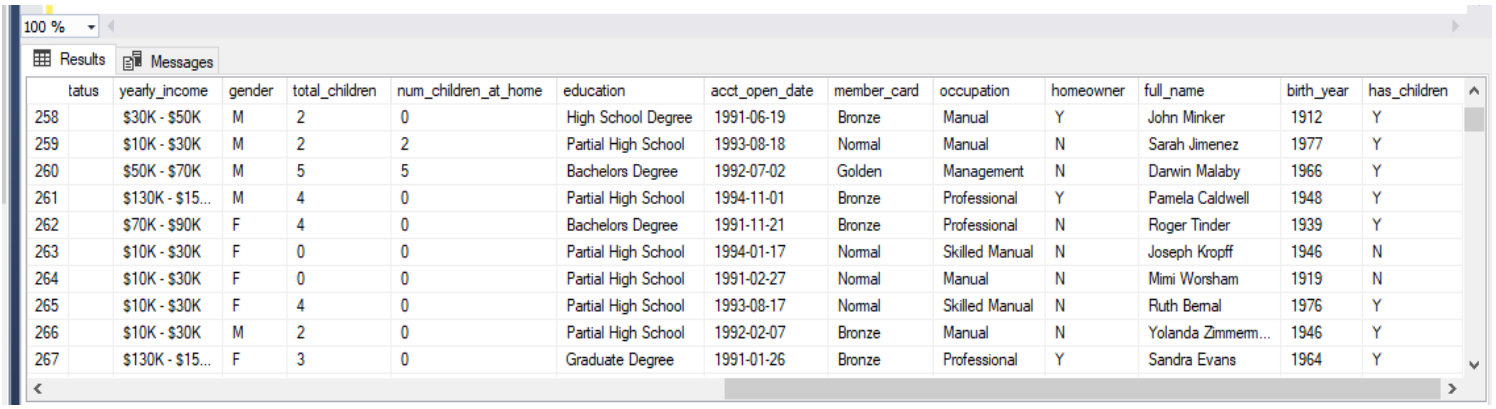
```sql
use Commercial_Project

-- CUSTOMER TABLE
select * from Customers
-- Add a new column named 'full_name' to merge the 'first_name' and 'last_name' columns, separated by a
space

ALTER TABLE Customers
ADD full_name VARCHAR(50)

UPDATE Customers
SET full_name = CONCAT_WS(' ', first_name, last_name)

-- Create a new column named 'birth_year' to extract the year from the 'birthdate' column, and format
as varchar

ALTER TABLE Customers
ADD birth_year VARCHAR(50)

UPDATE Customers
SET birth_year = DATEPART(YEAR, birthdate)

-- Create a conditional column named 'has_children' which equals 'N' if 'total_children' = 0, otherwise
'Y'

ALTER TABLE Customers
ADD has_children VARCHAR(50)

UPDATE Customers
SET has_children =

(SELECT CASE WHEN total_children = 0 THEN 'N'
ELSE 'Y'
END AS has_children)
```



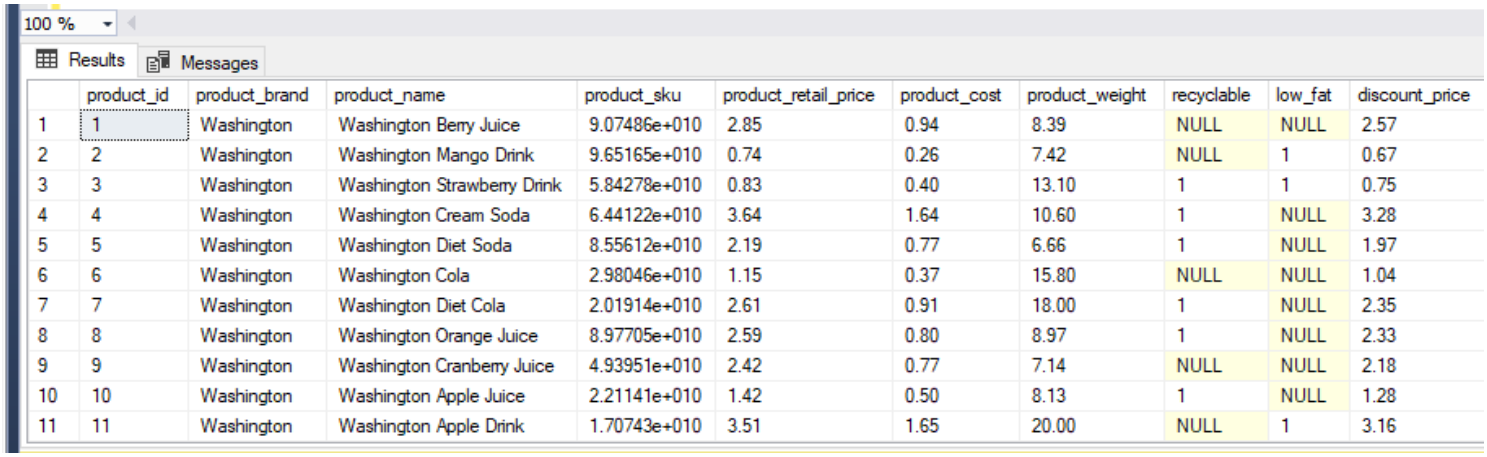
100 %

Results | Messages

	tatus	yearly_income	gender	total_children	num_children_at_home	education	acct_open_date	member_card	occupation	homeowner	full_name	birth_year	has_children
258		$30K - $50K	M	2	0	High School Degree	1991-06-19	Bronze	Manual	Y	John Minker	1912	Y
259		$10K - $30K	M	2	2	Partial High School	1993-08-18	Normal	Manual	N	Sarah Jimenez	1977	Y
260		$50K - $70K	M	5	5	Bachelors Degree	1992-07-02	Golden	Management	N	Darwin Malaby	1966	Y
261		$130K - $15...	M	4	0	Partial High School	1994-11-01	Bronze	Professional	Y	Pamela Caldwell	1948	Y
262		$70K - $90K	F	4	0	Bachelors Degree	1991-11-21	Bronze	Professional	N	Roger Tinder	1939	Y
263		$10K - $30K	F	0	0	Partial High School	1994-01-17	Normal	Skilled Manual	N	Joseph Kropff	1946	N
264		$10K - $30K	F	0	0	Partial High School	1991-02-27	Normal	Manual	N	Mimi Worsham	1919	N
265		$10K - $30K	F	4	0	Partial High School	1993-08-17	Normal	Skilled Manual	N	Ruth Bernal	1976	Y
266		$10K - $30K	M	2	0	Partial High School	1992-02-07	Bronze	Manual	N	Yolanda Zimmerm...	1946	Y
267		$130K - $15...	F	3	0	Graduate Degree	1991-01-26	Bronze	Professional	Y	Sandra Evans	1964	Y

```sql
use Commercial_Project

-- PRODUCTS TABLE
select * from Products

-- Use the statistics tools to return the number of distinct product brands, followed by distinct
product names

select COUNT(DISTINCT product_brand) AS number_of_unique_product_brands,
       COUNT(DISTINCT product_name) AS number_of_unique_product_names
from Products

-- Add a calculated column named 'discount_price', equal to 90% of the original retail price
-- Format as a fixed decimal number, and then use the rounding tool to round to 2 digits
```

```sql
ALTER TABLE products
ADD discount_price DECIMAL(10,2)

UPDATE Products
SET discount_price = CAST((product_retail_price * 0.90) AS DECIMAL(10,2))
```

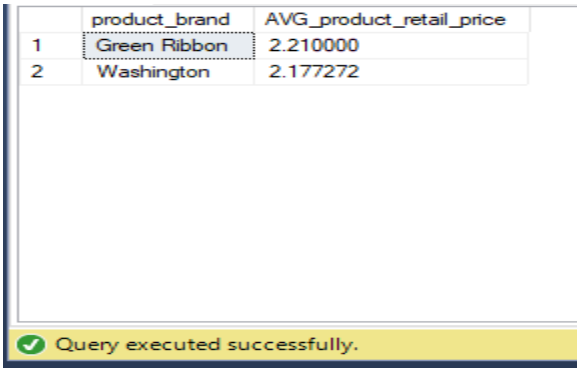


100 %

Results | Messages

	product_id	product_brand	product_name	product_sku	product_retail_price	product_cost	product_weight	recyclable	low_fat	discount_price
1	1	Washington	Washington Berry Juice	9.07486e+010	2.85	0.94	8.39	NULL	NULL	2.57
2	2	Washington	Washington Mango Drink	9.65165e+010	0.74	0.26	7.42	NULL	1	0.67
3	3	Washington	Washington Strawberry Drink	5.84278e+010	0.83	0.40	13.10	1	1	0.75
4	4	Washington	Washington Cream Soda	6.44122e+010	3.64	1.64	10.60	1	NULL	3.28
5	5	Washington	Washington Diet Soda	8.55612e+010	2.19	0.77	6.66	1	NULL	1.97
6	6	Washington	Washington Cola	2.98046e+010	1.15	0.37	15.80	NULL	NULL	1.04
7	7	Washington	Washington Diet Cola	2.01914e+010	2.61	0.91	18.00	1	NULL	2.35
8	8	Washington	Washington Orange Juice	8.97705e+010	2.59	0.80	8.97	1	NULL	2.33
9	9	Washington	Washington Cranberry Juice	4.93951e+010	2.42	0.77	7.14	NULL	NULL	2.18
10	10	Washington	Washington Apple Juice	2.21141e+010	1.42	0.50	8.13	1	NULL	1.28
11	11	Washington	Washington Apple Drink	1.70743e+010	3.51	1.65	20.00	NULL	1	3.16

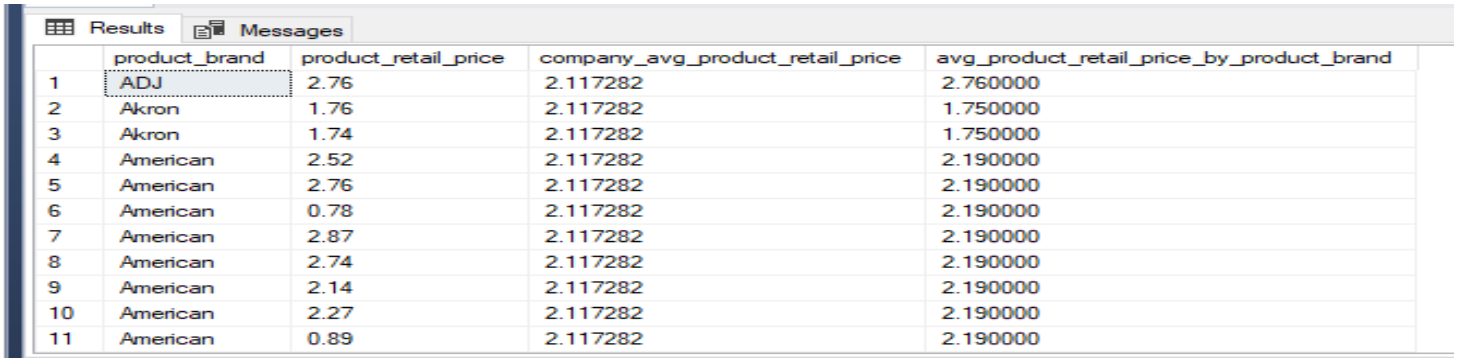
```sql
-- Select "product_brand" and use the Group By option to calculate the average retail price by brand
-- You should see an average retail price of $2.18 for Washington products, and $2.21 for Green Ribbon

select product_brand, AVG(product_retail_price) as AVG_product_retail_price from Products
where product_brand IN ('Washington', 'Green Ribbon')
group by product_brand

-- BY USING A COMMON TABLE EXPRESSION

WITH CTE AS (select product_brand, AVG(product_retail_price) as AVG_product_retail_price from Products
group by product_brand)

select * from CTE
where product_brand IN ('Washington', 'Green Ribbon')
```



	product_brand	AVG_product_retail_price
1	Green Ribbon	2.210000
2	Washington	2.177272

Query executed successfully.

```sql
-- OVER() CLAUSE WINDOW FUNCTION

select product_brand, product_retail_price,
AVG(product_retail_price) OVER() AS company_avg_product_retail_price,
AVG(product_retail_price) OVER(PARTITION BY product_brand) AS avg_product_retail_price_by_product_brand
from Products
```



Results | Messages

	product_brand	product_retail_price	company_avg_product_retail_price	avg_product_retail_price_by_product_brand
1	ADJ	2.76	2.117282	2.760000
2	Akron	1.76	2.117282	1.750000
3	Akron	1.74	2.117282	1.750000
4	American	2.52	2.117282	2.190000
5	American	2.76	2.117282	2.190000
6	American	0.78	2.117282	2.190000
7	American	2.87	2.117282	2.190000
8	American	2.74	2.117282	2.190000
9	American	2.14	2.117282	2.190000
10	American	2.27	2.117282	2.190000
11	American	0.89	2.117282	2.190000

```sql
-- Replace "null" values with zeros in both the "recyclable" and "low-fat" columns

UPDATE Products
SET recyclable = 0
WHERE recyclable IS NULL

UPDATE Products
SET low_fat = REPLACE(low_fat, NULL, 0)


-- STORES TABLE
select * from Stores

-- Add a calculated column named 'full_address', by merging 'store_city', 'store_state', and
'store_country', separated by a comma and space

ALTER TABLE Stores
ADD full_address VARCHAR(50)

UPDATE stores
SET full_address = CONCAT(store_city, ', ', store_state, ', ', store_country)

-- Add a column named "area_code", by extracting the characters before the dash ("-") in the
"store_phone" field

ALTER TABLE stores
ADD area_code VARCHAR(50)

UPDATE stores
SET area_code = SUBSTRING(store_phone, 1, 3)
```

full_address	area_code
Acapulco, Guerrero, Mexico	262
Bellingham, WA, USA	605
Bremerton, WA, USA	509
Camacho, Zacatecas, Mexico	304
Guadalajara, Jalisco, Mexico	801
Beverly Hills, CA, USA	958
Los Angeles, CA, USA	477
Merida, Yucatan, Mexico	797
Mexico City, DF, Mexico	439
Orizaba, Veracruz, Mexico	212

```sql
-- CALENDAR TABLE

select * from Calendar

-- Add these columns: Start of Week (starting Sunday), Name of Day, Start of Month, Name of Month,
Quarter of Year, Year
```

```sql
ALTER TABLE Calendar
ADD Start_Of_Week Date

UPDATE Calendar
SET start_of_week = DATETRUNC(WEEK, Date)

ALTER TABLE Calendar
ADD Name_Of_day VARCHAR(50)

UPDATE Calendar
SET Name_Of_day = DATENAME(DW, Date)

-- OR I could have used the FORMAT function too

UPDATE Calendar
SET Name_Of_day = FORMAT(date, 'dddd')


ALTER TABLE Calendar
ADD Start_Of_Month Date

UPDATE Calendar
SET Start_of_Month = DATETRUNC(MONTH, Date)

ALTER TABLE Calendar
ADD Name_Of_Month Date

ALTER TABLE Calendar
ALTER COLUMN Name_Of_Month VARCHAR(50)

UPDATE Calendar
SET Name_Of_Month = DATENAME(MONTH, Date)

ALTER TABLE Calendar
ADD Quarter_Of_Year INTEGER

UPDATE Calendar
SET Quarter_Of_Year = DATEPART(QUARTER, Date)

ALTER TABLE Calendar
ADD Year INTEGER

UPDATE Calendar
SET Year = DATEPART(YEAR, Date)
```

Results | Messages

	Date	Start_Of_Week	Name_Of_day	Start_Of_Month	Name_Of_Month	Quarter_Of_Year	Year
1	1997-01-01	1996-12-29	Wednesday	1997-01-01	January	1	1997
2	1997-01-02	1996-12-29	Thursday	1997-01-01	January	1	1997
3	1997-01-03	1996-12-29	Friday	1997-01-01	January	1	1997
4	1997-01-04	1996-12-29	Saturday	1997-01-01	January	1	1997
5	1997-01-05	1997-01-05	Sunday	1997-01-01	January	1	1997
6	1997-01-06	1997-01-05	Monday	1997-01-01	January	1	1997
7	1997-01-07	1997-01-05	Tuesday	1997-01-01	January	1	1997
8	1997-01-08	1997-01-05	Wednesday	1997-01-01	January	1	1997
9	1997-01-09	1997-01-05	Thursday	1997-01-01	January	1	1997
10	1997-01-10	1997-01-05	Friday	1997-01-01	January	1	1997
11	1997-01-11	1997-01-05	Saturday	1997-01-01	January	1	1997

Query executed successfully. DESKTOP-BV

```sql
-- Update the date fields e.g. across the Customers table to the "M/d/yyyy" format

select FORMAT(birthdate, 'M/d/yyyy') from Customers
```