Damir Sudarevic <damir.sudarevic@gmail.com>

Below is a SQL solution to the challenge of June 2019 (Map Coloring with Violations).

The txt below can be copied and pasted into Postgres or SQL Server.

Result: There are 12 solutions

BE	DK	FR	DE	LU	NL	cost
B	B	G	R	G	G	257
B	B	R	G	R	R	257
B	G	G	R	G	G	257
B	R	R	G	R	R	257
G	B	B	R	B	B	257
G	G	B	R	B	B	257
G	G	R	B	R	R	257
G	R	R	B	R	R	257
R	B	B	G	B	B	257
R	G	G	B	G	G	257
R	R	B	G	B	B	257
R	R	G	B	G	G	257

```
--      File: map_coloring_with_violations.sql
--      Version: 1.1
--     Last Changed: 2020-02-27
--          by: Damir;  https://www.damirsystems.com
--      Project: Map Coloring With Violations
--  Description: Fun challenge from Decision Management Community
--              https://dmcommunity.org/challenge/challenge-june-2019/
--          DB: PostgreSQL, MS SQL Server


WITH
color AS ( -- Red, Green, Blue
   SELECT p FROM (
      VALUES ('R'), ('G'), ('B')
   ) AS x(p)
),
possible AS ( -- generate 4^6 = 729 possible combinations
   SELECT
      a.p AS BE  -- Belgium
    , b.p AS DK  -- Denmark
    , c.p AS FR  -- France
    , d.p AS DE  -- Germany
    , e.p AS LU  -- Luxembourg
```

```
     , f.p AS NL  -- Netherlands
   FROM      color AS a
   CROSS JOIN color AS b
   CROSS JOIN color AS c
   CROSS JOIN color AS d
   CROSS JOIN color AS e
   CROSS JOIN color AS f
),
q_00 AS ( -- apply mandatory constraints
   SELECT BE, DK, FR, DE, LU, NL
   FROM possible

   WHERE (1=1) -- mandatory constraints

   --  BE borders FR, DE, NL, LU; LU allowed with penalty
   AND BE NOT IN (FR, DE, NL)

   --  DK borders DE
   AND DK NOT IN (DE)

   --  FR borders BE, DE, LU; LU allowed with penalty
   AND FR NOT IN (BE, DE)

   --  DE borders FR, BE, NL, DK, LU; LU allowed with penalty
   AND DE NOT IN (FR, BE, NL, DK)

   --  LU borders FR, DE, BE; all are allowed with penalty

   --  NL borders BE, DE
   AND NL NOT IN (BE, DE)
),
q_01 AS ( -- calculate penalty cost
   SELECT BE, DK, FR, DE, LU, NL,
        (case when (LU = FR) then 257 else 0 end)
      + (case when (LU = DE) then 904 else 0 end)
      + (case when (LU = BE) then 568 else 0 end) AS cost

   FROM q_00
),
q_02 AS ( -- find minimum penalty cost
   SELECT min(cost) AS min_cost
   FROM q_01
)
-- select all solutions with minimum cost
SELECT BE, DK, FR, DE, LU, NL, cost
FROM q_01 AS a
JOIN q_02 AS b ON a.cost = b.min_cost
```

```
-- order results
ORDER BY BE, DK, FR, DE, LU, NL
;
```