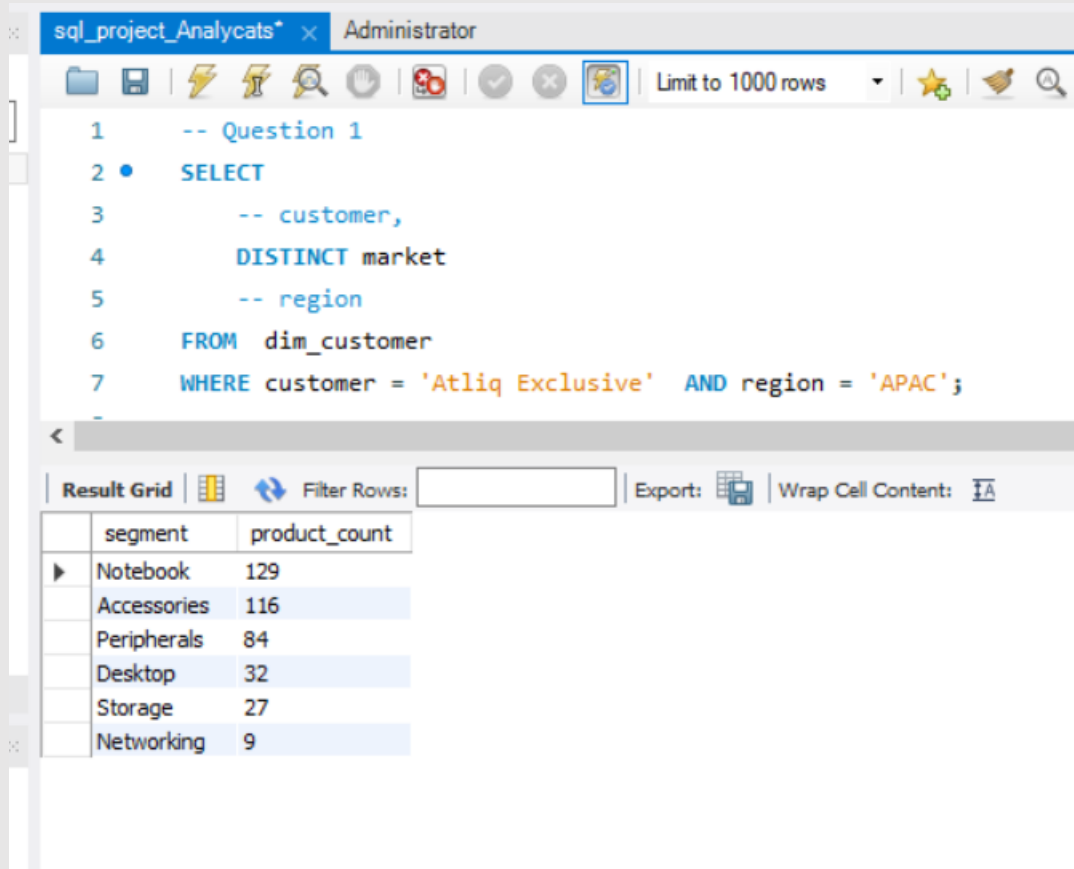# CODEBASICS SQL CHALLENGE SOLUTION

Question1

Provide the list of markets in which customer "Atliq Exclusive" operates its business in the APAC region.

Solution

```
sql_project_Analycats*    Administrator

                                              Limit to 1000 rows
  1      -- Question 1
  2 •    SELECT
  3          -- customer,
  4          DISTINCT market
  5          -- region
  6      FROM  dim_customer
  7      WHERE customer = 'Atliq Exclusive'  AND region = 'APAC';
```

| Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

segment	product_count
Notebook	129
Accessories	116
Peripherals	84
Desktop	32
Storage	27
Networking	9

Question 2.

What is the percentage of unique product increase in 2021 vs. 2020? The final output contains these fields,

- unique_products_2020
- unique_products_2021
- percentage_chg

```
 9        -- Question 2
10  ●  ⊖  SELECT (
11              SELECT
12                  count(distinct product_code)
13              FROM fact_sales_monthly f_2020
14              WHERE fiscal_year = '2020') AS unique_products_2020,
15      ⊖     (
16                SELECT
17                  count(distinct product_code)
18              FROM fact_sales_monthly f_2021
19              WHERE fiscal_year = '2021') AS unique_products_2021,
20              CONCAT(Round ((SELECT unique_products_2021 - unique_products_2020)/(SELECT unique_products_2020)*100,2 ),'%')AS percentage_chg
21      FROM dim_product
22      LIMIT 1
```

Result Grid | 🔡 | ↻ Filter Rows: [   ] | Export: 🔚 | Wrap Cell Content: 🔳 | Fetch rows: 🔽

unique_products_2020	unique_products_2021	percentage_chg
245	334	36.33%

Question 3.

Provide a report with all the unique product counts for each segment and sort them in descending order of product counts. The final output contains 2 fields,

- segment
- product_count

```
Query 1  ×   dim_product        dim_product        fact_gross_price        dim_customer        SQL File 11*

  1          -- Question 3
  2  •       SELECT |
  3              segment,
  4              COUNT(DISTINCT product_code) AS product_count
  5          FROM dim_product
  6          GROUP BY segment
  7          ORDER BY product_count DESC
  8
  9
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

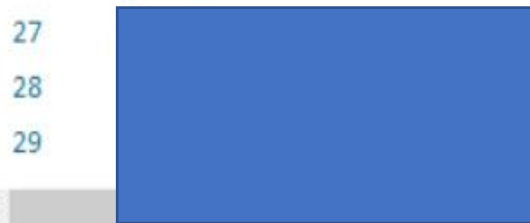| segment	product_count
Notebook	129
Accessories	116
Peripherals	84
Desktop	32
Storage	27
Networking	9

Result 11 ×

Output

QUESTION 4.

Follow-up: Which segment had the most increase in unique products in 2021 vs 2020? The final output contains these fields,

- segment
- product_count_2020
- product_count_2021
- difference

Limit to 1000 rows

```sql
8    -- Quetion 4
9  ⊖ WITH Counts_2020 AS (
10        SELECT
11            segment,
12            COUNT(DISTINCT d.product_code) AS product_count_2020
13        FROM dim_product d
14        JOIN fact_sales_monthly fm USING (product_code)
15        WHERE fiscal_year = 2020
16        GROUP BY segment
17   ),
18 ⊖  Counts_2021 AS (
19        SELECT
20            segment,
21            COUNT(DISTINCT d.product_code) AS product_count_2021
22        FROM dim_product d
23        JOIN fact_sales_monthly fm USING (product_code)
24        WHERE fiscal_year = 2021
25        GROUP BY segment
26   )
27
28
29
```

```
27      SELECT
28              c2.segment,
29              c1.product_count_2020,
30              c2.product_count_2021,
31              c2.product_count_2021 - c1.product_count_2020 AS difference
32      FROM Counts_2020 c1
33      JOIN Counts_2021 c2
34      ON c1.segment = c2.segment
35      ORDER BY difference DESC;
36
```

Result Grid | ▦ Filter Rows: [   ] | Export: ▦ | Wrap Cell Content: ‡A

segment	product_count_2020	product_count_2021	difference
Accessories	69	103	34
Notebook	92	108	16
Peripherals	59	75	16
Desktop	7	22	15
Storage	12	17	5
Networking	6	9	3

Question 5.

Get the products that have the highest and lowest manufacturing costs. The final output should contain these fields,

- product_code
- product
- manufacturing_cost

Limit to 1000 rows

```sql
1    -- Question 5
2    WITH highest_MCOST AS (
3        SELECT
4            product_code,
5            product,
6            MAX(fmc.manufacturing_cost) AS manufacturing_cost
7        FROM fact_manufacturing_cost fmc
8        JOIN dim_product USING (product_code)
9        GROUP BY product_code, product
10       ORDER BY manufacturing_cost DESC
11       LIMIT 1
12   ),
13   lowest_MCOST AS(
14   SELECT
15       product_code,
16       product,
17       MIN(fmc.manufacturing_cost) AS manufacturing_cost
18
19   FROM fact_manufacturing_cost fmc
```

```
18
19        FROM fact_manufacturing_cost fmc
20        JOIN dim_product USING (product_code)
21        GROUP BY product_code, product
22        ORDER BY manufacturing_cost
23        LIMIT 1
24      )
25      SELECT
26          *
27      FROM lowest_MCOST lm
28      UNION
29      SELECT
30          *
31      FROM highest_MCOST hm
32
```

Result Grid | Filter Rows: [   ] | Export: | Wrap Cell Content: ᶦA

product_code	product	manufacturing_cost
A2118150101	AQ Master wired x1 Ms	0.8920
A6120110206	AQ HOME Allin1 Gen 2	240.5364

QUESTION 6.

Generate a report which contains the top 5 customers who received an average high pre_invoice_discount_pct for the fiscal year 2021 and in the Indian market. The final output contains these fields,

- customer_code
- customer
- average_discount_percentage

Limit to 1000 rows

```
 1      -- Question  6
 2
 3 •    SELECT
 4              customer_code,
 5              customer,
 6              ROUND(AVG(pre_invoice_discount_pct),5) AS Average_discount_percentage
 7      FROM fact_pre_invoice_deductions fd
 8      JOIN dim_customer dc USING (customer_code)
 9      WHERE fiscal_year = '2021' AND market = 'India'
10      GROUP BY customer_code,customer
11      ORDER BY Average_discount_percentage DESC
12      LIMIT 5
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

customer_code	customer	Average_discount_percentage
90002009	Flipkart	0.30830
90002006	Viveks	0.30380
90002003	Ezone	0.30280
90002002	Croma	0.30250
90002016	Amazon	0.29330

.

QUESTION 7.

Get the complete report of the Gross sales amount for the customer "Atliq Exclusive" for each month. This analysis helps to get an idea of low and high-performing months and take strategic decisions. The final report contains these columns:
- Month
- Year
- Gross sales Amount

Limit to 1000 rows

```sql
1    -- Question 7
2
3 •  SELECT
4          MONTHNAME(date) AS MONTH,
5          YEAR(date) AS YEAR,
6          SUM(gross_price * sold_quantity) AS `Gross sales Amount`
7    FROM fact_sales_monthly fm
8    LEFT JOIN fact_gross_price fp USING (product_code)
9    LEFT JOIN dim_customer dc USING (customer_code)
10   WHERE customer = 'Atliq Exclusive'
11   GROUP BY MONTH,YEAR
12   ORDER BY YEAR
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: IA

MONTH	YEAR	Gross sales Amount
September	2019	9092670.3392
October	2019	10378637.5961
November	2019	15231894.9669
December	2019	9755795.0577
January	2020	9584951.9393
February	2020	8083995.5479
March	2020	766976.4531
April	2020	800071.9543
May	2020	1586964.4768
June	2020	3429736.5712
July	2020	5151815.4020
August	2020	5638281.8287

MONTH	YEAR	Gross sales Amount
September	2019	9092670.339
October	2019	10378637.6
November	2019	15231894.97
December	2019	9755795.058
January	2020	9584951.939
February	2020	8083995.548
March	2020	766976.4531
April	2020	800071.9543
May	2020	1586964.477
June	2020	3429736.571
July	2020	5151815.402
August	2020	5638281.829
September	2020	19530271.3
October	2020	21016218.21
November	2020	32247289.79
December	2020	20409063.18
January	2021	19570701.71
February	2021	15986603.89
March	2021	19149624.92
April	2021	11483530.3
May	2021	19204309.41
June	2021	15457579.66
July	2021	19044968.82
August	2021	11324548.34

.

QUESTION 8.

In which quarter of 2020, got the maximum total_sold_quantity? The final output contains these fields sorted by the total_sold_quantity,
- Quarter
- total_sold_quantity

```
118
119        -- Question 8
120
121 •    SELECT
122              CONCAT('Q',QUARTER (date)) AS Quarter,
123              SUM(sold_quantity) AS total_sold_quantity
124      FROM fact_sales_monthly fm
125      WHERE YEAR(date) = '2020'
126      GROUP BY Quarter
127      ORDER BY total_sold_quantity DESC
128      LIMIT 1
```

Quarter	total_sold_quantity
Q4	17447125

QUESTION 9.

Which channel helped to bring more gross sales in the fiscal year 2021 and the percentage of contribution? The final output contains these fields,

- Channel
- gross_sales_mln
- percentage

```
 1    -- Question 9
 2  • ⊖ WITH total_sales_by_channel AS (
 3              SELECT
 4                  channel,
 5                  ROUND(SUM(gross_Price * sold_quantity )/1000000,2) AS gross_sales_mln
 6              FROM dim_customer
 7              JOIN fact_sales_monthly fm USING (customer_code)
 8              JOIN fact_gross_price USING (product_code)
 9              WHERE fm.fiscal_year = 2021
10              GROUP BY channel
11      )
12      SELECT
13              channel,
14  ⊖         CONCAT(ROUND(gross_sales_mln / (SELECT SUM(gross_sales_mln)
15                                      FROM total_sales_by_channel) * 100 ,2), '%'
16              ) AS percentage
17      FROM  total_sales_by_channel
18      ORDER BY percentage DESC;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: ⌶A

channel	percentage
Retailer	73.22%
Direct	15.48%
Distributor	11.31%

QUESTION 10.
 Get the Top 3 products in each division that have a high total_sold_quantity in the fiscal_year 2021? The final output contains these fields,
- division
- product_code
- product
- total_sold_quantity
- rank order

```
1      -- Question 10
2  ●  ⊖ WITH total_products_sold AS (
3              select
4                      division,
5                      product_code,
6                      product,
7                      sum(sold_quantity) AS total_sold_quantity,
8              RANK() OVER(PARTITION BY division ORDER BY sum(sold_quantity) DESC) AS rank_order
9              FROM fact_sales_monthly fm
10             JOIN dim_product dp using (product_code)
11             WHERE fiscal_year = 2021
12             GROUP BY product_code,division,product
13     )
14     SELECT *
15     FROM  total_products_sold tpd
16     WHERE rank_order IN (1,2,3)
17
```

Result Grid | Filter Rows: [   ] | Export: | Wrap Cell Content: ‡A

division	product_code	product	total_sold_quantity	rank_order
N & S	A6720160103	AQ Pen Drive 2 IN 1	701373	1
N & S	A6818160202	AQ Pen Drive DRC	688003	2
N & S	A6819160203	AQ Pen Drive DRC	676245	3
P & A	A2319150302	AQ Gamers Ms	428498	1
P & A	A2520150501	AQ Maxima Ms	419865	2
P & A	A2520150504	AQ Maxima Ms	419471	3
PC	A4218110202	AQ Digit	17434	1
PC	A4319110306	AQ Velocity	17280	2
PC	A4218110208	AQ Digit	17275	3