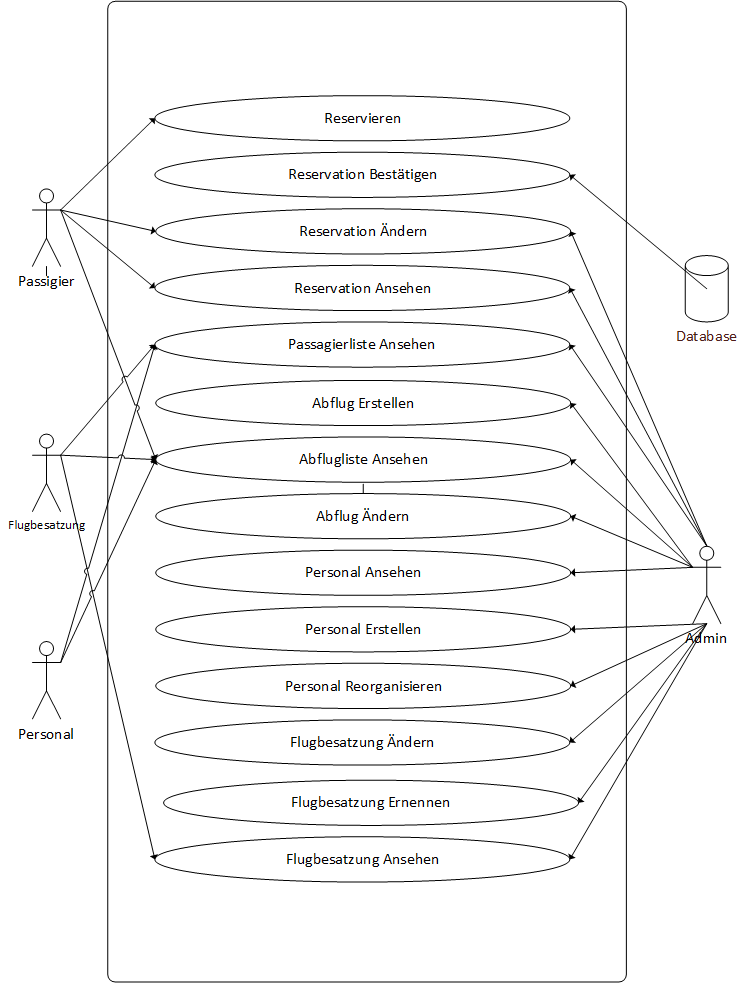
**1.EINLEITUNG**

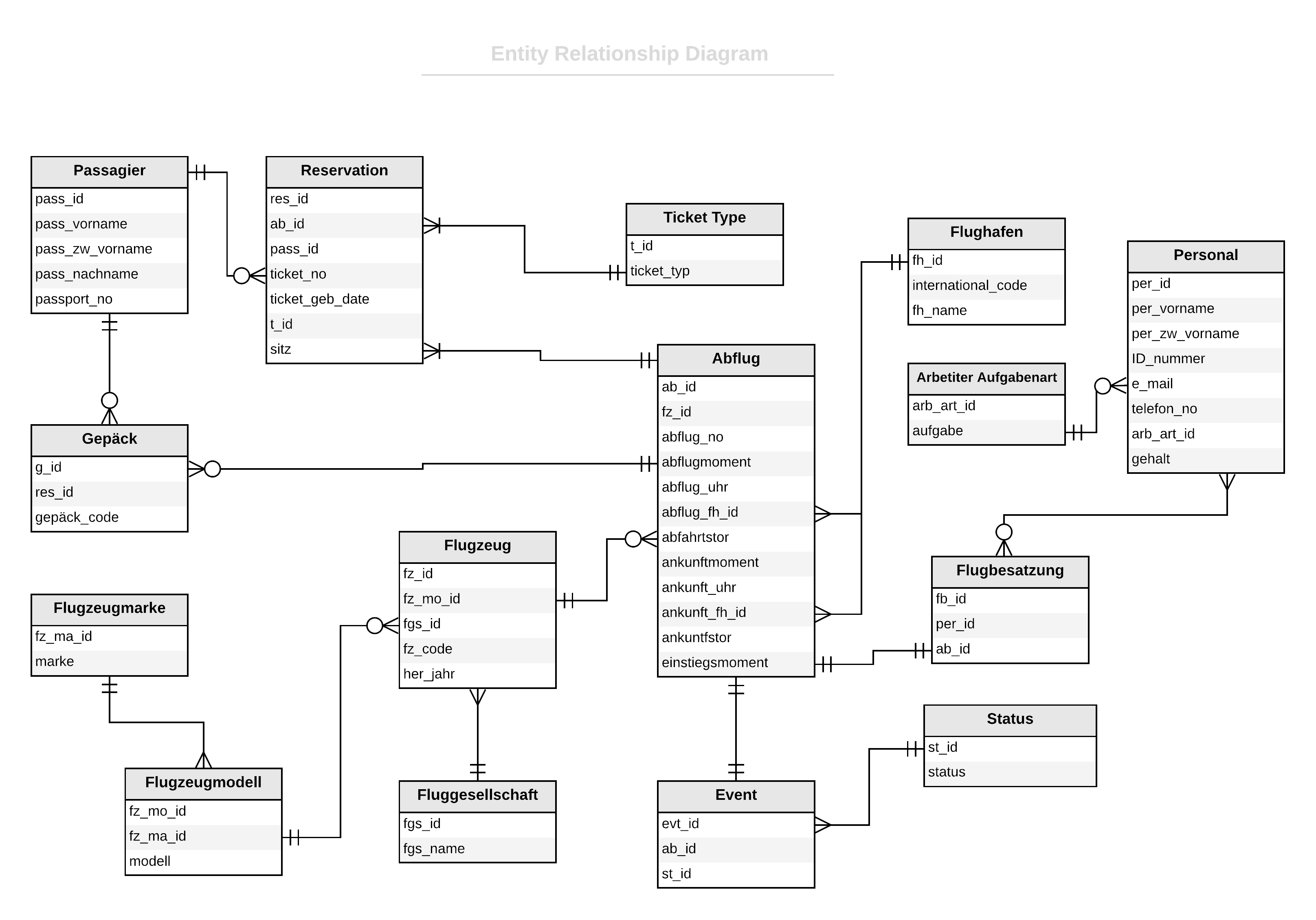
Bei diesem Projekt wurde ein einfache Airline-System Database entwickelt.

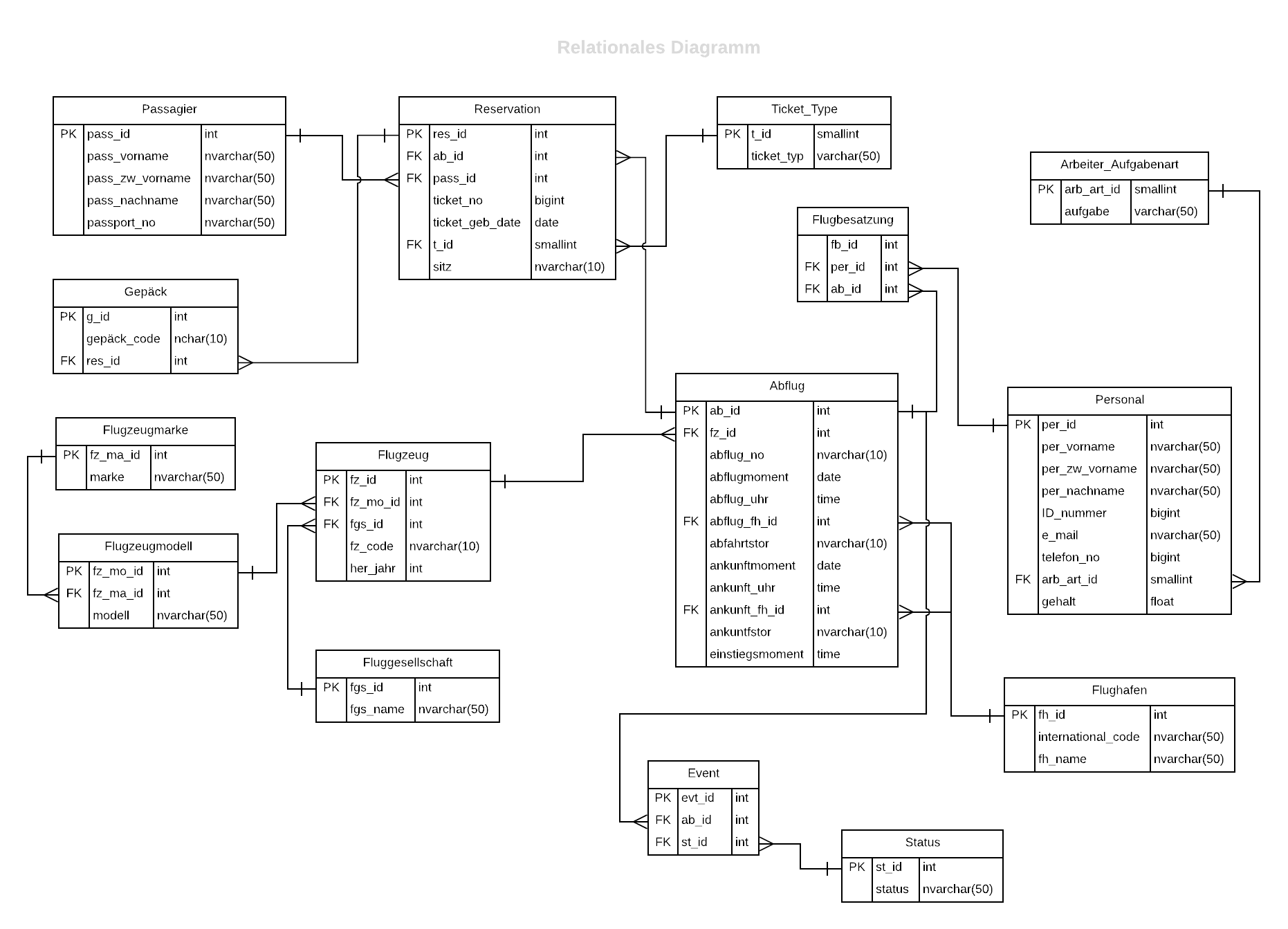
Bei erstem Teil haben wir die Dinge geplant, die sich in einem Airline-System befinden sollten, und zwar mit unseren Recherchen und unseren eigenen Erfahrungen.

Bei zweitem Teil wurde das Diagramm skizzieren. Für Diagramm wurde Lucidchart.com und MS Visio verwandet.

Bei der Vorbereitung dieses Projekts dachten wir, dass alle Fluggesellschaften und Flughäfen ein gemeinsames Datenbanksystem verwenden. Aus diesem Grund haben wir nur eine Tabelle für alle Fluggesellschaften und das Flughafenpersonal erstellt.

**2.ANWENDUNGSFALLDIAGRAMM**

**3.E-R DIAGRAMM**

**4.RELATIONALES MODELL**

1) Die folgende Abfrage zeigt den Status der allen Flüge.

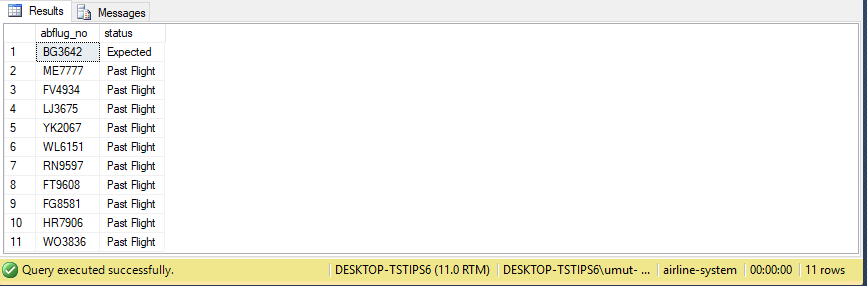
SELECT dbo.Abflug.abflug\_no, dbo.Status.status

FROM dbo.Abflug

INNER JOIN dbo.Event ON dbo.Abflug.ab\_id = dbo.Event.ab\_id

INNER JOIN dbo.Status ON dbo.Event.st\_id = dbo.Status.st\_id

ORDER BY dbo.Abflug.ab\_id DESC;



2) Die folgende Abfrage zeigt alle Passagierlisten von allen Flug, die Sortierung nach Tickettype.

SELECT dbo.Abflug.abflug\_no, dbo.Passagier.pass\_vorname, dbo.Passagier.pass\_zw\_vorname, dbo.Passagier.pass\_nachname, dbo.Reservation.ticket\_no, dbo.Reservation.ticket\_geb\_date, dbo.Ticket\_Type.ticket\_typ, dbo.Reservation.sitz

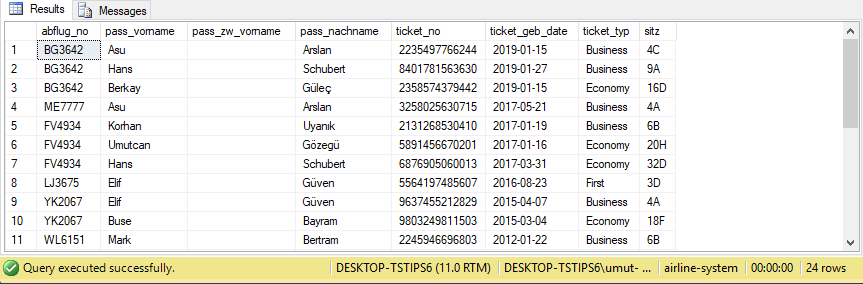
FROM dbo.Passagier

INNER JOIN dbo.Reservation ON dbo.Passagier.pass\_id = dbo.Reservation.pass\_id

INNER JOIN dbo.Ticket\_Type ON dbo.Reservation.t\_id = dbo.Ticket\_Type.t\_id

INNER JOIN dbo.Abflug ON dbo.Reservation.ab\_id = dbo.Abflug.ab\_id

ORDER BY dbo.Abflug.ab\_id DESC, dbo.Ticket\_Type.t\_id ASC;



3) Die folgende Abfrage zeigt die Liste der Flugzeug der Fluggesellschaften.

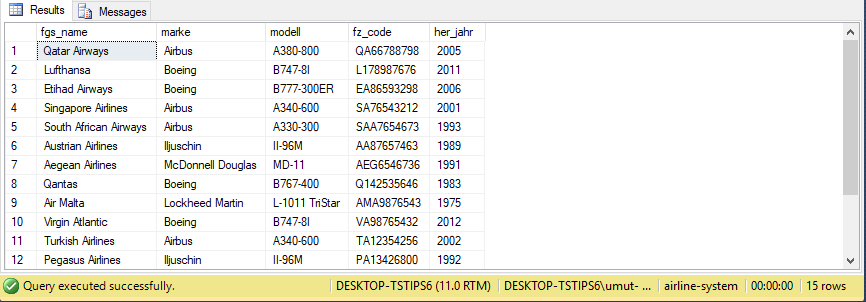
SELECT dbo.Fluggesellschaft.fgs\_name, dbo.Flugzeugmarke.marke, dbo.Flugzeugmodell.modell, dbo.Flugzeug.fz\_code, dbo.Flugzeug.her\_jahr

FROM dbo.Fluggesellschaft

INNER JOIN dbo.Flugzeug ON dbo.Fluggesellschaft.fgs\_id = dbo.Flugzeug.fgs\_id

INNER JOIN dbo.Flugzeugmodell ON dbo.Flugzeug.fz\_mo\_id = dbo.Flugzeugmodell.fz\_mo\_id

INNER JOIN dbo.Flugzeugmarke ON dbo.Flugzeugmodell.fz\_ma\_id = dbo.Flugzeugmarke.fz\_ma\_id;



4) Die folgende Abfrage zeigt alle Flugbesatzungen vom alten Flug zum neuen Flug.

SELECT dbo.Abflug.abflug\_no, dbo.Personal.per\_vorname, dbo.Personal.per\_zw\_vorname, dbo.Personal.per\_nachname, dbo.Arbeiter\_Aufgabenart.aufgabe

FROM dbo.Abflug

INNER JOIN dbo.Flugbesatzung ON dbo.Abflug.ab\_id = dbo.Flugbesatzung.ab\_id

INNER JOIN dbo.Personal ON dbo.Flugbesatzung.per\_id = dbo.Personal.per\_id

INNER JOIN dbo.Arbeiter\_Aufgabenart ON dbo.Personal.arb\_art\_id = dbo.Arbeiter\_Aufgabenart.arb\_art\_id;



5) Die folgende Abfrage zeigt die Gepäckliste anhand der Ticketnummer der Passagiere vom neuen Flug zum ältesten Flug.

SELECT dbo.Gepäck.gepäck\_code, dbo.Reservation.ticket\_no, dbo.Passagier.passport\_no, dbo.Abflug.abflug\_no

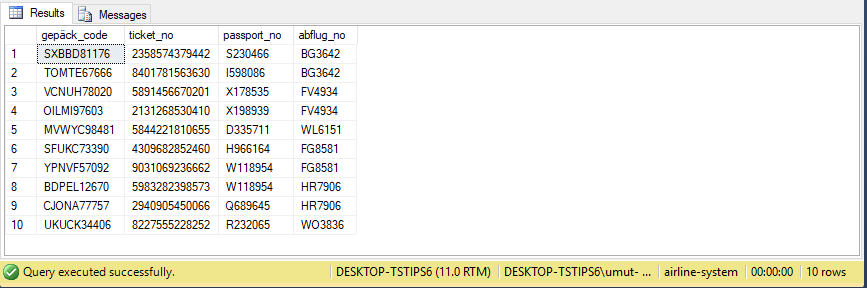
FROM dbo.Abflug

INNER JOIN dbo.Reservation ON dbo.Abflug.ab\_id = dbo.Reservation.ab\_id

INNER JOIN dbo.Gepäck ON dbo.Reservation.res\_id = dbo.Gepäck.res\_id

INNER JOIN dbo.Passagier ON dbo.Reservation.pass\_id = dbo.Passagier.pass\_id

ORDER BY dbo.Abflug.ab\_id DESC, dbo.Reservation.res\_id ASC;



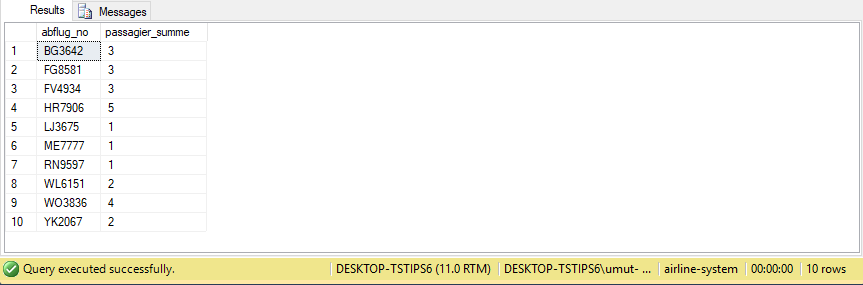
6) Die folgende Abfrage zeigt die Anzahl der Passagiere auf allen bisher durchgeführten Flügen.

SELECT dbo.Abflug.abflug\_no, COUNT(\*) AS passagier\_summe

FROM dbo.Abflug

INNER JOIN dbo.Reservation ON dbo.Abflug.ab\_id = dbo.Reservation.ab\_id

GROUP BY abflug\_no;



7) Die folgende Abfrage zeigt die Gehälter von Mitarbeitern, die für einen Flug über 4000€ eingesetzt werden können.

SELECT ID\_nummer, aufgabe, gehalt

FROM Personal

INNER JOIN Arbeiter\_Aufgabenart ON dbo.Personal.arb\_art\_id=dbo.Arbeiter\_Aufgabenart.arb\_art\_id

WHERE dbo.Personal.arb\_art\_id

IN(

SELECT arb\_art\_id

FROM Arbeiter\_Aufgabenart

WHERE Arbeiter\_Aufgabenart.aufgabe = 'Captain Pilot'

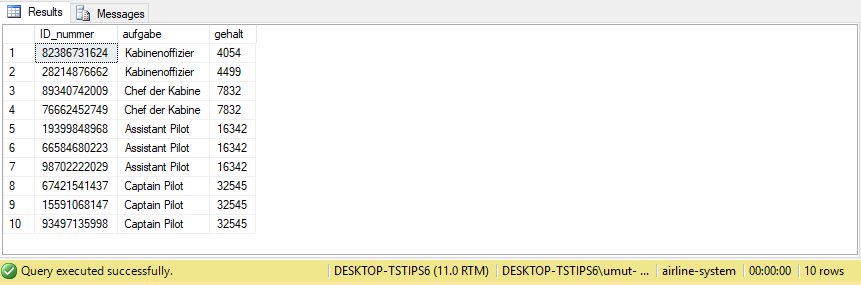
OR Arbeiter\_Aufgabenart.aufgabe = 'Assistant Pilot'

OR Arbeiter\_Aufgabenart.aufgabe = 'Chef der Kabine'

OR Arbeiter\_Aufgabenart.aufgabe = 'Kabinenoffizier'

)AND gehalt>4000

ORDER BY gehalt ASC;



8) Die folgende Abfrage listet Passagiere auf, die noch nie gebucht haben.

SELECT pass\_vorname + ' ' + pass\_zw\_vorname + ' ' + pass\_nachname AS pass\_name , passport\_no

FROM Passagier

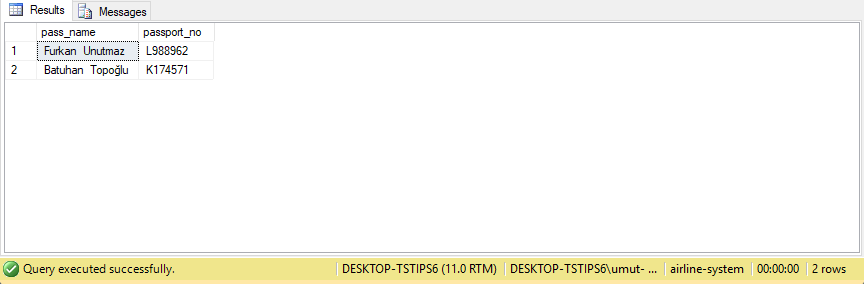
WHERE NOT EXISTS (

SELECT pass\_id

FROM Reservation

WHERE dbo.Passagier.pass\_id=dbo.Reservation.pass\_id

);



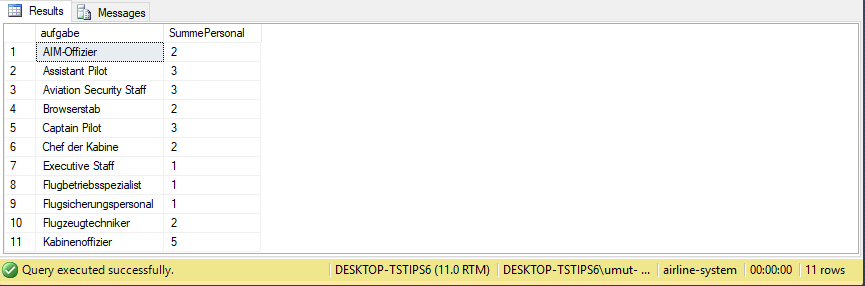
9) Die folgende Abfrage listet die Anzahl der Mitarbeiter nach Rang.

SELECT aufgabe, COUNT(\*) AS SummePersonal

FROM Personal

INNER JOIN Arbeiter\_Aufgabenart ON dbo.Personal.arb\_art\_id=dbo.Arbeiter\_Aufgabenart.arb\_art\_id

GROUP BY aufgabe;



10) Die folgende Abfrage listet die Gesamtzahl der Flugzeuge in allen Fluggesellschaften nach Flugzeugmodellen auf.

SELECT dbo.Flugzeugmarke.marke + ' ' + dbo.Flugzeugmodell.modell

AS FlugzeugModell,

(

SELECT COUNT(fz\_mo\_id)

FROM Flugzeug

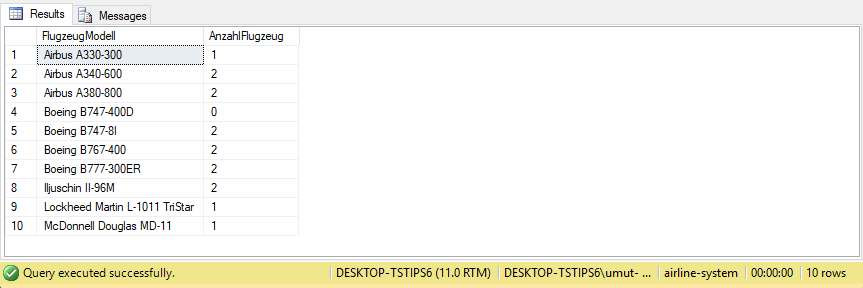
WHERE Flugzeug.fz\_mo\_id = Flugzeugmodell.fz\_mo\_id

)AS AnzahlFlugzeug

FROM Flugzeugmarke

INNER JOIN Flugzeugmodell ON dbo.Flugzeugmodell.fz\_ma\_id=dbo.Flugzeugmarke.fz\_ma\_id

ORDER BY FlugzeugModell ASC;



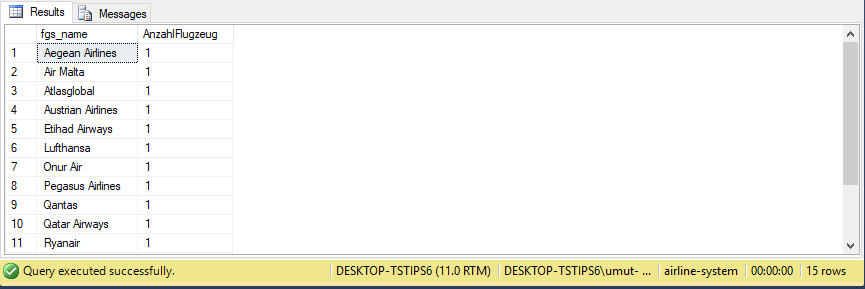
11) Die folgende Abfrage listet die Anzahl der Flugzeuge der Fluggesellschaft auf.

SELECT fgs\_name, COUNT (\*) AS AnzahlFlugzeug

FROM Flugzeug

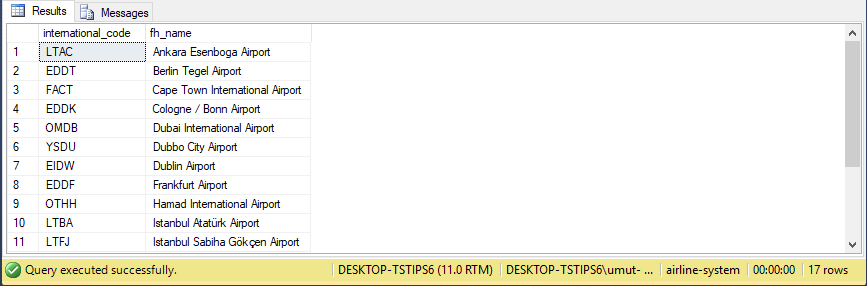
INNER JOIN Fluggesellschaft ON dbo.Fluggesellschaft.fgs\_id=dbo.Flugzeug.fgs\_id

GROUP BY fgs\_name;



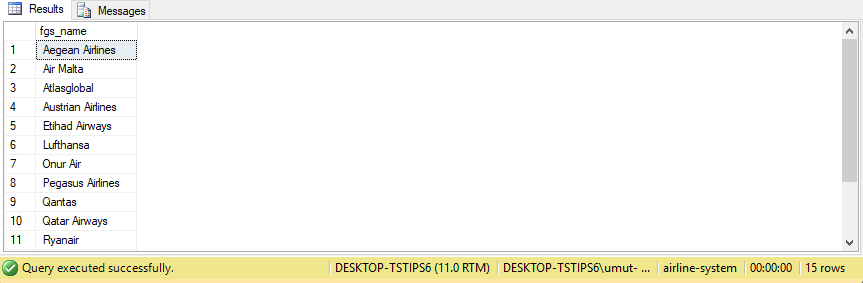
12) Die folgende Abfrage listet Flughäfen auf.

SELECT international\_code, fh\_name FROM dbo.Flughafen ORDER BY fh\_name ASC;



13) Die folgende Abfrage listet Fluggesellschaften auf.

SELECT fgs\_name FROM dbo.Fluggesellschaft ORDER BY fgs\_name ASC;



14) Die folgende Abfrage listet die Gesamtzahl der Gepäckstücke bei Flügen auf.

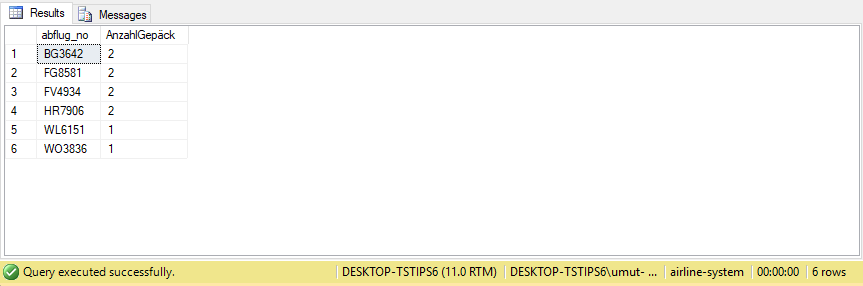
SELECT dbo.Abflug.abflug\_no, COUNT(\*) AS AnzahlGepäck

FROM dbo.Abflug

INNER JOIN dbo.Reservation ON dbo.Abflug.ab\_id = dbo.Reservation.ab\_id

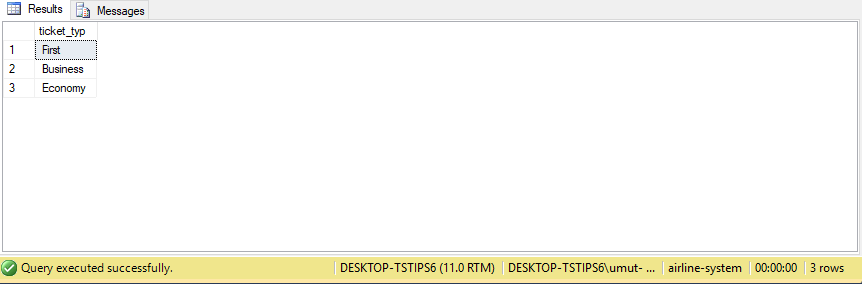
INNER JOIN dbo.Gepäck ON dbo.Reservation.res\_id = dbo.Gepäck.res\_id

GROUP BY abflug\_no;



15) Die folgende Abfrage zeigt die Tickettypen.

SELECT ticket\_typ FROM dbo.Ticket\_Type

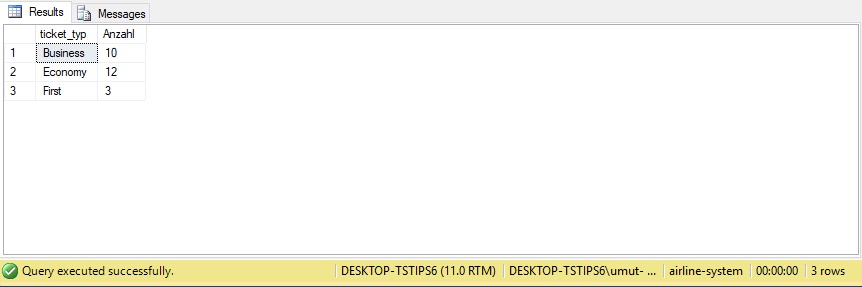


16) Die folgende Abfrage listet die Anzahl der nach Ticketarten verkauften Tickets auf.

SELECT ticket\_typ, COUNT(\*) AS Anzahl FROM dbo.Ticket\_Type

INNER JOIN Reservation ON dbo.Reservation.t\_id = dbo.Ticket\_Type.t\_id

GROUP BY ticket\_typ;



17) Die folgende Abfrage zeigt die letzten 10 Flüge.

SELECT TOP 10 abflug\_no,

abflug\_flughafen.fh\_name AS 'Abflug Flughafen',

abflugmoment, abflug\_uhr,

ankunft\_flughafen.fh\_name AS 'AnkunftFlughafen',

ankunftmoment, ankunft\_uhr

FROM Abflug

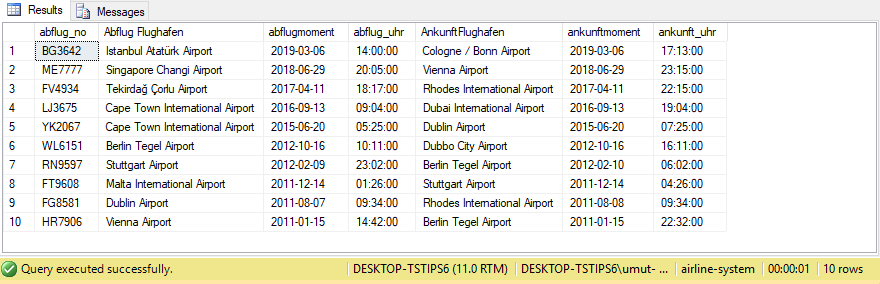
INNER JOIN Flughafen AS abflug\_flughafen

On abflug\_flughafen.fh\_id = Abflug.abflug\_fh\_id

Inner JOIN Flughafen AS ankunft\_flughafen

ON Abflug.ankunft\_fh\_id= ankunft\_flughafen.fh\_id

ORDER BY Abflug.ab\_id DESC;

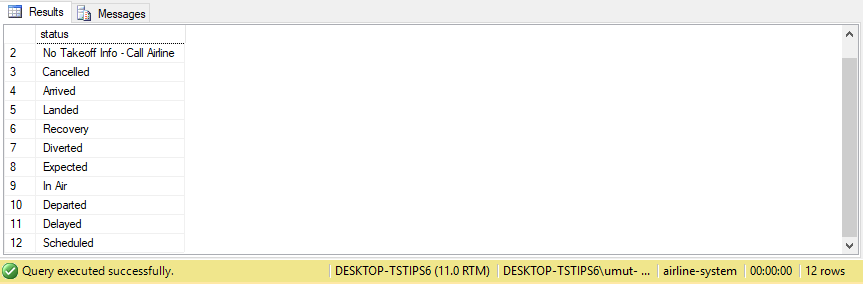


18) Die folgende Abfrage listet alle Situationen auf, die bei Flügen auftreten können.

SELECT status

FROM dbo.Status

ORDER BY st\_id DESC;



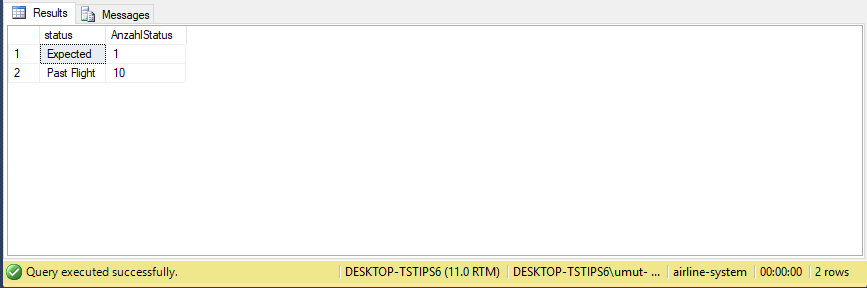
19) Die folgende Abfrage zeigt, wie oft der Flugstatus bei allen Flügen aufgetreten ist.

SELECT status, COUNT(\*) AS AnzahlStatus

FROM dbo.Status

INNER JOIN dbo.Event ON dbo.Event.st\_id = dbo.Status.st\_id

GROUP BY dbo.Status.status;



20) Die folgende Abfrage listet alle Personalinformationen auf.

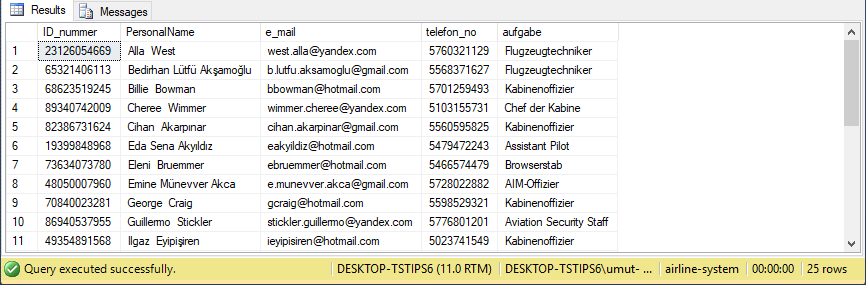
SELECT ID\_nummer, per\_vorname + ' ' + per\_zw\_vorname + ' ' + per\_nachname AS PersonalName, e\_mail, telefon\_no, aufgabe

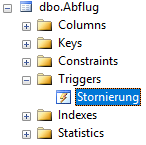
FROM dbo.Personal

INNER JOIN dbo.Arbeiter\_Aufgabenart ON

dbo.Personal.arb\_art\_id = dbo.Arbeiter\_Aufgabenart.arb\_art\_id

ORDER BY PersonalName ASC;



21) Die folgende Abfrage erstellt das "TRIGGER", das aktiviert wird, wenn "DELETE" in Tabelle "Abflug" abgefragt wird.

CREATE TRIGGER Stornierung

ON dbo.Abflug

INSTEAD OF DELETE

AS UPDATE dbo.Event

SET dbo.Event.st\_id= 10

WHERE dbo.Event.ab\_id IN(SELECT ab\_id FROM deleted)

22) Die folgende Abfrage löscht die Zeile mit der Abflug-ID 5.

DELETE FROM dbo.Abflug WHERE ab\_id = 5;

23) Die folgende Abfrage löscht die Zeile mit der Abflug-ID 10.

DELETE FROM dbo.Abflug WHERE ab\_id = 10;

24) Die folgende Abfrage fügt der Flughafen-Tabelle eine neue Zeile hinzu.

INSERT INTO Flughafen VALUES ('LTBJ', 'Adnan Menderes Airport')

25) Die folgende Abfrage fügt der Fluggesellschaft-Tabelle eine neue Zeile hinzu

INSERT INTO Fluggesellschaft VALUES ('Aer Lingus')

26) Die folgende Abfrage fügt der Flugzeug-Tabelle eine neue Zeile hinzu

INSERT INTO Flugzeug

VALUES ((SELECT fz\_mo\_id FROM Flugzeugmodell WHERE modell='A330-300'),

(SELECT fgs\_id FROM Fluggesellschaft WHERE fgs\_name= 'Aer Lingus'),

'AL23143567', 1994)

27) Die folgende Abfrage korrigiert die falschen Daten mithilfe von TRANSACTION.

BEGIN TRAN

UPDATE Event

SET st\_id = (SELECT st\_id FROM Status WHERE status='Past Flight')

WHERE Event.st\_id = 10

SAVE TRAN InfoCorrection

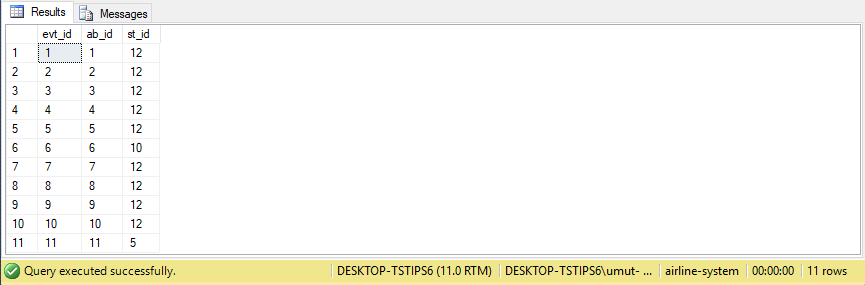
DELETE FROM Abflug WHERE dbo.Abflug.abflug\_no = 'YK2067'

ROLLBACK TRAN InfoCorrection

DELETE FROM Abflug WHERE dbo.Abflug.abflug\_no LIKE 'WL%'

SELECT \* FROM Event

COMMIT TRAN



28) Die folgende Abfrage wird auf kontrollierte Weise mit der TRANSAKTION durchgeführt, um einen neuen Passagier und eine neue Reservierung hinzuzufügen.

BEGIN TRANSACTION

INSERT INTO Passagier VALUES('Hasan', '', 'Yücel', 'H123456')

SAVE TRAN AddPass

INSERT INTO Reservation

VALUES (11, (SELECT pass\_id FROM Passagier WHERE passport\_no= 'H123456'),

(SELECT CONVERT(bigint, RAND()\*1000000000000)),

(SELECT GETDATE()), 1, '2A')

SAVE TRAN AddRes UPDATE Reservation SET sitz= '36A', t\_id= 3

WHERE pass\_id

IN (SELECT pass\_id FROM Passagier WHERE passport\_no= 'H123456')

COMMIT TRAN

29) Die folgende Abfrage aktualisiert die Flugbesatzung.

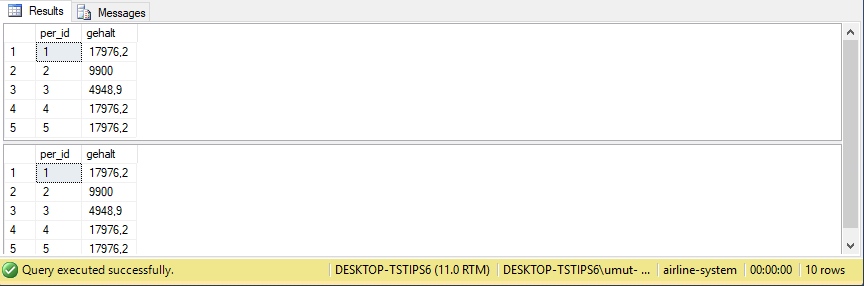
UPDATE Flugbesatzung SET per\_id = 25 WHERE per\_id = 24 AND ab\_id = 10

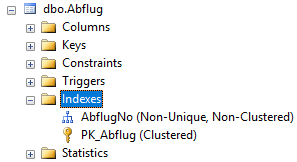
30) Die folgende Abfrage erhöht das Gehalt des Personals um 10%.

SELECT TOP 5 per\_id, gehalt FROM Personal

UPDATE Personal SET gehalt = gehalt + (gehalt\*0.1)

SELECT TOP 5 per\_id, gehalt FROM Personal

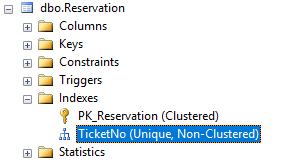




31) Die folgende Abfrage erzeugt den INDEX, um die Suche in der Abflug-Tabelle über abflug\_no zu beschleunigen.

CREATE NONCLUSTERED INDEX [AbflugNo]

ON [dbo].[Abflug] ( abflug\_no ASC)



32) Die folgende Abfrage erzeugt den INDEX, um die Suche in der Reservation-Tabelle über ticket\_no zu beschleunigen.

CREATE UNIQUE NONCLUSTERED INDEX [TicketNo]

ON [dbo].[Reservation

] (ticket\_no ASC)

33) AbflugStat View

CREATE VIEW [dbo].[AbflugStat]

AS

SELECT dbo.Abflug.abflug\_no, dbo.Status.status

FROM dbo.Abflug

INNER JOIN dbo.Event ON dbo.Abflug.ab\_id = dbo.Event.ab\_id

INNER JOIN dbo.Status ON dbo.Event.st\_id = dbo.Status.st\_id

ORDER BY dbo.Abflug.ab\_id DESC;

34) PassagierListe View

CREATE VIEW [dbo].[PassagierListe] AS

SELECT dbo.Abflug.abflug\_no, dbo.Passagier.pass\_vorname, dbo.Passagier.pass\_zw\_vorname, dbo.Passagier.pass\_nachname, dbo.Reservation.ticket\_no, dbo.Reservation.ticket\_geb\_date, dbo.Ticket\_Type.ticket\_typ, dbo.Reservation.sitz

FROM dbo.Passagier

INNER JOIN dbo.Reservation ON dbo.Passagier.pass\_id = dbo.Reservation.pass\_id

INNER JOIN dbo.Ticket\_Type ON dbo.Reservation.t\_id = dbo.Ticket\_Type.t\_id

INNER JOIN dbo.Abflug ON dbo.Reservation.ab\_id = dbo.Abflug.ab\_id

ORDER BY dbo.Abflug.ab\_id DESC, dbo.Ticket\_Type.t\_id ASC;

35) AblugListe View

CREATE VIEW [dbo].[AbflugListe] AS

SELECT dbo.Abflug.abflug\_no,

abflug\_flughafen.fh\_name AS 'Abflug Flughafen',

dbo.Abflug.abflugmoment, dbo.Abflug.abflug\_uhr,

ankunft\_flughafen.fh\_name AS 'AnkunftFlughafen',

dbo.Abflug.ankunftmoment, dbo.Abflug.ankunft\_uhr

FROM dbo.Passagier

INNER JOIN dbo.Reservation ON dbo.Passagier.pass\_id = dbo.Reservation.pass\_id

INNER JOIN dbo.Ticket\_Type ON dbo.Reservation.t\_id = dbo.Ticket\_Type.t\_id

INNER JOIN dbo.Abflug ON dbo.Reservation.ab\_id = dbo.Abflug.ab\_id

ORDER BY dbo.Abflug.ab\_id DESC, dbo.Ticket\_Type.t\_id ASC;

36) FlugbesatzungListe View

CREATE VIEW [dbo].[FlugbesatzungListe] AS

SELECT dbo.Abflug.abflug\_no, dbo.Personal.per\_vorname, dbo.Personal.per\_zw\_vorname, dbo.Personal.per\_nachname, dbo.Arbeiter\_Aufgabenart.aufgabe

FROM dbo.Abflug

INNER JOIN dbo.Flugbesatzung ON dbo.Abflug.ab\_id = dbo.Flugbesatzung.ab\_id

INNER JOIN dbo.Personal ON dbo.Flugbesatzung.per\_id = dbo.Personal.per\_id

INNER JOIN dbo.Arbeiter\_Aufgabenart ON dbo.Personal.arb\_art\_id = dbo.Arbeiter\_Aufgabenart.arb\_art\_id;

37) GepäckListe View

CREATE VIEW [dbo].[GepäckListe] AS

SELECT dbo.Gepäck.gepäck\_code, dbo.Reservation.ticket\_no, dbo.Passagier.passport\_no, dbo.Abflug.abflug\_no

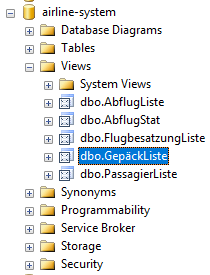
FROM dbo.Abflug

INNER JOIN dbo.Reservation ON dbo.Abflug.ab\_id = dbo.Reservation.ab\_id

INNER JOIN dbo.Gepäck ON dbo.Reservation.res\_id = dbo.Gepäck.res\_id

INNER JOIN dbo.Passagier ON dbo.Reservation.pass\_id = dbo.Passagier.pass\_id

ORDER BY dbo.Abflug.ab\_id DESC, dbo.Reservation.res\_id ASC;



38) ALTER & DROP Abfrages

ALTER TABLE Flugbesatzung DROP COLUMN fb\_id

ALTER TABLE Passagier ALTER COLUMN passport\_no nvarchar(7)

ALTER TABLE Reservation ALTER COLUMN sitz nvarchar(4)

ALTER TABLE Flughafen ALTER COLUMN international\_code nvarchar(4)

ALTER TABLE Personal ALTER COLUMN telefon\_no nvarchar(11)

BEGIN TRAN

DROP TABLE Flugbesatzung

ROLLBACK TRAN

39) Personal Addieren Procedure

CREATE PROCEDURE AddPersonal

( @per\_vn NVARCHAR(50),

@per\_zwn NVARCHAR(50),

@per\_nn NVARCHAR(50),

@per\_in BIGINT,

@per\_mail NVARCHAR(100),

@per\_tel NVARCHAR(11),

@per\_aai SMALLINT,

@per\_g FLOAT

)

AS

INSERT INTO Personal

VALUES (@per\_vn,@per\_zwn,@per\_nn,@per\_in,@per\_mail,@per\_tel,@per\_aai,@per\_g)

SELECT [per\_id]

,[per\_vorname]

,[per\_zw\_vorname]

,[per\_nachname]

,[ID\_nummer]

,[e\_mail]

,[telefon\_no]

,[aufgabe]

,[gehalt]

FROM [airline-system].[dbo].[Personal]

INNER JOIN Arbeiter\_Aufgabenart ON Personal.arb\_art\_id = Arbeiter\_Aufgabenart.arb\_art\_id

ORDER BY Personal.per\_id DESC;

EXECUTE AddPersonal Merve,Fatma,Eser,21123456722,'mfatmaeser@hotmail.com',05536667766,2,3547.2