



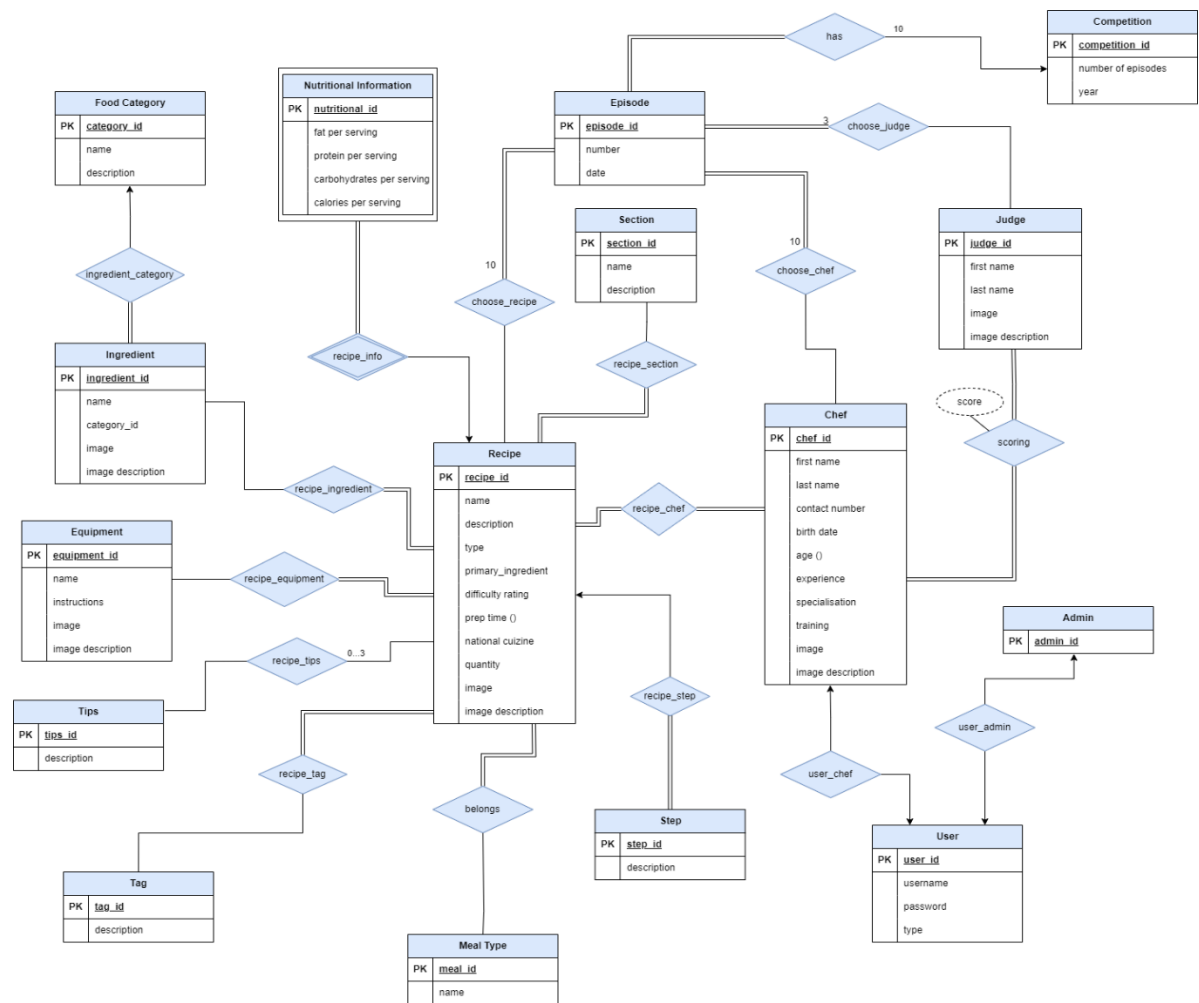
**ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ**  
**ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ -**  
**ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ**

**Βάσεις Δεδομένων**  
**6ο εξάμηνο, Ακαδημαϊκή περίοδος 2023-2024**

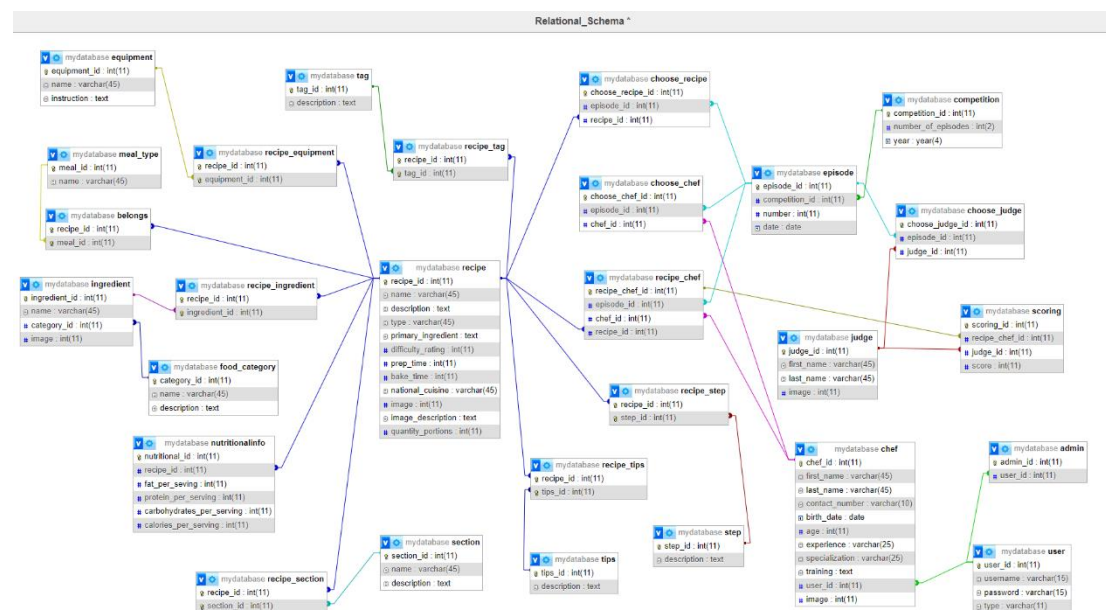
**Εξαμηνιαία Εργασία**  
**Ομάδα 4**

Φώτιος Κούτσικος : 03121082  
Μάριος Κορομπέλης : 03121064

# 1. Διάγραμμα ER



# 2. Σχεσιακό Διάγραμμα



## 2.1. Περιορισμοί ακεραιότητας Βάσης Δεδομένων

- Δεν μπορούν να συμμετέχουν τα ίδια άτομα σε ένα επεισόδιο ως μάγεις και ως κριτές.
- Κάθε επεισόδιο περιέχει 10 μάγεις, καθένas με διαφορετική εξειδίκευση.
- Κάθε επεισόδιο διαλέγει 10 εθνικές κουζίνες.
- Κάθε επεισόδιο περιέχει 10 συνταγές, μία από κάθε εθνική κουζίνα.
- Κάθε επεισόδιο περιέχει 3 κριτές.
- Κάθε μάγισσας σε ένα επεισόδιο βαθμολογείται από όλους τους κριτές αυτού του επεισοδίου.
- Μια συνταγή διαθέτει τουλάχιστον ένα μάγισσα.
- Οι μάγεις έχουν πάντα ηλικία πάνω από 20 έτη.
- Οι μάγεις δεν μπορούν να έχουν περισσότερα χρόνια εμπειρίας από την ηλικία τους (εμείς το θέσαμε χρόνια εμπειρίας  $\leq$  ηλικία - 15).
- Τα επεισόδια μίας σεζόν γίνονται όλα την ίδια χρονιά.
- Ένα επεισόδιο δεν μπορεί να έχει μεταγενέστερη ημερομηνία προβολής από ένα επόμενο επεισόδιο.
- Η επόμενη σεζόν θα πρέπει να ξεκινήσει αφού τελειώσει το τελευταίο επεισόδιο της προηγούμενης σεζόν.

## 2.2. Indexing Βάσης Δεδομένων

Για να ορίσουμε indexes στη βάση μας αρχικά λαμβάνουμε υπόψη μας ότι στο phpMyAdmin που χρησιμοποιούμε για να δημιουργήσουμε τους πίνακες, τα indexes ορίζονται αυτόματα για τα primary keys. Αυτά τα primary keys αποτελούν γενικά τα μοναδικά αναγνωριστικά κάθε στοιχείου κάθε πίνακα και τα χρησιμοποιούμε σε όλα τα ερωτήματα και γι' αυτό το λόγο έχουμε ορίσει ευρετήρια ώστε να επιτύχουμε τη βέλτιστη απόδοση της βάσης μας κατά την πρόσβασή τους. Επιπλέον χρησιμοποιούνται ευρετήρια σε foreign keys τα οποία κάνουν reference στοιχεία από άλλους πίνακες προκειμένου η ενημέρωση / διαγραφή τους να γίνεται όσο το δυνατό γρηγορότερα. Τέλος, έχουμε ορίσει indexes για τις εθνικές κουζίνες των συνταγών και την εξειδίκευση των μαγείρων εφόσον απαιτούνται σε πολλά ερωτήματα, καθώς και στις βαθμολογήσεις τους ανά επεισόδιο. Όλα τα ευρετήρια βρίσκονται στο DDL script DDL.sql ενώ μπορεί να φανούν και στο σχεσιακό διάγραμμα.

### DDL scripts

Στη βάση δεδομένων (στο git repository) υπάρχει το DDL script με ονομασία *DDL.sql* το οποίο δημιουργεί όλους τους πίνακες που έχουμε ορίσει, τις primary key εξαρτήσεις καθώς και τις προβολές.

```
1 CREATE TABLE `admin` (  
2   `admin_id` int(11) NOT NULL,  
3   `user_id` int(11) NOT NULL,  
4   PRIMARY KEY (`admin_id`),  
5   KEY `user_id` (`user_id`),  
6   CONSTRAINT `user_id` FOREIGN KEY (`user_id`) REFERENCES `user` (`user_id`) ON UPDATE CASCADE  
7 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci  
8  
9  
10 CREATE TABLE `belongs` (  
11   `recipe_id` int(11) NOT NULL,  
12   `meal_id` int(11) NOT NULL,  
13   PRIMARY KEY (`recipe_id`, `meal_id`),  
14   KEY `meal_id` (`meal_id`),  
15   CONSTRAINT `belongs_ibfk_1` FOREIGN KEY (`recipe_id`) REFERENCES `recipe` (`recipe_id`) ON UPDATE CASCADE,  
16   CONSTRAINT `belongs_ibfk_2` FOREIGN KEY (`meal_id`) REFERENCES `meal_type` (`meal_id`)  
17 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci  
18  
19  
20 CREATE TABLE `chef` (  
21   `chef_id` int(11) NOT NULL,  
22   `first_name` varchar(45) NOT NULL,  
23   `last_name` varchar(45) NOT NULL,  
24   `contact_number` varchar(10) NOT NULL,  
25   `birth_date` date NOT NULL,  
26   `age` int(11) NOT NULL,  
27   `experience` varchar(25) NOT NULL,  
28   `specialization` varchar(25) NOT NULL,  
29   `training` text NOT NULL,  
30   `user_id` int(11) NOT NULL,  
31   `image` int(11) NOT NULL,  
32   PRIMARY KEY (`chef_id`),  
33   KEY `user_id_fk` (`user_id`),  
34   CONSTRAINT `user_id_fk` FOREIGN KEY (`user_id`) REFERENCES `user` (`user_id`) ON UPDATE CASCADE  
35 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci  
36  
37  
38 CREATE TABLE `choose_chef` (  
39   `choose_chef_id` int(11) NOT NULL AUTO_INCREMENT,  
40   `episode_id` int(11) NOT NULL,  
41   `chef_id` int(11) NOT NULL,  
42   PRIMARY KEY (`choose_chef_id`),  
43   KEY `chef_id_fk` (`chef_id`),  
44   KEY `episode_id` (`episode_id`),  
45   CONSTRAINT `chef_id_fk` FOREIGN KEY (`chef_id`) REFERENCES `chef` (`chef_id`) ON UPDATE CASCADE,  
46   CONSTRAINT `episode_id_fk` FOREIGN KEY (`episode_id`) REFERENCES `episode` (`episode_id`) ON UPDATE CASCADE  
47 ) ENGINE=InnoDB AUTO_INCREMENT=6437 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci  
48  
49  
50 CREATE TABLE `choose_judge` (  
51   `choose_judge_id` int(11) NOT NULL AUTO_INCREMENT,  
52   `episode_id` int(11) NOT NULL,  
53   `judge_id` int(11) NOT NULL,  
54   PRIMARY KEY (`choose_judge_id`),  
55   KEY `judge_id` (`judge_id`),  
56   KEY `judge_id_2` (`judge_id`),  
57   KEY `episode_id` (`episode_id`),  
58   CONSTRAINT `choose_judge_ibfk_1` FOREIGN KEY (`episode_id`) REFERENCES `episode` (`episode_id`) ON UPDATE CASCADE,  
59   CONSTRAINT `choose_judge_ibfk_2` FOREIGN KEY (`judge_id`) REFERENCES `judge` (`judge_id`) ON UPDATE CASCADE  
60 ) ENGINE=InnoDB AUTO_INCREMENT=718 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci  
61  
62  
63 CREATE TABLE `choose_recipe` (  
64   `choose_recipe_id` int(11) NOT NULL AUTO_INCREMENT,  
65   `episode_id` int(11) NOT NULL,  
66   `recipe_id` int(11) NOT NULL,  
67   PRIMARY KEY (`choose_recipe_id`),  
68   KEY `episode_id` (`episode_id`),  
69   KEY `recipe_id` (`recipe_id`),  
70   CONSTRAINT `choose_recipe_ibfk_1` FOREIGN KEY (`recipe_id`) REFERENCES `recipe` (`recipe_id`) ON UPDATE CASCADE,  
71   CONSTRAINT `choose_recipe_ibfk_2` FOREIGN KEY (`episode_id`) REFERENCES `episode` (`episode_id`) ON UPDATE CASCADE  
72 ) ENGINE=InnoDB AUTO_INCREMENT=993 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci  
73  
74
```

```

74
75 CREATE TABLE `competition` (
76   `competition_id` int(11) NOT NULL,
77   `number_of_episodes` int(2) NOT NULL,
78   `year` year(4) NOT NULL,
79   PRIMARY KEY (`competition_id`)
80 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
81
82
83 CREATE TABLE `episode` (
84   `episode_id` int(11) NOT NULL,
85   `competition_id` int(11) NOT NULL,
86   `number` int(11) NOT NULL,
87   `date` date NOT NULL,
88   PRIMARY KEY (`episode_id`),
89   KEY `competition_id_fk` (`competition_id`),
90   CONSTRAINT `competition_id_fk` FOREIGN KEY (`competition_id`) REFERENCES `competition` (`competition_id`) ON UPDATE CASCADE
91 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
92
93
94 CREATE TABLE `equipment` (
95   `equipment_id` int(11) NOT NULL,
96   `name` varchar(45) NOT NULL,
97   `instruction` text NOT NULL,
98   PRIMARY KEY (`equipment_id`)
99 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
100
101
102 CREATE TABLE `food_category` (
103   `category_id` int(11) NOT NULL,
104   `name` varchar(45) NOT NULL,
105   `description` text NOT NULL,
106   PRIMARY KEY (`category_id`)
107 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
108
109
110 CREATE TABLE `ingredient` (
111   `ingredient_id` int(11) NOT NULL,
112   `name` varchar(45) NOT NULL,
113   `category_id` int(11) NOT NULL,
114   `image` int(11) NOT NULL,
115   PRIMARY KEY (`ingredient_id`),
116   KEY `ingredient_category_fk` (`category_id`),
117   CONSTRAINT `ingredient_category_fk` FOREIGN KEY (`category_id`) REFERENCES `food_category` (`category_id`) ON UPDATE CASCADE
118 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
119
120
121 CREATE TABLE `judge` (
122   `judge_id` int(11) NOT NULL,
123   `first_name` varchar(45) NOT NULL,
124   `last_name` varchar(45) NOT NULL,
125   `image` int(11) NOT NULL,
126   PRIMARY KEY (`judge_id`)
127 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
128
129
130 CREATE TABLE `meal_type` (
131   `meal_id` int(11) NOT NULL,
132   `name` varchar(45) NOT NULL,
133   PRIMARY KEY (`meal_id`)
134 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
135
136
137 CREATE TABLE `nutritionalinfo` (
138   `nutritional_id` int(11) NOT NULL,
139   `recipe_id` int(11) NOT NULL,
140   `fat_per_serving` int(11) NOT NULL,
141   `protein_per_serving` int(11) NOT NULL,
142   `carbohydrates_per_serving` int(11) NOT NULL,
143   `calories_per_serving` int(11) NOT NULL,
144   PRIMARY KEY (`nutritional_id`),
145   KEY `recipe_info_fk` (`recipe_id`),
146   CONSTRAINT `recipe_info_fk` FOREIGN KEY (`recipe_id`) REFERENCES `recipe` (`recipe_id`) ON UPDATE CASCADE
147 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
148

```

```

150 CREATE TABLE `recipe` (
151     `recipe_id` int(11) NOT NULL,
152     `name` varchar(45) NOT NULL,
153     `description` text NOT NULL,
154     `type` varchar(45) NOT NULL,
155     `primary_ingredient` text NOT NULL,
156     `difficulty_rating` int(11) NOT NULL,
157     `prep_time` int(11) NOT NULL,
158     `bake_time` int(11) NOT NULL,
159     `national_cuisine` varchar(45) NOT NULL,
160     `image` int(11) NOT NULL,
161     `image_description` text NOT NULL,
162     `quantity_portions` int(11) NOT NULL,
163     PRIMARY KEY (`recipe_id`)
164 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
165
166
167 CREATE TABLE `recipe_chef` (
168     `recipe_chef_id` int(11) NOT NULL AUTO_INCREMENT,
169     `episode_id` int(11) NOT NULL,
170     `chef_id` int(11) NOT NULL,
171     `recipe_id` int(11) NOT NULL,
172     PRIMARY KEY (`recipe_chef_id`),
173     KEY `episode_id` (`episode_id`),
174     KEY `recipe_id` (`recipe_id`),
175     KEY `chef_id` (`chef_id`),
176     CONSTRAINT `chef_id` FOREIGN KEY (`chef_id`) REFERENCES `chef` (`chef_id`) ON UPDATE CASCADE,
177     CONSTRAINT `episode_id` FOREIGN KEY (`episode_id`) REFERENCES `episode` (`episode_id`) ON UPDATE CASCADE,
178     CONSTRAINT `recipe_id` FOREIGN KEY (`recipe_id`) REFERENCES `recipe` (`recipe_id`) ON UPDATE CASCADE
179 ) ENGINE=InnoDB AUTO_INCREMENT=1024 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
180
181
182 CREATE TABLE `recipe_equipment` (
183     `recipe_id` int(11) NOT NULL,
184     `equipment_id` int(11) NOT NULL,
185     PRIMARY KEY (`recipe_id`,`equipment_id`),
186     KEY `recipe_equipment_ibfk_2` (`equipment_id`),
187     CONSTRAINT `recipe_equipment_ibfk_1` FOREIGN KEY (`recipe_id`) REFERENCES `recipe` (`recipe_id`),
188     CONSTRAINT `recipe_equipment_ibfk_2` FOREIGN KEY (`equipment_id`) REFERENCES `equipment` (`equipment_id`)
189 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
190
191
192 CREATE TABLE `recipe_ingredient` (
193     `recipe_id` int(11) NOT NULL,
194     `ingredient_id` int(11) NOT NULL,
195     PRIMARY KEY (`recipe_id`,`ingredient_id`),
196     KEY `ingredient_id` (`ingredient_id`),
197     CONSTRAINT `recipe_ingredient_ibfk_1` FOREIGN KEY (`recipe_id`) REFERENCES `recipe` (`recipe_id`) ON DELETE CASCADE ON UPDATE CASCADE,
198     CONSTRAINT `recipe_ingredient_ibfk_2` FOREIGN KEY (`ingredient_id`) REFERENCES `ingredient` (`ingredient_id`) ON DELETE CASCADE ON UPDATE CASCADE
199 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
200
201
202 CREATE TABLE `recipe_section` (
203     `recipe_id` int(11) NOT NULL,
204     `section_id` int(11) NOT NULL,
205     PRIMARY KEY (`recipe_id`,`section_id`),
206     KEY `section_id` (`section_id`),
207     CONSTRAINT `recipe_id_fk` FOREIGN KEY (`recipe_id`) REFERENCES `recipe` (`recipe_id`),
208     CONSTRAINT `section_id` FOREIGN KEY (`section_id`) REFERENCES `section` (`section_id`)
209 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
210

```

```

211
212 CREATE TABLE `recipe_step` (
213     `recipe_id` int(11) NOT NULL,
214     `step_id` int(11) NOT NULL,
215     PRIMARY KEY (`recipe_id`,`step_id`),
216     KEY `step_id_fk` (`step_id`),
217     CONSTRAINT `recipe_step_ibfk_1` FOREIGN KEY (`recipe_id`) REFERENCES `recipe` (`recipe_id`) ON UPDATE CASCADE,
218     CONSTRAINT `recipe_step_ibfk_2` FOREIGN KEY (`step_id`) REFERENCES `step` (`step_id`) ON UPDATE CASCADE
219 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
220
221
222 CREATE TABLE `recipe_tag` (
223     `recipe_id` int(11) NOT NULL,
224     `tag_id` int(11) NOT NULL,
225     PRIMARY KEY (`recipe_id`,`tag_id`),
226     KEY `tag_id` (`tag_id`),
227     CONSTRAINT `recipe_tag_ibfk_1` FOREIGN KEY (`recipe_id`) REFERENCES `recipe` (`recipe_id`) ON DELETE CASCADE ON UPDATE CASCADE,
228     CONSTRAINT `recipe_tag_ibfk_2` FOREIGN KEY (`tag_id`) REFERENCES `tag` (`tag_id`) ON DELETE CASCADE ON UPDATE CASCADE
229 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
230
231
232 CREATE TABLE `recipe_tips` (
233     `recipe_id` int(11) NOT NULL,
234     `tips_id` int(11) NOT NULL,
235     PRIMARY KEY (`recipe_id`,`tips_id`),
236     KEY `tips_id` (`tips_id`),
237     CONSTRAINT `recipe_tips_ibfk_1` FOREIGN KEY (`tips_id`) REFERENCES `tips` (`tips_id`) ON DELETE CASCADE ON UPDATE CASCADE,
238     CONSTRAINT `recipe_tips_ibfk_2` FOREIGN KEY (`recipe_id`) REFERENCES `recipe` (`recipe_id`) ON DELETE CASCADE ON UPDATE CASCADE
239 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
240
241
242 CREATE TABLE `scoring` (
243     `scoring_id` int(11) NOT NULL AUTO_INCREMENT,
244     `recipe_chef_id` int(11) NOT NULL,
245     `judge_id` int(11) NOT NULL,
246     `score` int(11) NOT NULL,
247     PRIMARY KEY (`scoring_id`),
248     KEY `judge_id` (`judge_id`),
249     KEY `recipe_chef_id` (`recipe_chef_id`),
250     CONSTRAINT `scoring_ibfk_1` FOREIGN KEY (`recipe_chef_id`) REFERENCES `recipe_chef` (`recipe_chef_id`) ON DELETE CASCADE ON UPDATE CASCADE,
251     CONSTRAINT `scoring_ibfk_2` FOREIGN KEY (`judge_id`) REFERENCES `judge` (`judge_id`) ON UPDATE CASCADE
252 ) ENGINE=InnoDB AUTO_INCREMENT=4096 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
253
254
255 CREATE TABLE `section` (
256     `section_id` int(11) NOT NULL,
257     `name` varchar(45) NOT NULL,
258     `description` text NOT NULL,
259     PRIMARY KEY (`section_id`)
260 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
261
262
263 CREATE TABLE `step` (
264     `step_id` int(11) NOT NULL,
265     `description` text NOT NULL,
266     PRIMARY KEY (`step_id`)
267 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
268
269
270 CREATE TABLE `tag` (
271     `tag_id` int(11) NOT NULL,
272     `description` text NOT NULL,
273     PRIMARY KEY (`tag_id`)
274 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
275
276
277 CREATE TABLE `tips` (
278     `tips_id` int(11) NOT NULL,
279     `description` text NOT NULL,
280     PRIMARY KEY (`tips_id`)
281 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
282
283
284 CREATE TABLE `user` (
285     `user_id` int(11) NOT NULL,
286     `username` varchar(15) NOT NULL,
287     `password` varchar(15) NOT NULL,
288     `type` varchar(11) NOT NULL,
289     PRIMARY KEY (`user_id`)
290 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
291

```

## DML scripts

Στο DML script γίνεται η εισαγωγή των δεδομένων στη βάση μας και πρόκειται για το αρχείο *DML.sql* (στο git repository). Η εισαγωγή των αρχικών στοιχείων έγινε με τη χρήση της ιστοσελίδας Mockaroo, η οποία εξειδικεύεται στη δημιουργία dummy data και μέσω αυτής «γεμίσαμε» πίνακες όπως μάγειρας, συνταγή, χρήστης, υλικά κ.λπ.. Στη συνέχεια μέσω των διαδικασιών InsertChefs, InsertJudges, InsertRecipes φορτώσαμε με δεδομένα πίνακες όπως choose\_chef, choose\_judge, choose\_recipe έτσι ώστε να πάρουμε τα ήδη υπάρχοντα στοιχεία και να τα αναθέσουμε σε κάθε επεισόδιο με τους περιορισμούς που φάνηκαν παραπάνω.

Ενδεικτικά δείχνουμε μερικώς το συμπλήρωμα των πινάκων chef και user.

```
133 • INSERT INTO `chef` (`chef_id`, `first_name`, `last_name`, `contact_number`, `birth_date`, `age`, `experience`, `specialization`, `training`, `user_id`, `image`) VALUES
134 (1, 'Shaun', 'Hallows', '596-235-73', '1986-12-13', 37, '17', 'Mexican-American', 'B Cook', 1, 0),
135 (2, 'Bill', 'Seabourne', '244-897-45', '2002-03-14', 22, '4', 'Mexican-American', 'Chef', 2, 0),
136 (3, 'Mikol', 'Saggers', '848-592-60', '2002-06-07', 21, '4', 'Japanese', 'Souchef', 3, 0),
137 (4, 'Rasia', 'Snoday', '812-828-79', '1992-11-05', 31, '21', 'Moroccan', 'Souchef', 4, 0),
138 (5, 'Lucina', 'Wintle', '665-323-58', '1978-09-15', 45, '17', 'Mexican', 'B Cook', 5, 0),
139 (6, 'Sansone', 'Heningham', '800-231-36', '1996-03-19', 28, '18', 'Asian Fusion', 'B Cook', 6, 0),
140 (7, 'Karlán', 'Sorensen', '611-823-58', '1981-02-24', 43, '21', 'Thai', 'Chef', 7, 0),
141 (8, 'Sterne', 'Blackford', '493-940-21', '1991-06-06', 32, '22', 'Japanese', 'C Cook', 8, 0),
142 (9, 'Alethea', 'Copcote', '570-878-76', '1974-04-16', 50, '5', 'American', 'Chef', 9, 0),
143 (10, 'Jefferey', 'Hannaby', '837-477-91', '1973-05-14', 51, '15', 'Italian-American', 'C Cook', 10, 0),
```

```
7419 • INSERT INTO `user` (`user_id`, `username`, `password`, `type`) VALUES
7420 (1, 'lgerard0', 'nC6!1eX(4', 'chef'),
7421 (2, 'gkirkam1', 'eC1#}.Qo@RgKo{', 'chef'),
7422 (3, 'lspickett2', 'eO6$)eOF0y>A', 'chef'),
7423 (4, 'dreadwin3', 'nZ2~$gIrLdy6', 'chef'),
7424 (5, 'mclipson4', 'dL5=B\'}DKy=hObN', 'chef'),
7425 (6, 'vvolonte5', 'xB0%Yhc,~fw', 'chef'),
7426 (7, 'kmealand6', 'eY8~!AnKLuQuK5s', 'chef'),
7427 (8, 'dkidston7', 'iG7`)/0te\\_Lj,d', 'chef'),
7428 (9, 'egoede8', 'uR7\"5.%>n#', 'chef'),
7429 (10, 'alorie9', 'rT8&J%b9rrMU', 'chef'),
7430 (11, 'cseelya', 'zY9\"<osn', 'chef'),
7431 (12, 'eramsbyb', 'vL7=&FK72', 'chef'),
7432 (13, 'lgervaise', 'rJ7`&scyP{F', 'chef'),
7433 (14, 'lcowsbyd', 'hG3)_&jR#IcUA', 'chef'),
7434 (15, 'gdiblinge', 'aT9,xux4`xirhzi', 'chef'),
7435 (16, 'ecalteronef', 'yU2|\"(H(>', 'chef'),
7436 (17, 'fwoodg', 'jW7!zRKDA7A', 'chef'),
7437 (18, 'locahsedyh', 'uZ1`JJH5', 'chef'),
7438 (19, 'jeffordi', 'kL3*Q8h7/jw', 'chef'),
7439 (20, 'hbankhurstj', 'uQ8/yq*+IL7u\"E|', 'chef'),
7440 (21, 'anorburyk', 'dE3_,\"@Q!\\', 'chef'),
7441 (22, 'swarwick1', 'oQ2|Hni62yX', 'chef'),
7442 (23, 'kgogginm', 'lX4>F$QF{<|oE', 'chef'),
7443 (24, 'ccanton', 'jL7~JN.n599<', 'chef'),
7444 (25, 'eleismano', 'kT2==LrK\"+k=6Qc', 'chef'),
7445 (26, 'acraggsp', 'xJ0>kS0}.', 'chef'),
7446 (27, 'bkuschelq', 'mN3_609yk*B=u*', 'chef'),
7447 (28, 'bnootr', 'mN0{KL_cY>sFYA', 'chef'),
```



### 3. Queries

1) Μέσος Όρος Αξιολογήσεων (σκορ) ανά μάγειρα και Εθνική κουζίνα.

Ο μέσος όρος αξιολογήσεων ανά μάγειρα:

```
1 SELECT
2     rc.chef_id,
3     AVG(s.score) AS average_score
4 FROM
5     scoring s
6 JOIN
7     recipe_chef rc ON s.recipe_chef_id = rc.recipe_chef_id
8 GROUP BY
9     rc.chef_id;
```

Ο μέσος όρος αξιολογήσεων ανά εθνική κουζίνα:

```
1 SELECT
2     r.national_cuisine,
3     AVG(s.score) AS average_score
4 FROM
5     scoring s
6 JOIN
7     recipe_chef rc ON s.recipe_chef_id = rc.recipe_chef_id
8 JOIN
9     recipe r ON rc.recipe_id = r.recipe_id
10 GROUP BY
11     r.national_cuisine;
```

2) Για δεδομένη Εθνική κουζίνα και έτος, ποιοι μάγειρες ανήκουν σε αυτήν και ποιοι μάγειρες συμμετείχαν σε επεισόδια;

```
1 SELECT
2     r.national_cuisine,
3     YEAR(e.date) AS year,
4     c.chef_id,
5     c.first_name,
6     c.last_name,
7     c.specialization
8 FROM
9     chef c
10 JOIN
11     choose_chef cc ON c.chef_id = cc.chef_id
12 JOIN
13     episode e ON cc.episode_id = e.episode_id
14 JOIN
15     choose_recipe cr ON e.episode_id = cr.episode_id
16 JOIN
17     recipe r ON cr.recipe_id = r.recipe_id
18 WHERE
19     c.specialization = r.national_cuisine
20 GROUP BY
21     r.national_cuisine, YEAR(e.date), c.chef_id
22 ORDER BY
23     r.national_cuisine, YEAR(e.date), c.chef_id;
```

3) Βρείτε τους νέους μάγειρες (ηλικία < 30 ετών) που έχουν τις περισσότερες συνταγές.

```
1 SELECT
2     c.chef_id,
3     c.first_name,
4     c.last_name,
5     c.age,
6     COUNT(rc.recipe_id) AS recipe_count
7 FROM
8     chef c
9 JOIN
10    recipe_chef rc ON c.chef_id = rc.chef_id
11 WHERE
12     c.age < 30
13 GROUP BY
14     c.chef_id, c.first_name, c.last_name
15 HAVING
16     recipe_count = (
17         SELECT
18             MAX(recipe_count)
19         FROM (
20             SELECT
21                 chef_id,
22                 COUNT(recipe_id) AS recipe_count
23             FROM
24                 recipe_chef
25             GROUP BY
26                 chef_id
27         ) AS subquery
28     )
29 ORDER BY
30     recipe_count DESC;
```

4) Βρείτε τους μάγειρες που δεν έχουν συμμετάσχει ποτέ σε ως κριτές σε κάποιο επεισόδιο.

```
1 SELECT
2     c.chef_id,
3     c.first_name,
4     c.last_name
5 FROM
6     chef c
7 LEFT JOIN
8     choose_judge cj ON c.chef_id = cj.judge_id
9 WHERE
10    cj.judge_id IS NULL;
```

5) Ποιοι κριτές έχουν συμμετάσχει στον ίδιο αριθμό επεισοδίων σε διάστημα ενός έτους με περισσότερες από 3 εμφανίσεις;

```
1 SELECT
2     pc.year,
3     pc.episode_count,
4     pc.judges
5 FROM
6     ParticipationCount pc;
```

Όπου το ParticipationCount είναι μία προβολή ορισμένη ως εξής:

```
CREATE VIEW ParticipationCount AS (
    SELECT
        jp.year,
        jp.episode_count,
        GROUP_CONCAT(jp.judge_id) AS judges
    FROM
        JudgeParticipation jp
    GROUP BY
        jp.year, jp.episode_count
    HAVING
        COUNT(jp.judge_id) > 1
);
```

Και με τη σειρά του το JudgeParticipation:

```
CREATE VIEW JudgeParticipation AS (
    SELECT
        cj.judge_id,
        YEAR(e.date) AS year,
        COUNT(DISTINCT e.episode_id) AS episode_count
    FROM
        choose_judge cj
    JOIN
        episode e ON cj.episode_id = e.episode_id
    GROUP BY
        cj.judge_id, year
    HAVING
        COUNT(DISTINCT e.episode_id) > 3
);
```

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

```
1 SELECT
2     cp.chef_id,
3     cp.participation_count
4 FROM
5     ChefParticipation cp,
6     MaxParticipation mp
7 WHERE
8     cp.participation_count <= mp.max_participation - 5
9 ORDER BY
10    cp.participation_count;
```

Με τα δύο views να είναι τα ακόλουθα:

```
CREATE VIEW ChefParticipation AS (
    SELECT
        cc.chef_id,
        COUNT(cc.episode_id) AS participation_count
    FROM
        choose_chef cc
    GROUP BY
        cc.chef_id
);

CREATE VIEW MaxParticipation AS (
    SELECT
        MAX(participation_count) AS max_participation
    FROM
        ChefParticipation
);
```

8) Σε ποιο επεισόδιο χρησιμοποιήθηκαν τα περισσότερα εξαρτήματα (εξοπλισμός);

```
1 SELECT
2     e.episode_id,
3     COUNT(re.equipment_id) AS equipment_count
4 FROM
5     choose_recipe cr
6 JOIN
7     recipe_equipment re ON cr.recipe_id = re.recipe_id
8 JOIN
9     episode e ON cr.episode_id = e.episode_id
10 GROUP BY
11     e.episode_id
12 ORDER BY
13     equipment_count DESC
14 LIMIT 1;
```

9) Λίστα με μέσο όρο αριθμού γραμμάρων υδατανθράκων στο διαγωνισμό ανά έτος;

```
1 SELECT
2     YEAR(e.date) AS year,
3     AVG(ni.carbohydrates_per_serving) AS avg_carbohydrates
4 FROM
5     episode e
6 JOIN
7     choose_recipe cr ON e.episode_id = cr.episode_id
8 JOIN
9     nutritionalinfo ni ON cr.recipe_id = ni.recipe_id
10 GROUP BY
11     YEAR(e.date)
12 ORDER BY
13     year;
```

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

```
1 SELECT
2     national_cuisine,
3     competition_id1,
4     competition_id2,
5     participations1,
6     participations2
7 FROM
8     ConsecutiveCompetitions
9 WHERE
10     participations1 = participations2
11     AND participations1 > 3;
```

Με την προβολή ConsecutiveCompetitions:

```
CREATE VIEW ConsecutiveCompetitions AS (
    SELECT
        cp1.national_cuisine,
        cp1.competition_id AS competition_id1,
        cp1.participations AS participations1,
        cp2.competition_id AS competition_id2,
        cp2.participations AS participations2
    FROM
        CuisineParticipations cp1
    JOIN
        CuisineParticipations cp2
    ON
        cp1.national_cuisine = cp2.national_cuisine
        AND cp1.competition_id + 1 = cp2.competition_id
);
```

11) Βρείτε τους top-5 κριτές που έχουν δώσει συνολικά την υψηλότερη βαθμολόγηση σε ένα μάγειρα. (όνομα κριτή, όνομα μάγειρα και συνολικό σκορ βαθμολόγησης)

```
1 SELECT
2     j.first_name AS judge_first_name,
3     j.last_name AS judge_last_name,
4     c.first_name AS chef_first_name,
5     c.last_name AS chef_last_name,
6     tjs.total_score
7 FROM
8     TopJudgeScores tjs
9 JOIN
10    judge j ON tjs.judge_id = j.judge_id
11 JOIN
12    chef c ON tjs.chef_id = c.chef_id
13 WHERE
14    tjs.rn <= 5;
```

Όπου χρησιμοποιούμε το view:

```
CREATE VIEW TopJudgeScores AS (
    SELECT
        js.judge_id,
        js.chef_id,
        js.total_score,
        ROW_NUMBER() OVER (ORDER BY js.total_score DESC) AS
    FROM
        JudgeScores js
);
```

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

```
1 SELECT
2     e.episode_id,
3     MAX(r.difficulty_rating) AS max_difficulty_rating
4 FROM
5     episode e
6 JOIN
7     choose_recipe cr ON e.episode_id = cr.episode_id
8 JOIN
9     recipe r ON cr.recipe_id = r.recipe_id
10 GROUP BY
11     e.episode_id
12 ORDER BY
13     max_difficulty_rating DESC
14 LIMIT 1;
```

14) Ποια θεματική ενότητα έχει εμφανιστεί τις περισσότερες φορές στο διαγωνισμό;

```
1 SELECT
2     section_id,
3     name,
4     COUNT(*) AS appearance_count
5 FROM
6     section
7 GROUP BY
8     section_id
9 ORDER BY
10    appearance_count DESC
11 LIMIT 1;
```

## 4. Οδηγίες Εγκατάστασης

Το repository της βάσης δεδομένων στο GitHub είναι:

[https://github.com/fotiskoutsikos/Data\\_Bases\\_2024](https://github.com/fotiskoutsikos/Data_Bases_2024)

Αρχικά, πρέπει να γίνει clone το git repo είτε μέσω της εφαρμογής GitHub desktop, είτε μέσω κάποιου terminal με την εντολή:

```
git clone https://github.com/fotiskoutsikos/Data_Bases_2024
```

στο τοπικό directory όπου θα εγκατασταθεί η εφαρμογή.

Έπειτα, προκειμένου να εγκατασταθεί η βάση στον υπολογιστή μας πρέπει να διαθέτουμε έναν sql server (εμείς χρησιμοποιήσαμε το xampp) και ένα DBMS (εδώ χρησιμοποιήσαμε το phpMyAdmin). Η βάση θα εγκατασταθεί αν τρέξουμε τα scripts DDL.sql και DML.sql που βρίσκονται στο repository.

Έχοντας εκτελέσει αυτά τα βήματα, η βάση θα είναι έτοιμη για χρήση.