НПГ ПО КТС - ГРАД ПРАВЕЦ

**КУРСОВА РАБОТА**

**по**

**Бази данни**

**Ученик:***<Виктор>, <19205>*  **Преподавател:** ................

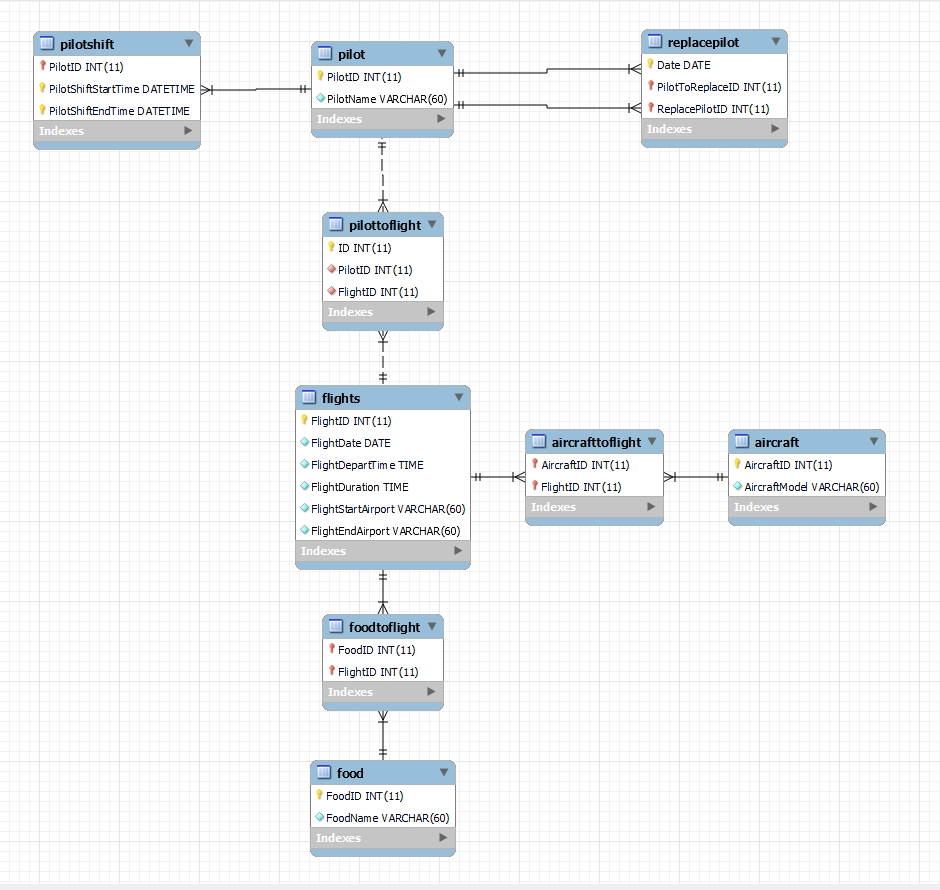
*/ас. инж. Елизабет Михайлова/*

**<01.04.23>**

**Част „Проектиране“**

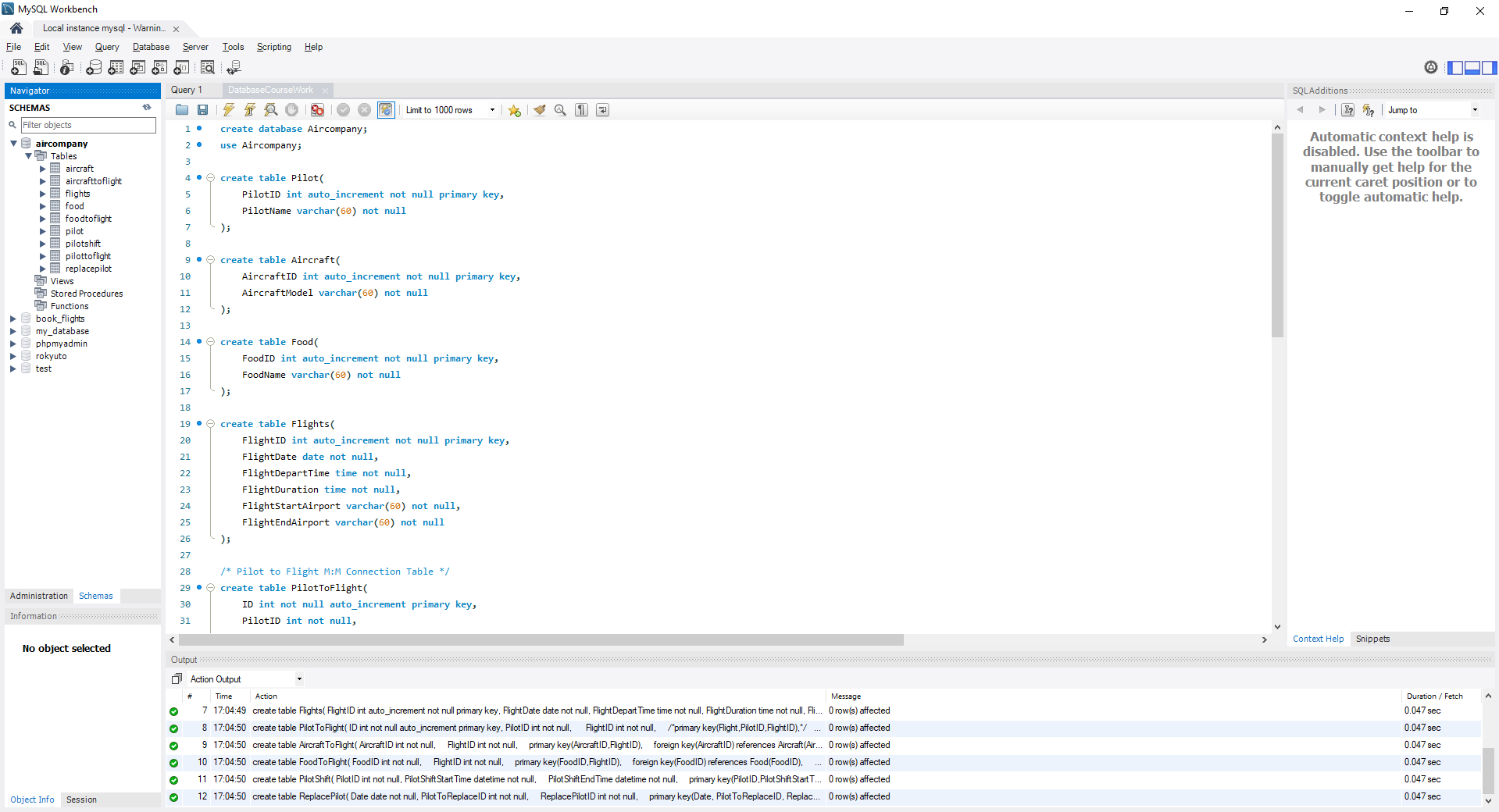
**Тема: Разработете база данни за система за менажиране на полетите на авиокомпания като се съхраняват деня, часа и продължителността на полета, начално и крайно летище, самолет, пилот, тип на предлаганата храна. Да може да се извежда разписанията на компанията, нарядите на пилотите, възможностите за заместване и др.**

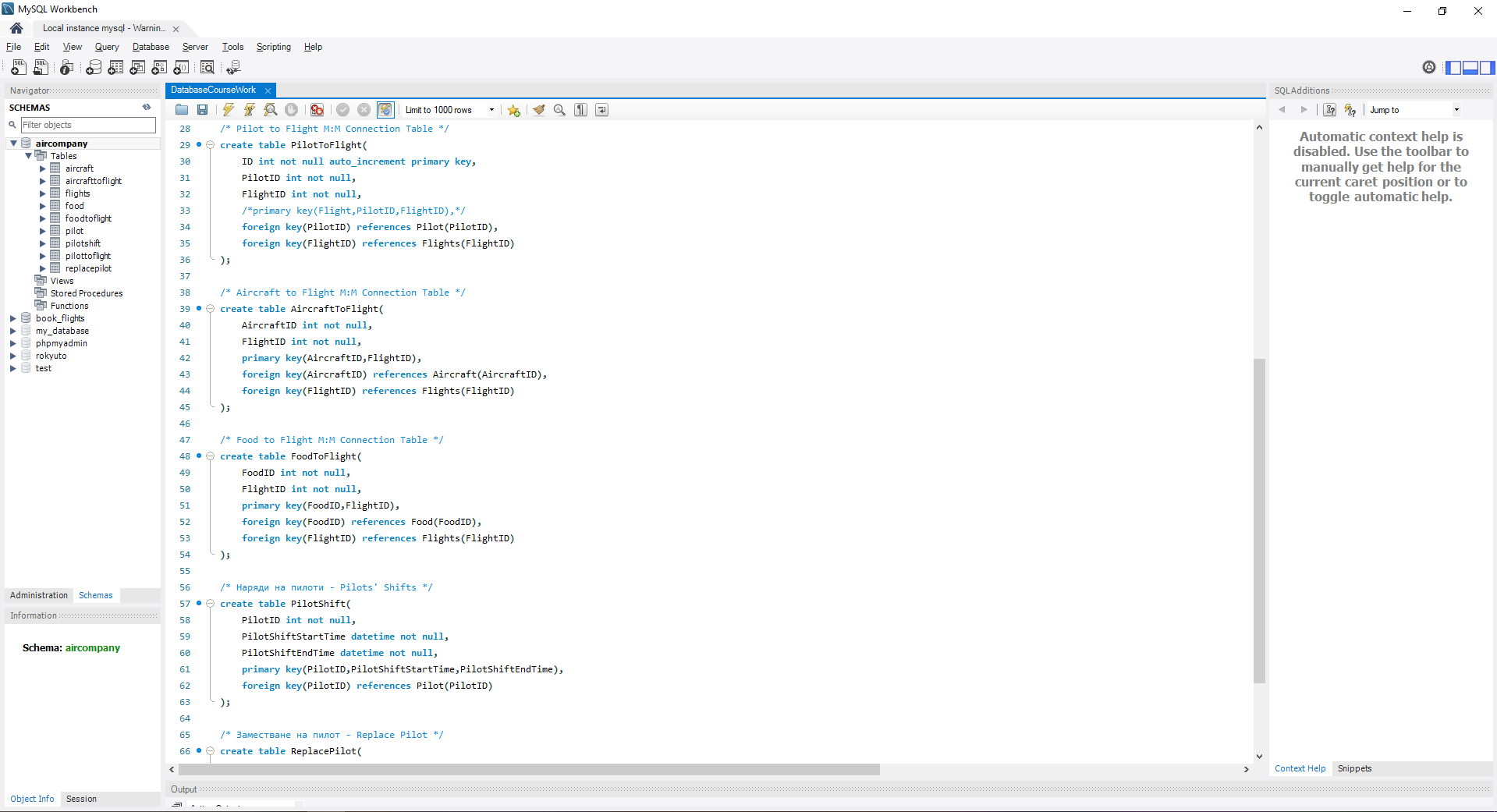
**Задача 1: Да се проектира база от данни и да се представи ER диаграма със съответни CREATE TABLE заявки в средата MySQL .**

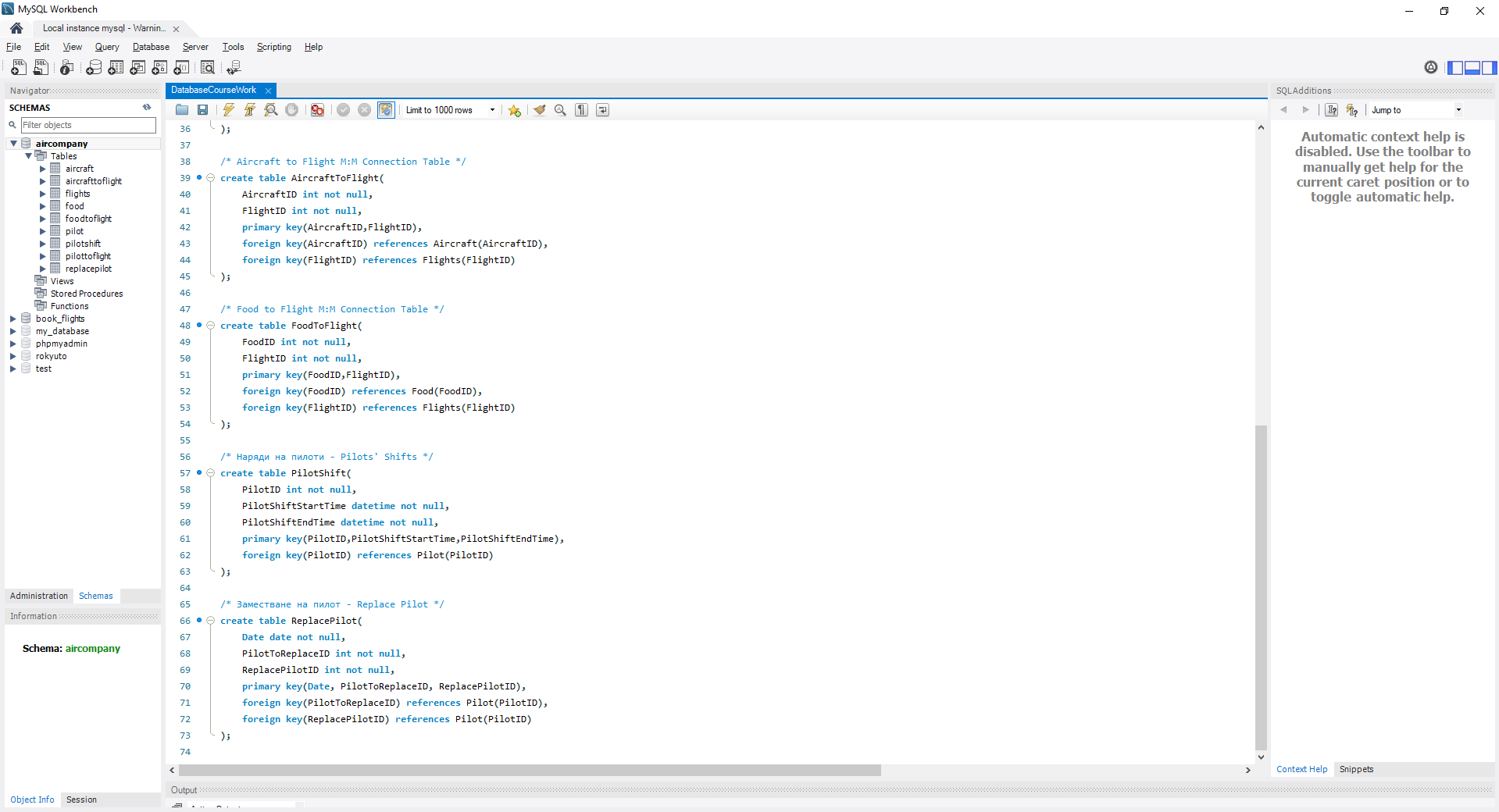
****

Правим CREATE TABLE заявки като създаваме следните таблици:

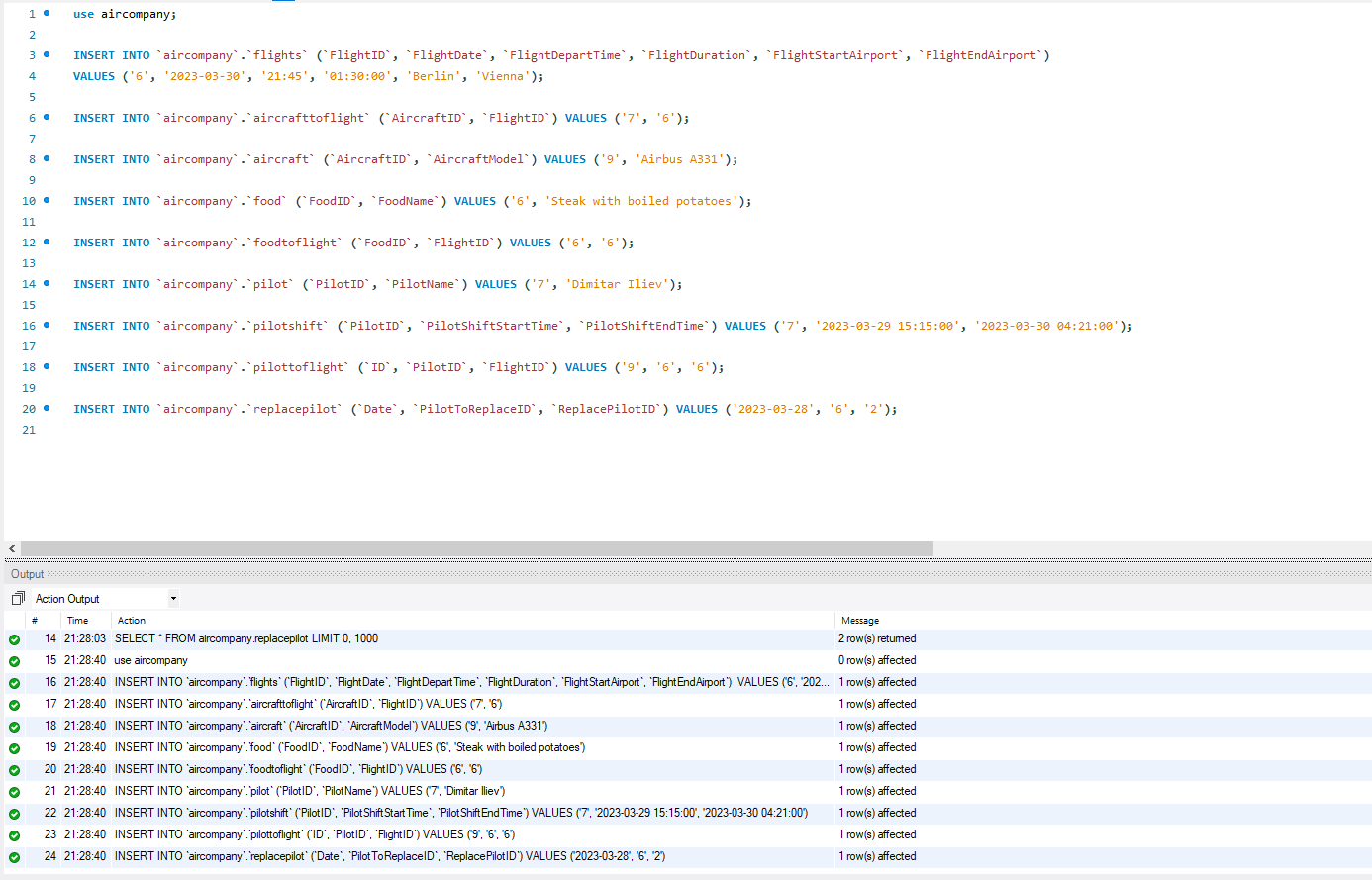
* Полети (flights) – съдържаща ID (primary key), дата на полета (FlightDate), час на излитане (FlightDepartTime), продължителност на полета (FlightDuration), начално (FlightStartAirport) и крайно (FlightEndAirport) летище;
* Пилот (pilot) – съдържаща ID (primary key) и име на пилота (PilotName);
* Междинна таблица (pilottoflight) - за връзка много към много (М:М) между таблиците flights и pilot. Съдържа външни ключове сочещи към съответните primary ключове на таблиците;
* Самолет (aircraft) – съдържаща ID (primary key) и модел на самолета (AircraftModel);
* Междинна таблица (aircrafttoflights) - за връзка много към много (М:М) между таблиците flights и aircraft. Съдържа външни ключове сочещи към съответните primary ключове на таблиците;
* Предлагана храна (food) – съдържаща ID (primary key) и име на предлаганата храна (FoodName);
* Междинна таблица (foodtoflights) - за връзка много към много (М:М) между таблиците flights и food. Съдържа външни ключове сочещи към съответните primary ключове на таблиците;
* Таблица за наряди на пилоти (pilotshift) – съдържа ID (primary key) на пилот като външен ключ сочещ към primary ключа на таблицата pilot (връзка едно към много – 1:М ), начало на наряда (PilotShiftStartTime) и край на наряда (PilotShiftEndTime) на пилота;
* Таблица за заместване на пилот (replacepilot) – съдържа Дата (primary key) на извършеното заместване на пилот и външни ключове PilotToReplaceID и ReplacePilotID, сочещи към ID-та на пилоти от таблицата pilot.

****

****

****

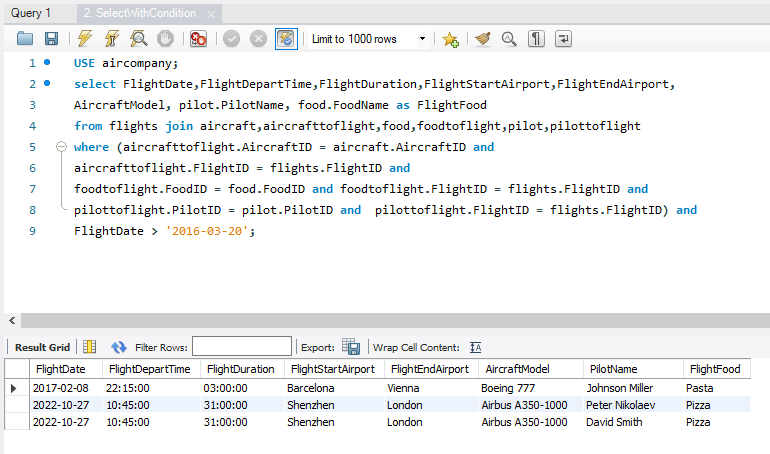
Insert заявките са със следният синтаксис:

****

**Част „Задачи“**

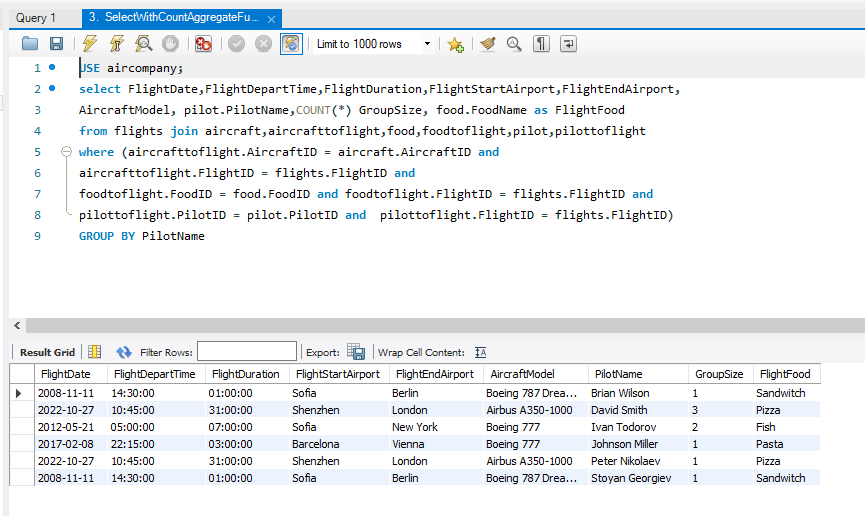
**Задача 2: Напишете заявка, в която демонстрирате SELECT с ограничаващо условие по избор .**

Правим SELECT заявка, включвайки таблиците flights, aircraft, aircrafttoflights, food, foodtoflights, pilot, pilottoflight чрез командите from и join, реализирайки ограничаващо условие: датата на полета да е след 20.03.2016г. чрез where. Демонстрираме подробна информация за полета (дата, час на излитане, продължителност на полета, начално и крайно летище, самолет, пилот и предлагана храна) чрез сравняване на ID ключове на таблиците.



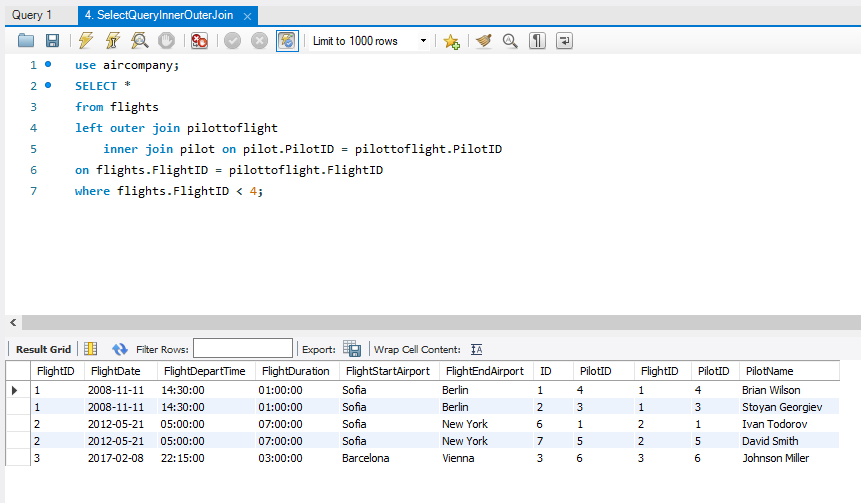
**Задача 3: Напишете заявка, в която използвате агрегатна функция и GROUP BY по ваш избор .**

Правим SELECT заявка, включвайки същите таблици чрез командите from и join, реализирайки групиране по колоната PilotName чрез group by, като представяме броят повторения на всеки пилот с помощта на агрегатната функция count(), именувайки колоната GroupSize. Демонстрираме същата подробна информация за полета чрез сравняване на ID ключове на таблиците.



**Задача 4: Напишете заявка, в която демонстрирате INNER и OUTER JOIN по ваш избор .**

Правим SELECT заявка, върху таблицата flights чрез командата from като присъединяваме към заявката чрез left outer (външно присъединява подадената таблица към първата, връщайки като резултат всички данни при съвпадение в която и да е от двете таблици и подравнявайки резултата по лявата таблица) и inner join (вътрешно присъединяваме подадената таблица към предишната, връщайки като резултат всички съвпадения в двете таблици едновременно) таблиците pilottoflight и pilot. Използваме on клауза за сравняване на ID ключовете на отделните таблици.



**Задача 5: Напишете заявка, в която демонстрирате едновременно JOIN и агрегатна функция .**

Правим SELECT заявка, включвайки flights, food и foodtoflight таблици чрез командите from и join, реализирайки групиране по колоната FoodName чрез group by, като представяме броят повторения на всяка храна с помощта на агрегатната функция count(), именувайки колоната FoodGroupCount. Демонстрираме подробна информация за полета чрез сравняване на ID ключове на таблиците.

