/\* Query 1 EVERYONE \*/

SELECT Recipe.name

FROM RecipeIngredient

JOIN Recipe ON Recipe.id = RecipeIngredient.recipeId

WHERE RecipeIngredient.ingredientName = "tomatoes"

GROUP BY Recipe.id

ORDER BY Recipe.name;

/\* Query 2 FILIP \*/

SELECT Recipe.name, COUNT(\*) AS Ingredients

FROM RecipeIngredient

JOIN Recipe ON RecipeIngredient.recipeid = Recipe.id

GROUP BY Recipe.id;

/\* Query 3 FILIP \*/

SELECT

    Recipe.name,

    CASE

        WHEN Recipe.complexity > 3 AND COUNT(\*) > 5 THEN 'High'

        WHEN Recipe.complexity < 3 AND COUNT(\*) < 5 THEN 'Low'

        ELSE 'Medium'

    END AS complexity

FROM recipeIngredient

JOIN recipe ON recipeIngredient.recipeid = recipe.id

GROUP BY Recipe.id;

/\* Query 4 WILLIAM \*/

SELECT

    Recipe.name AS recipeName,

    Author.name AS authorName,

    RecipeCategory.categoryName AS category

FROM Recipe

JOIN RecipeAuthor ON Recipe.id = RecipeAuthor.recipeId

JOIN Author ON Author.id = RecipeAuthor.authorId

JOIN RecipeCategory ON Recipe.id = RecipeCategory.recipeId

WHERE

    Author.name = 'Jamie Oliver' AND

    RecipeCategory.categoryName = 'Christmas Food'

ORDER BY Recipe.name;

/\* Query 5 OLIVER \*/

SELECT Recipe.name AS recipeName, COUNT(\*) AS amountOfFavorites

FROM FavoriteRecipe

JOIN Recipe

ON Recipe.id = FavoriteRecipe.recipeId

GROUP BY FavoriteRecipe.recipeId

ORDER BY COUNT(\*) DESC

LIMIT 10;

/\* Query 6 ALEXANDER \*/

SELECT User.firstName, User.lastName, COUNT(\*) AS "totalFavorites"

FROM FavoriteRecipe

JOIN User ON User.id = FavoriteRecipe.recipeId

GROUP BY FavoriteRecipe.userId

ORDER BY COUNT(\*) DESC

LIMIT 3;

/\* Query 7 MINHUI \*/

SELECT RecipeCategory.categoryName, COUNT(\*) AS FavoriteCount

FROM RecipeCategory

JOIN FavoriteRecipe ON RecipeCategory.recipeId = FavoriteRecipe.recipeId

GROUP BY categoryName

ORDER BY FavoriteCount DESC

LIMIT 1;

/\* Query 8 MARCUS \*/

SELECT

    Author.name,

    ROUND(AVG(Recipe.minutes),    2) AS AverageMinutes,

    ROUND(AVG(Recipe.complexity), 2) AS AverageComplexity,

    AverageIngredientCount

FROM RecipeAuthor

    JOIN Author ON RecipeAuthor.authorId = Author.id

    JOIN Recipe ON RecipeAuthor.recipeId = Recipe.id

    JOIN

        (SELECT

            authorId AS innerAuthorId,

            ROUND(AVG(IngredientCount), 2) AS AverageIngredientCount

        FROM

            (SELECT

                RecipeAuthor.authorId,

                RecipeIngredient.recipeId,

                COUNT(\*) AS IngredientCount

            FROM RecipeIngredient

                JOIN RecipeAuthor ON RecipeIngredient.recipeId = RecipeAuthor.recipeId

            GROUP BY authorId, RecipeIngredient.recipeId)

        GROUP BY authorId)

    ON RecipeAuthor.authorId = innerAuthorId

GROUP BY Author.id;

/\* Query 9 EVERYONE \*/

SELECT \*

FROM User

WHERE

    /\* User has at least one favorite recipe \*/

    User.id IN (SELECT userId FROM FavoriteRecipe)

    AND

    /\* User is not one of the users with at least one favorite recipe containing lactose \*/

    User.id NOT IN

    (SELECT DISTINCT userId

    FROM FavoriteRecipe

    WHERE recipeId IN

    /\* Recipes with lactose \*/

    (SELECT DISTINCT RecipeIngredient.recipeId

    FROM RecipeIngredient

    WHERE RecipeIngredient.ingredientName IN

    ('milk', 'cream', 'cheese', 'butter',

    'parmesan', 'yoghurt', 'margarine', 'cheddar')

    GROUP BY RecipeIngredient.recipeId)

    GROUP BY userId)

/\* Query 10 EVERYONE \*/

SELECT Recipe.id, Recipe.name, FavoriteRecipe.userId

FROM FavoriteRecipe

JOIN Recipe ON Recipe.id = FavoriteRecipe.recipeId

WHERE FavoriteRecipe.recipeId NOT IN

    /\* Favorites from one user (user 5) \*/

    (SELECT FavoriteRecipe.recipeId FROM FavoriteRecipe WHERE FavoriteRecipe.userId = 5)

    AND FavoriteRecipe.userId IN

    (SELECT userId

    FROM FavoriteRecipe

    WHERE

        FavoriteRecipe.userId IN (

        /\* Users with at least 2 same favorite recipes \*/

        SELECT

            userId

        FROM FavoriteRecipe

        WHERE FavoriteRecipe.recipeId IN

             /\* Favorites from one user (user 5) \*/

            (SELECT FavoriteRecipe.recipeId FROM FavoriteRecipe WHERE FavoriteRecipe.userId = 5)

        GROUP BY userId HAVING COUNT(\*) > 1)

    GROUP BY userId

    ORDER BY COUNT(\*)

    LIMIT 3) /\* Amount of "recommendation users" + 1 \*/

ORDER BY RANDOM()