1. How many total products are in the Products table?

Considerations
Distinct products are determined by the item_description column in the products table. However, the description column contains a few entry errors which would lead to multiple counts for one product. The number of entry errors are considered negligible compared to total number of different products.

Query
```
SELECT COUNT(DISTINCT item_description) AS total_products
FROM products
```

Data Output

	total_products bigint
1	7276

2. Who are the top most diverse vendors (i.e. they have the highest number of distinct products)? How many different products do they have?

Query
```
SELECT vendor_name, COUNT(DISTINCT item_description) AS num_of_different_products
FROM products
GROUP BY 1
ORDER BY 2 DESC
LIMIT 10
```

Data Output

	vendor_name text	num_of_different_products bigint
1	Jim Beam Brands	680
2	Diageo Americas	654
3	Pernod Ricard Usa/austin Nichols	473
4	Heaven Hill Distilleries Inc.	283
5	Mhw Ltd	254
6	Bacardi U.s.a. Inc.	238
7	Sazerac Co. Inc.	217
8	Luxco-st Louis	207
9	Brown-forman Corporation	198
10	Sazerac North America	185

3. Which products sell the best by total number of unit sales?

Considerations
Distinct products are determined by the description column in the sales table. However, the description column contains a few entry errors which would lead to multiple counts for one product. The number of entry errors are considered negligible compared to total number of different products.

Query

```
SELECT description, SUM(bottle_qty) AS total_num_of_unit_sales
FROM sales
GROUP BY 1
ORDER BY 2 DESC
```

Data Output

	description text	total_num_of_unit_sales bigint
1	Black Velvet	1672344
2	Hawkeye Vodka	1192739
3	Captain Morgan Spiced Rum	766901
4	Fireball Cinnamon Whiskey	666062
5	Jack Daniels Old #7 Black Lbl	546234
6	Five O'clock	503006
7	Mccormick Vodka Pet	487426
8	Barton Vodka	450451
9	Smirnoff Vodka 80 Prf	425220
10	Crown Royal Canadian Whisky	376931
11	Mccormick Vodka	373696
12	Phillips Vodka	366488

Total rows: 2958 Query complete 00:00:00.894

4. Which products sell the best by total dollar value of sales?

Query

```
SELECT description, SUM(total) AS total_dollar_value_of_sales
FROM sales
GROUP BY 1
ORDER BY 2 DESC
```

Data Output

	description text	total_dollar_value_of_sales numeric
1	Black Velvet	18315550.64
2	Captain Morgan Spiced Rum	13772752.25
3	Jack Daniels Old #7 Black Lbl	13701656.82
4	Fireball Cinnamon Whiskey	10622967.34
5	Crown Royal Canadian Whisky	9714022.78
6	Hawkeye Vodka	8730823.84
7	Absolut Swedish Vodka 80 Prf	7431864.05
8	Grey Goose Vodka	6444939.38
9	Jagermeister Liqueur	6298430.40
10	Jim Beam	5163920.19
11	Bacardi Superior Rum	4895721.66
12	Smirnoff Vodka 80 Prf	4876655.59

Total rows: 2958 Query complete 00:00:01.057

5. What are the top 10 categories of liquor sold based on the total amount of sales revenue?

Considerations
There is 1 null entry in the category column, and 11 null entries in the category_name column in the sales table. However, they do not affect the top 10 categories of liquor sold based on total sales revenue.

Query
```
SELECT category_name, SUM(total) AS total_sales_revenue
FROM sales
GROUP BY 1
ORDER BY 2 DESC
LIMIT 10
```

Data Output

	category_name text	total_sales_revenue numeric
1	CANADIAN WHISKIES	48053061.91
2	80 PROOF VODKA	48045532.51
3	SPICED RUM	31600618.50
4	IMPORTED VODKA	23879524.63
5	TEQUILA	21411263.64
6	STRAIGHT BOURBON WHISKIES	20924480.19
7	WHISKEY LIQUEUR	19339201.42
8	TENNESSEE WHISKIES	17647970.35
9	PUERTO RICO & VIRGIN ISLANDS RUM	12729072.76
10	BLENDED WHISKIES	12037250.55

6. Which rum products have sales greater than $10,000? How about whiskey or vodka products?

Considerations
Only products which fall explicitly into the Rum, Whiskey/Whiskies, or Vodka category under category_name in the sales table are being considered for this question. Null categories are not considered.

Query 1
```
SELECT description as rum_products, SUM(total) AS total_sales_revenue
FROM sales
WHERE category_name ILIKE '%rum%'
GROUP BY description
HAVING SUM(total) > 10000
ORDER BY 2 DESC
```

Data Output 1

	rum_products text	total_sales_revenue numeric
1	Captain Morgan Spiced Rum	13772752.25
2	Bacardi Superior Rum	4895721.66
3	Captain Morgan Spiced Barrel	3774545.06
4	Admiral Nelson Spiced Rum	3650784.34
5	Captain Morgan Original Spiced	3497803.08
6	Malibu Coconut Rum	3431143.75
7	Paramount White Rum	2287962.99
8	Bacardi Limon	1498386.52
9	Captain Morgan Original Spiced Rum Pet	1334951.04
10	Sailor Jerry Spiced Navy Rum	1167578.50
11	Bacardi Gold Rum	838135.45
12	Paramount Gold Rum	744953.37

Total rows: 122 Query complete 00:00:00.780

Query 2

```
SELECT description as whiskey_products, SUM(total) AS total_sales_revenue
FROM sales
WHERE category_name ILIKE '%whisk%'
GROUP BY description
HAVING SUM(total) > 10000
ORDER BY 2 DESC
```

Data Output 2

	whiskey_products text	total_sales_revenue numeric
1	Black Velvet	18315550.64
2	Jack Daniels Old #7 Black Lbl	13701656.82
3	Fireball Cinnamon Whiskey	10622967.34
4	Crown Royal Canadian Whisky	9714022.78
5	Jim Beam	5163920.19
6	Jameson	4711290.31
7	Seagrams 7 Crown Bl Whiskey	4330957.37
8	Crown Royal	4124703.73
9	Southern Comfort	3571641.63
10	Templeton Rye	3052082.98
11	Maker's Mark	2963128.62
12	Canadian Ltd Whisky	2405739.82

Total rows: 293 Query complete 00:00:01.571

Query 3

```
SELECT description as vodka_products, SUM(total) AS total_sales_revenue
FROM sales
WHERE category_name ILIKE '%vodka%'
GROUP BY description
HAVING SUM(total) > 10000
ORDER BY 2 DESC
```

Data Output 3

	vodka_products text	total_sales_revenue numeric
1	Hawkeye Vodka	8730823.84
2	Absolut Swedish Vodka 80 Prf	7431864.05
3	Grey Goose Vodka	6444939.38
4	Smirnoff Vodka 80 Prf	4876655.59
5	Barton Vodka	3236508.16
6	Phillips Vodka	3093016.65
7	Five O'clock	2724203.37
8	Uv Blue (raspberry) Vodka	2547677.99
9	Ketel One Imported Vodka	2334630.05
10	UV Vodka PET	2308126.71
11	Smirnoff Vodka 80 Prf Pet	2129087.76
12	Mccormick Vodka Pet	2115611.57

Total rows: 291 Query complete 00:00:01.648

7. Which county sold the most amount of vodka during February 2014?

Considerations

For each county, only the total sales revenue from vodka was considered.

Query

```
SELECT county, SUM(total) AS total_sales_revenue_from_vodka
FROM sales
WHERE category_name ILIKE '%vodka%' AND date BETWEEN '2014-02-01' AND '2014-02-28'
GROUP BY 1
ORDER BY 2 DESC
LIMIT 1
```

Data Output

	county text	total_sales_revenue_from_vodka numeric
1	Polk	2259389.14

8. Which counties were in the top 10 counties for vodka sales in any month in 2014?

Query

```
WITH ranked_sales AS
(
SELECT county, to_char(date, 'MM-YYYY') AS sale_month, SUM(total) AS monthly_vodka_sales,
        RANK() OVER(PARTITION BY to_char(date, 'MM-YYYY') ORDER BY SUM(total) DESC) AS
sales_rank
FROM sales
WHERE category_name ILIKE '%vodka%' AND date BETWEEN '2014-01-01' AND '2014-12-31'
GROUP BY 2, 1
)
SELECT county, sale_month, monthly_vodka_sales
FROM ranked_sales
WHERE sales_rank <= 10
ORDER BY 2, sales_rank
```

Data Output

	county text	sale_month text	monthly_vodka_sales numeric
1	Polk	01-2014	2188692.56
2	Linn	01-2014	975547.22
3	Scott	01-2014	764583.12
4	Johnson	01-2014	687857.74
5	Black Hawk	01-2014	611497.16
6	Pottawattamie	01-2014	425965.76
7	Story	01-2014	358245.36
8	Dubuque	01-2014	348632.82
9	Woodbury	01-2014	289273.18
10	Cerro Gordo	01-2014	222337.54
11	Polk	02-2014	2259389.14
12	Linn	02-2014	737865.34

Total rows: 120 Query complete 00:00:01.639

9. Create a report that shows how many times a county appeared in the "top 10 counties for vodka sales in a month" list over the course of 2014.

Query

```
WITH top_ten_list AS
(
WITH ranked_sales AS
(
SELECT county, to_char(date, 'MM-YYYY') AS sale_month, SUM(total) AS monthly_vodka_sales,
        RANK() OVER(PARTITION BY to_char(date, 'MM-YYYY') ORDER BY SUM(total) DESC) AS
sales_rank
FROM sales
WHERE category_name ILIKE '%vodka%' AND date BETWEEN '2014-01-01' AND '2014-12-31'
GROUP BY 2, 1
)
SELECT county, sale_month, monthly_vodka_sales
FROM ranked_sales
WHERE sales_rank <= 10
ORDER BY 2, sales_rank
)
SELECT county, COUNT(county) AS num_of_times_in_top_ten
FROM top_ten_list
GROUP BY county
ORDER BY 2 DESC
```

Data Output

	county text	num_of_times_in_top_ten bigint
1	Black Hawk	12
2	Dubuque	12
3	Johnson	12
4	Linn	12
5	Polk	12
6	Pottawattamie	12
7	Scott	12
8	Woodbury	11
9	Story	10
10	Cerro Gordo	7
11	Dallas	6
12	Des Moines	1
13	Dickinson	1

Total rows: 13 | Query complete 00:00:01.580

Q10. What is the trend of sales by month? Break up variables such as bottle_price or liter_size into categories (for example: cheap, medium, or expensive).

Considerations

The variables btl_price and liter_size were separated into 4 and 3 different categories, respectively, using the ntiles function. However, because the btl_price column contains discrete data, the 25%, 50, and 75% percentile price points had to be manually assigned to 'Average', 'Expensive' and 'Very Expensive' categories to avoid a spread of the percentile price points in 2 different categories. The same treatment was done for the 'Small', 'Medium', and 'Large' categories for the liter_size column.

Query 1

```
WITH btl_price_cat_table2 AS
(
WITH btl_price_cat_table AS
(
SELECT btl_price, date, total,
            NTILE(4) OVER (ORDER BY btl_price) as cat_num
FROM sales
)
,
percentiles AS
(
SELECT PERCENTILE_DISC(0.25) WITHIN GROUP (ORDER BY btl_price) AS p25,
            PERCENTILE_DISC(0.50) WITHIN GROUP (ORDER BY btl_price) AS median,
            PERCENTILE_DISC(0.75) WITHIN GROUP (ORDER BY btl_price) AS p75
FROM sales
)
SELECT date, total, btl_price, cat_num,
CASE
   WHEN cat_num = 4 OR btl_price = p75
         THEN 'Very Expensive'
         WHEN cat_num = 3 OR btl_price = median
         THEN 'Expensive'
         WHEN cat_num = 2 OR btl_price = p25
         THEN 'Average'
         ELSE 'Cheap'
END AS btl_price_categories
FROM btl_price_cat_table, percentiles
)
SELECT btl_price_categories, to_char(date, 'MM-YYYY') AS sale_month, SUM(total) AS monthly_sales,
            RANK() OVER(PARTITION BY to_char(date, 'MM-YYYY') ORDER BY SUM(total) DESC) AS
sales_rank
FROM btl_price_cat_table2
WHERE date BETWEEN '2014-01-01' AND '2014-12-31'
GROUP BY 2, 1
```

Data Output 1

	btl_price_categories text	sale_month text	monthly_sales numeric	sales_rank bigint
1	Very Expensive	01-2014	16717332.20	1
2	Expensive	01-2014	11154099.76	2
3	Average	01-2014	7241342.28	3
4	Cheap	01-2014	5218249.72	4
5	Very Expensive	02-2014	16547516.42	1
6	Expensive	02-2014	10967888.94	2
7	Average	02-2014	6179241.62	3
8	Cheap	02-2014	4848941.44	4
9	Very Expensive	03-2014	17282020.42	1
10	Expensive	03-2014	9048810.62	2
11	Average	03-2014	7061995.18	3
12	Cheap	03-2014	5210717.60	4

Total rows: 48 Query complete 00:00:10.935

Query 2

```
WITH liter_size_cat_table2 AS
(
WITH liter_size_cat_table AS
(
SELECT liter_size, date, total,
        NTILE(3) OVER (ORDER BY liter_size) as cat_num
FROM sales
)
,
percentiles AS
(
SELECT PERCENTILE_CONT(0.333) WITHIN GROUP (ORDER BY liter_size) AS p33,
        PERCENTILE_CONT(0.667) WITHIN GROUP (ORDER BY liter_size) AS p67
FROM sales
)
SELECT date, total, liter_size, cat_num,
CASE
        WHEN cat_num = 3 OR liter_size = p67
        THEN 'Large'
        WHEN cat_num = 2 OR liter_size = p33
        THEN 'Medium'
        ELSE 'Small'
END AS liter_size_categories
FROM liter_size_cat_table, percentiles
)
SELECT liter_size_categories, to_char(date, 'MM-YYYY') AS sale_month, SUM(total) AS monthly_sales,
        RANK() OVER(PARTITION BY to_char(date, 'MM-YYYY') ORDER BY SUM(total) DESC) AS
sales_rank
FROM liter_size_cat_table2
WHERE date BETWEEN '2014-01-01' AND '2014-12-31'
GROUP BY 2, 1
```

Data Output 2

	liter_size_categories text	sale_month text	monthly_sales numeric	sales_rank bigint
1	Large	01-2014	20991650.96	1
2	Medium	01-2014	16286928.78	2
3	Small	01-2014	3052444.22	3
4	Large	02-2014	20345777.68	1
5	Medium	02-2014	15141755.12	2
6	Small	02-2014	3056055.62	3
7	Large	03-2014	20625882.94	1
8	Medium	03-2014	14725402.40	2
9	Small	03-2014	3252258.48	3
10	Large	04-2014	23747393.20	1
11	Medium	04-2014	17737936.06	2
12	Small	04-2014	3903165.38	3

Total rows: 36 Query complete 00:00:10.811

11b. How many stores have more than $2,000,000 in total sales?

Query

```
WITH stores_making_above_2mil AS
(
SELECT store, SUM(total)
FROM sales
GROUP BY 1
HAVING SUM(total) > 2000000
ORDER BY 2
)
SELECT COUNT(store) as num_of_stores_making_above_2mil
FROM stores_making_above_2mil
```

Data Output

	num_of_stores_making_above_2mil bigint
1	24

Bonus Q1. We think the data might have been corrupted in some way. The category listed in the sales table doesn't always match up with the category in the products table. How many times has this happened, and can you find any patterns to it?

Query

```
WITH mismatch_counter_table AS
(
SELECT s.category_name as sales_category, p.category_name as products_category, s.vendor as sales_vendor, s.county as sales_county, s.store as sales_store,
CASE
        WHEN s.category_name != p.category_name
        THEN 'Mismatch'
        ELSE 'Correct'
END AS mismatch_counter
FROM sales s
INNER JOIN products p ON s.item = p.item_no
WHERE s.category_name IS NOT NULL AND p.category_name IS NOT NULL
)
SELECT mismatch_counter, COUNT(*)
FROM mismatch_counter_table
GROUP BY 1
```

Data Output

	mismatch_counter text	count bigint
1	Correct	3046390
2	Mismatch	2020

Alternative Query 1 (needs to be combined with mismatch_counter_table)

```
SELECT sales_category, products_category, COUNT(mismatch_counter) AS num_of_mismatches
FROM mismatch_counter_table
WHERE mismatch_counter = 'Mismatch'
GROUP BY 1, 2
ORDER BY 3 DESC
```

Data Output

	sales_category text	products_category text	num_of_mismatches bigint
1	DECANTERS & SPECIALTY PACKAGES	MISC. IMPORTED CORDIALS & LIQUEURS	718
2	IMPORTED GRAPE BRANDIES	DECANTERS & SPECIALTY PACKAGES	286
3	SCOTCH WHISKIES	SINGLE MALT SCOTCH	184
4	SPICED RUM	DECANTERS & SPECIALTY PACKAGES	177
5	STRAIGHT RYE WHISKIES	CANADIAN WHISKIES	136
6	MISC. IMPORTED CORDIALS & LIQUEURS	COFFEE LIQUEURS	96
7	IMPORTED GRAPE BRANDIES	MISC. IMPORTED CORDIALS & LIQUEURS	90
8	80 PROOF VODKA	IMPORTED VODKA	74
9	SINGLE MALT SCOTCH	DECANTERS & SPECIALTY PACKAGES	72
10	TEQUILA	COFFEE LIQUEURS	66
11	CANADIAN WHISKIES	DECANTERS & SPECIALTY PACKAGES	49
12	MISC. AMERICAN CORDIALS & LIQUEURS	CREAM LIQUEURS	44
13	WHISKEY LIQUEUR	MISC. AMERICAN CORDIALS & LIQUEURS	18
14	TENNESSEE WHISKIES	DISTILLED SPIRITS SPECIALTY	8
15	OTHER PROOF VODKA	80 PROOF VODKA	2

Total rows: 15 | Query complete 00:00:00.785

The highest number of mismatches were found in the Decanters & Speciality Packages category.

Alternative Query 2 (needs to be combined with mismatch_counter_table)

```
SELECT sales_vendor, COUNT(mismatch_counter) AS num_of_mismatches
FROM mismatch_counter_table
WHERE mismatch_counter = 'Mismatch'
GROUP BY 1
ORDER BY 2 DESC
```

Data Output 2

	sales_vendor text	num_of_mismatches bigint
1	Sidney Frank Importing Co.	718
2	Moet Hennessy USA Inc.	358
3	Diageo Americas	226
4	U.S. Distilled Prod. Co	184
5	The Patron Spirits Company	162
6	Pernod Ricard USA/Austin Nich…	136
7	Bacardi U.S.A., Inc.	90
8	WILLIAM GRANT AND SONS INC.	74
9	Duggan's Distillers Products Corp	44
10	Broadbent Distillery	18
11	Ole Smoky Distillery LLC	8
12	Blaum Bros. Distilling Co.	2

Total rows: 12 Query complete 00:00:00.821

The highest number of mismatches came from the vendor Sidney Frank Importing Co.

Alternative Query 3 (needs to be combined with mismatch_counter_table)

```
SELECT sales_county, COUNT(mismatch_counter) AS num_of_mismatches
FROM mismatch_counter_table
WHERE mismatch_counter = 'Mismatch'
GROUP BY 1
ORDER BY 2 DESC
```

Data Output 3

	sales_county text	num_of_mismatches bigint
1	Polk	422
2	Scott	183
3	Linn	168
4	Black Hawk	153
5	Johnson	100
6	Story	89
7	Pottawattamie	67
8	Woodbury	53
9	Dickinson	39
10	Des Moines	38
11	Dubuque	36
12	Cerro Gordo	35
13	Jasper	32
14	Bremer	26
15	Muscatine	22

Total rows: 86 Query complete 00:00:00.

The highest number of mismatches came from Polk county.

Alternative Query 4 (needs to be combined with mismatch_counter_table)

```
SELECT sales_store, COUNT(mismatch_counter) AS num_of_mismatches
FROM mismatch_counter_table
WHERE mismatch_counter = 'Mismatch'
GROUP BY 1
ORDER BY 2 DESC
```

Data Output

	sales_store integer	num_of_mismatches bigint
1	2614	38
2	2620	35
3	2666	33
4	2629	32
5	2633	28
6	4604	25
7	4129	24
8	3869	23
9	2500	23
10	2603	19
11	2567	19
12	2572	19
13	2501	19
14	2524	18
15	2651	18

Total rows: 398 Query complete 00:00:0

The highest number of mismatches came from store 2614.

Bonus Q2. Store 2238 (Adventureland Inn at 3200 Adventureland Dr) sold $883.24 in April and $27,526.38 in May, for a 3017% growth rate. That was the highest percentage month-on-month growth rate. Create a query that shows this and the next 9 highest after that.

Query

```
WITH growth_rate_table AS
(
WITH monthly_sales_table AS
(
SELECT store, to_char(date, 'YYYY-MM') AS sale_year_month, SUM(total) AS monthly_sales
FROM sales
WHERE store = 2238
GROUP BY 2, 1
ORDER BY 2
)
SELECT store, sale_year_month, monthly_sales,
        ROUND(100*((monthly_sales-LAG(monthly_sales)OVER(PARTITION BY store ORDER BY sale_year_month))/
        LAG(monthly_sales)OVER(PARTITION BY store ORDER BY sale_year_month)),0) AS growth_rate
FROM monthly_sales_table
ORDER BY 2
)
SELECT sale_year_month, growth_rate
FROM growth_rate_table
WHERE growth_rate IS NOT NULL
ORDER BY 2 DESC
LIMIT 10
```

Data Output

	sale_year_month text	growth_rate numeric
1	2014-05	3017
2	2015-01	465
3	2014-02	77
4	2014-11	17
5	2014-07	14
6	2014-06	-7
7	2014-04	-34
8	2014-10	-41
9	2014-03	-47
10	2015-02	-49

Bonus Q3. The store_address field in the stores table actually contains three rows of text. Quite often the latitude and longitude are in the last line of text. Create a query that shows the geo-locatable stores in latitude order (i.e. show the stores from the most northerly to the most southerly).

Query

```
WITH coordinates AS
(
WITH geolocation_column AS
(
SELECT store, name,
        SUBSTRING(store_address, STRPOS(store_address, '('), STRPOS(store_address, ')')) AS geolocation
FROM stores
)
SELECT store, name,
        TRIM('(' FROM SPLIT_PART(geolocation, ',', 1)) AS latitude,
        TRIM(')' FROM SPLIT_PART(geolocation, ' ', 2)) AS longitude
FROM geolocation_column
WHERE geolocation != ''
)
SELECT store, name, latitude
FROM coordinates
ORDER BY 3 DESC
```

Data Output

	store [PK] integer	name text	latitude text
1	3596	Laddy's Bar And Grill	43.49114240400007
2	3914	Market Street Market	43.45480634900008
3	4030	Buy Rite Foods	43.45480634900008
4	4006	Larchwood Offsale	43.45384534900006
5	4904	Larchwood Quick Stop	43.45355826500003
6	4337	Grand Falls Casino Resort	43.45331135500004
7	4839	DS Services LLC	43.44498284600007
8	3664	Shade Tree Liquors	43.443225246000054
9	3705	Liquor Locker	43.43186091100006
10	3657	Sportsmans Short Stop / Rock Rapi...	43.43104975500006
11	4434	todd's	43.43008068600005
12	2522	Hy-vee Wine and Spirits / Spirit Lak	43.42301360600004

Total rows: 1972 | Query complete 00:00:00.610