SQL QUERIES

/\* Convert Avg.Cost for 2 into USD dollars \*/

SELECT m.RestaurantName, m.Currency, m.Average\_Cost\_for\_two AS LocalCost, c.`USD

Rate`, ROUND(m.Average\_Cost\_for\_two \* c.`USD Rate`, 2) AS CostInUSD

FROM main m JOIN currency c ON m.Currency = c.Currency

WHERE m.Average\_Cost\_for\_two IS NOT NULL;

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # RestaurantName | Currency | LocalCost | USD Rate | CostInUSD |
| Punjabi's Veg Grill | Indian Rupees(Rs.) | 300 | 0.012 | 3.6 |
| Tasty Tandoor | Indian Rupees(Rs.) | 200 | 0.012 | 2.4 |
| Taste of Spice | Indian Rupees(Rs.) | 400 | 0.012 | 4.8 |
| Kolcata Bengali Dhaba | Indian Rupees(Rs.) | 100 | 0.012 | 1.2 |
| Sunil Punjabi Dhaba | Indian Rupees(Rs.) | 150 | 0.012 | 1.8 |
| Bikaner Restaurant | Indian Rupees(Rs.) | 100 | 0.012 | 1.2 |
| Just In | Indian Rupees(Rs.) | 200 | 0.012 | 2.4 |
| Shri Shyam Ji Shudh Shakahari Bhojnalaya | Indian Rupees(Rs.) | 150 | 0.012 | 1.8 |
| Shree Bhojnalaya | Indian Rupees(Rs.) | 100 | 0.012 | 1.2 |
| Baba Ka Dhaba | Indian Rupees(Rs.) | 200 | 0.012 | 2.4 |

/\* No. of restaurants city wise \*/

SELECT City,COUNT(\*) AS NumberOfRestaurants

FROM main GROUP BY City

ORDER BY NumberOfRestaurants DESC limit 5;

|  |  |
| --- | --- |
| # Countryname | NumberOfRestaurants |
| India | 759 |
| United States | 22 |
| United Kingdom | 9 |
| Singapore | 5 |
| New Zealand | 5 |

/\* No. of restaurants country wise \*/

SELECT c.Countryname, COUNT(\*) AS NumberOfRestaurants FROM main m

JOIN country c ON m.CountryCode = c.CountryCode

GROUP BY c.Countryname ORDER BY NumberOfRestaurants DESC limit 5;

|  |  |
| --- | --- |
| Countryname | NumberOfRestaurants |
| India | 759 |
| United States | 22 |
| United Kingdom | 9 |
| Singapore | 5 |
| New Zealand | 5 |

/\* Number of restaurants openings based on year, month and quarter \*/

SELECT `Year Opening` AS Year,COUNT(\*) AS NumRestaurants

FROM main GROUP BY `Year Opening` ORDER BY Year;

|  |  |
| --- | --- |
| # Year | NumRestaurants |
| 2010 | 85 |
| 2011 | 88 |
| 2012 | 89 |
| 2013 | 90 |
| 2014 | 89 |
| 2015 | 87 |
| 2016 | 73 |
| 2017 | 93 |
| 2018 | 107 |

/\* Restaurants by Year and Quarter \*/

SELECT `Year Opening` AS Year, CEIL(`Month Opening` / 3) AS Quarter, COUNT(\*) AS NumRestaurants

FROM main WHERE `Month Opening` IS NOT NULL AND `Year Opening` IS NOT NULL

GROUP BY `Year Opening`, CEIL(`Month Opening` / 3) ORDER BY Year, Quarter limit 10;

|  |  |  |
| --- | --- | --- |
| Year | Quarter | NumRestaurants |
| 2010 | 1 | 27 |
| 2010 | 2 | 15 |
| 2010 | 3 | 23 |
| 2010 | 4 | 20 |
| 2011 | 1 | 14 |
| 2011 | 2 | 27 |
| 2011 | 3 | 31 |
| 2011 | 4 | 16 |
| 2012 | 1 | 22 |
| 2012 | 2 | 15 |

/\* Restaurants by Year and Month \*/

SELECT `Year Opening` AS Year, `Month Opening` AS MonthNumber,

COUNT(\*) AS NumRestaurants FROM main

WHERE `Month Opening` IS NOT NULL AND `Year Opening` IS NOT NULL

GROUP BY `Year Opening`, `Month Opening` ORDER BY Year, MonthNumber limit 10;

|  |  |  |
| --- | --- | --- |
| # Year | MonthNumber | NumRestaurants |
| 2010 | 1 | 8 |
| 2010 | 2 | 12 |
| 2010 | 3 | 7 |
| 2010 | 4 | 3 |
| 2010 | 5 | 5 |
| 2010 | 6 | 7 |
| 2010 | 7 | 2 |
| 2010 | 8 | 10 |
| 2010 | 9 | 11 |
| 2010 | 10 | 4 |

/\* Restaurants based on Avg. Rating 0–2 (Poor) 2–3.5 (Average)

3.5–4.5 (Good) 4.5–5 (Excellent)\*/

SELECT CASE

WHEN Rating < 2 THEN 'Poor'

WHEN Rating < 3.5 THEN 'Average'

WHEN Rating < 4.5 THEN 'Good'

ELSE 'Excellent'

END AS RatingCategory,

COUNT(\*) AS NumRestaurants FROM main

WHERE Rating IS NOT NULL

GROUP BY RatingCategory

ORDER BY FIELD(RatingCategory, 'Poor', 'Average', 'Good', 'Excellent');

|  |  |
| --- | --- |
| RatingCategory | NumRestaurants |
| Poor | 714 |
| Average | 23 |
| Good | 57 |
| Excellent | 7 |

/\* Create buckets based on Avg. price of reasonable size and find count of restaurants \*/

SELECT CASE

WHEN Average\_Cost\_for\_two <= 250 THEN '₹0 - ₹250'

WHEN Average\_Cost\_for\_two <= 500 THEN '₹251 - ₹500'

WHEN Average\_Cost\_for\_two <= 1000 THEN '₹501 - ₹1000'

WHEN Average\_Cost\_for\_two <= 2000 THEN '₹1001 - ₹2000'

ELSE '₹2001+' END AS PriceBucket,

COUNT(\*) AS NumRestaurants FROM main

WHERE Average\_Cost\_for\_two IS NOT NULL

GROUP BY PriceBucket ORDER BY MIN(Average\_Cost\_for\_two);

|  |  |
| --- | --- |
| PriceBucket | NumRestaurants |
| ₹0 - ₹250 | 379 |
| ₹251 - ₹500 | 332 |
| ₹501 - ₹1000 | 89 |
| ₹1001 - ₹2000 | 1 |

/\* Percentage of restaurants based on table booking \*/

SELECT Has\_Table\_booking, COUNT(\*) AS NumRestaurants,

ROUND(100.0 \* COUNT(\*) / (SELECT COUNT(\*) FROM main

WHERE Has\_Table\_booking IS NOT NULL), 2) AS Percentage

FROM main WHERE Has\_Table\_booking IS NOT NULL

GROUP BY Has\_Table\_booking;

|  |  |  |
| --- | --- | --- |
| # Has\_Table\_booking | NumRestaurants | Percentage |
| No | 777 | 97.00 |
| Yes | 24 | 3.00 |

/\* Percentage of Restaurants based on online delivery \*/

SELECT Has\_Online\_delivery, COUNT(\*) AS NumRestaurants,

ROUND(100.0 \* COUNT(\*) / (SELECT COUNT(\*) FROM main WHERE Has\_Online\_delivery IS NOT NULL),2) AS Percentage

FROM main WHERE Has\_Online\_delivery IS NOT NULL

GROUP BY Has\_Online\_delivery;

|  |  |  |
| --- | --- | --- |
| # Has\_Online\_delivery | NumRestaurants | Percentage |
| No | 753 | 94.01 |
| Yes | 48 | 5.99 |

/\* creation of calendar table \*/

SELECT DISTINCT

STR\_TO\_DATE(CONCAT(`Year Opening`, '-', `Month Opening`, '-', `Day Opening`), '%Y-%m-%d') AS DateKey\_Opening,

-- (a) Year

`Year Opening` AS Year,

-- (b) Month Number (1–12)

`Month Opening` AS MonthNo,

-- (c) Month Name

MONTHNAME(STR\_TO\_DATE(CONCAT(`Year Opening`, '-', `Month Opening`, '-', `Day Opening`), '%Y-%m-%d')) AS MonthName,

-- (d) Calendar Quarter

CONCAT('Q', QUARTER(STR\_TO\_DATE(CONCAT(`Year Opening`, '-', `Month Opening`, '-', `Day Opening`), '%Y-%m-%d'))) AS Quarter,

-- (e) Year-Month (e.g., 2022-August)

DATE\_FORMAT(STR\_TO\_DATE(CONCAT(`Year Opening`, '-', `Month Opening`, '-', `Day Opening`), '%Y-%m-%d'), '%Y-%M') AS YearMonth,

-- (f) Weekday Number (1 = Sunday)

DAYOFWEEK(STR\_TO\_DATE(CONCAT(`Year Opening`, '-', `Month Opening`, '-', `Day Opening`), '%Y-%m-%d')) AS WeekdayNo,

-- (g) Weekday Name

DAYNAME(STR\_TO\_DATE(CONCAT(`Year Opening`, '-', `Month Opening`, '-', `Day Opening`), '%Y-%m-%d')) AS WeekdayName,

-- (h) Financial Month (April = FM-1 ... March = FM-12)

CONCAT('FM-', CASE

WHEN `Month Opening` >= 4 THEN `Month Opening` - 3

ELSE `Month Opening` + 9

END ) AS FinancialMonth,

-- (i) Financial Quarter (FQ1 = Apr–Jun, etc.)

CONCAT('FQ-', CASE

WHEN `Month Opening` BETWEEN 4 AND 6 THEN 1

WHEN `Month Opening` BETWEEN 7 AND 9 THEN 2

WHEN `Month Opening` BETWEEN 10 AND 12 THEN 3

ELSE 4 END ) AS FinancialQuarter

FROM main WHERE `Year Opening` IS NOT NULL

AND `Month Opening` IS NOT NULL AND `Day Opening` IS NOT NULL

ORDER BY DateKey\_Opening limit 10;

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| # DateKey\_Opening | Year | MonthNo | MonthName | Quarter | YearMonth | WeekdayNo | WeekdayName | FinancialMonth | FinancialQuarter |
| 2010-01-04 | 2010 | 1 | January | Q1 | 2010-January | 2 | Monday | FM-10 | FQ-4 |
| 2010-01-06 | 2010 | 1 | January | Q1 | 2010-January | 4 | Wednesday | FM-10 | FQ-4 |
| 2010-01-14 | 2010 | 1 | January | Q1 | 2010-January | 5 | Thursday | FM-10 | FQ-4 |
| 2010-01-16 | 2010 | 1 | January | Q1 | 2010-January | 7 | Saturday | FM-10 | FQ-4 |
| 2010-01-17 | 2010 | 1 | January | Q1 | 2010-January | 1 | Sunday | FM-10 | FQ-4 |
| 2010-01-18 | 2010 | 1 | January | Q1 | 2010-January | 2 | Monday | FM-10 | FQ-4 |
| 2010-01-19 | 2010 | 1 | January | Q1 | 2010-January | 3 | Tuesday | FM-10 | FQ-4 |
| 2010-01-24 | 2010 | 1 | January | Q1 | 2010-January | 1 | Sunday | FM-10 | FQ-4 |
| 2010-02-02 | 2010 | 2 | February | Q1 | 2010-February | 3 | Tuesday | FM-11 | FQ-4 |
| 2010-02-09 | 2010 | 2 | February | Q1 | 2010-February | 3 | Tuesday | FM-11 | FQ-4 |

/\* Count Restaurants by Cusines \*/

SELECT Cuisines, COUNT(\*) AS RestaurantCount

FROM main GROUP BY Cuisines

ORDER BY RestaurantCount DESC limit 5;

|  |  |
| --- | --- |
| # Cuisines | RestaurantCount |
| North Indian | 174 |
| Chinese | 46 |
| Fast Food | 43 |
| Bakery | 28 |
| Mughlai | 27 |

/\* Top 10 Cusines with most restaurants \*/

SELECT Cuisines, COUNT(\*) AS RestaurantCount

FROM main

GROUP BY Cuisines

ORDER BY RestaurantCount DESC

LIMIT 10;

|  |  |
| --- | --- |
| # Cuisines | RestaurantCount |
| North Indian | 174 |
| Chinese | 46 |
| Fast Food | 43 |
| Bakery | 28 |
| Mughlai | 27 |
| Bakery, Desserts | 26 |
| North Indian, Chinese | 23 |
| Street Food | 23 |
| Chinese, Fast Food | 22 |
| North Indian, Mughlai | 18 |

/\* Best-rated restaurant(s) in each city \*/

SELECT m.City, m.RestaurantName, m.Rating, m.Votes

FROM main m

JOIN ( SELECT City, MAX(Rating) AS MaxRating FROM main

GROUP BY City ) AS t

ON m.City = t.City AND m.Rating = t.MaxRating

ORDER BY m.City limit 8;

|  |  |  |  |
| --- | --- | --- | --- |
| # City | RestaurantName | Rating | Votes |
| Albany | Henry Campbell's Steakhouse | 4 | 51 |
| Auckland | Tucks and Bao | 4 | 86 |
| Birmingham | Lasan Restaurant | 4 | 213 |
| Boise | Chandlers Steakhouse | 4 | 379 |
| Boise | Barbacoa Restaurant | 4 | 538 |
| Davenport | Duck City Bistro | 4 | 201 |
| Des Moines | Tursi's Latin King | 4 | 476 |
| Dubuque | Woodfire Grille | 4 | 72 |