Deforestation SQL Queries

SQL Queries Used Global Situation

```
CREATE VIEW Forestation AS
SELECT F.country_code, l.country_name,
                r.region, f.year, f.forest_area_sqkm, l.total_area_sq_mi,
                (f.forest_area_sqkm/(l.total_area_sq_mi*2.59))*100 as forest_percent ,
        r.income_group FROM forest_area f
        JOIN land_area l
        ON f.country_code = l.country_code AND f.year = l.year
        JOIN regions r
       ON f.country_code = r.country_code AND;
```

1.

```
SELECT year, region, forest_area_sqkm as total_forest_area
        FROM Forestation
        WHERE year=1990 AND region='World'
```

2.

```
SELECT year, region, forest_area_sqkm as total_forest_area
        FROM Forestation
        WHERE year=2016 AND region='World'
```

3.

```
DROP VIEW IF EXISTS Forestation cascade;

        CREATE VIEW Forestation AS
                SELECT F.country_code, l.country_name,
                        r.region, f.year, f.forest_area_sqkm, l.total_area_sq_mi,
                        (f.forest_area_sqkm/(l.total_area_sq_mi*2.59))*100 as forest_percent ,
                        r.income_group
                FROM forest_area f
                  JOIN land_area l
                    ON f.country_code = l.country_code AND
                                f.year=l.year
                  JOIN regions r
                    ON f.country_code = r.country_code;

        WITH forest_world_2016 AS(
                SELECT region, forest_area_sqkm AS new_forest
                  FROM forestation
                        WHERE year=2016 and region='World'),
forest_world_1990 AS(
                SELECT region, forest_area_sqkm AS old_forest
```

```
                FROM forestation
               WHERE year=1990 and region='World')

SELECT region, (new_forest - old_forest) AS forest_world_change
  FROM forest_world_2016
    JOIN forest_world_1990 USING(region);
```

answer: 1324449 Km lost from 1990 to 2016

4. DROP VIEW IF EXISTS Forestation cascade;

CREATE VIEW Forestation AS

```
SELECT F.country_code, l.country_name, r.region, f.year, f.forest_area_sqkm,
                l.total_area_sq_mi,
               (f.forest_area_sqkm/(l.total_area_sq_mi*2.59))*100 as forest_percent ,
                r.income_group
        FROM forest_area f
          JOIN land_area l
            ON f.country_code = l.country_code AND
        f.year=l.year
          JOIN regions r
            ON f.country_code = r.country_code;

WITH forest_world_2016 AS(
        SELECT region, forest_area_sqkm AS new_forest
                FROM forestation
                WHERE year=2016 and region='World'),

        forest_world_1990 AS(
        SELECT region, forest_area_sqkm AS old_forest
                FROM forestation
                WHERE year=1990 and region='World')
SELECT region, ((new_forest-old_forest)/old_forest)*100 AS forest_world_change
FROM forest_world_2016
JOIN forest_world_1990 USING(region);
```

answers: 3.2 % lost from 1990 to 2016

5.
```
SELECT year, region,COUNTRY_name, total_area_sq_mi*2.59 as total_area_km
        FROM Forestation
        WHERE year=2016 AND total_area_sq_mi*2.59 < 1324449
        ORDER BY total_area_sq_mi*2.59 desc;
```

Answer: Peru with 1279999.9891 km; a bit larger than PERU

Regional Outlook

```
CREATE VIEW REGIONAL_OUTLOOK AS
SELECT region,year,
        sum(forest_area_sqkm)/sum(total_area_sq_mi *2.59) as forest_percent
        FROM Forestation
        WHERE year= 1990 or year=2016
        GROUP BY region,year
        ORDER BY region,year DESC
```

1.

```
SELECT region,ROUND(forest_percent::numeric,2) AS world_forest_percent
        FROM regional_outlook
        WHERE year=2016 AND region='World';
```

1.

```
SELECT region,ROUND(MAX(forest_percent)::NUMERIC,2) AS max_forest_percent
        FROM regional_outlook
        WHERE year=2016
        GROUP BY region
        ORDER BY ROUND (MAX (forest_percent)::NUMERIC,2) DESC LIMIT
        1;
```

1.

```
SELECT region,ROUND(MIN(forest_percent)::NUMERIC,2) AS  min_forest_percent
        FROM regional_outlook
        WHERE year=2016
        GROUP BY region
        ORDER BY ROUND (MIN (forest_percent)::NUMERIC,2)
        LIMIT 1;
```

Answers: HIGHEST: Latin America & Caribbean 0.51 percent
LOWEST: Middle East & North Africa 0.02 percent
WORLD: 0.31 PERCENT

2.

Answers: HIGHEST: Latin America & Caribbean 0.46 percent
LOWEST: Middle East & North Africa 0.02 percent
WORLD: 0.33 PERCENT

3.

Latin America & Caribbean, South Asia, Sub-Saharan Africa

Country Level Detail:

```
DROP VIEW IF EXISTS Forestation cascade;

CREATE VIEW Forestation AS
        SELECT F.country_code, l.country_name, r.region,
                f.year, f.forest_area_sqkm, l.total_area_sq_mi,
                (f.forest_area_sqkm/(l.total_area_sq_mi*2.59))*100 as forest_percent ,
                r.income_group FROM forest_area f
                JOIN land_area l
                ON f.country_code = l.country_code AND
                f.year = l.year
                JOIN regions r
                ON f.country_code = r.country_code;

DROP VIEW IF EXISTS REGIONAL_OUTLOOK;

CREATE VIEW regional_outlook AS
        SELECT region,year,
                Sum(forest_area_sqkm)/sum(total_area_sq_mi *2.59) as forest_percent
                FROM Forestation
                WHERE year= 1990 or year=2016
                GROUP BY region,year
                ORDER BY region,year DESC;

WITH forest_area_2016 AS(
        SELECT country_name,year, forest_area_sqkm AS new_forest
                FROM forestation
                WHERE year=2016),

        forest_area_1990 AS(
        SELECT country_name,year, forest_area_sqkm AS old_forest
                FROM forestation
                WHERE year=1990)
SELECT country_name, (new_forest - old_forest) AS forest_area_change
        FROM forest_area_2016
        JOIN forest_area_1990 USING(country_name)
        ORDER BY (new_forest - old_forest) LIMIT
        6;
```

Table 3.1

```
DROP VIEW IF EXISTS Forestation cascade;

CREATE VIEW Forestation AS
        SELECT F.country_code, l.country_name, r.region, f.year, f.forest_area_sqkm,
                l.total_area_sq_mi,
                (f.forest_area_sqkm/(l.total_area_sq_mi*2.59))*100 as forest_percent ,
                r.income_group FROM forest_area f
                JOIN land_area l
                ON   f.country_code   =   l.country_code   AND
                        f.year = l.year
                JOIN regions r
                ON f.country_code = r.country_code;

WITH forest_area_2016 AS(
        SELECT country_name,region, year,
forest_area_sqkm AS new_forest
        FROM forestation
        WHERE year=2016),

        forest_area_1990 AS(
         SELECT country_name, region, year,
forest_area_sqkm AS old_forest
                FROM forestation
                WHERE year=1990)

SELECT country_name, region,
                ABS(new_forest-old_forest) AS forest_abs_change FROM
        forest_area_2016
        JOIN forest_area_1990 USING(country_name,region)
        ORDER BY ABS(new_forest-old_forest) DESC LIMIT
        100;
```

Table 3.2

```
DROP VIEW IF EXISTS Forestation cascade;

CREATE VIEW Forestation AS
        SELECT F.country_code, l.country_name,
                 r.region, f.year, f.forest_area_sqkm, l.total_area_sq_mi,
                (f.forest_area_sqkm/(l.total_area_sq_mi*2.59))*100 as forest_percent ,
                r.income_group FROM forest_area f
                JOIN land_area l
                ON   f.country_code   =   l.country_code   AND
                        f.year = l.year
                JOIN regions r
                ON f.country_code = r.country_code;

WITH forest_area_2016 AS(
        SELECT country_name,region, year, forest_area_sqkm AS new_forest
                FROM forestation
                WHERE year=2016),

        forest_area_1990 AS(
        SELECT country_name, region, year, forest_area_sqkm AS old_forest
                FROM forestation
                WHERE year=1990)
SELECT country_name, region, ((new_forest-old_forest)/old_forest) AS forest_percent_change
        FROM forest_area_2016
        JOIN forest_area_1990 USING(country_name,region)
        where ((new_forest-old_forest)/old_forest)<0
        ORDER BY ((new_forest-old_forest)/old_forest);
```

QUARTILE

1.

```
DROP VIEW IF EXISTS Forestation cascade;

CREATE VIEW Forestation AS
SELECT F.country_code, l.country_name,
                r.region, f.year, f.forest_area_sqkm, l.total_area_sq_mi,
                (f.forest_area_sqkm/(l.total_area_sq_mi*2.59))*100 as forest_percent ,
                r.income_group
        FROM forest_area f
        JOIN land_area l
        ON f.country_code = l.country_code AND
                f.year = l.year
        JOIN regions r
       ON f.country_code = r.country_code;

WITH forest_percent_2016 AS(
        SELECT country_name,region, forest_percent
                FROM forestation
                WHERE year=2016
                ORDER BY forest_percent DESC),

  num_quartile AS(
        SELECT country_name,forest_percent,
        CASE
                WHEN forest_percent/25 <= 1 THEN 'First Quartile'
                WHEN forest_percent/25 <= 2 THEN 'Second Quartile'
                WHEN forest_percent/25 <= 3 THEN 'Third Quartile'
                WHEN forest_percent/25 <= 4 THEN 'Fourth Quartile'
                ELSE 'Null'
        END AS quartile
        FROM forest_percent_2016)

        SELECT country_name, forest_percent
                FROM num_quartile
                WHERE quartile = 'Fourth Quartile';

        SELECT count(country_name)
                FROM num_quartile
                WHERE quartile = 'Fourth Quartile' ;
```