SQL Challenge 1

Based on a given dataset, I have created the following queries. The datasets (named Orders, Vendors) are also available in the folder.

1 - List all the orders that were made in Taiwan

```
SELECT * FROM orders WHERE country_name = 'Taiwan';
```

Answer:

Row	rdbms_id	country_name	date_local	vendor_id	customer_id	gmv_local	is_voucher_used	is_successful_order
1	18	Taiwan	2012-10-02T00:00:00	2870	3983	713.7	false	false
2	18	Taiwan	2012-10-02T00:00:00	2870	3010	270.3	false	false
3	18	Taiwan	2012-10-02T00:00:00	3812	3983	551.1	false	false
4	18	Taiwan	2014-10-02T00:00:00	2870	43008	2532.33	false	false
5	18	Taiwan	2014-10-02T00:00:00	3812	42901	332.15	false	false
6	18	Taiwan	2012-10-02T00:00:00	2870	3977	629.2	false	true
7	18	Taiwan	2012-10-02T00:00:00	3812	3524	45.0	false	true
8	18	Taiwan	2014-10-02T00:00:00	3812	15797	1228.5	false	true

2 - Find the Total GMV by country

```
SELECT country_name, SUM(gmv_local) AS total_gmv
FROM orders
GROUP BY country_name
ORDER BY total_gmv DESC;
```

Answer:

Row	country_name	total_gmv
1	Taiwan	17889.23
2	Singapore	3424.75
3	Thailand	1658.26

3 - Find the top active vendor by GMV in each country

```
SELECT o.country_name, vendor_name, SUM(gmv_local) AS total_gmv
FROM orders AS o
INNER JOIN vendors AS v
ON v.id = o.vendor_id
WHERE v.is_active = true
GROUP BY o.country_name, vendor_name
ORDER BY total_gmv DESC, country_name
LIMIT 3;
```

Answer:

Row	country_name	vendor_name	total_gmv
1	Singapore	IC House	1924.96
2	Taiwan	Sweet Chinese Desserts	13491.41
3	Thailand	9th Cafe House	1658.26

4 - Find the top 2 vendors per country, in each year available in the dataset

```
DROP TABLE IF EXISTS temp;
CREATE TEMPORARY TABLE temp AS
SELECT date_trunc ('year', (date_local::date))::date AS year ,
v.country_name AS country_name , vendor_name AS vendor_name ,
SUM(o.gmv_local) AS total_gmv,
ROW_NUMBER () OVER (PARTITION BY v.country_name, date_trunc ('year',
(date_local::date))::date ORDER BY SUM(o.gmv_local) desc) AS POSITION
FROM vendors AS v
INNER JOIN orders AS o
ON v.id = o.vendor_id
GROUP BY YEAR, v.vendor_name , v.country_name
ORDER BY YEAR, country_name, total_gmv desc;

SELECT YEAR, country_name, vendor_name, total_gmv FROM TEMP
WHERE POSITION IN (1,2);
```

Answer:

Row	year	country_name	vendor_name	total_gmv
1	2012-01-01T00:00:00	Singapore	IC House	119.34
2	2012-01-01T00:00:00	Singapore	Canada Paste	89.64
3	2012-01-01T00:00:00	Taiwan	I LOVE PIZZAAAAAAAAAAA	1613.2
4	2012-01-01T00:00:00	Taiwan	Sweet Chinese Desserts	596.1
5	2012-01-01T00:00:00	Thailand	9th Cafe House	1658.26
6	2014-01-01T00:00:00	Singapore	IC House	1805.62
7	2014-01-01T00:00:00	Singapore	WS (BM)	768.83
8	2014-01-01T00:00:00	Taiwan	Sweet Chinese Desserts	12895.31
9	2014-01-01T00:00:00	Taiwan	I LOVE PIZZAAAAAAAAAAA	2784.62

Alternatively, for Problem 4, this is another solution, with a RANK() function.

```
WITH main_calc as(
 SELECT date_part('year', date_local::date) AS YEAR, o.country_name AS country,
 vendor_name AS vendor,
 sum (gmv_local) AS total_gmv
 FROM orders AS o
 LEFT JOIN vendors AS v
 ON  o.vendor_id = v.id
 GROUP BY 1,2,3
 ),


ranked AS (
 SELECT *, RANK() OVER (PARTITION BY year, country ORDER BY total_gmv DESC) AS ranked
 FROM main_calc
 )

 SELECT YEAR, country, vendor, total_gmv
 FROM ranked
 WHERE ranked IN (1,2);
```