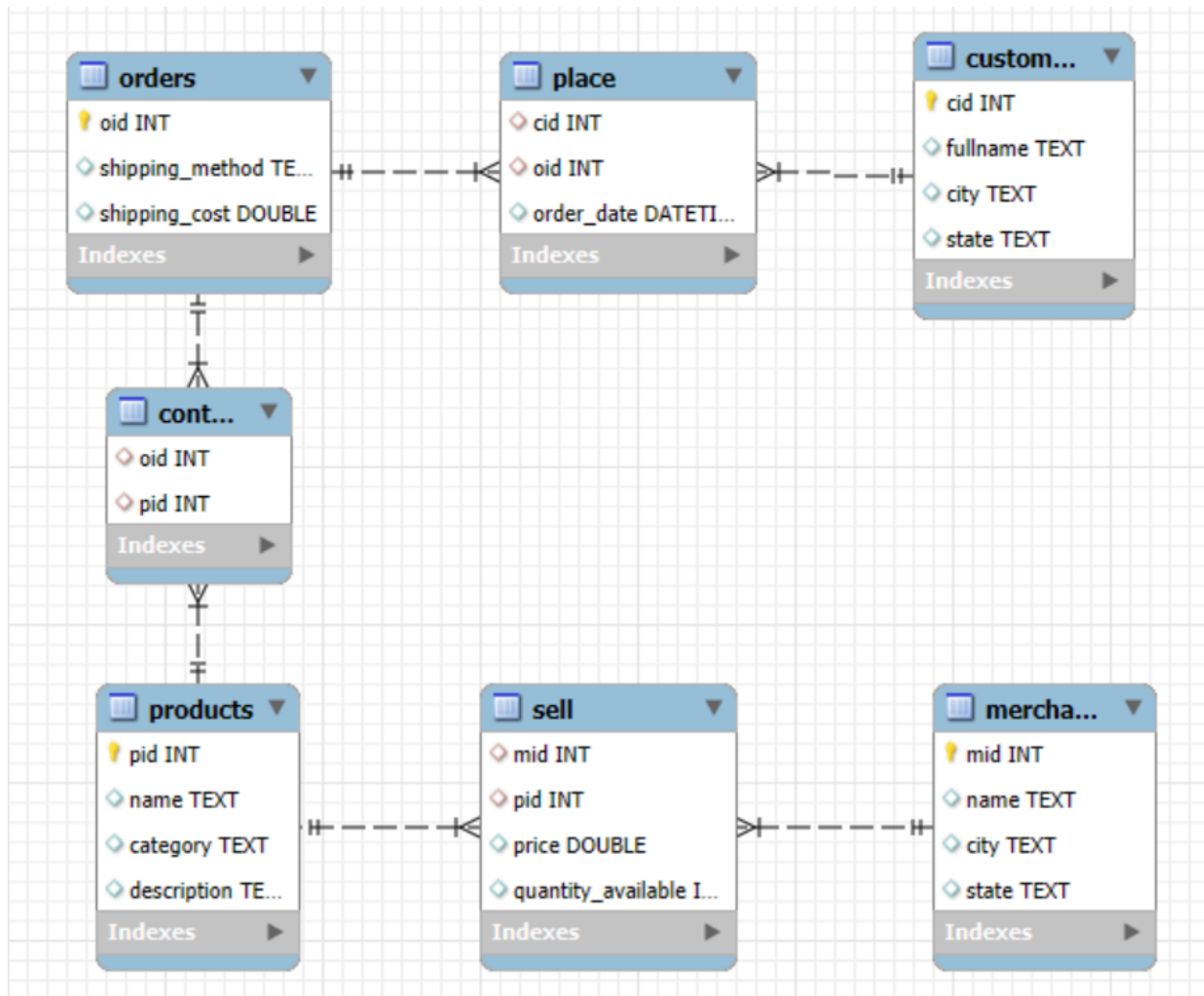


Title: Database Assignment 3

Your Name: Jack Saunders

Date: 10/12/2024

## ER Diagram



1. List names and sellers of products that are no longer available (quantity=0)

This query shows all the items with quantity=0  
Inner joins on merchants, sell, and products

**Answer:**

```

25 -- Queries Homework 3
26 -- 1) List names and sellers of products that are no longer available (quantity=0)
27 • select distinct merchants.name, products.name, quantity_available
28 from merchants inner join sell inner join products
29 on sell.pid = products.pid
30 where quantity_available = 0;
31

```

name	name	quantity_available
Lenovo	Router	0
Dell	Router	0
HP	Router	0
Apple	Router	0
Acer	Router	0
Lenovo	Network Card	0

**Table continues...**

## 2. List names and descriptions of products that are not sold

When an item is not sold, it's quantity\_available is zero in the sell table as shown by line 35. So showing only items with quantity of zero shows not sold items.

Inner join on sell and products.

**Answer:**

```

32 -- 2) List names and descriptions of products that are not sold.
33 • select distinct quantity_available as qty, products.name, description
34 from sell inner join products
35 where quantity_available = 0;

```

qty	name	description
0	Hard Drive	640GB USB 2.0 Portable
0	Monitor	LED 22-inch Backlit
0	Printer	All-in-one
0	Super Drive	8X DVD-ROM 32X CD-ROM SATA Internal Blu-ra...
0	Hard Drive	2TB Internal SATA
0	Laptop	Intel Core i5-2410M 2.3GHz / 4GB RAM / 640GB...

**Table continues...**

## 3. How many customers bought SATA drives but not any routers?

I Googled SATA Drive, so I assumed a SATA Drive is a Super Drive. SQL visual error in line 42 for the except clause. It still works. The except clause subtracts all items from the second half of the statement that are within the first half of the statement, thus excluding customers who have bought a Router.

**Answer:**

```

37 -- 3) How many customers bought SATA drives but not any routers?
38 • select distinct customers.fullname, products.name, contain.oid
39 from customers inner join place inner join orders inner join contain inner join products
40 on customers.cid = place.cid and place.oid = orders.oid and orders.oid = contain.oid and contain.pid = products.pid
41 where products.name = 'Super Drive'
42 except
43 select distinct customers.fullname, products.name, contain.oid
44 from customers inner join place inner join orders inner join contain inner join products
45 on customers.cid = place.cid and place.oid = orders.oid and orders.oid = contain.oid and contain.pid = products.pid
46 where products.name = 'Router';

```

fullname	name	oid
Caryn Merritt	Super Drive	12
Inez Long	Super Drive	131
Charde McGee	Super Drive	140
Clementine Travis	Super Drive	140
Gray Perkins	Super Drive	140

**Table continues...**

Inner joins on customers, place, orders, contain, and products. Lines 38-40 and 43-45 are identical. The difference is line 41 that looks for Super Drive and line 46 which looks for Router.

Lines 48-59 was failed attempt. But, other calculations during this failed attempt lead to finding the pids of super drives and oids of routers. A more manual approach would have used these numbers.

```
49  /*select oid, contain.pid, products.name
50      from contain inner join products
51      on contain.pid = products.pid
52      where products.name = 'Super Drive'
53      order by oid asc;
54  except
55      select oid
56      from contain inner join products
57      on contain.pid = products.pid
58      where products.name = 'router'
59      order by oid asc;
60
61  -- SATA drive pids 4, 7, 11, 21, 22, 30, 31, 32
62  -- router oids 8, 18, 19, 20, 23*/
```

#### 4. HP has a 20% sale on all its Networking products.

Since this isn't a question, I assumed all of HP's Networking products should have a 20% off sale. Shows only HP items in the networking category. Calculates the sale price ( 20% sale is 80% of the original price) using \*.8 for sale price.

**Answer:**

```

64 -- 4) HP has a 20% sale on all its Networking products.
65 • select merchants.name, products.name, products.category, sell.price * .8
66 from merchants inner join sell inner join products
67 on merchants.mid = sell.mid and sell.pid = products.pid
68 where category = 'networking' and merchants.name = 'HP';
69

```

name	name	category	sell.price *.8
HP	Router	Networking	827.5680000000001
HP	Network Card	Networking	923.7440000000001
HP	Network Card	Networking	276.008
HP	Network Card	Networking	209.76
HP	Ethernet Adapter	Networking	1008.3600000000001
HP	Router	Networking	164.448

**Table continues...**

## 5. What did Uriel Whitney order from Acer? (make sure to at least retrieve product names and prices).

Perform inner join on customers, place, contain, products, sell, and merchants. Return only unique or distinct products along with customer fullname, sell price and company or merchant name.

The magic is the where clause that looks for truples in all the joined datasets with both a customer's fullname of Uriel Whitney and company's name of Acer.

**Answer:**

```

78 -- 5) What did Uriel Whitney order from Acer? (make sure to at least retrieve product names and prices).
79 • select distinct products.name, sell.price, merchants.name, fullname
80 from customers inner join place inner join contain inner join products inner join sell inner join merchants
81 on customers.cid = place.cid and place.cid = contain.cid and contain.pid = products.pid and products.pid = sell.pid and sell.mid = merchants.mid
82 where fullname = 'Uriel Whitney' and merchants.name = 'Acer';
83

```

name	price	name	fullname
Monitor	1435.38	Acer	Uriel Whitney
Router	521.07	Acer	Uriel Whitney
Router	1256.57	Acer	Uriel Whitney
Monitor	1103.47	Acer	Uriel Whitney
Super Drive	356.13	Acer	Uriel Whitney
Printer	1345.37	Acer	Uriel Whitney

**Table continues...**

## 6. List the annual total sales for each company (sort the results along the company and the year attributes).

Generally, the higher quantity\_available is a higher price because more items were bought. There is some price variation coming from the markup of different companies like Acer's monitor vs HP's monitor. So all we need is the

**Answer:**

```

76 -- 6) List the annual total sales for each company (sort the results along the company and the year attributes).
77 • select distinct merchants.name, avg(price)/9.986381 as 'avg price per year', max(order_date), min(order_date)
78 from merchants inner join sell inner join contain inner join orders inner join place
79 on merchants.mid = sell.mid and sell.pid = contain.pid and contain.oid = place.oid
80 group by merchants.name
81 order by avg(price) desc;
82

```

name	avg price per year	max(order_date)	min(order_date)
Lenovo	86.62806754907642	2020-12-24 00:00:00	2011-01-02 00:00:00
Apple	80.13597634162296	2020-12-24 00:00:00	2011-01-02 00:00:00
Dell	77.37787683632025	2020-12-24 00:00:00	2011-01-02 00:00:00
HP	71.8247683406127	2020-12-24 00:00:00	2011-01-02 00:00:00
Acer	71.64728885263914	2020-12-24 00:00:00	2011-01-02 00:00:00

price and no shipping cost, which is extra and paid by customers and is assumed to not become revenue for the company.

These side calculations in lines 85-91 and with the help of a calculator, I found that the range of dates is **9.986301 years**

```
85  /*select max(order_date)
86  from place;-- 12/24/2020
87  select min(order_date)
88  from place;-- 1/2/2011
89  SELECT datediff(max(order_date), min(order_date)) as datediff
90  from place;*/
91  -- actual difference between 12/24/2020 and 1/2/2011 is 9.986301 years
```

This is helpful to find the average revenue of each company **per year**. Thus not in total. The prices are divided by 9.986301 years to find the average revenue per year in the middle of line 77. I assumed all companies started at the same time and ended at the same time for fairness to all the companies and since it's unknown when exactly a company started. It's possible that they didn't get any sales in days or even years at first. Also, line 81 puts the list in descending order by price, so the highest average is at the top. Plus, the full range of dates is on the right.

7. Which company had the highest annual revenue and in what year?

Line 94 shows the average price as in average revenue per year. It also shows the year only of each year of revenue.

Inner join on merchants, sell, contain, orders, and place.

Ordered by early to later years on the right side.

Line 99 limits the table to a single tuple, which has the highest price of all years.

### Answer:

```

93 -- 7) Which company had the highest annual revenue and in what year?
94 • select distinct merchants.name, avg(price) as 'avg price per year', year(order_date)
95 from merchants inner join sell inner join contain inner join orders inner join place
96 on merchants.mid = sell.mid and sell.pid = contain.pid and contain.oid = place.oid
97 group by merchants.name, year(order_date)
98 order by year(order_date), avg(price) desc
99 limit 1;

```

Result Grid		
name	avg price per year	year(order_date)
Lenovo	906.5657352947308	2011

Table continues...

## 8. On average, what was the cheapest shipping method used ever?

“ever” means for a single purchase as in lowest shipping cost of any oid.

Only the orders table is required for the data that the question is asking. Ordered from lowest to highest minimum costs per shipping method. There are only 3 shipping methods, but the lowest price **only** of these 3 is still shown due to line 106.

### Answer:

```

101 -- 8) On average, what was the cheapest shipping method used ever?
102 • select shipping_method, min(shipping_cost)
103 from orders
104 group by shipping_method
105 order by min(shipping_cost) asc
106 limit 1;

```

Result Grid	
shipping_method	min(shipping_cost)
USPS	0.02
FedEx	0.03
UPS	0.07

## 9. What is the best sold (\$) category for each company?

Inner join on merchants, sell, and products.

Orders price in descending order with the category of the products and the company name.

The top results are the most sold category.

### Answer:

```

108 -- 9) What is the best sold ($) category for each company?
109 • select distinct merchants.name, category, price
110 from merchants inner join sell inner join products
111 on merchants.mid = sell.mid and sell.pid = products.pid
112 order by price desc;

```

Result Grid			Filter Rows:	Export:	Wrap Cell Content:
name	category	price			
HP	Computer	1490.37			
Lenovo	Peripheral	1487.03			
HP	Networking	1474.87			
Dell	Peripheral	1450.74			
Acer	Peripheral	1435.38			

**Table continues...**

10. For each company find out which customers have spent the most and the least amounts.

Inner join on everything except products because don't need data or connection to products.

Finds sum of prices for customer's purchases and the min and max of them in line 116.

Orders prices in ascending order in line 122.

**Answer:**

```

114 -- 10) For each company find out which customers have spent the most and
115 -- the least amounts.
116 • select fullname, sum(price) as min, sum(price) as max
117 from customers inner join place inner join orders inner join contain inner join
118 sell inner join merchants
119 on customers.cid = place.cid and place.oid = orders.oid and orders.oid = contain.oid
120 and contain.pid = sell.pid and sell.mid = merchants.mid
121 group by fullname
122 order by sum(price) asc;

```

Result Grid			Filter Rows:	Export:	Wrap Cell Content:
fullname	min	max			
Inez Long	155299.66	155299.66			
Wynne McKinney	166074.87999999998	166074.87999999998			
Charde McGee	198456.83999999994	198456.83999999994			
Abraham Sears	203470.73999999996	203470.73999999996			
Gray Perkins	232513.00999999995	232513.00999999995			