**Ministerul Educaţiei și Cercetării al Republicii Moldova**

**Universitatea Tehnică a Moldovei**

**Facultatea Calculatoare, Informatică și Microelectronică**

**RAPORT**

Lucrarea de laborator nr.10

Disciplina: *Baza de date*

Tema**:** Crearea si actualizarea tabelelor in SQL.

**A efectuat: Vlasitchi Stefan**

**st. gr. TI-216**

**A verificat:**

**asist. univ. Cebotar Gabriela**

**Chișinău - 2023**

**Lucrare de laborator nr. 10**

**Sarcina 1:**

În cadrul acestei sarcini, am asigurat diversitatea tipurilor de date în definirea atributelor. Am utilizat tipuri precum INT pentru numere întregi (de exemplu, PilotID), VARCHAR(255) pentru șiruri de caractere (cum ar fi FirstName și LastName), DATE pentru date calendaristice (DOB, LicenseExpireDate), DATETIME pentru date și timp (DateTimeLeave, DateTimeArrive), și DECIMAL(10,2) pentru numere cu zecimale (Fare, ServicePrice).

-- Definirea atributelor cu tipuri de date diverse

CREATE TABLE Pilot (

PilotID INT IDENTITY(1,1) PRIMARY KEY,

FirstName VARCHAR(255),

LastName VARCHAR(255),

DOB DATE,

HoursFlown INT,

LicenseNo INT UNIQUE,

FOREIGN KEY (LicenseNo) REFERENCES License(LicenseNo)

);

-- Alte tabele cu diverse tipuri de date

CREATE TABLE PlaneModel (

ModelNumber INT IDENTITY(1,1) PRIMARY KEY,

ManufacturerName VARCHAR(255),

-- Alte atribute cu tipuri de date diverse

);

CREATE TABLE License (

LicenseNo INT PRIMARY KEY,

LicenseType VARCHAR(255),

LicenseExpireDate DATE

);

-- ... (Restul tabelelor cu tipuri de date diverse)

**Sarcina 2:**

Am inclus proprietățile IDENTITY pentru autoincrementare în script. De exemplu, coloane precum PilotID, ModelNumber, sau PassengerID sunt definite cu IDENTITY(1,1). De asemenea, am introdus o coloană calculată, cum ar fi Age în tabela PassengerDetails, care se actualizează automat bazându-se pe data de naștere a pasagerului.

-- Sarcina 2

-- Exemple de utilizare a IDENTITY și Coloane Calculate

CREATE TABLE PassengerDetails (

PassengerID INT IDENTITY(1,1) PRIMARY KEY,

PassengerFirstName VARCHAR(255),

PassengerLastName VARCHAR(255),

PassengerDOB DATE,

PassportNo VARCHAR(255) UNIQUE,

Age AS (DATEDIFF(year, PassengerDOB, GETDATE()))

-- Alte atribute cu diverse tipuri de date

);

**Sarcina 3:**

Scriptul conține toate tipurile de constrângeri enumerate. Cheile primare (PRIMARY KEY) sunt definite pentru coloanele precum ModelNumber, LicenseNo, PilotID, FlightNo etc. Unicitatea (UNIQUE) este asigurată pentru coloanele LicenseNo, RegistrationNo, PassportNo. Constrângerile NOT NULL sunt aplicate în principal pe cheile primare și externe. De exemplu, ModelNumber, PilotID, FlightNo etc. nu pot accepta valori nule. Există, de asemenea, o constrângere CHECK în tabela Flight care verifică dacă FlightArriveFrom nu este egal cu FlightDepartTo.

-- Sarcina 3

-- Exemple de constrângeri

CREATE TABLE Flight (

FlightNo INT IDENTITY(1,1),

FlightDate DATE,

FlightDepartTo VARCHAR(255),

FlightArriveFrom VARCHAR(255),

Distance INT,

CHECK (FlightArriveFrom != FlightDepartTo),

PRIMARY KEY (FlightNo, FlightDate)

-- Alte constrângeri precum PRIMARY KEY, UNIQUE, NOT NULL, FOREIGN KEY

-- sunt prezente în celelalte tabele conform cerințelor.

**Sarcina 4:**

Despre cheile compuse și cheile externe compuse, scriptul include un exemplu în tabela InstanceAttendant, unde ambele coloane InstanceID și AttendantID formează o cheie externă compusă care face referire la cheile primare din tabelele FlightInstance și FlightAttendant, respectiv.

-- Sarcina 4

-- Exemple de chei compuse și chei externe compuse

CREATE TABLE InstanceAttendant (

InstanceID INT,

AttendantID INT,

PRIMARY KEY (InstanceID, AttendantID),

FOREIGN KEY (InstanceID) REFERENCES FlightInstance(InstanceID),

FOREIGN KEY (AttendantID) REFERENCES FlightAttendant(AttendantID)

**Sarcina 5:**

Tipurile de asocieri (1:1, 1:N, N:M) sunt prezente în script. De exemplu, relația 1:1 între FlightInstance și FlightAttendant prin intermediul coloanei FSM\_AttendantID. Relația 1:N între Flight și FlightInstance prin cheia externă FlightNo și FlightDate. Relația N:M este ilustrată în tabela InstanceAttendant, legând FlightInstance și FlightAttendant.

Aceste elemente sunt integrate în scriptul SQL, asigurând îndeplinirea cerințelor specifice fiecărei sarcini.

-- Sarcina 5

-- Tipuri de asocieri 1:1, 1:N, N:M

-- Exemple de relații

-- 1:1 între FlightInstance și FlightAttendant

-- 1:N între Flight și FlightInstance

-- N:M în tabela InstanceAttendant

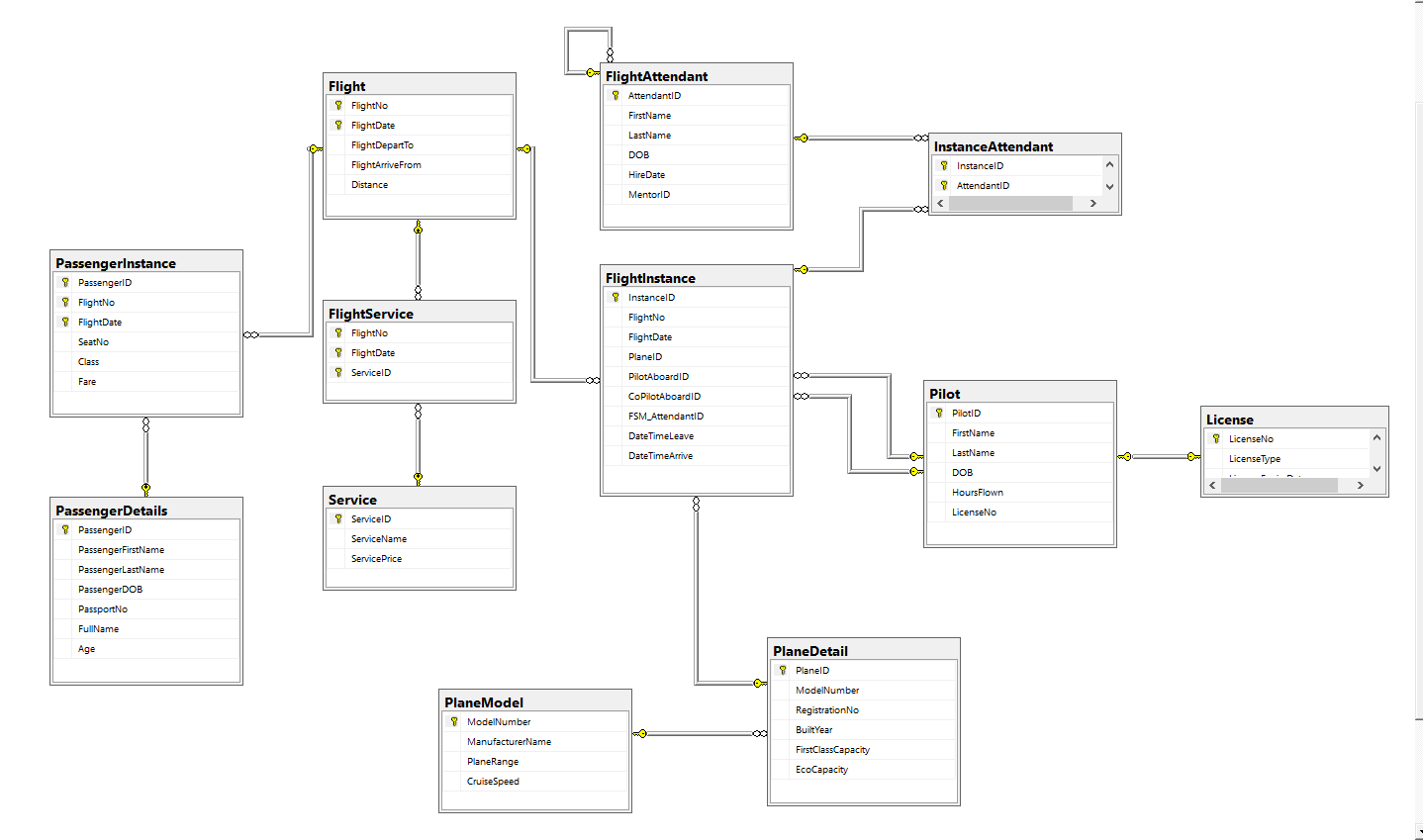


Figura 1 - Diagrama bazei de date AirLine2

**Concluzie**

In cadrul acestui laborator, am dezvoltat o baza de date pentru o linie aeriana, respectand diversitatea tipurilor de date si includerea de constrangeri precum PRIMARY KEY, FOREIGN KEY, UNIQUE, CHECK, si DEFAULT. Am implementat si chei simple si compuse, precum si tipuri variate de asociere (1:1, 1:n, n:m). Am populat baza de date cu informatii de test folosind comenzi INSERT in SQL pentru a crea 15 de tupluri sau mai multe in fiecare relatie. Scriptul creeaza tabele pentru modelele de avioane, piloti, detalii ale avioanelor, pasageri, zboruri, instante de pasageri, insotitori de zbor, instante de zbor, si asocieri intre instante si insotitori. Scopul acestui proiect este sa ofere o structura solida pentru o aplicatie de management al unei linii aeriene, cu date realiste si relevante pentru domeniu.