

Managing Data Structures

Final Project Team Project

QSN Retail (E-commerce)

Professor Helen Yelluas

Project Team:
Quinton Lopez
Sung Keum
Nikita Sarda
David Smarsh

Agenda

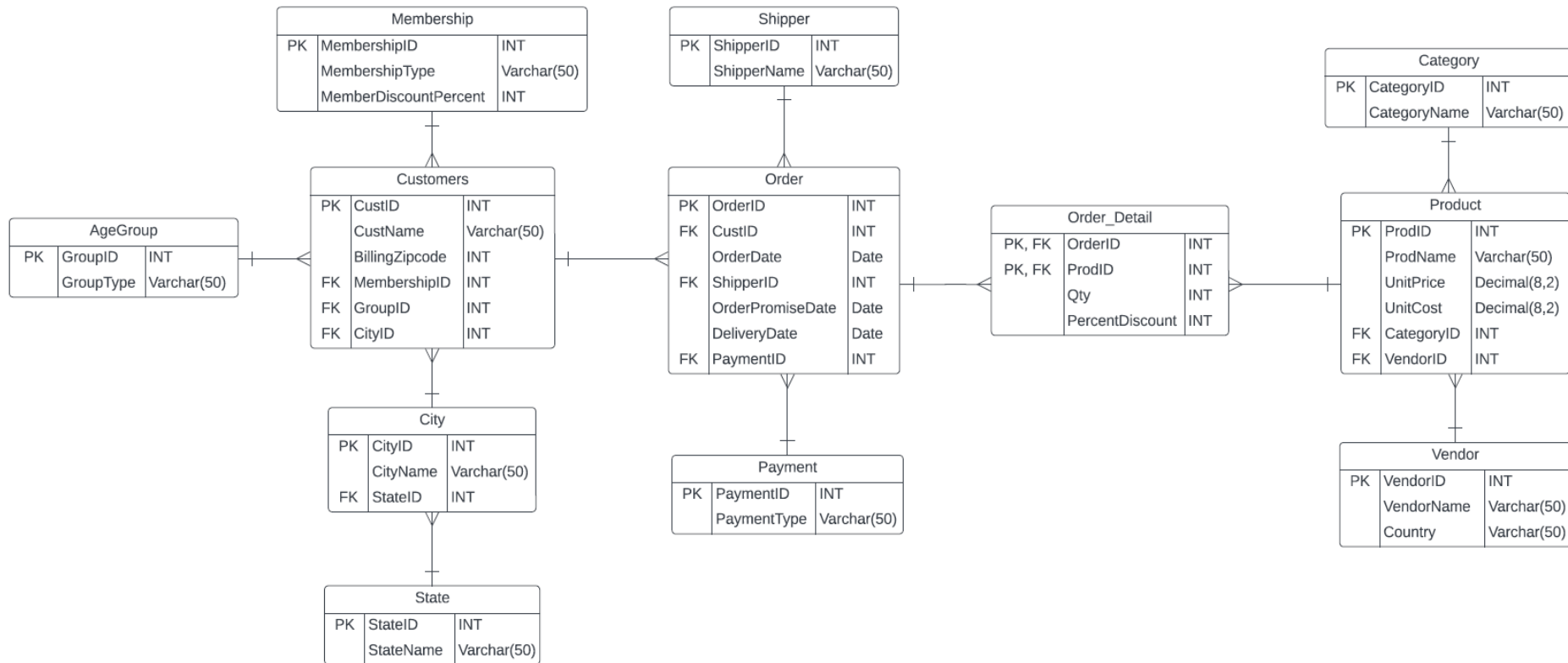
- QSNRetail Description
- Entity Relationship Diagram (ERD)
- 18 Analytical Questions
- Each Analytical Question Details
 - Table Data
 - SQL Query
 - Graph

QSNRetail Description

QSN Retail Inc., is an e-commerce company that sells variety of products to everyday shoppers. Think of Amazon.com but we do it better and faster. Our goal is to provide an exceptional service and high-quality products to our customers to meet their everyday needs and wants. We encourage customers to return to us for their future purchases through our generous reward system. Our products are supplied by the finest vendors from all over world, whom we handpicked through rigorous process, and routinely undergo quality assurance testing. Our company offers many different payment options for our customers and utilizes third-party logistics companies to facilitate transactions. We are QSNRetail!

QSN Retail Entity Relationship Diagram (ERD)

QSN Retail Inc.



Analytical Questions

QSN Retail

(1 of 2)

18 Analytical Questions (1 – 9)

- 1. Total sales by year by age group**
- 2. Total sales by month and year**
- 3. Total sales by year by membership type**
- 4. Number of late deliveries for each shipper by year**
- 5. Total sales by year by different payment type**
- 6. Number of deliveries by each shipper by year**
- 7. Total profit by month and year**
- 8. Total sales by category by year**
- 9. Total profit by category by year**

Analytical Questions

QSN Retail

(2 of 2)

Analytical Questions (10 – 18)

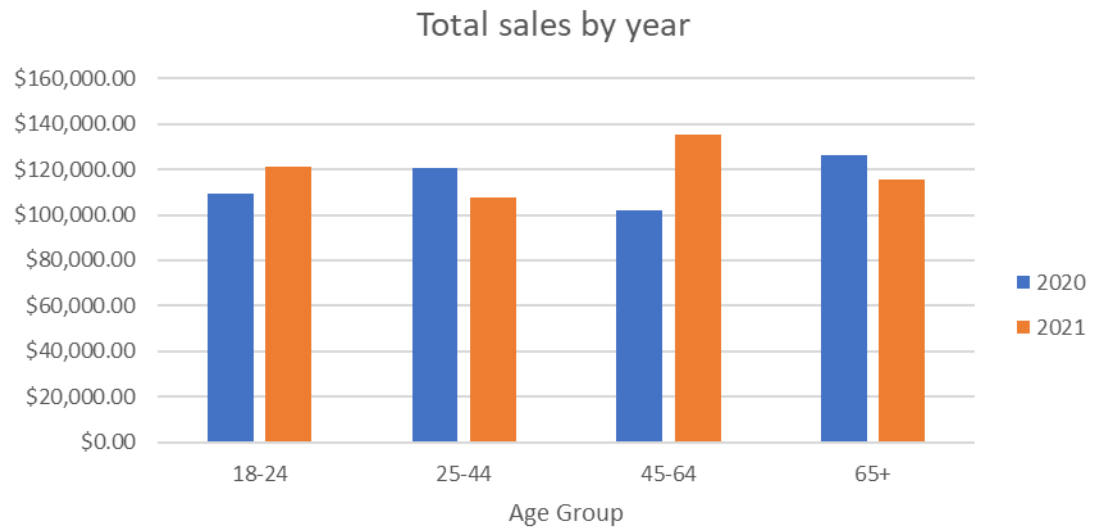
- 10. Average number of products per basket by month and year**
- 11. Average ticket per basket by month and year**
- 12. Total discounts received by our customers by month and year**
- 13. Number of products sold by year**
- 14. Number of products sold by category by year**
- 15. Payment to each supplier by year**
- 16. Total number of products provided by each supplier by year**
- 17. Total payment to suppliers by country by year**
- 18. Total discounts received by membership type by year**

Analytical Question 1

Total sales by year by age group

Data Graph and SQL Query

AgeGroup	Year	Total_sale
18-24	2020	109076.01
18-24	2021	121459.06
25-44	2020	120507.56
25-44	2021	107568.51
45-64	2020	101967.55
45-64	2021	135200.37
65+	2020	126496.39
65+	2021	115623.82



```

SELECT AgeGroup,
       year(OrderDate) as Year,
       SUM(ROUND((Quantity * (UnitPrice * (( 100 - PercentDiscount ) / 100))) * ((100 - MemberDiscountPercent) / 100), 2)) as Total_sale

FROM denormalized
GROUP BY AgeGroup, Year
    
```

Analytical Question 2

Total sales by month and year

Data Graph and SQL Query

Month	Year	Total_sale
1	2020	\$23,256.74
1	2021	\$52,902.80
2	2020	\$37,698.08
2	2021	\$37,372.53
3	2020	\$29,194.48
3	2021	\$46,809.60
4	2020	\$47,863.24
4	2021	\$33,409.97
5	2020	\$38,483.62
5	2021	\$50,765.88
6	2020	\$40,500.17
6	2021	\$30,130.27
7	2020	\$32,674.07
7	2021	\$46,989.64
8	2020	\$43,206.05
8	2021	\$27,069.61
9	2020	\$34,826.30
9	2021	\$42,227.26
10	2020	\$43,135.49
10	2021	\$27,041.11
11	2020	\$62,057.86
11	2021	\$43,937.86
12	2020	\$25,150.87
12	2021	\$41,194.84



```

SELECT
    Month(OrderDate) as Month,
    year(OrderDate) as Year,
    CONCAT("$", FORMAT(SUM((Quantity * (UnitPrice * (( 100 - PercentDiscount ) / 100))) * ((100 - MemberDiscountPercent) / 100)), 2)) as Total_sale

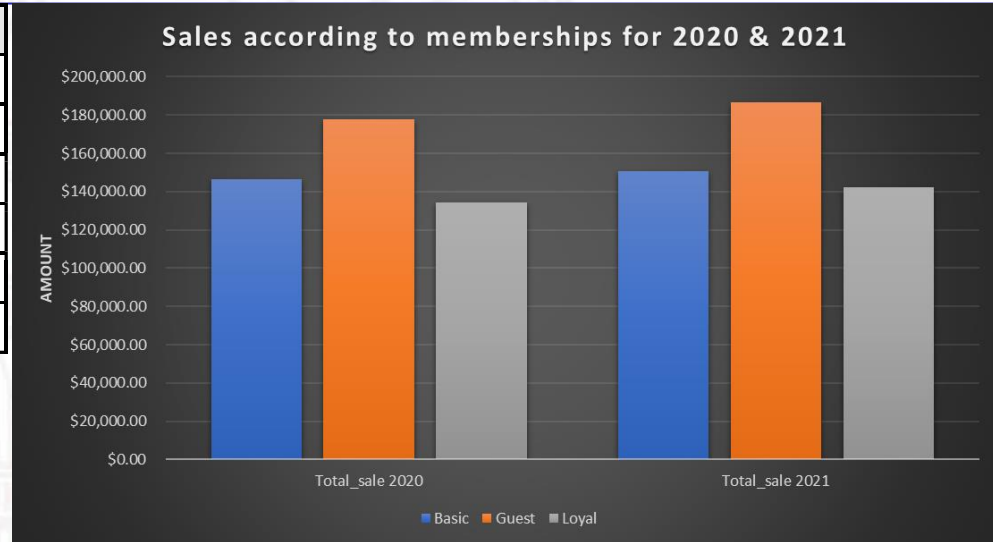
FROM denormalized
GROUP BY Month, Year
    
```


Analytical Question 3

Total sales by year by membership type

Data Graph and SQL Query

Year	MembershipType	Total_sale
2020	Basic	\$146,425.28
2021	Basic	\$150,826.86
2020	Guest	\$177,532.58
2021	Guest	\$186,815.71
2020	Loyal	\$134,089.14
2021	Loyal	\$142,208.82



```

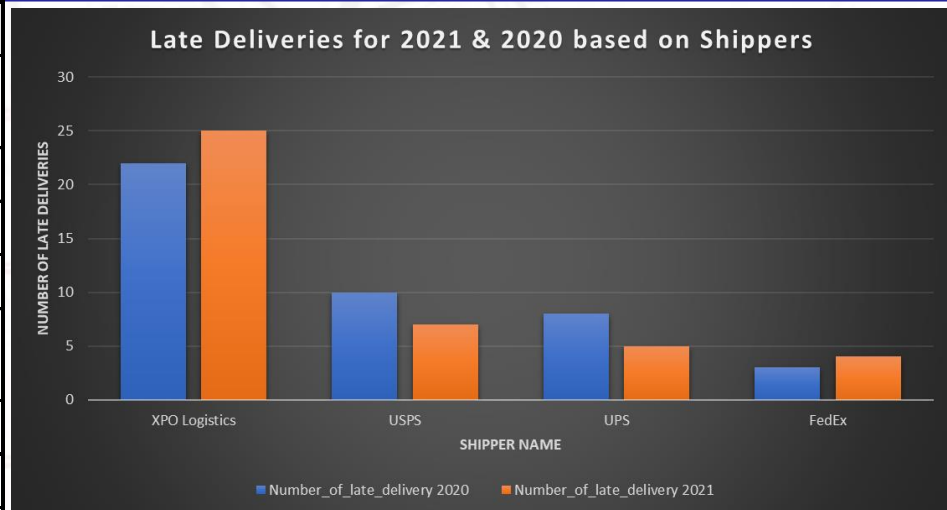
SELECT
    year(OrderDate) as Year,
    MembershipType,
    CONCAT("$", FORMAT(SUM((Quantity * (UnitPrice * (( 100 - PercentDiscount ) / 100))) * ((100 - MemberDiscountPercent) / 100)), 2)) as Total_sale

FROM denormalized
GROUP BY Year, MembershipType
ORDER BY MembershipType, Year
    
```

Analytical Question 4

Number of late deliveries for each shipper by year Data Graph and SQL Query

Year	ShipperName	Number_of_late_delivery
2020	XPO Logistics	22
2020	USPS	10
2020	UPS	8
2020	FedEx	3
2021	XPO Logistics	25
2021	USPS	7
2021	UPS	5
2021	FedEx	4



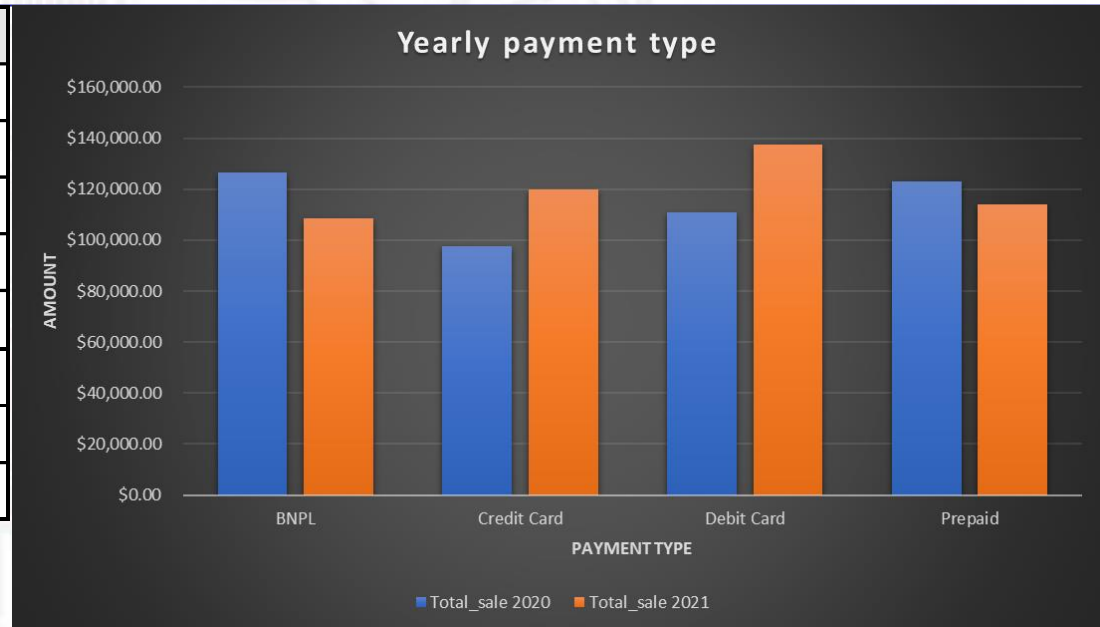
```
SELECT Year(OrderDate) as Year,
       ShipperName,
       Count(Distinct OrderID) as Number_of_late_delivery
FROM denormalized Where
       DeliveryDate > OrderPromiseDate
Group by ShipperName, Year
Order by Year, Number_of_late_delivery DESC
```

Analytical Question 5

Total sales by year by different payment type

Data Graph and SQL Query

Year	PaymentType	Total_sale
2020	BNPL	\$126,513.71
2020	Credit Card	\$97,625.59
2020	Debit Card	\$110,906.36
2020	Prepaid	\$123,001.34
2021	BNPL	\$108,382.86
2021	Credit Card	\$120,031.17
2021	Debit Card	\$137,298.51
2021	Prepaid	\$114,138.84



```

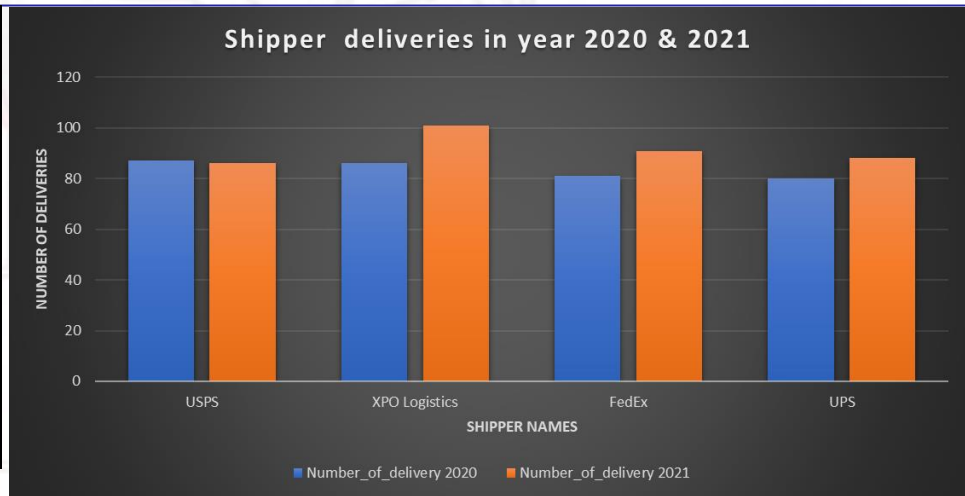
SELECT year(OrderDate) as Year, PaymentType,
       CONCAT("$", FORMAT(SUM((Quantity * (UnitPrice * ((100 - PercentDiscount) / 100))) * ((100 -
MemberDiscountPercent) / 100)), 2)) as Total_sale
FROM denormalized
GROUP BY Year, PaymentType
    
```

Analytical Question 6

Number of deliveries by each shipper by year

Data Graph and SQL Query

Year	ShipperName	Number_of_delivery
2020	USPS	87
2020	XPO Logistics	86
2020	FedEx	81
2020	UPS	80
2021	XPO Logistics	101
2021	FedEx	91
2021	UPS	88
2021	USPS	86



```
SELECT Year(OrderDate) as Year, ShipperName,  
       Count(Distinct OrderID) as Number_of_delivery  
FROM denormalized Group by ShipperName, Year  
Order by Year, Number_of_delivery DESC
```


Analytical Question 7

Total profit by month and year

Data Graph and SQL Query

Year	Month	Total_profit	Year	Month	Total profit
2020	1	\$1,872.79	2021	1	\$4,178.82
2020	2	\$3,176.16	2021	2	\$2,890.57
2020	3	\$2,744.56	2021	3	\$3,496.61
2020	4	\$4,314.33	2021	4	\$2,084.99
2020	5	\$3,005.65	2021	5	\$4,811.94
2020	6	\$2,781.22	2021	6	\$2,325.30
2020	7	\$2,619.11	2021	7	\$3,584.66
2020	8	\$3,499.05	2021	8	\$2,545.62
2020	9	\$3,199.31	2021	9	\$3,260.33
2020	10	\$3,522.50	2021	10	\$2,398.16
2020	11	\$4,381.93	2021	11	\$3,297.89
2020	12	\$2,090.90	2021	12	\$3,084.87



```

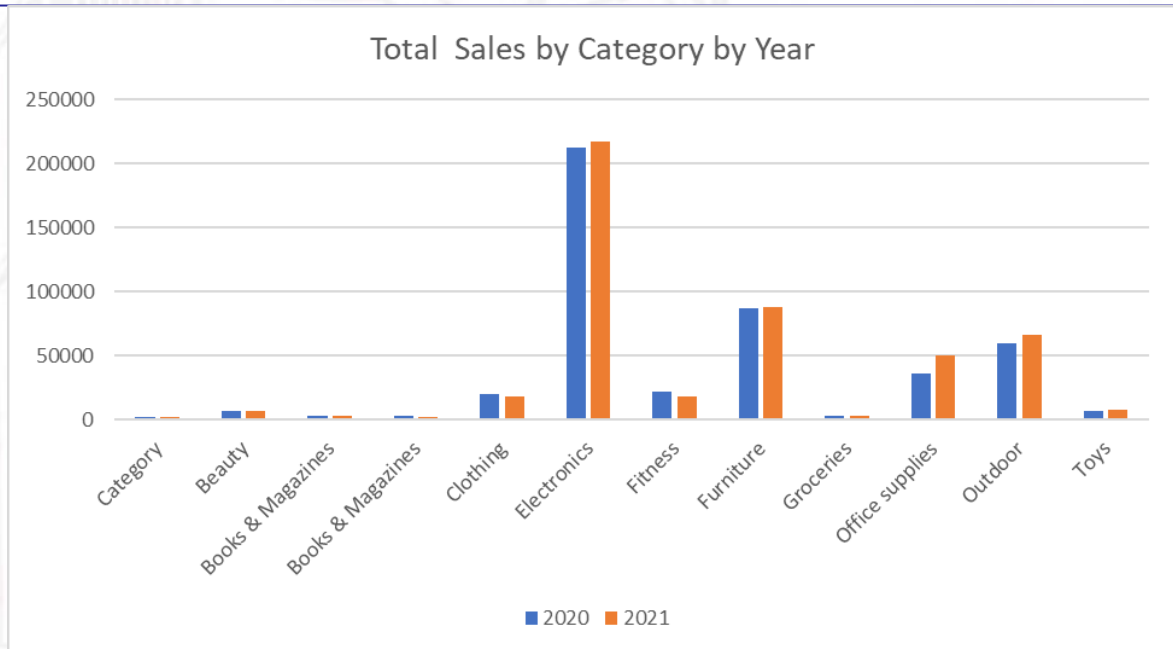
SELECT
    year(OrderDate) as Year,
    MONTH(OrderDate) as Month,
    CONCAT("$", FORMAT(SUM(ROUND((Quantity * (UnitPrice * (( 100 - PercentDiscount ) / 100))) * ((100 -
MemberDiscountPercent) / 100), 2)) - SUM(ROUND((Quantity * UnitCost))), 2)) as Total_profit
FROM denormalized GROUP BY Year, Month
    
```

Analytical Question 8

Total sales by category by year

Data Graph and SQL Query

Category	Year	Total_sale
Beauty	2020	\$6,656.35
Beauty	2021	\$6,935.46
Books & Magazines	2020	\$3,562.36
Books & Magazines	2021	\$2,860.31
Clothing	2020	\$20,093.36
Clothing	2021	\$17,867.59
Electronics	2020	\$212,773.48
Electronics	2021	\$217,475.09
Fitness	2020	\$21,915.81
Fitness	2021	\$18,624.46
Furniture	2020	\$87,358.76
Furniture	2021	\$87,842.52
Groceries	2020	\$2,771.05
Groceries	2021	\$2,808.70
Office supplies	2020	\$36,432.01
Office supplies	2021	\$50,811.00
Outdoor	2020	\$59,387.07
Outdoor	2021	\$66,843.00
Toys	2020	\$7,096.75
Toys	2021	\$7,783.24



```
SELECT Category,
       year(OrderDate) as Year,
       CONCAT("$", FORMAT(SUM((Quantity * (UnitPrice * ((100 - PercentDiscount) / 100))) * ((100 - MemberDiscountPercent) / 100)), 2)) as Total_sale
```

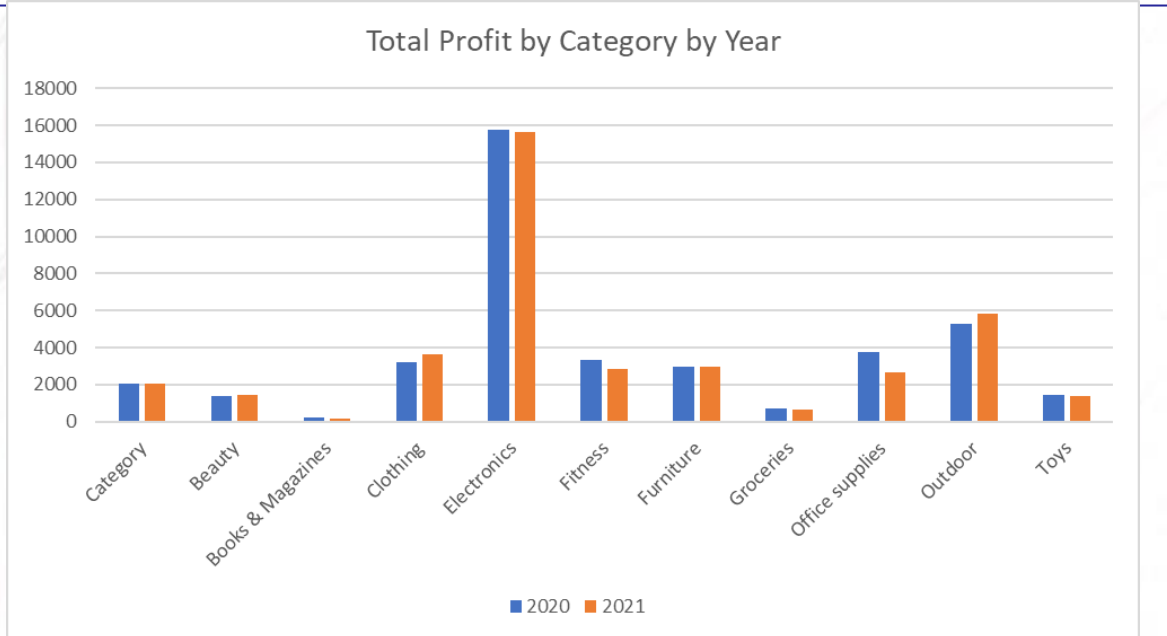
From denormalized
Group by Category, Year

Analytical Question 9

Total profit by category by year

Data Graph and SQL Query

Category	Year	Total_profit
Beauty	2020	\$1,372.37
Beauty	2021	\$1,466.51
Books & Magazines	2020	\$199.33
Books & Magazines	2021	\$159.25
Clothing	2021	\$3,192.59
Clothing	2020	\$3,653.46
Electronics	2020	\$15,773.53
Electronics	2021	\$15,625.15
Fitness	2020	\$3,310.96
Fitness	2021	\$2,829.51
Furniture	2021	\$2,942.57
Furniture	2020	\$2,938.81
Groceries	2021	\$679.71
Groceries	2020	\$665.04
Office supplies	2021	\$3,753.08
Office supplies	2020	\$2,645.10
Outdoor	2020	\$5,297.13
Outdoor	2021	\$5,843.17
Toys	2021	\$1,468.22
Toys	2020	\$1,351.78



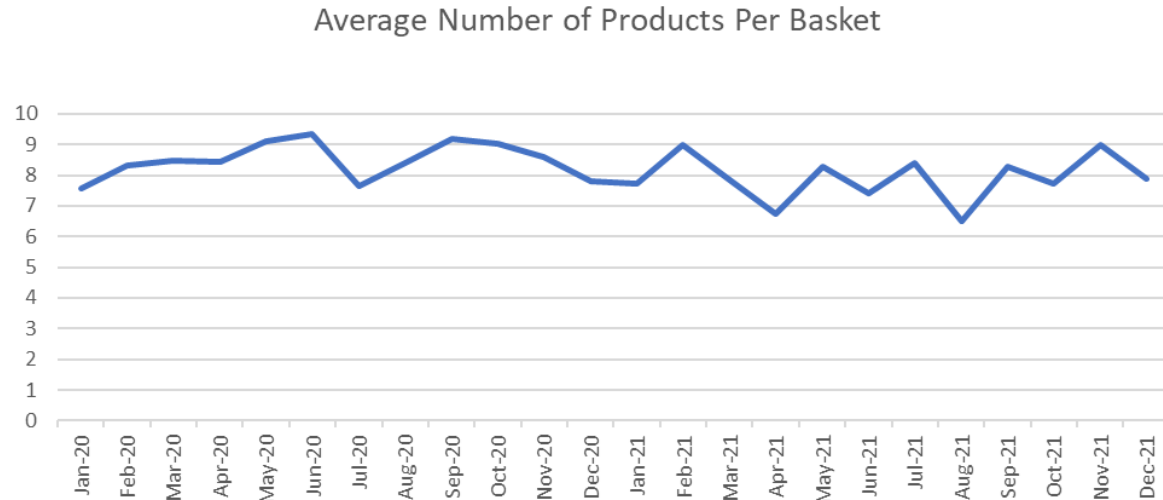
```

SELECT
    Category,
    year(OrderDate) as Year,
    CONCAT("$", FORMAT(SUM(ROUND((Quantity * (UnitPrice * ((100 - PercentDiscount) / 100))) * ((100 -
MemberDiscountPercent) / 100), 2)) -
    SUM(ROUND((Quantity * UnitCost))), 2)) as Total_profit
FROM denormalized
GROUP BY Year, Category
ORDER BY Category
    
```

Analytical Question 10

Average number of products per basket by month and year - Data Graph and SQL Query

Year	Month	avg_no_products_per_basket
2020	1	7.5714
2020	2	8.3000
2020	3	8.4815
2020	4	8.4324
2020	5	9.1250
2020	6	9.3548
2020	7	7.6400
2020	8	8.3939
2020	9	9.2000
2020	10	9.0400
2020	11	8.5882
2020	12	7.8148
2021	1	7.7297
2021	2	9.0000
2021	3	7.8333
2021	4	6.7308
2021	5	8.2750
2021	6	7.4286
2021	7	8.3939
2021	8	6.5000
2021	9	8.2632
2021	10	7.7273
2021	11	9.0000
2021	12	7.8857



```

SELECT Year, Month,
        AVG(number_of_products) as avg_no_products_per_basket
FROM
    (
        SELECT o.OrderID, SUM(QTY) as number_of_products,
               MONTH(OrderDate) as Month,
               Year(OrderDate) as Year
        FROM
            Order_Detail od,
            `Order` o
        WHERE
            o.OrderID = od.OrderID
        GROUP BY o.OrderID, Year, Month) as a GROUP BY Year, Month
    
```

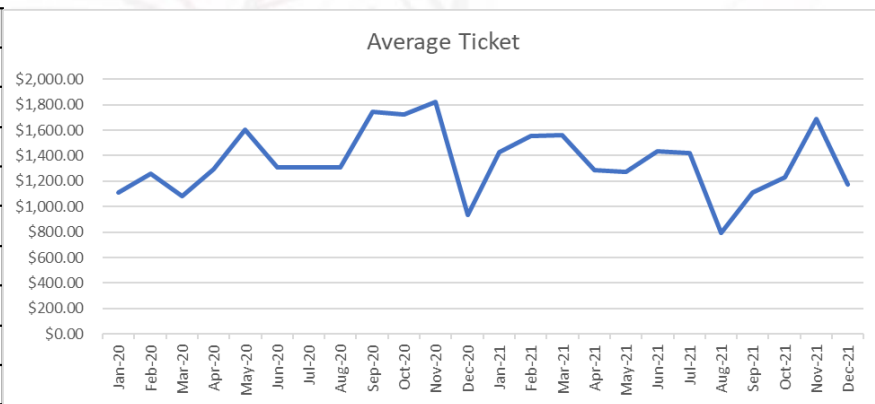

Analytical Question 11

Average ticket per basket by month and year

Data Graph and SQL Query

Month	Year	avg_ticket
1	2020	\$1,107.46
1	2021	\$1,429.81
2	2020	\$1,256.60
2	2021	\$1,557.19
3	2020	\$1,081.28
3	2021	\$1,560.32
4	2020	\$1,293.60
4	2021	\$1,285.00
5	2020	\$1,603.48
5	2021	\$1,269.15
6	2020	\$1,306.46
6	2021	\$1,434.77
7	2020	\$1,306.96
7	2021	\$1,423.93

Month	Year	avg_ticket
8	2020	\$1,309.27
8	2021	\$796.17
9	2020	\$1,741.31
9	2021	\$1,111.24
10	2020	\$1,725.42
10	2021	\$1,229.14
11	2020	\$1,825.23
11	2021	\$1,689.92
12	2020	\$931.51
12	2021	\$1,177.0



```

SELECT
    Month(OrderDate) as Month,
    year(OrderDate) as Year,
    CONCAT("$", FORMAT(SUM((Quantity * (UnitPrice * (( 100 - PercentDiscount ) / 100))) * ((100 -
MemberDiscountPercent) / 100)) / COUNT(DISTINCT OrderID), 2)) as avg_ticket

FROM denormalized
GROUP BY Month, Year
    
```

Analytical Question 12

Total discounts received by our customers by month and year - Data Graph and SQL Query

Month	Year	Total_discount_received
1	2020	\$605.59
1	2021	\$1,417.82
2	2020	\$787.86
2	2021	\$868.34
3	2020	\$702.82
3	2021	\$1,068.35
4	2020	\$1,093.68
4	2021	\$932.06
5	2020	\$1,074.61
5	2021	\$1,212.50
6	2020	\$1,307.51
6	2021	\$864.66
7	2020	\$924.30
7	2021	\$1,736.55
8	2020	\$1,261.39
8	2021	\$819.29
9	2020	\$763.48
9	2021	\$1,443.16
10	2020	\$1,067.36
10	2021	\$730.06
11	2020	\$1,514.50
11	2021	\$1,172.68
12	2020	\$627.56
12	2021	\$1,533.78



```

SELECT
    Month(Order Date) as Month,
    year(OrderDate) as Year,
    CONCAT("$", FORMAT(SUM(((UnitPrice * (PercentDiscount / 100)) + ((UnitPrice * (PercentDiscount / 100)) *
    (MemberDiscountPercent / 100))), 2)) as Total_discount_received

FROM denormalized
GROUP BY Month, Year
    
```

Analytical Question 13

Number of products sold by year

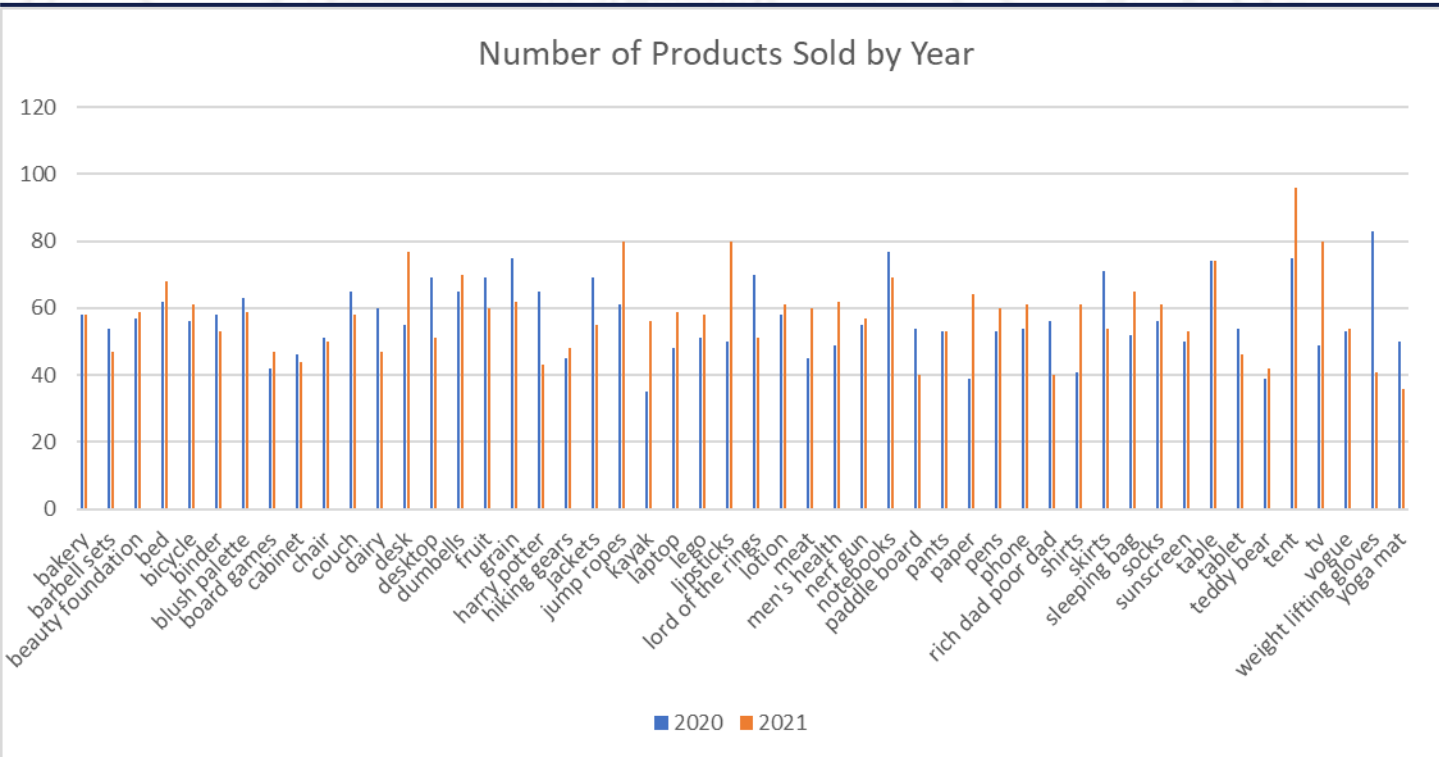
Data Graph and SQL Query (1 of 2)

Year	Product	Num_of_prod_sold	Year	Product	Num_of_prod_sold
2020	bakery	58	2020	lord of the rings	70
2021	bakery	58	2021	lord of the rings	51
2020	barbell sets	54	2020	lotion	58
2021	barbell sets	47	2021	lotion	61
2020	beauty foundation	57	2020	meat	45
2021	beauty foundation	59	2021	meat	60
2020	bed	62	2020	men's health	49
2021	bed	68	2021	men's health	62
2020	bicycle	56	2020	nerf gun	55
2021	bicycle	61	2021	nerf gun	57
2020	binder	58	2020	notebooks	77
2021	binder	53	2021	notebooks	69
2020	blush palette	63	2020	paddle board	54
2021	blush palette	59	2021	paddle board	40
2020	board games	42	2020	pants	53
2021	board games	47	2021	pants	53
2020	cabinet	46	2020	paper	39
2021	cabinet	44	2021	paper	64
2020	chair	51	2020	pens	53
2021	chair	50	2021	pens	60
2020	couch	65	2020	phone	54
2021	couch	58	2021	phone	61
2020	dairy	60	2020	rich dad poor dad	56
2021	dairy	47	2021	rich dad poor dad	40
2020	desk	55	2020	shirts	41
2021	desk	77	2021	shirts	61
2020	desktop	69	2020	skirts	71
2021	desktop	51	2021	skirts	54
2020	dumbbells	65	2020	sleeping bag	52
2021	dumbbells	70	2021	sleeping bag	65
2020	fruit	69	2020	socks	56
2021	fruit	60	2021	socks	61
2020	grain	75	2020	sunscreen	50
2021	grain	62	2021	sunscreen	53
2020	harry potter	65	2020	table	74
2021	harry potter	43	2021	table	74
2020	hiking gears	45	2020	tablet	54
2021	hiking gears	48	2021	tablet	46
2020	jackets	69	2020	teddy bear	39
2021	jackets	55	2021	teddy bear	42
2020	jump ropes	61	2020	tent	75
2021	jump ropes	80	2021	tent	96
2020	kayak	35	2020	tv	49
2021	kayak	56	2021	tv	80
2020	laptop	48	2020	vogue	53
2021	laptop	59	2021	vogue	54
2020	lego	51	2020	weight lifting gloves	83
2021	lego	58	2021	weight lifting gloves	41
2020	lipsticks	50	2020	yoga mat	50
2021	lipsticks	80	2021	yoga mat	36

Analytical Question 13

Number of products sold by year

Data Graph and SQL Query (2 of 2)



```

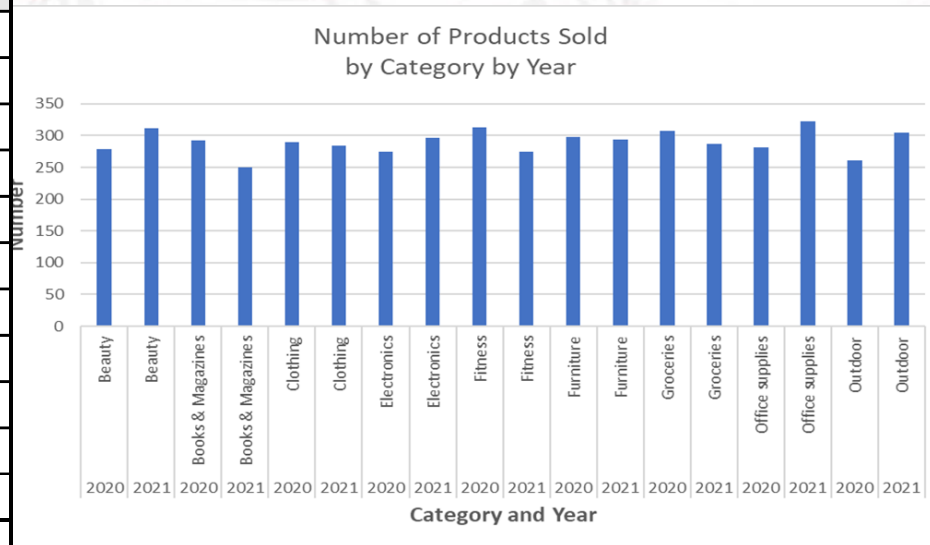
Select
    Year(OrderDate) as Year,
    Product,
    SUM(Quantity) as Number_of_prod_sold
From
    denormalized
Group By
    Product,
    Year
    
```


Analytical Question 14

Number of products sold by category by year

Data Graph and SQL Query

Year	Category	Number_of_prod_sold
2020	Beauty	278
2021	Beauty	312
2020	Books & Magazines	293
2021	Books & Magazines	250
2020	Clothing	290
2021	Clothing	284
2020	Electronics	274
2021	Electronics	297
2020	Fitness	313
2021	Fitness	274
2020	Furniture	298
2021	Furniture	294
2020	Groceries	307
2021	Groceries	287
2020	Office supplies	282
2021	Office supplies	323
2020	Outdoor	261
2021	Outdoor	305
2020	Toys	243
2021	Toys	265



```

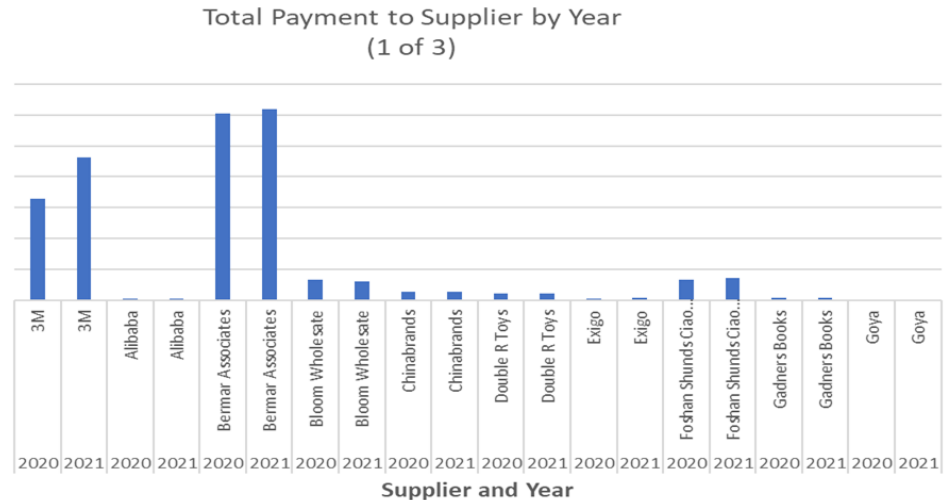
Select
    Year(OrderDate) as Year,
    Category,
    SUM(Quantity) as Number_of_prod_sold
From
    denormalized
Group By
    Category,
    Year
    
```

Analytical Question 15

Payment to each supplier by year

Data Graph and SQL Query(1 of 3)

Year	Supplier	Payment_to_supplier
2020	3M	\$33,000.00
2021	3M	\$46,200.00
2020	Alibaba	\$552.00
2021	Alibaba	\$480.00
2020	Bermar Associates	\$60,400.00
2021	Bermar Associates	\$61,900.00
2020	Bloom Wholesate	\$6,610.00
2021	Bloom Wholesate	\$6,220.00
2020	Chinabrands	\$2,650.00
2021	Chinabrands	\$2,650.00
2020	Double R Toys	\$2,100.00
2021	Double R Toys	\$2,300.00
2020	Exigo	\$610.00
2021	Exigo	\$800.00
2020	Foshan Shunds Ciao Furniture	\$6,750.00
2021	Foshan Shunds Ciao Furniture	\$7,200.00
2020	Gadners Books	\$791.00
2021	Gadners Books	\$754.00
2020	Goya	\$354.00
2021	Goya	\$315.00



Select

Year(OrderDate) as Year,

Supplier,

CONCAT("\$", FORMAT(SUM(UnitCost * Quantity), 2)) as Payment_to_supplier

From denormalized

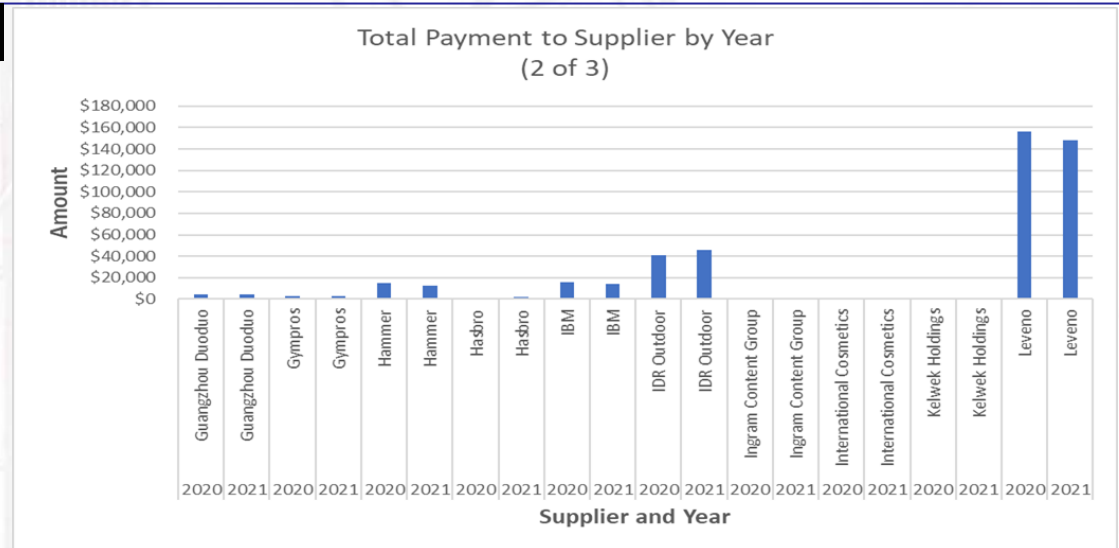
Group By Supplier, Year

Analytical Question 15

Payment to each supplier by year

Data Graph and SQL Query (2 of 3)

Year	Supplier	Payment_to_supplier
2020	Guangzhou Duoduo	\$4,170.00
2021	Guangzhou Duoduo	\$4,130.00
2020	Gympros	\$3,195.00
2021	Gympros	\$2,715.00
2020	Hammer	\$14,800.00
2021	Hammer	\$12,280.00
2020	Hasbro	\$1,605.00
2021	Hasbro	\$1,695.00
2020	IBM	\$16,200.00
2021	IBM	\$13,800.00
2020	IDR Outdoor	\$41,100.00
2021	IDR Outdoor	\$46,000.00
2020	Ingram Content Group	\$1,597.00
2021	Ingram Content Group	\$1,302.00
2020	International Cosmetics	\$464.00
2021	International Cosmetics	\$488.00
2020	Kelwek Holdings	\$265.00
2021	Kelwek Holdings	\$300.00
2020	Leveno	\$156,300.00
2021	Leveno	\$148,050.00



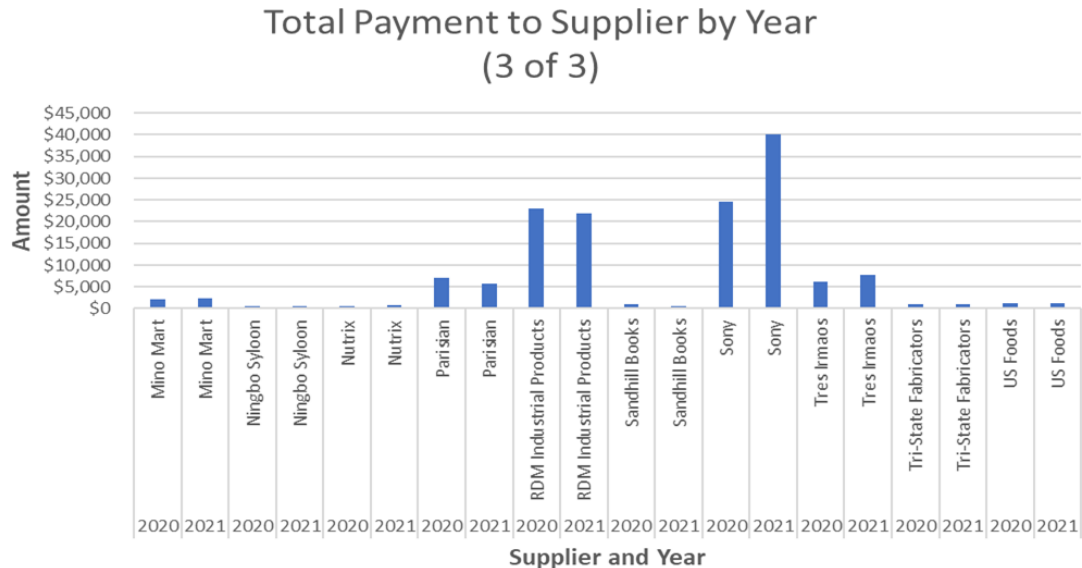
Select
 Year(OrderDate) as Year,
 Supplier,
 CONCAT("\$", FORMAT(SUM(UnitCost * Quantity), 2)) as Payment_to_supplier
 From denormalized
 Group By Supplier, Year

Analytical Question 15

Payment to each supplier by year

Data Graph and SQL Query(3 of 3)

Year	Supplier	Payment_to_supplier
2020	Mino Mart	\$2,040.00
2021	Mino Mart	\$2,320.00
2020	Ningbo Syloon	\$522.00
2021	Ningbo Syloon	\$558.00
2020	Nutrix	\$650.00
2021	Nutrix	\$851.00
2020	Parisian	\$7,180.00
2021	Parisian	\$5,805.00
2020	RDM Industrial Products	\$23,000.00
2021	RDM Industrial Products	\$22,000.00
2020	Sandhill Books	\$975.00
2021	Sandhill Books	\$645.00
2020	Sony	\$24,500.00
2021	Sony	\$40,000.00
2020	Tres Irmaos	\$6,240.00
2021	Tres Irmaos	\$7,800.00
2020	Tri-State Fabricators	\$1,020.00
2021	Tri-State Fabricators	\$1,000.00
2020	US Foods	\$1,200.00
2021	US Foods	\$1,334.00



Select

Year(OrderDate) as Year,

Supplier,

CONCAT("\$", FORMAT(SUM(UnitCost * Quantity), 2)) as Payment_to_supplier

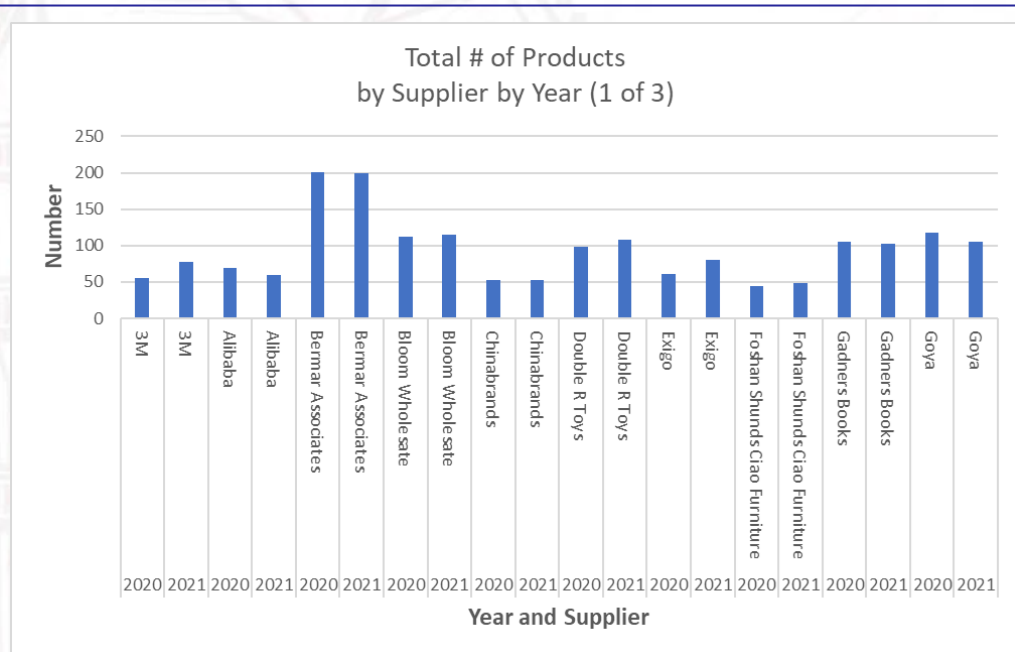
From denormalized

Group By Supplier, Year

Analytical Question 16

Total number of products provided by each supplier by year - Data Graph and SQL Query (1 of 3)

Year	Supplier	num_of_products
2020	3M	55
2021	3M	77
2020	Alibaba	69
2021	Alibaba	60
2020	Bermar Associates	201
2021	Bermar Associates	200
2020	Bloom Wholesate	112
2021	Bloom Wholesate	115
2020	Chinabrands	53
2021	Chinabrands	53
2020	Double R Toys	98
2021	Double R Toys	108
2020	Exigo	61
2021	Exigo	80
2020	Foshan Shunds Ciao Furniture	45
2021	Foshan Shunds Ciao Furniture	48
2020	Gadners Books	105
2021	Gadners Books	102
2020	Goya	118
2021	Goya	105

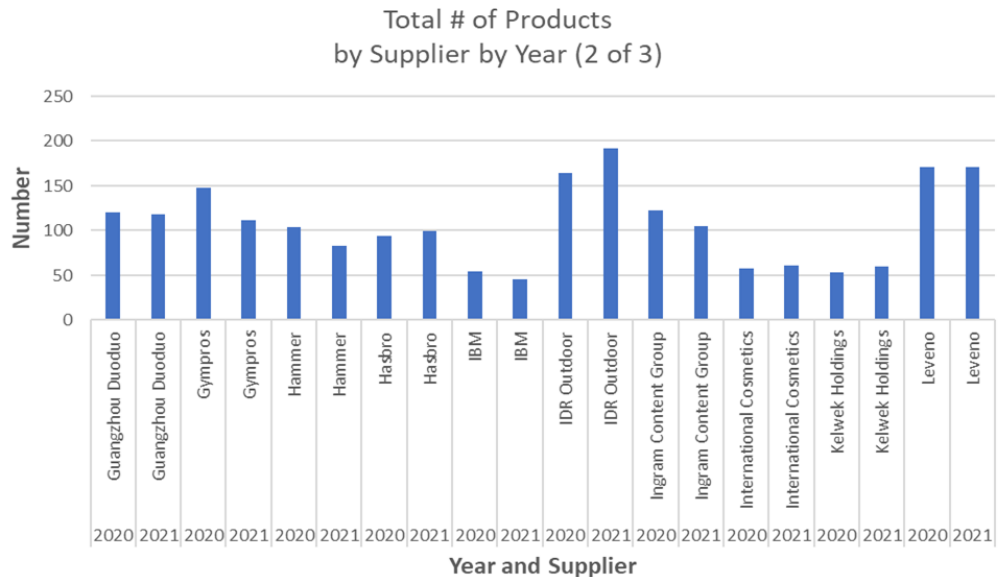


Select
 Year(OrderDate) as Year,
 Supplier,
 SUM(Quantity) as num_of_products
 From denormalized
 Group By Supplier, Year

Analytical Question 16

Total number of products provided by each supplier by year - Data Graph and SQL Query (2 of 3)

Year	Supplier	num_of_products
2020	Guangzhou Duoduo	120
2021	Guangzhou Duoduo	118
2020	Gympros	148
2021	Gympros	111
2020	Hammer	104
2021	Hammer	83
2020	Hasbro	94
2021	Hasbro	99
2020	IBM	54
2021	IBM	46
2020	IDR Outdoor	164
2021	IDR Outdoor	192
2020	Ingram Content Group	123
2021	Ingram Content Group	105
2020	International Cosmetics	58
2021	International Cosmetics	61
2020	Kelwek Holdings	53
2021	Kelwek Holdings	60
2020	Leveno	171
2021	Leveno	171

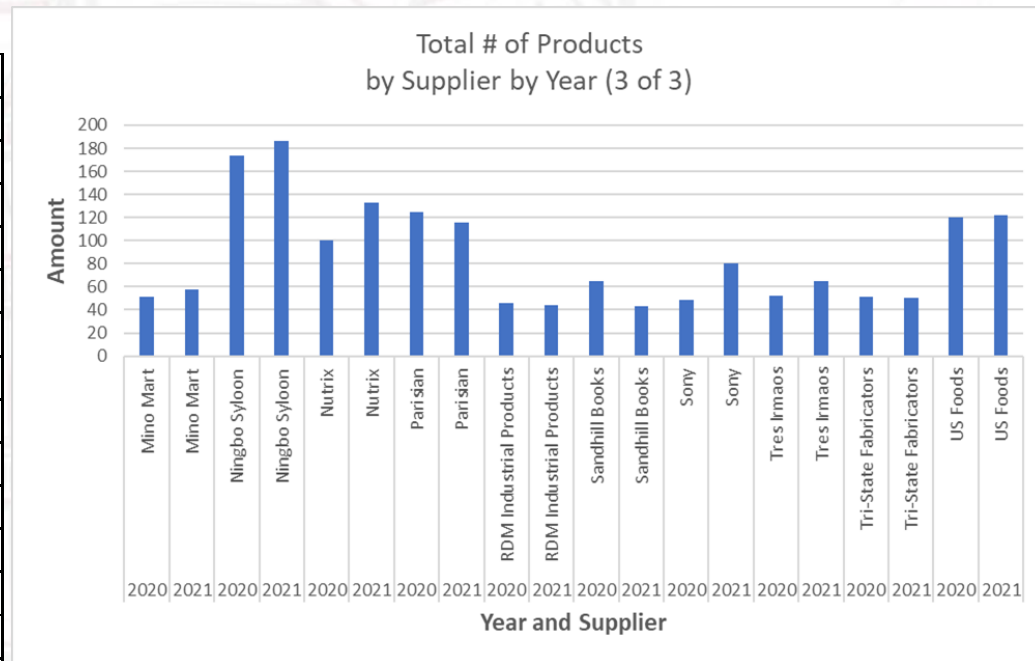


Select
 Year(OrderDate) as Year,
 Supplier,
 SUM(Quantity) as num_of_products
 From denormalized
 Group By Supplier, Year

Analytical Question 16

Total number of products provided by each supplier by year - Data Graph and SQL Query (3 of 3)

Year	Supplier	num_of_products
2020	Mino Mart	51
2021	Mino Mart	58
2020	Ningbo Syloon	174
2021	Ningbo Syloon	186
2020	Nutrix	100
2021	Nutrix	133
2020	Parisian	125
2021	Parisian	116
2020	RDM Industrial Products	46
2021	RDM Industrial Products	44
2020	Sandhill Books	65
2021	Sandhill Books	43
2020	Sony	49
2021	Sony	80
2020	Tres Irmaos	52
2021	Tres Irmaos	65
2020	Tri-State Fabricators	51
2021	Tri-State Fabricators	50
2020	US Foods	120
2021	US Foods	122



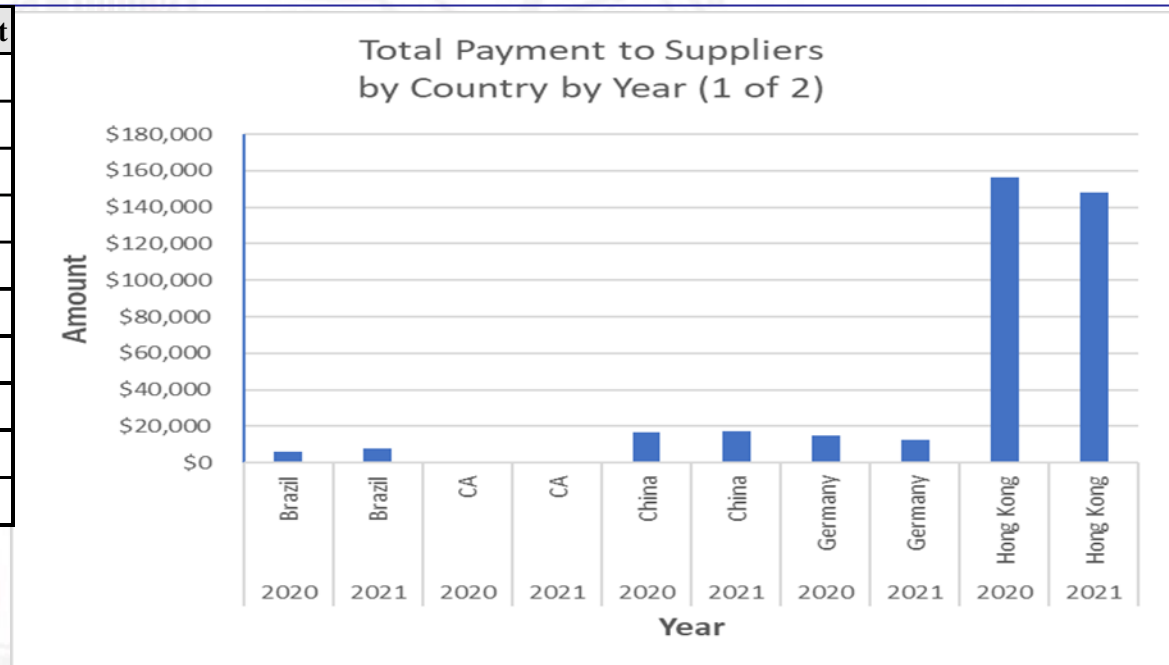
Select
 Year(OrderDate) as Year,
 Supplier,
 SUM(Quantity) as num_of_products
 From denormalized
 Group By Supplier, Year

Analytical Question 17

Total payment to suppliers by country by year

Data Graph and SQL Query (1 of 2)

Year	SupplierCountry	Total_payment
2020	Brazil	\$6,240.00
2021	Brazil	\$7,800.00
2020	CA	\$975.00
2021	CA	\$645.00
2020	China	\$16,744.00
2021	China	\$17,318.00
2020	Germany	\$14,800.00
2021	Germany	\$12,280.00
2020	Hong Kong	\$156,300.00
2021	Hong Kong	\$148,050.00



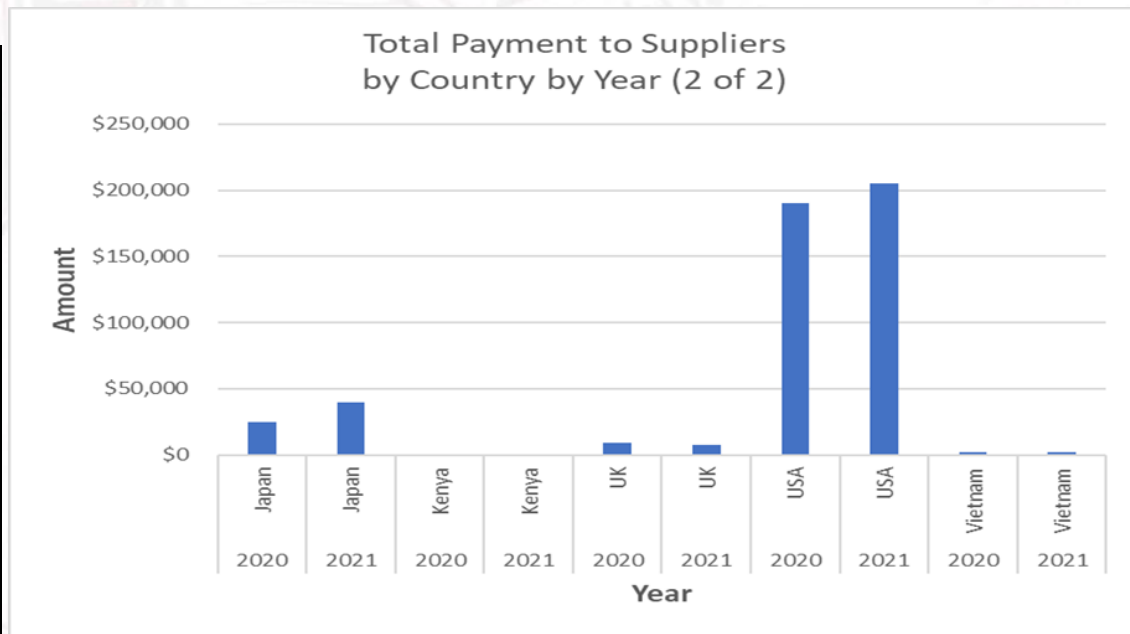
Select Year(OrderDate) as Year,
 SupplierCountry,
 CONCAT("\$", FORMAT(SUM(UnitCost * Quantity), 2)) as Total_payment
 From denormalized
 Group By SupplierCountry, Year

Analytical Question 17

Total payment to suppliers by country by year

Data Graph and SQL Query (2 of 2)

Year	SupplierCountry	Total_payment
2020	Japan	\$24,500.00
2021	Japan	\$40,000.00
2020	Kenya	\$265.00
2021	Kenya	\$300.00
2020	UK	\$9,045.00
2021	UK	\$7,847.00
2020	USA	\$189,931.00
2021	USA	\$205,332.00
2020	Vietnam	\$2,040.00
2021	Vietnam	\$2,320.00



```

Select Year(OrderDate) as Year,
       SupplierCountry,
       CONCAT("$", FORMAT(SUM(UnitCost * Quantity), 2)) as Total_payment
From denormalized
Group By SupplierCountry, Year
    
```

Analytical Question 18

Total discounts received by membership type by year – Data Graph and SQL Query

Year	MembershipType	Total_discount_received
2020	Basic	\$3,855.63
2020	Guest	\$4,489.93
2020	Loyal	\$3,385.09
2021	Basic	\$4,257.92
2021	Guest	\$5,004.09
2021	Loyal	\$4,537.26



```
SELECT
    year(OrderDate) as Year,
    MembershipType,
    CONCAT("$", FORMAT(SUM(((UnitPrice * (PercentDiscount / 100)) + ((UnitPrice * (PercentDiscount / 100)) *
(MemberDiscountPercent / 100))), 2)) as Total_discount_received

FROM denormalized
GROUP BY Year, MembershipType
```

The background of the slide features a large, faint, circular seal of Golden Gate University. The seal contains a detailed illustration of the Golden Gate Bridge, with the university's name "GOLDEN GATE UNIVERSITY" arched across the top and the year "1901" at the bottom. The word "Questions?" is centered over the seal in a large, bold, black font.

Questions?