M1B1T1_AG1_19. PROCESAMIENTO DE DATOS CON SPARK 2.X. DATOS DE VENTAS

SERGI SOLÉ BERTRAN Y MARC ERICSSON NAVARRO

1. **Los 10 productos más comprados.**

API

```
+----------+-----+
|product_id|count|
+----------+-----+
|        64|   50|
|        81|   45|
|         3|   44|
|        61|   43|
|        36|   43|
|        60|   43|
|        31|   43|
|        30|   42|
|        69|   42|
|        58|   41|
+----------+-----+
only showing top 10 rows
```

SQL

```
+----------+-------------+
|product_id|Total_vendido|
+----------+-------------+
|        64|           50|
|        81|           45|
|         3|           44|
|        36|           43|
|        60|           43|
|        61|           43|
|        31|           43|
|        30|           42|
|        69|           42|
|         7|           41|
+----------+-------------+
only showing top 10 rows
```

2. **Porcentaje de compra de cada tipo de producto (item_type).**

API

item_type	count	Porcentaje_compra
shirt	650	19.584212112081953
shoe	693	20.87978306718891
trouser	649	19.55408255498644
jean	689	20.75926483880687
jacket	638	19.222657426935825

SQL

item_type	count	Porcentaje_compra
shirt	650	19.584212112081953
shoe	693	20.87978306718891
trouser	649	19.55408255498644
jean	689	20.75926483880687
jacket	638	19.222657426935825

3. **Obtener los 3 productos más comprados por cada tipo de producto.**

API

item_type	product_id	count	row
trouser	23	13	1
trouser	25	12	2
trouser	68	12	3
shoe	85	16	1
shoe	96	15	2
shoe	69	15	3
shirt	3	13	1
shirt	59	12	2
shirt	54	12	3
jean	76	14	1
jean	90	12	2
jean	47	11	3
jacket	12	12	2
jacket	64	12	1
jacket	92	11	3

SQL

```
+---------+----------+-------------+---+
|item_type|product_id|Total_product|row|
+---------+----------+-------------+---+
|  trouser|        23|           13|  1|
|  trouser|        25|           12|  2|
|  trouser|        68|           12|  3|
|     shoe|        85|           16|  1|
|     shoe|        96|           15|  2|
|     shoe|        69|           15|  3|
|    shirt|         3|           13|  1|
|    shirt|        59|           12|  2|
|    shirt|        54|           12|  3|
|     jean|        76|           14|  1|
|     jean|        90|           12|  2|
|     jean|        47|           11|  3|
|   jacket|        12|           12|  2|
|   jacket|        64|           12|  1|
|   jacket|        92|           11|  3|
+---------+----------+-------------+---+
```

4. Obtener los productos que son más caros que la media del precio de los productos.

   API

```
|product_id|        avg(price)|
+----------+------------------+
|        90| 70.28797499999999|
|        95| 65.01621666666668|
|        17|62.274790322580635|
|        68|61.318492307692324|
|        13| 60.19272352941178|
|        42|59.744032000000004|
|        81| 59.31631111111109|
|        94|58.857958064516126|
|         5| 58.51494242424243|
|        18| 57.36564444444444|
|         3|56.245918181818176|
|        31| 55.65601395348836|
|        71|  54.9160303030303|
|        33| 54.63238378378379|
|        97| 54.63164857142858|
|        11| 54.47455405405405|
|        44| 54.41726153846153|
|        27|54.061468000000005|
|        78|53.652037931034485|
|        23| 53.44257714285715|
+----------+------------------+
only showing top 20 rows
```

   SQL

```
+----------+-----------------+
|product_id|    price_average|
+----------+-----------------+
|        90| 70.28797499999999|
|        95| 65.01621666666668|
|        17|62.274790322580635|
|        68|61.318492307692324|
|        13| 60.19272352941178|
|        42|59.744032000000004|
|        81| 59.31631111111109|
|        94|58.857958064516126|
|         5| 58.51494242424243|
|        18| 57.36564444444444|
|         3|56.245918181818176|
|        31| 55.65601395348836|
|        71|  54.9160303030303|
|        33| 54.63238378378379|
|        97| 54.63164857142858|
|        11| 54.47455405405405|
|        44| 54.41726153846153|
|        27|54.061468000000005|
|        78|53.652037931034485|
|        23| 53.44257714285715|
+----------+-----------------+
only showing top 20 rows
```

5. **Indicar la tienda que ha vendido más productos.**

   API

```
+-------+-----+
|shop_id|count|
+-------+-----+
|     69|   47|
+-------+-----+
only showing top 1 row
```

   SQL

```
+-------+----------------+
|shop_id|product_quantity|
+-------+----------------+
|     69|              47|
+-------+----------------+
only showing top 1 row
```

6. **Indicar la tienda que ha facturado más dinero.**

   API

```
+-------+------------------+
|shop_id|       sum(price)|
+-------+------------------+
|     69|2444.8898000000013|
+-------+------------------+
only showing top 1 row
```

SQL

```
+-------+------------------+
|shop_id|   Total_facturado|
+-------+------------------+
|     69|2444.8898000000013|
+-------+------------------+
only showing top 1 row
```

7. **Dividir el mundo en 5 áreas geográficas iguales según la longitud (location.lon) y agregar una columna con el nombre del área geográfica, por ejemplo, "área1", "área2", ...**
    - **¿En qué área se utiliza más PayPal?**

API

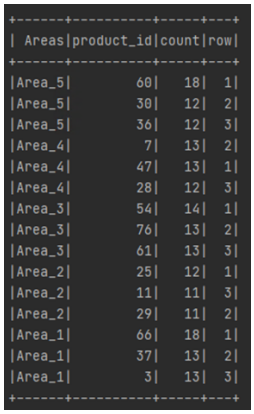
```
+------+-----+
| Areas|count|
+------+-----+
|Area_4|  241|
+------+-----+
only showing top 1 row
```

SQL

```
+------+------------+
| Areas|Total_paypal|
+------+------------+
|Area_4|         241|
+------+------------+
only showing top 1 row
```

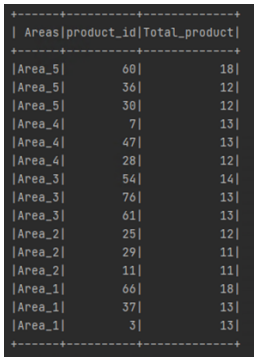
    - **¿Cuáles son los 3 productos más comprados en cada área?**

API



Areas	product_id	count	row
Area_5	60	18	1
Area_5	30	12	2
Area_5	36	12	3
Area_4	7	13	2
Area_4	47	13	1
Area_4	28	12	3
Area_3	54	14	1
Area_3	76	13	2
Area_3	61	13	3
Area_2	25	12	1
Area_2	11	11	3
Area_2	29	11	2
Area_1	66	18	1
Area_1	37	13	2
Area_1	3	13	3

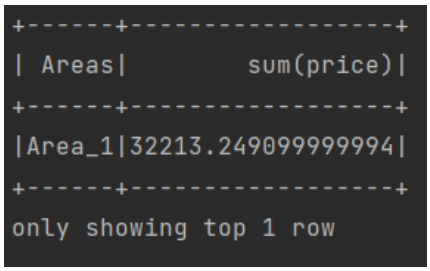
SQL



Areas	product_id	Total_product
Area_5	60	18
Area_5	36	12
Area_5	30	12
Area_4	7	13
Area_4	47	13
Area_4	28	12
Area_3	54	14
Area_3	76	13
Area_3	61	13
Area_2	25	12
Area_2	29	11
Area_2	11	11
Area_1	66	18
Area_1	37	13
Area_1	3	13

- **¿Qué área ha facturado menos dinero?**

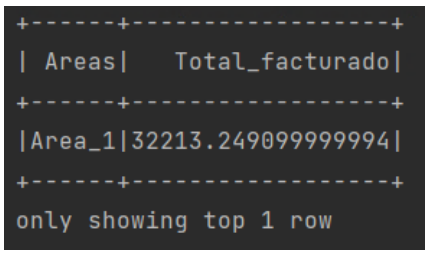
API



Areas	sum(price)
Area_1	32213.249099999994

only showing top 1 row

SQL



Areas	Total_facturado
Area_1	32213.249099999994

only showing top 1 row

8. **Indicar los productos que no tienen stock suficiente para las compras realizadas.**

API

product_id	ventas	quantity	(quantity - ventas)
1	39	34	-5
29	38	25	-13
37	38	22	-16

SQL

product_id	product_sold	stock	Rotura_Stock
1	39	34	-5
29	38	25	-13
37	38	22	-16