions) \bowtie_{<ft.AccountID=f.AccountID>} \rho_{f}(FinancialTransactionEntry))$

$\bowtie_{<f.AccountID=A.AccountID>}\rho_{A}(Account))\ \bowtie_{<A.AccountGroupId=AG.AccountGroupId>}\rho_{AG}(AccountGroup))$
$\bowtie_{<AG.CompanyID=C.CompanyID>}\rho_{C}(Company))\ \cup\ \rho_{DrTradingSum}(0)$

DrTable<-$\mathcal{F}_{SUM(DrTradingSum)->DrSide}$

r4<-$_{A.AccountId}\mathcal{F}_{SUM(F.Amount)->CrTradingSum}(\sigma_{(C.CompanyName = 'Myntra' AND AG.Name= 'Trading Account' AND AG.Header = False AND (FT.TransactionDate >= '04-01-2019' AND FT.TransactionDate <= '03-31-2020'))}$
$(((((\rho_{ft}(FinancialTransactions)\bowtie_{<ft.AccountID=f.AccountID>}\rho_{f}(FinancialTransactionEntry))$
$\bowtie_{<f.AccountID=A.AccountID>}\rho_{A}(Account))\ \bowtie_{<A.AccountGroupId=AG.AccountGroupId>}\rho_{AG}(AccountGroup))$
$\bowtie_{<AG.CompanyID=C.CompanyID>}\rho_{C}(Company))\ \cup\ \rho_{CrTradingSum}(0)$

CrTable<-$\mathcal{F}_{SUM(CrTradingSum)->CrSide}$

GrossProfitTable<-$\sigma_{(r4.CrSide - r3.DrSide)}$(DrTable x CrTable)

Net Profit<-$\sigma_{(CrSide + GrossProfit - DrSide)}$(DrTable X CrTable X GrossProfitTable)

SQL Query:

```
Select (CrSide + GrossProfit - DrSide) as "Net Profit"
From
        (
                Select SUM(DrPNLSum) as DrSide
                From(
                        Select SUM(F.Amount) as DrPNLSum
                        From FinancialTransaction as FT JOIN FinancialTransactionEntry as F ON
(FT.TransactionNumber = F.TransactionNumber)
                        JOIN Account as A ON(F.AccountId=A.AccountId)
                        JOIN AccountGroup as AG ON (AG.AccountGroupId = A.AccountGroupId)
                        JOIN Company as C ON (C.CompanyID = AG.CompanyId)
                        Where C.CompanyName = 'Myntra' AND AG.ShowIn = 'Profit and Loss
Account' AND AG.Header = True
                        AND FT.TransactionDate BETWEEN '01-04-2019' AND '31-03-2020'
                        GROUP BY A.AccountId
                        UNION
                        Select 0 as DrPNLSum
                ) as r1
        ) as DrTable,
        (
                Select SUM(CrPNLSum) as CrSide
                From(
                        Select SUM(F.Amount) as CrPNLSum
                        From FinancialTransaction as FT JOIN FinancialTransactionEntry as F ON
```

```
(FT.TransactionNumber = F.TransactionNumber)
                        JOIN Account as A ON(F.AccountId=A.AccountId)
                        JOIN AccountGroup as AG ON (AG.AccountGroupId = A.AccountGroupId)
                        JOIN Company as C ON (C.CompanyID = AG.CompanyId)
                        Where C.CompanyName = 'Myntra' AND AG.ShowIn = 'Profit and Loss
Account' AND AG.Header = False
                        AND FT.TransactionDate BETWEEN '01-04-2019' AND '31-03-2020'
                        GROUP BY A.AccountId
                        UNION
                        Select 0 as DrPNLSum
                ) as r2
        ) as CrTable,
        (
                Select (CrSide - DrSide) as GrossProfit
                From
                (
                        Select SUM(DrTradingSum) as DrSide
                        From(
                                Select SUM(F.Amount) as DrTradingSum
                                From FinancialTransaction as FT JOIN FinancialTransactionEntry as F
ON (FT.TransactionNumber = F.TransactionNumber)
                                JOIN Account as A ON(F.AccountId=A.AccountId)
                                JOIN AccountGroup as AG ON (AG.AccountGroupId =
A.AccountGroupId)
                                JOIN Company as C
                                ON (C.CompanyID = AG.CompanyId)
                                Where C.CompanyName = 'Myntra' AND AG.ShowIn = 'Trading
Account' AND AG.Header = True
                                AND FT.TransactionDate BETWEEN '01-04-2019' AND '31-03-2020'
                                GROUP BY A.AccountId
                                UNION
                                Select 0 as DrTradingSum
                        ) as r1
                ) as DrTable,
                (
                        Select SUM(CrTradingSum) as CrSide
                        From(
                                Select SUM(F.Amount) as CrTradingSum
                                From FinancialTransaction as FT JOIN FinancialTransactionEntry as F
ON (FT.TransactionNumber = F.TransactionNumber)
                                JOIN Account as A ON(F.AccountId=A.AccountId)
                                JOIN AccountGroup as AG ON (AG.AccountGroupId =
A.AccountGroupId)
                                JOIN Company as C ON (C.CompanyID = AG.CompanyId)
                                Where C.CompanyName = 'Myntra' AND AG.ShowIn = 'Trading
Account' AND AG.Header = False
                                AND FT.TransactionDate BETWEEN '01-04-2019' AND '31-03-2020'
                                GROUP BY A.AccountId
                                UNION
```

```
                    Select 0 as DrTradingSum
                ) as r2
            ) as CrTable
        ) as GrossProfitTable;
```

	Net Profit numeric
1	31200.00

21. Prepare the final statement of a given company for a given year.
Relational Algebra:

SQL Query:

```
Select AG.ShowIn, A.AccountName, SUM(F.Amount) as "Amount", AG.Header
From FinancialTransaction as FT JOIN FinancialTransactionEntry as F ON (FT.TransactionNumber =
F.TransactionNumber)
JOIN Account as A ON(F.AccountId=A.AccountId)
JOIN AccountGroup as AG ON (AG.AccountGroupId = A.AccountGroupId)
JOIN Company as C ON (C.CompanyID = AG.CompanyId)
Where C.CompanyName = 'Myntra' AND AG.Name <> 'Capital'
Group By AG.ShowIn, AG.Header, A.AccountName
UNION
--To add all previuous year profit or loss to captial account
Select r1.ShowIn, r1.AccountName, ("Amount" +"Net Profit"), r1.Header
FROM (
        Select AG.ShowIn, A.AccountName, SUM(F.Amount) as "Amount", AG.Header
        From FinancialTransaction as FT JOIN FinancialTransactionEntry as F ON
(FT.TransactionNumber = F.TransactionNumber)
        JOIN Account as A ON(F.AccountId=A.AccountId)
        JOIN AccountGroup as AG ON (AG.AccountGroupId = A.AccountGroupId)
        JOIN Company as C ON (C.CompanyID = AG.CompanyId)
        Where C.CompanyName = 'Myntra' AND AG.Name = 'Capital'
        Group By AG.ShowIn, AG.Header, A.AccountName
        ) as r1
        ,(
                Select (CrSide + GrossProfit - DrSide) as "Net Profit"
                From
                        (
                                Select SUM(DrPNLSum) as DrSide
                                From(
                                        Select SUM(F.Amount) as DrPNLSum
                                        From FinancialTransaction as FT JOIN FinancialTransactionEntry
as F ON (FT.TransactionNumber = F.TransactionNumber)
                                        JOIN Account as A ON(F.AccountId=A.AccountId)
                                        JOIN AccountGroup as AG ON (AG.AccountGroupId =
A.AccountGroupId)
                                        JOIN Company as C ON (C.CompanyID = AG.CompanyId)
```

```
                                        Where C.CompanyName = 'Myntra' AND AG.ShowIn = 'Profit
and Loss Account' AND AG.Header = True
                                        AND FT.TransactionDate < '01-04-2020'
                                        GROUP BY A.AccountId
                                        UNION
                                        Select 0 as DrPNLSum
                                ) as r1
                        ) as DrTable,
                        (
                                Select SUM(CrPNLSum) as CrSide
                                From(
                                        Select SUM(F.Amount) as CrPNLSum
                                        From FinancialTransaction as FT JOIN FinancialTransactionEntry
as F ON (FT.TransactionNumber = F.TransactionNumber)
                                        JOIN Account as A ON(F.AccountId=A.AccountId)
                                        JOIN AccountGroup as AG ON (AG.AccountGroupId =
A.AccountGroupId)
                                        JOIN Company as C ON (C.CompanyID = AG.CompanyId)
                                        Where C.CompanyName = 'Myntra' AND AG.ShowIn = 'Profit
and Loss Account' AND AG.Header = False
                                        AND FT.TransactionDate < '01-04-2020'
                                        GROUP BY A.AccountId
                                        UNION
                                        Select 0 as DrPNLSum
                                ) as r2
                        ) as CrTable,
                        (
                                Select (CrSide - DrSide) as GrossProfit
                                From
                                (
                                        Select SUM(DrTradingSum) as DrSide
                                        From(
                                                Select SUM(F.Amount) as DrTradingSum
                                                From FinancialTransaction as FT JOIN
FinancialTransactionEntry as F ON (FT.TransactionNumber = F.TransactionNumber)
                                                JOIN Account as A ON(F.AccountId=A.AccountId)
                                                JOIN AccountGroup as AG ON (AG.AccountGroupId =
A.AccountGroupId)
                                                JOIN Company as C
                                                ON (C.CompanyID = AG.CompanyId)
                                                Where C.CompanyName = 'Myntra' AND AG.ShowIn =
'Trading Account' AND AG.Header = True
                                                AND FT.TransactionDate < '01-04-2020'
                                                GROUP BY A.AccountId
                                                UNION
                                                Select 0 as DrTradingSum
                                        ) as r1
                                ) as DrTable,
                                (
```

```
                                        Select SUM(CrTradingSum) as CrSide
                                        From(
                                                Select SUM(F.Amount) as CrTradingSum
                                                From FinancialTransaction as FT JOIN
FinancialTransactionEntry as F ON (FT.TransactionNumber = F.TransactionNumber)
                                                JOIN Account as A ON(F.AccountId=A.AccountId)
                                                JOIN AccountGroup as AG ON (AG.AccountGroupId =
A.AccountGroupId)
                                                JOIN Company as C ON (C.CompanyID =
AG.CompanyId)
                                                Where C.CompanyName = 'Myntra' AND AG.ShowIn =
'Trading Account' AND AG.Header = False
                                                AND FT.TransactionDate < '01-04-2020'
                                                GROUP BY A.AccountId
                                                UNION
                                                Select 0 as DrTradingSum
                                        ) as r2
                                ) as CrTable
                        ) as GrossProfitTable
        )as r2;
```

	showin accountingsystem.finalaccount	accountname character varying (50)	Amount numeric	header boolean
1	Profit and Loss Account	Carriage Outwards	60000.00	true
2	Balance Sheet	Furniture and Fixtures	101470.00	true
3	Profit and Loss Account	Salary	15000.00	true
4	Profit and Loss Account	Telephone Charges	51000.00	true
5	Balance Sheet	Debtors	23750.00	true
6	Profit and Loss Account	Depreciation	33530.00	true
7	Balance Sheet	Creditors	25000.00	false
8	Balance Sheet	Capital	1771200.00	false
9	Profit and Loss Account	Travelling Expenses	45000.00	true
10	Balance Sheet	Cash	257550.00	true

22. Rank companies based on their Net Profit.
Relational Algebra:

SQL Query:

```
Select (CrSide - DrSide) as "Net Profit",DrTable.CompanyID,DrTable.CompanyName
From(
Select coalesce(SUM(F.Amount),0) as DrSide,C.CompanyID,C.CompanyName
```

From FinancialTransaction as FT JOIN FinancialTransactionEntry as F ON (FT.TransactionNumber = F.TransactionNumber)
JOIN Account as A ON(F.AccountId=A.AccountId)
JOIN AccountGroup as AG ON (AG.AccountGroupId = A.AccountGroupId)
JOIN Company as C ON (C.CompanyID = AG.CompanyId)
Where (AG.ShowIn = 'Profit and Loss Account' OR AG.ShowIn = 'Trading Account') AND AG.Header = True AND FT.TransactionDate BETWEEN '01-04-2019' AND '31-03-2020'
Group by C.CompanyID) as DrTable JOIN
(Select coalesce(SUM(F.Amount),0) as CrSide, C.CompanyID,C.CompanyName
From FinancialTransaction as FT JOIN FinancialTransactionEntry as F ON (FT.TransactionNumber = F.TransactionNumber)
JOIN Account as A ON(F.AccountId=A.AccountId)
JOIN AccountGroup as AG ON (AG.AccountGroupId = A.AccountGroupId)
JOIN Company as C ON (C.CompanyID = AG.CompanyId)
Where (AG.ShowIn = 'Profit and Loss Account' OR AG.ShowIn = 'Trading Account') AND AG.Header = False
AND FT.TransactionDate BETWEEN '01-04-2019' AND '31-03-2020'
Group by C.CompanyID) as CrTable
ON DrTable.CompanyID=CrTable.CompanyID
ORDER BY "Net Profit" Desc;

	Net Profit numeric	companyid [PK] character varying (20)	companyname character varying (50)
1	26200.00	C00001	Myntra

23. Show the complete accounting process from Journal Entry to Balance Sheet.

Relational Algebra:

--Journal Entry

$\mathcal{F}_{(Ft.TransactionNumber,\ Ft.TransactionDate,A.AccountName,ABS(F.Amount)\ ->\ "Transaction\ Amount")}(\sigma_{(C.CompanyName = 'Myntra'\ AND\ Ft.TransactionDate\ BETWEEN\ '04\text{-}01\text{-}2019'\ AND\ '03\text{-}31\text{-}2020')}((((\rho_{ft}(FinancialTransactions)$
$\bowtie_{<ft.AccountID=f.AccountID>}\rho_{f}(FinancialTransactionEntry))\bowtie_{<f.AccountID=A.AccountID>}\rho_{A}(Account))$
$\bowtie_{<A.AccountGroupId=AG.AccountGroupId>}\rho_{AG}(AccountGroup))\ \bowtie_{<AG.CompanyID=C.CompanyID>}\rho_{C}(Company))$

--All Ledger accounts

$\pi_{(FT.TransactionDate,\ A.AccountName,\ F.Amount,\ AG.ShowIn)}(\sigma_{(C.CompanyName = 'Myntra'\ AND\ ((AG.ShowIn = 'Balance\ Sheet'\ AND\ FT.TransactionDate <= '31\text{-}03\text{-}2020')\ OR\ (AG.ShowIn <> 'Balance\ Sheet'\ AND\ FT.TransactionDate\ BETWEEN\ '01\text{-}04\text{-}2019'\ AND\ '31\text{-}03\text{-}2020')))}((((\rho_{ft}(FinancialTransactions)$
$\bowtie_{<ft.AccountID=f.AccountID>}\rho_{f}(FinancialTransactionEntry))\bowtie_{<f.AccountID=A.AccountID>}\rho_{A}(Account))$
$\bowtie_{<A.AccountGroupId=AG.AccountGroupId>}\rho_{AG}(AccountGroup))\ \bowtie_{<AG.CompanyID=C.CompanyID>}\rho_{C}(Company))$

SQL Query:

--Journal Entry

Select Ft.TransactionNumber, Ft.TransactionDate,A.AccountName,ABS(F.Amount) as "Transaction Amount"
From FinancialTransaction as FT JOIN FinancialTransactionEntry as F ON (FT.TransactionNumber = F.TransactionNumber)

```
JOIN Account as A ON (F.accountId = A.accountId)
JOIN AccountGroup as AG ON (AG.AccountGroupId = A.AccountGroupId)
JOIN Company as C ON (C.CompanyId = AG.CompanyId)
Where C.CompanyName = 'Myntra' AND Ft.TransactionDate BETWEEN '01-04-2019' AND '31-03-
2020'
Order By Ft.TransactionNumber;

--Accounts
Select FT.TransactionDate, A.AccountName, F.Amount, AG.ShowIn, AG.Name
From FinancialTransaction as FT JOIN FinancialTransactionEntry as F ON (FT.TransactionNumber =
F.TransactionNumber)
JOIN Account as A ON(F.AccountId=A.AccountId)
JOIN AccountGroup as AG ON (AG.AccountGroupId = A.AccountGroupId)
JOIN Company as C ON (C.CompanyID = AG.CompanyId)
Where  C.CompanyName = 'Myntra' AND ((AG.ShowIn = 'Balance Sheet' AND FT.TransactionDate
<= '31-03-2020') OR (AG.ShowIn <> 'Balance Sheet' AND FT.TransactionDate BETWEEN '01-04-
2019' AND '31-03-2020'))
Order By AG.Name, A.AccountName;

--Final Statements
Select AG.ShowIn, A.AccountName, SUM(F.Amount) as "Amount", AG.Header
From FinancialTransaction as FT JOIN FinancialTransactionEntry as F ON (FT.TransactionNumber =
F.TransactionNumber)
JOIN Account as A ON(F.AccountId=A.AccountId)
JOIN AccountGroup as AG ON (AG.AccountGroupId = A.AccountGroupId)
JOIN Company as C ON (C.CompanyID = AG.CompanyId)
Where C.CompanyName = 'Myntra' AND AG.Name <> 'Capital'
Group By AG.ShowIn, AG.Header, A.AccountName
UNION
--To add all previuous year profit or loss to captial account
Select r1.ShowIn, r1.AccountName, ("Amount" +"Net Profit"), r1.Header
FROM (
        Select AG.ShowIn, A.AccountName, SUM(F.Amount) as "Amount", AG.Header
        From FinancialTransaction as FT JOIN FinancialTransactionEntry as F ON
(FT.TransactionNumber = F.TransactionNumber)
        JOIN Account as A ON(F.AccountId=A.AccountId)
        JOIN AccountGroup as AG ON (AG.AccountGroupId = A.AccountGroupId)
        JOIN Company as C ON (C.CompanyID = AG.CompanyId)
        Where C.CompanyName = 'Myntra' AND AG.Name = 'Capital'
        Group By AG.ShowIn, AG.Header, A.AccountName
        ) as r1
        ,(
                Select (CrSide + GrossProfit - DrSide) as "Net Profit"
                From
                        (
                                Select SUM(DrPNLSum) as DrSide
                                From(
                                        Select SUM(F.Amount) as DrPNLSum
                                        From FinancialTransaction as FT JOIN FinancialTransactionEntry
```

```
as F ON (FT.TransactionNumber = F.TransactionNumber)
                                        JOIN Account as A ON(F.AccountId=A.AccountId)
                                        JOIN AccountGroup as AG ON (AG.AccountGroupId =
A.AccountGroupId)
                                        JOIN Company as C ON (C.CompanyID = AG.CompanyId)
                                        Where C.CompanyName = 'Myntra' AND AG.ShowIn = 'Profit
and Loss Account' AND AG.Header = True
                                        AND FT.TransactionDate < '01-04-2020'
                                        GROUP BY A.AccountId
                                        UNION
                                        Select 0 as DrPNLSum
                                ) as r1
                        ) as DrTable,
                        (
                                Select SUM(CrPNLSum) as CrSide
                                From(
                                        Select SUM(F.Amount) as CrPNLSum
                                        From FinancialTransaction as FT JOIN FinancialTransactionEntry
as F ON (FT.TransactionNumber = F.TransactionNumber)
                                        JOIN Account as A ON(F.AccountId=A.AccountId)
                                        JOIN AccountGroup as AG ON (AG.AccountGroupId =
A.AccountGroupId)
                                        JOIN Company as C ON (C.CompanyID = AG.CompanyId)
                                        Where C.CompanyName = 'Myntra' AND AG.ShowIn = 'Profit
and Loss Account' AND AG.Header = False
                                        AND FT.TransactionDate < '01-04-2020'
                                        GROUP BY A.AccountId
                                        UNION
                                        Select 0 as DrPNLSum
                                ) as r2
                        ) as CrTable,
                        (
                                Select (CrSide - DrSide) as GrossProfit
                                From
                                (
                                        Select SUM(DrTradingSum) as DrSide
                                        From(
                                                Select SUM(F.Amount) as DrTradingSum
                                                From FinancialTransaction as FT JOIN
FinancialTransactionEntry as F ON (FT.TransactionNumber = F.TransactionNumber)
                                                JOIN Account as A ON(F.AccountId=A.AccountId)
                                                JOIN AccountGroup as AG ON (AG.AccountGroupId =
A.AccountGroupId)
                                                JOIN Company as C
                                                ON (C.CompanyID = AG.CompanyId)
                                                Where C.CompanyName = 'Myntra' AND AG.ShowIn =
'Trading Account' AND AG.Header = True
                                                AND FT.TransactionDate < '01-04-2020'
                                                GROUP BY A.AccountId
```

```
                                        UNION
                                        Select 0 as DrTradingSum
                                ) as r1
                        ) as DrTable,
                        (
                                Select SUM(CrTradingSum) as CrSide
                                From(
                                        Select SUM(F.Amount) as CrTradingSum
                                        From FinancialTransaction as FT JOIN
FinancialTransactionEntry as F ON (FT.TransactionNumber = F.TransactionNumber)
                                        JOIN Account as A ON(F.AccountId=A.AccountId)
                                        JOIN AccountGroup as AG ON (AG.AccountGroupId =
A.AccountGroupId)
                                        JOIN Company as C ON (C.CompanyID =
AG.CompanyId)
                                        Where C.CompanyName = 'Myntra' AND AG.ShowIn =
'Trading Account' AND AG.Header = False
                                        AND FT.TransactionDate < '01-04-2020'
                                        GROUP BY A.AccountId
                                        UNION
                                        Select 0 as DrTradingSum
                                ) as r2
                        ) as CrTable
                ) as GrossProfitTable
        )as r2;
```

	showin accountingsystem.finalaccount	accountname character varying (50)	Amount numeric	header boolean
1	Profit and Loss Account	Carriage Outwards	60000.00	true
2	Balance Sheet	Furniture and Fixtures	101470.00	true
3	Profit and Loss Account	Salary	15000.00	true
4	Profit and Loss Account	Telephone Charges	51000.00	true
5	Balance Sheet	Debtors	23750.00	true
6	Profit and Loss Account	Depreciation	33530.00	true
7	Balance Sheet	Creditors	25000.00	false
8	Balance Sheet	Capital	1771200.00	false
9	Profit and Loss Account	Travelling Expenses	45000.00	true
10	Balance Sheet	Cash	257550.00	true

24. List all the item groups which doesn't belong to any other item group.

Relational Algebra:

$R1 \leftarrow \prod_{(parentItemGroupID)}(\sigma_{(parentItemGroupID\ is\ not\ null)}(itemGroup))$

$\prod_{(i.itemgroupId,i.category,i.companyId,i.parentItemGroupId)}(\sigma_{(i.itemGroupId\ not\ in\ R1)}(\rho_i(itemGroup)))$

SQL Query:

```
Select *
From ItemGroup as I
Where I.ItemGroupID NOT IN(
                Select ParentItemGroupID
                From ItemGroup
                Where ParentItemGroupID IS NOT NULL
        );
```

	itemgroupid [PK] character varying (20)	category character varying (100)	companyid character varying (20)	parentitemgroupid character varying (20)
1	F01	Fashions	C00001	[null]
2	W01	Women's wear	C00001	FS01
3	CW01	Children's wear	C00001	FS01
4	J01	Jeans	C00001	M01
5	S01	Suits	C00001	M01
6	WF01	Women's Footwear	C00001	FS01
7	MF01	Men's Footwear	C00001	FS01
8	CF01	Children's footwear wear	C00001	FS01
9	FG01	Fitness gadgets	C00001	GD01
10	SW01	Smart Wearables	C00001	GD01
11	FA01	Fitness Apparels	C00001	FS01
12	MB01	Mobiles	C00002	EC01
13	LP01	Laptops	C00002	EC01
14	TV01	TVs	C00002	EC01
15	MK01	MotorBikes	C00002	VH01

25. List Most Sold Items.

Relational Algebra:

$R1 \leftarrow {}_{(itemId)}F_{(SUM(qty)\ as\ total\_qty)}(\sigma_{(companyId='C00001')}\ salesItem)$

$Total\_Qty \leftarrow F_{(SUM(qty))}(salesItem)$

$R2 \leftarrow \prod_{(salesItem.itemId)}(\sigma_{(Total\_Qty\ =\ R1)}(salesItem))$

$\prod_{(itemId,itemName,barcode,price,itemGroupId)}(\sigma_{(item.itemId\ in\ R2)}(item))$

SQL Query:

```
Select *
From item
```

```
where item.itemid in (
        Select salesitem.itemid
        From salesitem
        Group By salesitem.itemid
        having sum(qty)=(
                Select sum(qty) as total_qty
                From salesitem
                where companyid='C00001'
                Group By itemid
                Order by total_qty desc limit 1));
```

	itemid [PK] character varying (20)	itemname character varying (100)	barcode character varying (40)	price numeric (15,2)	itemgroupid character varying (20)
1	I00012	Exercise Machines	BA1245495059	75000.00	FG01

26. Items not sold at all for a given amount of time. Example, items not sold in the last 6 months.
Relational Algebra:

$R1 \leftarrow \prod_{(salesItem.itemId)}(\sigma_{(salesInvoice.date > CURRENT\_DATE-180)}(salesInvoice \bowtie_{(salesItem.invoiceId = salesInvoice.invoiceId)} salesItem))$

$\prod_{(item.*)}(\sigma_{(itemId\ not\ in\ r1)}(item))$

SQL Query:

```
Select *
From item
where itemid NOT IN(
        Select salesitem.itemid
        From salesinvoice
        inner join salesitem on (salesitem.invoiceid=salesinvoice.invoiceid)
        where(salesinvoice.date>CURRENT_DATE-180));
```

	itemid [PK] character varying (20)	itemname character varying (100)	barcode character varying (40)	price numeric (15,2)	itemgroupid character varying (20)
1	I00001	HRX	BA1245495050	20500.00	TW01
2	I00002	Roadster	BA1245495050	5500.00	TW01
3	I00003	HRX Cyan	BA1245495050	5500.00	TW01
4	I00004	Women Beige Solid Felted Sh...	BA1245495053	3000.00	W01
5	I00005	Women Peach-Coloured Solid ...	BA1245495054	3000.00	W01
6	I00007	Sui Peach-Coloured Solid Ligh...	BA1245495054	2000.00	S01
7	I00008	Men Pink & Blue Slim Fit Strip...	BA1245495055	800.00	FS01
8	I00009	Blue & Green Two-Toned Slim ...	BA1245495057	800.00	FS01
9	I00011	Men Grey Casual Fit West-Ris...	BA1245495058	7100.00	J01
10	I00015	Exercise Cycles	BA1245495062	8000.00	FG01
11	I00017	Samsung Galaxy M51	SM4577823560	25000.00	MB01
12	I00019	DELL Inspiron 5370	DL2039407402	70000.00	LP01
13	I00022	Honda Shine 150	HO7821928190	65000.00	MK01
14	I00023	Maruti Suzuki Swift	MS2172726270	700000.00	CR01

27. Percentage of each item sold.
Relational Algebra:
Total_Amount <- $F_{SUM(amount * qty)}$(salesItem)

$\prod_{(salesItem.itemId,itemName)} F_{SUM(amount * qty*100/Total\_Amount) \text{ as } percentage\_sales}(\sigma(salesItem \bowtie_{(salesItem.itemId = item.itemId)} item))$

SQL Query:
Select salesitem.itemid,itemname,sum(amount*qty)*100/(Select sum(amount*qty) From salesitem) as percentage_sales
From salesitem inner join item on (salesitem.itemid=item.itemid) Group By salesitem.itemid,itemname
Order by "percentage_sales" DESC;

	itemid character varying (20)	itemname character varying (100)	percentage_sales double precision
1	I00012	Exercise Machines	68.62734059203045
2	I00013	Fitness Apparels	22.519225789017373
3	I00021	HP Pavilion	2.251922578901737
4	I00014	Treadmills	2.1393264499566507
5	I00015	Exercise Cycles	1.2194160764752908
6	I00016	One Plus Nord	1.1259612894508686
7	I00020	ASUS TUF	0.900769031560695
8	I00018	Vivo v20	0.4503845157803475
9	I00010	Men Grey Skinny Fit Low-Rise ...	0.4503845157803475
10	I00006	Suit Felted Shacket	0.31526916104624325