



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ
ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Βάσεων Δεδομένων
6ο Εξάμηνο, Ακαδημαϊκή περίοδος 2023-2024

Αναφορά
Εξαμηνιαίας Εργασίας

Διδάσκοντες: Δημήτριος Τσουμάκος, Μάριος Κόνιαρης

Σιμραντζίτ Σινγκ Ντούλκου AM:03121105

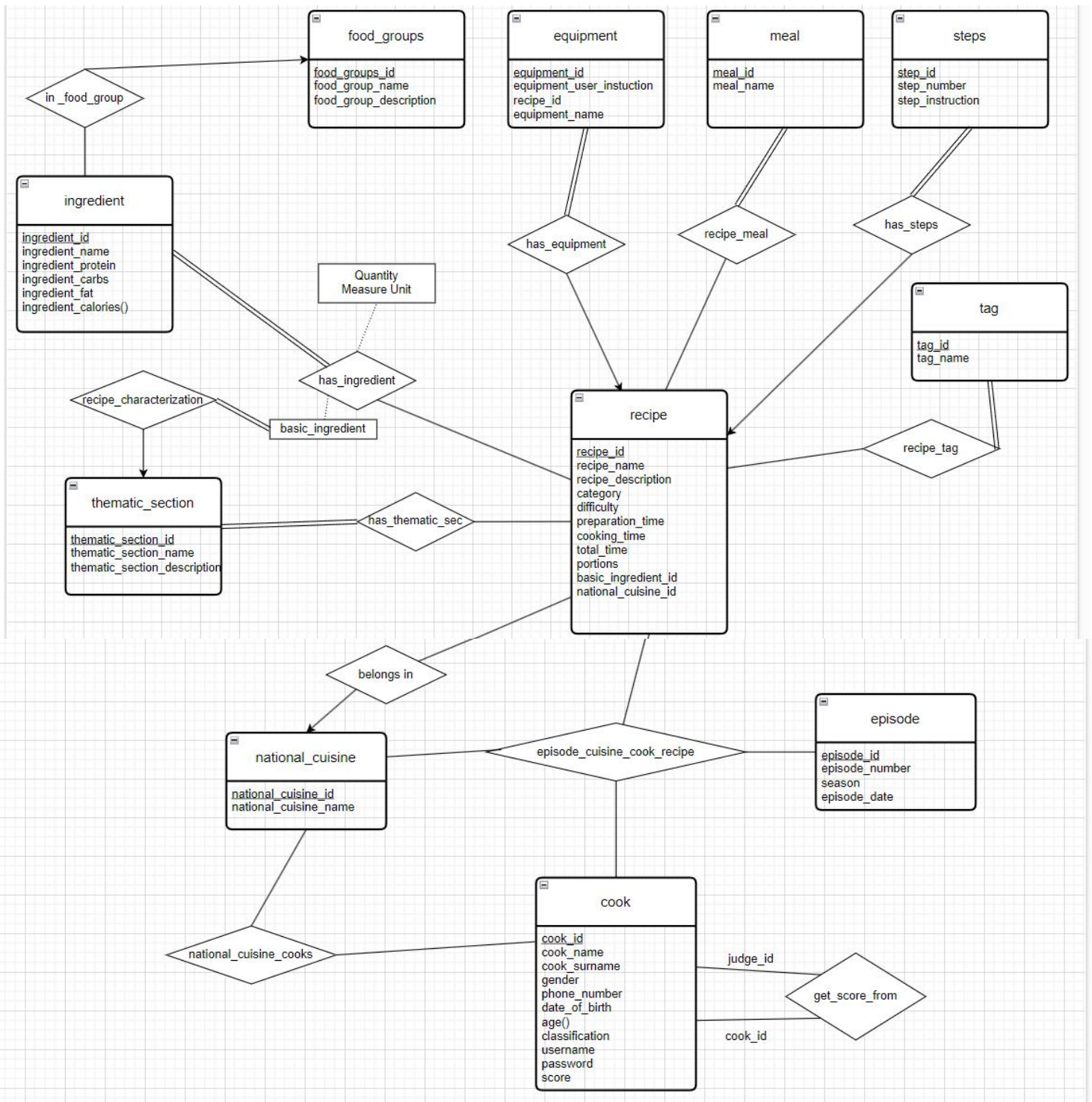
Αναστάσιος Κασιδόκωστας AM:03120834

Βασίλης Κοτρώνης AM:03121432

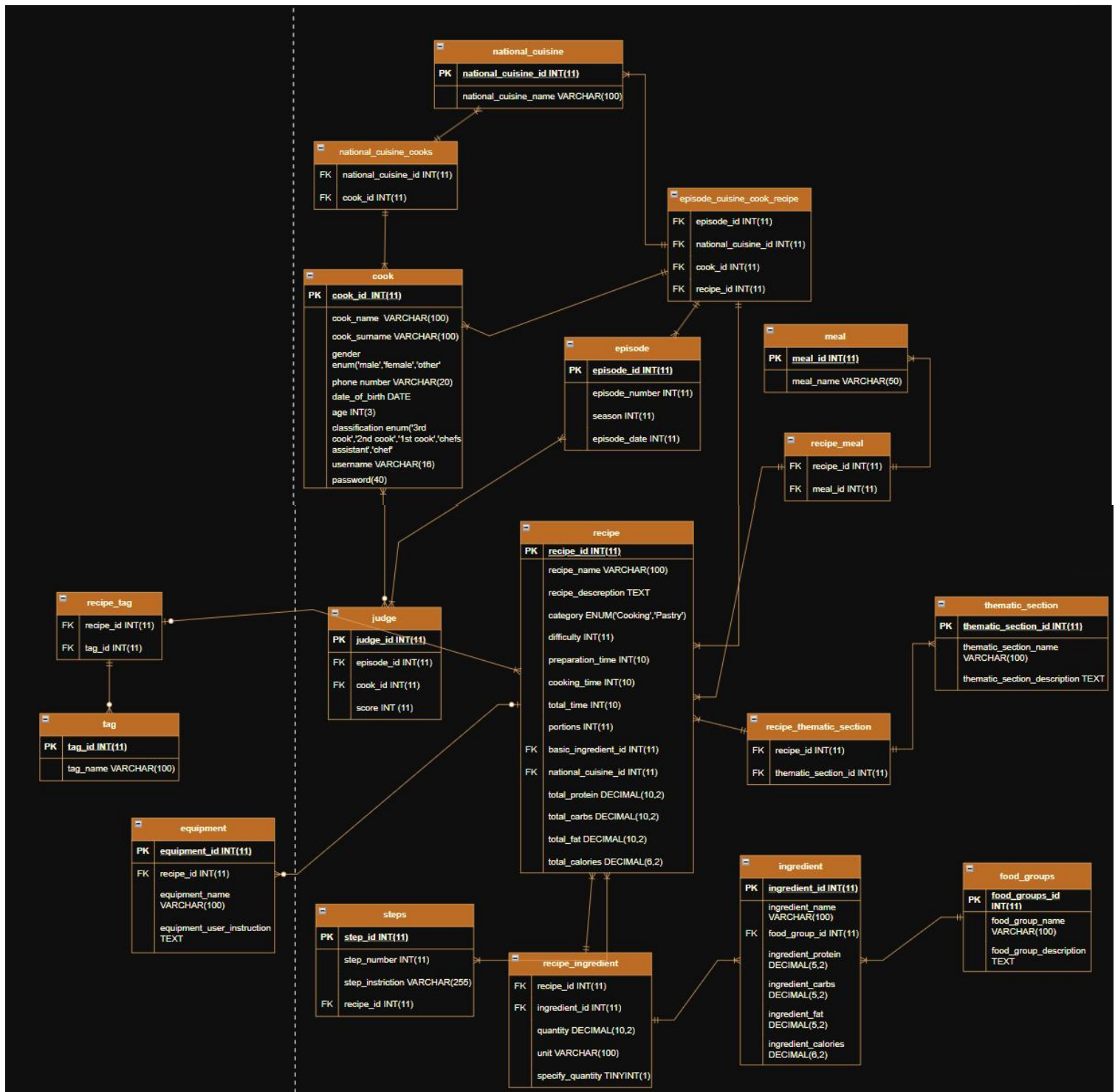
<https://github.com/dulkusim/masterchef-DB>

Άσκηση 1

ER Diagram



Relational Diagram



Άσκηση 2

- Τα DDL scripts βρίσκονται στα αρχεία: masterchef_drop_schema.sql και masterchef_create_schema.sql
- Το DML script βρίσκεται στο αρχείο: masterchef_insert_schema.sql

https://github.com/dulkusim/masterchef-DB/blob/main/SQL%20Codes/masterchef_create_schema.sql

https://github.com/dulkusim/masterchef-DB/blob/main/SQL%20Codes/masterchef_drop_schema.sql

https://github.com/dulkusim/masterchef-DB/blob/main/SQL%20Codes/masterchef_insert_schema.sql

Άσκηση 2.1

Περιορισμοί:

- Ένας μάγειρας είναι μεταξύ της ηλικίας των 18 και 90.
- Ένας κριτής δεν μπορεί να συμμετέχει ως μάγειρας στο ίδιο έτος.
- Τα γραμμάρια των πρωτεινών, υδατανθράκων και λιπιδίων είναι ≥ 0 ώστε να γίνεται σωστά ο υπολογισμός των θερμίδων.
- Κάθε νέο επεισόδιο πρέπει να έχει ημερομηνία μετάδοσης μεταγενέστερη από το τελευταίο επεισόδιο.
- Σε ένα επεισόδιο δεν μπορεί να συμμετέχει ο ίδιος μάγειρας και η ίδια εθνική κουζίνα και η ίδια συνταγή πάνω από μια φορά.
- Δεν μπορεί κάποιος μάγειρας/κριτής/ εθνική κουζίνα/ συνταγή να συμμετέχει συνεχόμενα σε περισσότερα από 3 επεισόδια.
- Το βασικό υλικό μιας συνταγής πρέπει να ανήκει στα υλικά της συνταγής.
- Οι χρόνοι μαγειρικής και προετοιμασίας δεν μπορεί να είναι αρνητικοί.
- Όταν ενημερώνεται ένα υλικό πρέπει να ενημερώνονται και οι διατροφικές αξίες των συνταγών στα οποία χρησιμοποιείται.

Άσκηση 2.2

Indexes χρησιμοποιήθηκαν μόνο σε primary keys και στα foreign keys όλων των πινάκων.

Άσκηση 3

3.1

Μέση Βαθμολογία καθε μάγειρα:

```
SELECT c.cook_id, c.cook_name, c.cook_surname, AVG(j.score) AS average_score
FROM cook c
JOIN judge j ON c.cook_id = j.cook_id
GROUP BY c.cook_id, c.cook_name, c.cook_surname;
```

Μέση Βαθμολογία ανά εθνική κουζίνα:

```
SELECT nc.national_cuisine_id, nc.national_cuisine_name, AVG(j.score) AS average_score
FROM judge j
JOIN national_cuisine_cooks cc ON j.cook_id = cc.cook_id
JOIN episode_cuisine_cook_recipe ce ON j.episode_id = ce.episode_id AND
cc.national_cuisine_id = ce.national_cuisine_id
JOIN national_cuisine nc ON nc.national_cuisine_id = cc.national_cuisine_id
GROUP BY nc.national_cuisine_id, nc.national_cuisine_name;
```

3.2

Σημείωση: Οι μάγειρες που ανήκουν σε μία κουζίνα δεν αλλάζουν με το έτος.

Οπότε: Ποιοί μάγειρες ανήκουν π.χ στην κουζίνα με id 4

```
SELECT c.cook_id, CONCAT(c.cook_name, ' ', c.cook_surname) AS cook_fullname
FROM cook c
JOIN national_cuisine_cooks ncc ON c.cook_id = ncc.cook_id
WHERE ncc.national_cuisine_id = 4;
```

Ποιοί μάγειρες συμμετείχαν σε επεισόδια από την κουζίνα με id=4 στο 2ο έτος

```
SELECT DISTINCT eccr.cook_id, CONCAT(c.cook_name, ' ', c.cook_surname) AS
cook_fullname, e.season
FROM episode_cuisine_cook_recipe eccr
JOIN episode e ON eccr.episode_id = e.episode_id
JOIN cook c ON eccr.cook_id = c.cook_id
WHERE eccr.national_cuisine_id = 4
AND e.season = 2;
```

3.3

Νέοι μάγειρες (ηλικία < 30 ετών) που έχουν τις περισσότερες συνταγές.

SELECT

```
c.cook_id, CONCAT(c.cook_name, ' ', c.cook_surname) AS cook_fullname,  
c.age, COUNT(eccr.recipe_id) AS recipe_count
```

FROM

```
cook c
```

JOIN

```
episode_cuisine_cook_recipe eccr ON c.cook_id = eccr.cook_id
```

WHERE

```
c.age < 30
```

GROUP BY

```
c.cook_id, c.cook_name, c.cook_surname, c.age
```

ORDER BY

```
recipe_count DESC;
```

3.4

Ποιοί μάγειρες δεν έχουν συμμετάσχει ποτέ ως κριτές

```
SELECT c.cook_id, CONCAT(c.cook_name, ' ', c.cook_surname) AS cook_fullname
```

```
FROM cook c
```

```
WHERE c.cook_id NOT IN (SELECT DISTINCT judge_id FROM judge);
```

3.5

Σημείωση: Επειδή έχουμε 100 μάγειρες δεν είναι πολύ πιθανό να συμμετείχαν πάνω από 3 φορές ως κριτές σε μία σεζόν, για αυτό το όριο έγινε πάνω από 1 φορά.

Ποιοι κριτές έχουν συμμετάσχει στον ίδιο αριθμό επεισοδίων σε διάστημα ενός έτους με περισσότερες από 1 εμφανίσεις;

SELECT

```
j1.judge_id AS judge1_id,  
j2.judge_id AS judge2_id,  
j1.season,  
j1.episode_count
```

FROM

```
(
```

SELECT

```
j.judge_id,  
e.season,  
COUNT(DISTINCT j.episode_id) AS episode_count
```

FROM

```
judge j
```



```

        JOIN
            episode e ON j.episode_id = e.episode_id
        GROUP BY
            j.judge_id, e.season
        HAVING
            COUNT(DISTINCT j.episode_id) > 1
    ) j1
JOIN
    (
        SELECT
            j.judge_id,
            e.season,
            COUNT(DISTINCT j.episode_id) AS episode_count
        FROM
            judge j
        JOIN
            episode e ON j.episode_id = e.episode_id
        GROUP BY
            j.judge_id, e.season
        HAVING
            COUNT(DISTINCT j.episode_id) > 1
    ) j2 ON j1.season = j2.season AND j1.episode_count = j2.episode_count AND
j1.judge_id < j2.judge_id
ORDER BY
    j1.season, j1.episode_count DESC, j1.judge_id, j2.judge_id;

```

3.6

Top-3 ζεύγη ετικετών που εμφανίστηκαν σε επεισόδια

```

SELECT
    t1.tag_id AS tag1_id,
    tag1.tag_name AS tag1_name,
    t2.tag_id AS tag2_id,
    tag2.tag_name AS tag2_name,
    COUNT(DISTINCT e.episode_id) AS episode_count
FROM
    recipe_tag t1
JOIN
    recipe_tag t2 ON t1.recipe_id = t2.recipe_id AND t1.tag_id < t2.tag_id
JOIN
    episode_cuisine_cook_recipe e ON t1.recipe_id = e.recipe_id
JOIN
    tag tag1 ON t1.tag_id = tag1.tag_id
JOIN
    tag tag2 ON t2.tag_id = tag2.tag_id
GROUP BY
    t1.tag_id, t2.tag_id, tag1.tag_name, tag2.tag_name
ORDER BY
    episode_count DESC
LIMIT 3;

```

3.7

Βρείτε όλους τους μάγειρες που συμμετείχαν τουλάχιστον 5 λιγότερες φορές από τον μάγειρα με τις περισσότερες συμμετοχές σε επεισόδια.

```
SELECT DISTINCT c.cook_id, CONCAT(c.cook_name, ' ', c.cook_surname) AS cook_fullname,
ec.appearances
FROM (
    SELECT cook_id, COUNT(*) AS appearances
    FROM episode_cuisine_cook_recipe
    GROUP BY cook_id
) AS ec
CROSS JOIN (
    SELECT MAX(appearances) AS max_appearances
    FROM (
        SELECT COUNT(*) AS appearances
        FROM episode_cuisine_cook_recipe
        GROUP BY cook_id
    ) AS max_count
) AS max_cook
JOIN cook c ON ec.cook_id = c.cook_id
WHERE max_cook.max_appearances - ec.appearances >= 5;
```

3.9

Λίστα με μέσο όρο αριθμού γραμμάρων υδατανθράκων στο διαγωνισμό ανά έτος

```
SELECT
    e.season,
    avg(r.total_carbs) AS average_carb_grams
FROM
    episode_cuisine_cook_recipe eccr
JOIN
    recipe r ON eccr.recipe_id = r.recipe_id
JOIN
    episode e ON eccr.episode_id = e.episode_id
GROUP BY
    e.season;
```


3.10

Σημείωση: Έγινε η υπόθεση πως το ερώτημα εννοεί σε κάθε μια από τις δύο σεζόν οι Εθνικές κουζίνες να έχουν τον ίδιο αριθμό συμμετοχών, π.χ να έχουν και οι δύο 3 συμμετοχές στην πρώτη σεζόν και να έχουν και οι δύο 5 συμμετοχές στην δεύτερη σεζόν.

Ποιες Εθνικές κουζίνες έχουν τον ίδιο αριθμό συμμετοχών σε διαγωνισμούς, σε διάστημα δύο συνεχόμενων ετών, με τουλάχιστον 3 συμμετοχές ετησίως.

```
SELECT
    sp1.cuisine1_id,
    sp1.cuisine2_id,
    sp1.appearance_count1 AS appearance_count_season1,
    sp1.season1,
    sp2.appearance_count1 AS appearance_count_season2,
    sp2.season1 AS season2
FROM (
    SELECT
        sa1.national_cuisine_id AS cuisine1_id,
        sa2.national_cuisine_id AS cuisine2_id,
        sa1.season AS season1,
        sa1.appearance_count AS appearance_count1
    FROM (
        SELECT
            eccr.national_cuisine_id,
            e.season,
            COUNT(eccr.episode_id) AS appearance_count
        FROM episode_cuisine_cook_recipe eccr
        JOIN episode e ON eccr.episode_id = e.episode_id
        GROUP BY eccr.national_cuisine_id, e.season
        HAVING COUNT(eccr.episode_id) >= 3
    ) sa1
    JOIN (
        SELECT
            eccr.national_cuisine_id,
            e.season,
            COUNT(eccr.episode_id) AS appearance_count
        FROM episode_cuisine_cook_recipe eccr
        JOIN episode e ON eccr.episode_id = e.episode_id
        GROUP BY eccr.national_cuisine_id, e.season
        HAVING COUNT(eccr.episode_id) >= 3
    ) sa2
    ON sa1.season = sa2.season
    AND sa1.appearance_count = sa2.appearance_count
    AND sa1.national_cuisine_id < sa2.national_cuisine_id
) sp1
JOIN (
    SELECT
        sa1.national_cuisine_id AS cuisine1_id,
```

```

sa2.national_cuisine_id AS cuisine2_id,
sa1.season AS season1,
sa1.appearance_count AS appearance_count1
FROM (
    SELECT
        eccr.national_cuisine_id,
        e.season,
        COUNT(eccr.episode_id) AS appearance_count
    FROM episode_cuisine_cook_recipe eccr
    JOIN episode e ON eccr.episode_id = e.episode_id
    GROUP BY eccr.national_cuisine_id, e.season
    HAVING COUNT(eccr.episode_id) >= 3
) sa1
JOIN (
    SELECT
        eccr.national_cuisine_id,
        e.season,
        COUNT(eccr.episode_id) AS appearance_count
    FROM episode_cuisine_cook_recipe eccr
    JOIN episode e ON eccr.episode_id = e.episode_id
    GROUP BY eccr.national_cuisine_id, e.season
    HAVING COUNT(eccr.episode_id) >= 3
) sa2
ON sa1.season = sa2.season
AND sa1.appearance_count = sa2.appearance_count
AND sa1.national_cuisine_id < sa2.national_cuisine_id
) sp2
ON sp1.cuisine1_id = sp2.cuisine1_id
AND sp1.cuisine2_id = sp2.cuisine2_id
AND sp1.season1 + 1 = sp2.season1
ORDER BY sp1.cuisine1_id, sp1.cuisine2_id, sp1.season1;

```

3.11

Βρείτε τους top-5 κριτές που έχουν δώσει συνολικά την υψηλότερη βαθμολόγηση σε ένα μάγειρα

```

SELECT
    judge.judge_id,
    CONCAT(cook_judge.cook_name, ' ', cook_judge.cook_surname) AS judge_name,
    judge.cook_id,
    CONCAT(cook.cook_name, ' ', cook.cook_surname) AS cook_name,
    SUM(judge.score) AS total_score
FROM
    judge
JOIN
    cook AS cook_judge ON judge.judge_id = cook_judge.cook_id
JOIN
    cook ON judge.cook_id = cook.cook_id
GROUP BY
    judge.judge_id, judge.cook_id
ORDER BY
    total_score DESC
LIMIT 5;

```

3.12

Ποιο ήταν το πιο τεχνικά δύσκολο, από πλευράς συνταγών, επεισόδιο του διαγωνισμού ανά έτος;

```
SELECT season, episode_id, total_difficulty
FROM (
    SELECT
        season,
        e.episode_id,
        SUM(r.difficulty) AS total_difficulty,
        ROW_NUMBER() OVER (PARTITION BY season ORDER BY SUM(r.difficulty) DESC) AS rn
    FROM episode e
    JOIN episode_cuisine_cook_recipe eccr ON e.episode_id = eccr.episode_id
    JOIN recipe r ON eccr.recipe_id = r.recipe_id
    GROUP BY season, e.episode_id
) AS sub
WHERE rn = 1;
```

3.13

Ποιο επεισόδιο συγκέντρωσε τον χαμηλότερο βαθμό επαγγελματικής κατάρτισης (κριτές και μάγειρες);

```
SELECT
    episode_id,
    SUM(classification_points) AS total_classification_points
FROM
    (
        -- Classification points for judges
        SELECT DISTINCT
            j.episode_id,
            j.judge_id AS cook_id,
            CASE c.classification
                WHEN '3rd cook' THEN 1
                WHEN '2nd cook' THEN 2
                WHEN '1st cook' THEN 3
                WHEN 'chefs assistant' THEN 4
                WHEN 'chef' THEN 5
                ELSE 0
            END AS classification_points
        FROM
            judge j
        JOIN
            cook c ON j.judge_id = c.cook_id

        UNION

        -- Classification points for cooks
        SELECT DISTINCT
            j.episode_id,
```

```

        j.cook_id,
        CASE c.classification
            WHEN '3rd cook' THEN 1
            WHEN '2nd cook' THEN 2
            WHEN '1st cook' THEN 3
            WHEN 'chefs assistant' THEN 4
            WHEN 'chef' THEN 5
            ELSE 0
        END AS classification_points
    FROM
        judge j
    JOIN
        cook c ON j.cook_id = c.cook_id
    ) AS unique_cooks_and_judges
GROUP BY
    episode_id
ORDER BY
    total_classification_points ASC
LIMIT 1;

```

3.14

Ποια θεματική ενότητα έχει εμφανιστεί τις περισσότερες φορές στο διαγωνισμό;

```

SELECT ts.thematic_section_id, ts.thematic_section_name, COUNT(rts.recipe_id) AS
appearance_count
FROM recipe_thematic_section rts
JOIN thematic_section ts ON rts.thematic_section_id = ts.thematic_section_id
GROUP BY ts.thematic_section_id, ts.thematic_section_name
ORDER BY appearance_count DESC
LIMIT 1;

```

3.15

Σημείωση: Έπειδη δεν είναι πολύ πιθανό μια από τις 10 μόνο ομάδες τροφίμων να μην συμμετέχουν στο διαγωνισμό, έγινε και υλοποίηση με βάση την ομάδα του βασικού υλικού των συνταγών.

Ποιες ομάδες τροφίμων δεν έχουν εμφανιστεί ποτέ στον διαγωνισμό, με βάση όλα τα υλικά;

```

SELECT fg.food_group_id, fg.food_group_name, COUNT(eccr.recipe_id) AS appearance_count
FROM food_groups fg
LEFT JOIN ingredient i ON fg.food_group_id = i.food_group_id
LEFT JOIN recipe_ingredient ri ON i.ingredient_id = ri.ingredient_id
LEFT JOIN episode_cuisine_cook_recipe eccr ON ri.recipe_id = eccr.recipe_id
GROUP BY fg.food_group_id, fg.food_group_name
HAVING appearance_count = 0;

```

Ποιες ομάδες τροφίμων δεν έχουν εμφανιστεί ποτέ στον διαγωνισμό, με βάση το βασικό υλικό;

```
SELECT fg.food_group_id, fg.food_group_name, COUNT(r.recipe_id) AS appearance_count
FROM food_groups fg
LEFT JOIN ingredient i ON fg.food_group_id = i.food_group_id
LEFT JOIN recipe r ON i.ingredient_id = r.basic_ingredient_id
GROUP BY fg.food_group_id, fg.food_group_name
HAVING appearance_count=0
```

Σύνοψη

- Για την λειτουργία του διαγωνισμού υποθέσαμε πως οι μάγειρες έχουν ηλικίες μεταξύ 18 και 90 για προφανείς λόγους, έλεγχος ο οποίος γίνεται με triggers before_insert_cook και before_update_cook.
- Η κάθε συνταγή ανήκει μόνο σε μια εθνική κουζίνα θεωρώντας πως διαφορετικές χώρες έχουν παραλλαγές του ίδιου φαγητού και όχι την ίδια συνταγή “κατά γράμμα”.
- Μέσω των stored procedures:
 - 1) RandomAssignEpisodeCuisineCookRecipe μπορούμε να κάνουμε την κλήρωση για το κάθε επεισόδιο, όπου διαλέγονται 10 διαφορετικές εθνικές κουζίνες, η κάθε κουζίνα διαλέγει των αντιπρόσωπό της διαφορετικό η κάθε μια (που έχει γνώση της κουζίνας) και μια συνταγή αυτής που καλείται να μαγειρέψει εκείνος ο μάγειρας.
 - 2) RandomAssignJudges μπορούμε σε κάθε επεισόδιο να διαλέξουμε 3 κριτές οι οποίοι δεν συμμετέχουν επίσης ως μάγειρες στην ίδια σεζόν.
- Οι τυχών αλλαγές σε υλικά, ενημέρωνουν τις αντίστοιχες συνταγές στις οποίες χρησιμοποιούνται μέσω κατάλληλων triggers.