

- Q1. Write a code to check NULL values
Q2. If NULL values are present, update them with zeros for all columns.

```
MariaDB [(none)]> use project1;
Database changed
MariaDB [project1]> show tables;
+-----+
| Tables_in_project1 |
+-----+
| daily_reports      |
+-----+
1 row in set (0.001 sec)

MariaDB [project1]> SELECT *
-> FROM daily_reports
-> WHERE Province IS NULL
->      OR Country_Region IS NULL
->      OR Latitude IS NULL
->      OR Longitude IS NULL
->      OR Date IS NULL
->      OR Confirmed IS NULL
->      OR Deaths IS NULL
->      OR Recovered IS NULL;
Empty set (0.105 sec)

MariaDB [project1]> UPDATE daily_reports
-> SET
->     Province = COALESCE(Province, ''),
->     Country_Region = COALESCE(Country_Region, ''),
->     Latitude = COALESCE(Latitude, 0),
->     Longitude = COALESCE(Longitude, 0),
->     Date = COALESCE(Date, '0000-00-00'),
->     Confirmed = COALESCE(Confirmed, 0),
->     Deaths = COALESCE(Deaths, 0),
->     Recovered = COALESCE(Recovered, 0);
Query OK, 0 rows affected (0.160 sec)
Rows matched: 78386  Changed: 0  Warnings: 0
```

- Q3. check total number of rows
 Q4. Check what is start_date and end_date
 Q5. Number of month present in dataset

```
MariaDB [project1]> SELECT COUNT(*) AS total_rows
-> FROM daily_reports;
+-----+
| total_rows |
+-----+
|      78386 |
+-----+
1 row in set (0.009 sec)
```

```
MariaDB [project1]> SELECT
->     MIN(Date) AS start_date,
->     MAX(Date) AS end_date
-> FROM daily_reports;
+-----+-----+
| start_date | end_date |
+-----+-----+
| 2020-01-22 | 2021-06-13 |
+-----+-----+
1 row in set (0.013 sec)
```

```
MariaDB [project1]> SELECT COUNT(DISTINCT DATE_FORMAT(Date, '%Y-%m')) AS number_of_months
-> FROM daily_reports;
+-----+
| number_of_months |
+-----+
|             18 |
+-----+
1 row in set (0.024 sec)
```

- Q6. Find monthly average for confirmed, deaths, recovered

```
MariaDB [project1]> SELECT
->     DATE_FORMAT(Date, '%Y-%m') AS month,
->     AVG(Confirmed) AS avg_confirmed,
->     AVG(Deaths) AS avg_deaths,
->     AVG(Recovered) AS avg_recovered
-> FROM daily_reports
-> GROUP BY DATE_FORMAT(Date, '%Y-%m')
-> ORDER BY month;
+-----+-----+-----+-----+
| month   | avg_confirmed | avg_deaths | avg_recovered |
+-----+-----+-----+-----+
| 2020-01 | 4.1455 | 0.1234 | 0.0929 |
| 2020-02 | 15.2960 | 0.5936 | 7.0320 |
| 2020-03 | 161.1303 | 8.6607 | 27.8739 |
| 2020-04 | 505.8004 | 41.5223 | 171.6422 |
| 2020-05 | 574.8498 | 30.2809 | 318.2964 |
| 2020-06 | 859.2281 | 29.8175 | 548.7916 |
| 2020-07 | 1432.3611 | 35.1096 | 983.0582 |
| 2020-08 | 1611.8429 | 37.5367 | 1299.2947 |
| 2020-09 | 1784.5874 | 34.7773 | 1438.9067 |
| 2020-10 | 2412.1996 | 36.7583 | 1420.6431 |
| 2020-11 | 3592.1944 | 56.7634 | 1985.3446 |
| 2020-12 | 4050.4397 | 71.2183 | 2497.8850 |
| 2021-01 | 3911.2285 | 84.1837 | 1919.6370 |
| 2021-02 | 2433.3636 | 69.1649 | 1558.3917 |
| 2021-03 | 2916.7972 | 59.1998 | 1652.2859 |
| 2021-04 | 4699.3552 | 78.4387 | 3074.7851 |
| 2021-05 | 4005.2541 | 76.7803 | 4007.5078 |
| 2021-06 | 2508.6324 | 66.2622 | 2769.4496 |
+-----+-----+-----+-----+
18 rows in set (0.065 sec)
```

Q7. Find most frequent value for confirmed, deaths, recovered each month

```
MariaDB [project1]> SELECT
-> MONTH(Date) AS Month,
-> YEAR(Date) AS Year,
-> SUBSTRING_INDEX(GROUP_CONCAT(Confirmed ORDER BY Confirmed DESC),',',1) AS Most_frequent_confirmed,
-> SUBSTRING_INDEX(GROUP_CONCAT(Deaths ORDER BY Deaths DESC),',',1) AS Most_frequent_deaths,
-> SUBSTRING_INDEX(GROUP_CONCAT(Recovered ORDER BY Recovered DESC),',',1) AS Most_frequent_recovered
-> FROM
-> daily_reports
-> GROUP BY
-> YEAR(Date),MONTH(Date)
-> ORDER BY
-> YEAR(Date),MONTH(Date);
```

Month	Year	Most_frequent_confirmed	Most_frequent_deaths	Most_frequent_recovered
1	2020	2131	49	51
2	2020	14840	242	3418
3	2020	26314	1085	4289
4	2020	50740	2607	33227
5	2020	34907	2309	51717
6	2020	54771	2003	94305
7	2020	75866	1595	140050
8	2020	85687	1505	95881
9	2020	97894	1703	101468
10	2020	99264	3351	388340
11	2020	207933	2259	139292
12	2020	823225	3752	1123456
1	2021	300462	4475	87090
2	2021	134975	3907	98389
3	2021	100158	3869	102138
4	2021	401993	4249	299988
5	2021	414188	4529	422436
6	2021	134154	7374	231456

18 rows in set (0.139 sec)

Q8. Find minimum values for confirmed, deaths, recovered per year

Q9. Find maximum values of confirmed, deaths, recovered per year

```
MariaDB [project1]> SELECT
->     YEAR(Date) AS year,
->     MIN(Confirmed) AS min_confirmed,
->     MIN(Deaths) AS min_deaths,
->     MIN(Recovered) AS min_recovered
-> FROM daily_reports
-> GROUP BY YEAR(Date);
```

year	min_confirmed	min_deaths	min_recovered
2020	0	0	0
2021	0	0	0

2 rows in set (0.022 sec)

```
MariaDB [project1]> SELECT
->     YEAR(Date) AS year,
->     MAX(Confirmed) AS max_confirmed,
->     MAX(Deaths) AS max_deaths,
->     MAX(Recovered) AS max_recovered
-> FROM daily_reports
-> GROUP BY YEAR(Date);
```

year	max_confirmed	max_deaths	max_recovered
2020	823225	3752	1123456
2021	414188	7374	422436

2 rows in set (0.023 sec)

Q10. The total number of case of confirmed, deaths, recovered each month

```
MariaDB [project1]> SELECT
->     DATE_FORMAT(Date, '%Y-%m') AS month,
->     SUM(Confirmed) AS total_confirmed,
->     SUM(Deaths) AS total_deaths,
->     SUM(Recovered) AS total_recovered
-> FROM daily_reports
-> GROUP BY DATE_FORMAT(Date, '%Y-%m');
```

month	total_confirmed	total_deaths	total_recovered
2020-01	6384	190	143
2020-02	68312	2651	31405
2020-03	769236	41346	133070
2020-04	2336798	191833	792987
2020-05	2744333	144561	1519547
2020-06	3969634	137757	2535417
2020-07	6838092	167613	4693120
2020-08	7694938	179200	6202833
2020-09	8244794	160671	6647749
2020-10	11515841	175484	6782150
2020-11	16595938	262247	9172292
2020-12	19336799	339996	11924903
2021-01	18672205	401893	9164347
2021-02	10492664	298239	6719785
2021-03	13924790	282620	7888013
2021-04	21711021	362387	14205507
2021-05	19121083	366549	19131842
2021-06	5022282	132657	5544438

18 rows in set (0.066 sec)

Q11. Check how corona virus spread out with respect to confirmed case
(Eg.: total confirmed cases, their average, variance & STDEV)

```
MariaDB [project1]> -- Calculate total confirmed cases
MariaDB [project1]> SELECT
    ->     COUNT(Confirmed) AS total_confirmed_cases
    -> FROM
    ->     daily_reports;
+-----+
| total_confirmed_cases |
+-----+
|           78386 |
+-----+
1 row in set (0.010 sec)
```

```
MariaDB [project1]>
MariaDB [project1]> -- Calculate average of confirmed cases
MariaDB [project1]> SELECT
    ->     AVG(Confirmed) AS average_confirmed_cases
    -> FROM
    ->     daily_reports;
+-----+
| average_confirmed_cases |
+-----+
|          2156.8283 |
+-----+
1 row in set (0.011 sec)
```

```
MariaDB [project1]>
MariaDB [project1]> -- Calculate variance of confirmed cases
MariaDB [project1]> SELECT
    ->     VARIANCE(Confirmed) AS variance_confirmed_cases
    -> FROM
    ->     daily_reports;
+-----+
| variance_confirmed_cases |
+-----+
|       157288925.0780 |
+-----+
1 row in set (0.009 sec)
```

```
MariaDB [project1]>
MariaDB [project1]> -- Calculate standard deviation of confirmed cases
MariaDB [project1]> SELECT
    ->     STDDEV(Confirmed) AS std_dev_confirmed_cases
    -> FROM
    ->     daily_reports;
+-----+
| std_dev_confirmed_cases |
+-----+
|          12541.4882 |
+-----+
1 row in set (0.009 sec)
```

Q12. Check how corona virus spread out with respect to death case per month

(Eg.: total confirmed cases, their average, variance & STDEV)

```
MariaDB [project1]> -- Calculate total death cases per month
```

```
MariaDB [project1]> SELECT
->     DATE_FORMAT(Date, '%Y-%m') AS month,
->     COUNT(Deaths) AS total_death_cases
-> FROM
->     daily_reports
-> GROUP BY
->     DATE_FORMAT(Date, '%Y-%m');
```

month	total_death_cases
2020-01	1540
2020-02	4466
2020-03	4774
2020-04	4620
2020-05	4774
2020-06	4620
2020-07	4774
2020-08	4774
2020-09	4620
2020-10	4774
2020-11	4620
2020-12	4774
2021-01	4774
2021-02	4312
2021-03	4774
2021-04	4620
2021-05	4774
2021-06	2002

```
18 rows in set (0.048 sec)
```

```
MariaDB [project1]>
```

```
MariaDB [project1]> -- Calculate average of death cases per month
```

```
MariaDB [project1]> SELECT
->     AVG(Deaths) AS average_death_cases
-> FROM
->     daily_reports;
```

average_death_cases
46.5376

```
1 row in set (0.013 sec)
```

```
MariaDB [project1]>
```

```
MariaDB [project1]> -- Calculate variance of death cases per month
```

```
MariaDB [project1]> SELECT
->     VARIANCE(Deaths) AS variance_death_cases
-> FROM
->     daily_reports;
```

variance_death_cases
45892.0189

```
1 row in set (0.009 sec)
```

```
MariaDB [project1]>
```

```
MariaDB [project1]> -- Calculate standard deviation of death cases per month
```

```
MariaDB [project1]> SELECT
->     STDDEV(Deaths) AS std_dev_death_cases
-> FROM
->     daily_reports;
```

std_dev_death_cases
214.2242

```
1 row in set (0.009 sec)
```

Q13. Check how corona virus spread out with respect to recovered case
(Eg.: total confirmed cases, their average, variance & STDEV)

```
MariaDB [project1]> -- Calculate total recovered cases
MariaDB [project1]> SELECT
  ->     COUNT(Recovered) AS total_recovered_cases
  -> FROM
  ->     daily_reports;
+-----+
| total_recovered_cases |
+-----+
|           78386      |
+-----+
1 row in set (0.010 sec)
```

```
MariaDB [project1]>
MariaDB [project1]> -- Calculate average of recovered cases
MariaDB [project1]> SELECT
  ->     AVG(Recovered) AS average_recovered_cases
  -> FROM
  ->     daily_reports;
+-----+
| average_recovered_cases |
+-----+
|          1442.7264      |
+-----+
1 row in set (0.011 sec)
```

```
MariaDB [project1]>
MariaDB [project1]> -- Calculate variance of recovered cases
MariaDB [project1]> SELECT
  ->     VARIANCE(Recovered) AS variance_recovered_cases
  -> FROM
  ->     daily_reports;
+-----+
| variance_recovered_cases |
+-----+
|        107029523.2623    |
+-----+
1 row in set (0.009 sec)
```

```
MariaDB [project1]>
MariaDB [project1]> -- Calculate standard deviation of recovered cases
MariaDB [project1]> SELECT
  ->     STDDEV(Recovered) AS std_dev_recovered_cases
  -> FROM
  ->     daily_reports;
+-----+
| std_dev_recovered_cases |
+-----+
|          10345.5074      |
+-----+
1 row in set (0.009 sec)
```


- Q14. Find Country having highest number of the Confirmed case
Q15. Find Country having lowest number of the death case

```
MariaDB [project1]> SELECT
->     Country_Region,
->     SUM(Confirmed) AS total_confirmed_cases
-> FROM
->     daily_reports
-> GROUP BY
->     Country_Region
-> ORDER BY
->     total_confirmed_cases DESC
-> LIMIT 1;
```

```
+-----+-----+
| Country_Region | total_confirmed_cases |
+-----+-----+
| US             |          33461982    |
+-----+-----+
1 row in set (0.037 sec)
```

```
MariaDB [project1]> SELECT
->     Country_Region,
->     SUM(Deaths) AS total_death_cases
-> FROM
->     daily_reports
-> GROUP BY
->     Country_Region
-> ORDER BY
->     total_death_cases ASC
-> LIMIT 1;
```

```
+-----+-----+
| Country_Region | total_death_cases |
+-----+-----+
| Kiribati       |          0         |
+-----+-----+
1 row in set (0.035 sec)
```

Q16. Find top 5 countries having highest recovered case

```
MariaDB [project1]> SELECT
->     Country_Region,
->     SUM(Recovered) AS total_recovered_cases
-> FROM
->     daily_reports
-> GROUP BY
->     Country_Region
-> ORDER BY
->     total_recovered_cases DESC
-> LIMIT 5;
```

Country_Region	total_recovered_cases
India	28089649
Brazil	15400169
US	6303715
Turkey	5202251
Russia	4745756

5 rows in set (0.037 sec)