НПГ ПО КТС - ГРАД ПРАВЕЦ

**КУРСОВА РАБОТА**

**по**

**Бази данни**

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

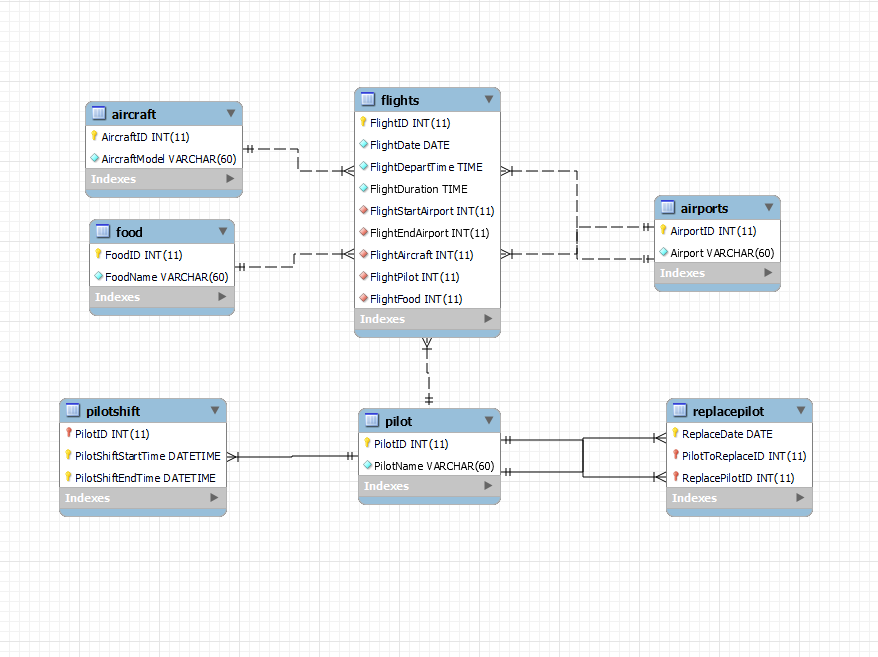
*/ас. инж. Елизабет Михайлова/*

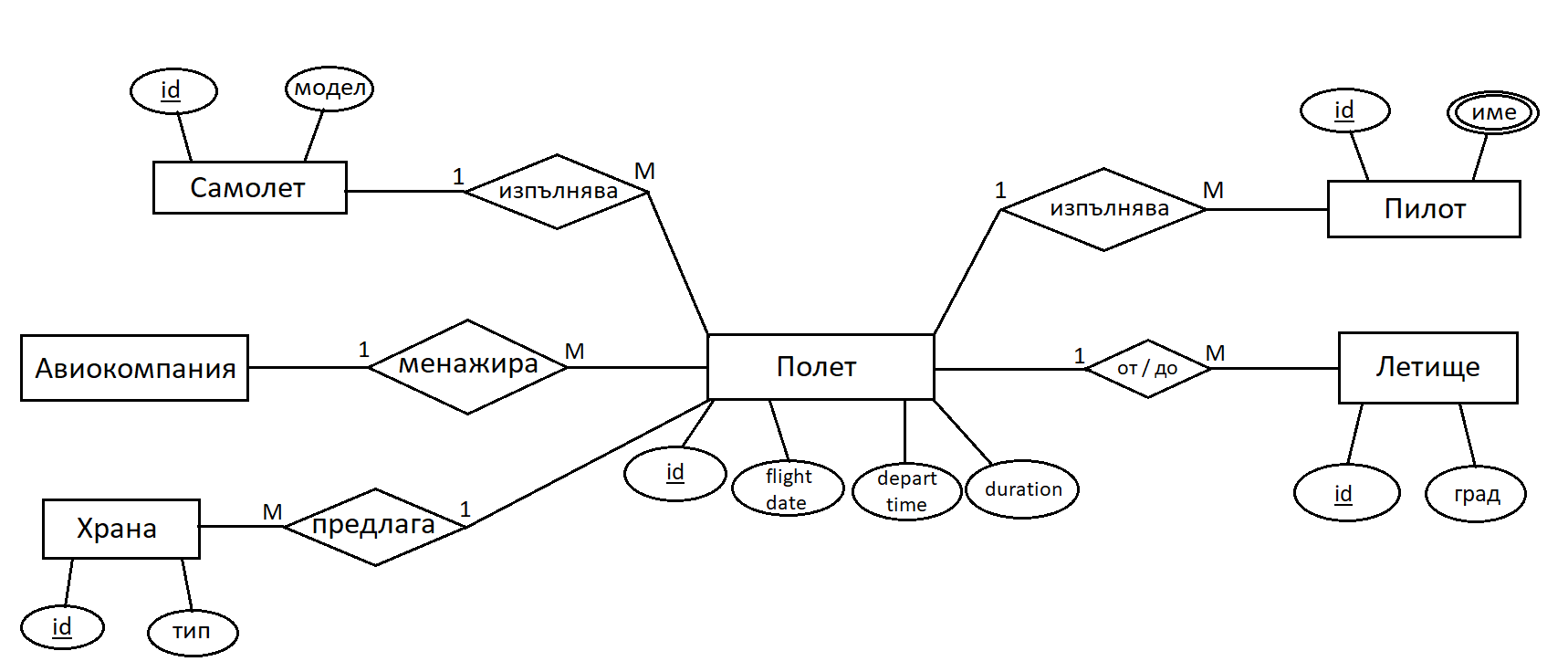
**<11.04.23>**

**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.**

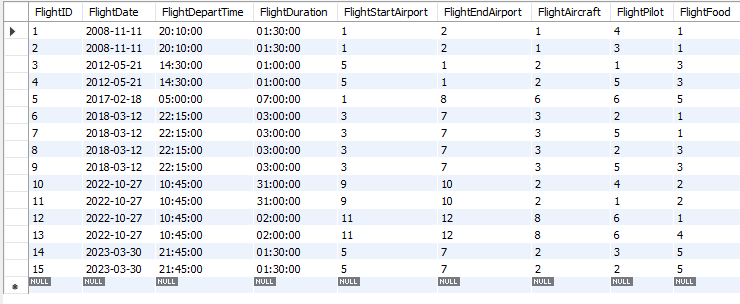
**Задача 1: Да се проектира база от данни и да се представи ER диаграма със съответни CREATE TABLE заявки в средата MySQL .**

****

****

Правим CREATE TABLE заявки като създаваме следните таблици:

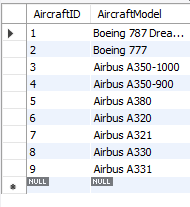
* Полети (flights) – съдържаща ID (primary key), дата на полета (FlightDate), час на излитане (FlightDepartTime), продължителност на полета (FlightDuration), начално (FlightStartAirport) и крайно (FlightEndAirport) летище - външни ключове, сочещи към id-тата на съответните летища от таблицата airport, самолет (FlightAircraft) – външен ключ, сочещ към id-то на съответният самолет от aircraft таблицата, пилот (FlightPilot) – външен кюч, сочещ към id-то на съответният пилот от pilot таблицата, предлагана храна (FlightFood) - външен ключ, сочещ към id-то на съответната храна от таблицата food;



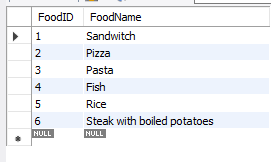
* Летище (airport) – съдържаща ID (primary key) и летище (Airport);



* Самолет (aircraft) – съдържаща ID (primary key) и модел на самолета (AircraftModel);



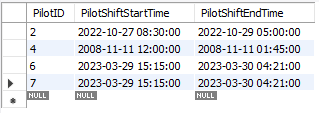
* Предлагана храна (food) – съдържаща ID (primary key) и име на предлаганата храна (FoodName);



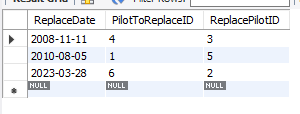
* Пилот (pilot) – съдържаща ID (primary key) и име на пилота (PilotName);

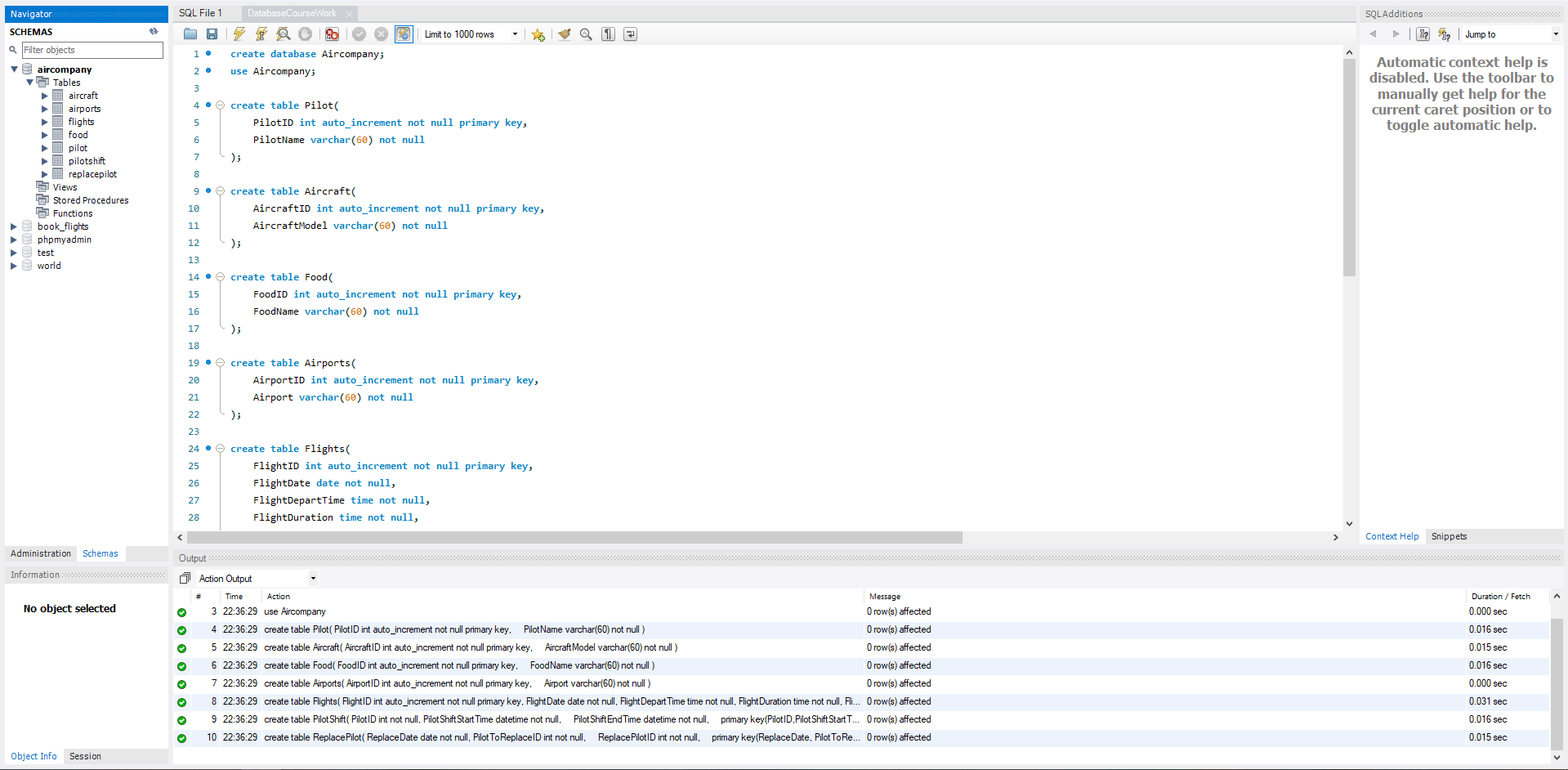


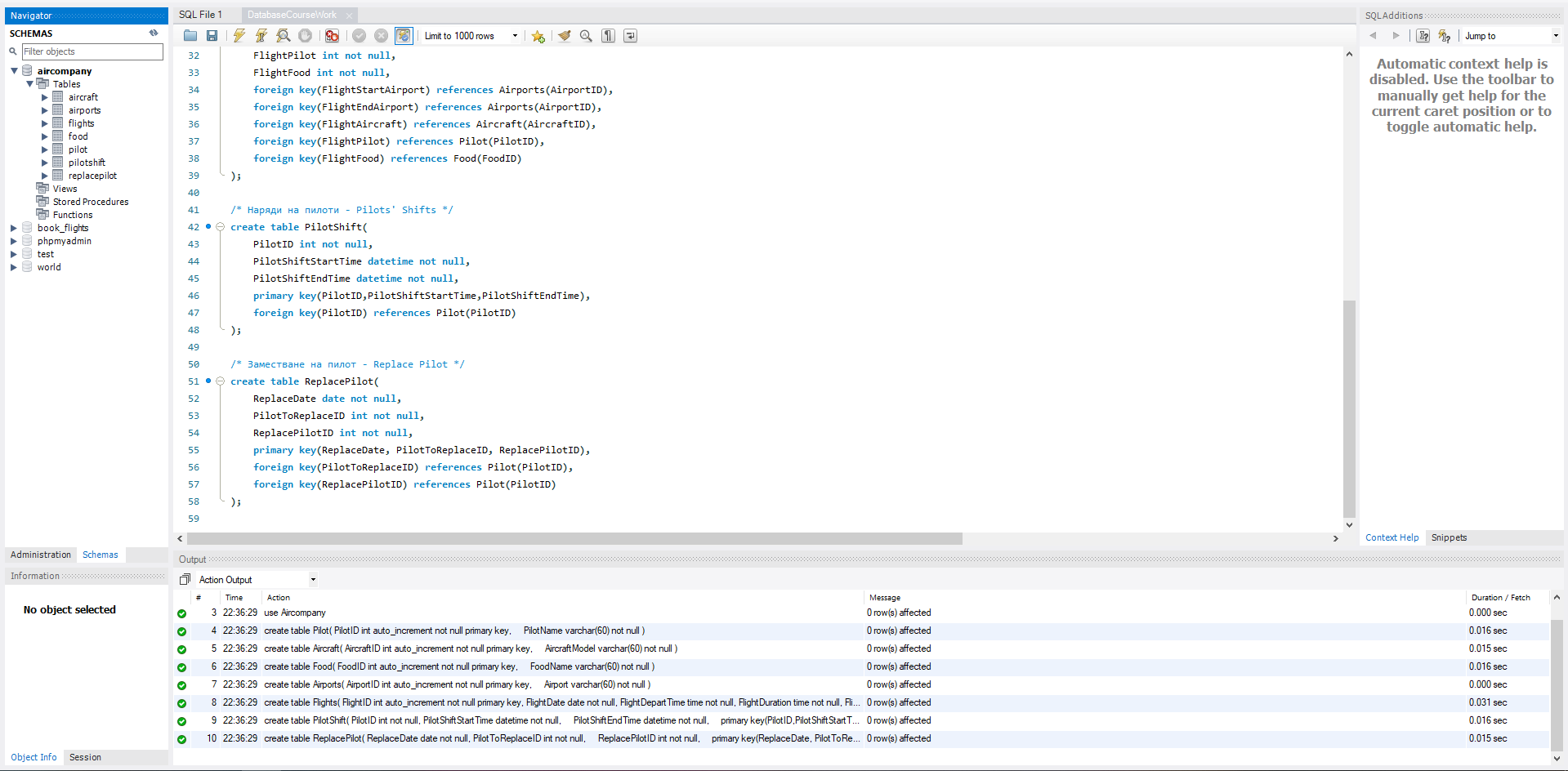
* Таблица за наряди на пилоти (pilotshift) – съдържа ID (primary key) на пилот като външен ключ сочещ към primary ключа на таблицата pilot (връзка едно към много – 1:М ), начало на наряда (PilotShiftStartTime) и край на наряда (PilotShiftEndTime) на пилота;



* Таблица за заместване на пилот (replacepilot) – съдържа Дата на извършеното заместване на пилот и външни ключове PilotToReplaceID и ReplacePilotID, сочещи към ID-та на пилоти от таблицата pilot. И трите полета са общ primary key.



****

****

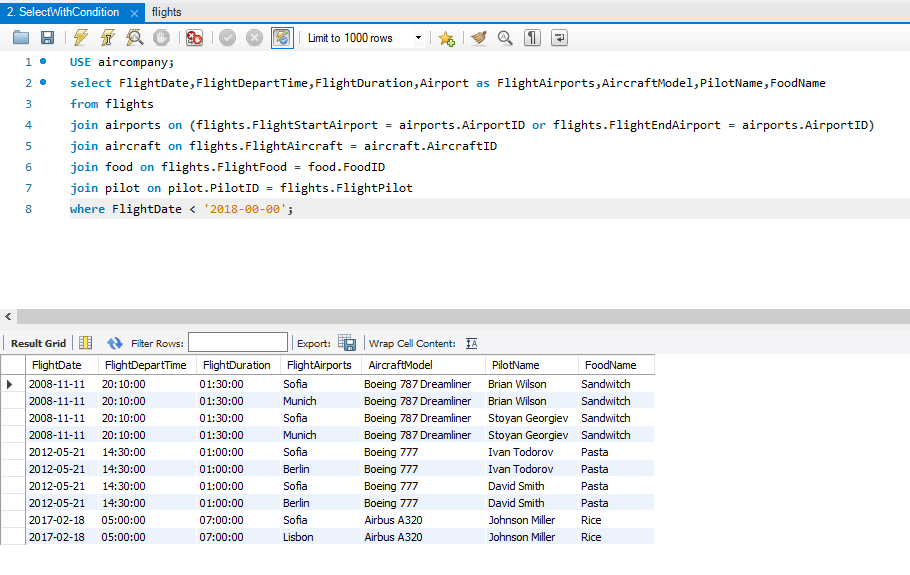
Insert заявките са със следният синтаксис:

****

**Част „Задачи“**

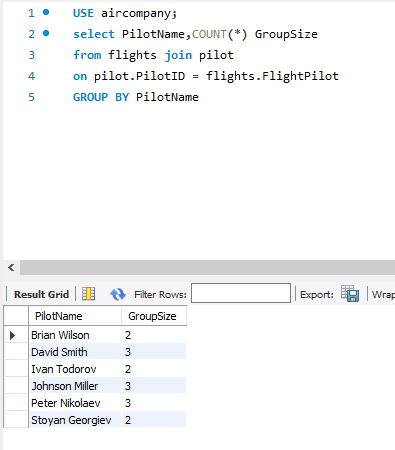
**Задача 2: Напишете заявка, в която демонстрирате SELECT с ограничаващо условие по избор .**

Правим SELECT заявка, включвайки таблиците flights, aircraft, airports, food и pilot чрез командите from и join, реализирайки ограничаващо условие: датата на полета да е преди 2018г. чрез where. Демонстрираме подробна информация за полета (дата, час на излитане, продължителност на полета, начално и крайно летище (FlightAirports – 1-ви ред показва летището на излитане, а 2-ри ред показва летището на кацане), самолет, пилот и предлагана храна) чрез сравняване на ID ключове на join таблиците с on клауза.



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

Правим SELECT заявка, включвайки flights и pilot 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 информация за името на пилота и колко пъти името му се среща в таблицата flights чрез сравняване на ID ключове на таблиците.

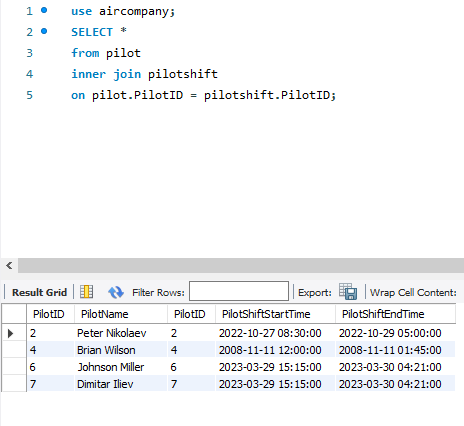
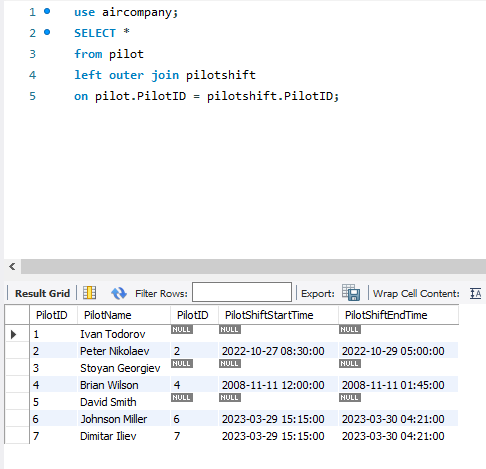


**Задача 4: Напишете заявка, в която демонстрирате INNER и OUTER JOIN по ваш избор .**

Правим SELECT заявки, върху таблицата pilot чрез командата from като присъединяваме към заявката чрез:

* inner join (вътрешно присъединяваме подадената таблица към предишната, връщайки като резултат само съвпаденията в двете таблици - общото)
* 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)

таблицата pilotshift. 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 и food 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 FoodGroupSize. Демонстрираме информация за името на храната и размера на групата чрез сравняване на ID ключове на таблиците.

