Databases Final Project

By

Mehmet Baran Ozkutuk

104085

Airport Management Database

Project Description

I was asked to create a database for the purpose of airport management which must hold the information about the airport, flights, passengers and employees and every table we create, each of them must hold at least 3000 data in them. Designing and ER diagram is also part of the project which are allowed to be created in any platform. The CPU usage, input/output cost and operation cost should also be included to the report.

Entities

Airplane

AIRLINEID	AI_NAME	THREE_DIGIT_CODE

Airport

AP_NAME	STATE	COUNTRY	CNAME

City

CNAME	STATE	COUNTRY

Employee1

SSN	FNAME	M	LNAME	ADDRESS	PHONE	AGE	SEX	JOBTYPE	ASTYPE
ETYPE	SHIFT	POSITION	AP_NAME						

Employee2

JOBTYPE	SALARY

Flight

FLIGHT_CODE	SOURCE	DESTINATION	ARRIVAL	DEPARTURE	STATUS	DURATION	FLIGHTTYPE	LAY_TIME
NO_OF_STEPS	AIRLINEID							

Passenger1

PID	PASSPORTNO

Passenger2

PASSPORTNO	FNAME	M	LNAME	ADDRESS	PHONE	AGE	SEX

Ticket1

TICKET_NUMBER	SOURCE	DESTINATION	DATE_OF_BOOK	DATE_OF_TRAVEL	SEATNO
CLASS	DATE_OF_CANC	PID	PASSPORTNO		

Ticket2

DATE_OF_BOOK	SOURCE	DESTINATION	CLASS	PRICE

Pilot

PILOT_ID	PNAME	M	PLNAME	REG_NUMBER	ADDRESS	AGE	SEX	INCIDENT_NO	AP_NAME

Plane

PLANE_ID	MODEL	CAPACITY	PLANE_REG_DATE	PILOT_ID

Main SQL Queries

Creating CITY Table

```
-- Airport Management System --
CREATE TABLE CITY
(CNAME VARCHAR(15) NOT NULL,
STATE VARCHAR(15),
COUNTRY VARCHAR(30),
PRIMARY KEY(CNAME));

-- Inserting values of Table: CITY--
INSERT INTO CITY (CNAME, STATE, COUNTRY) VALUES('Louisville','Kentucky','United States');
INSERT INTO CITY (CNAME, STATE, COUNTRY) VALUES ('Chandigarh','Chandigarh','India');
INSERT INTO CITY (CNAME, STATE, COUNTRY) VALUES ('Fort Worth','Texas','United States');
INSERT INTO CITY (CNAME, STATE, COUNTRY) VALUES('Delhi','Delhi','India');
INSERT INTO CITY (CNAME, STATE, COUNTRY) VALUES('Mumbai','Maharashtra','India');
INSERT INTO CITY (CNAME, STATE, COUNTRY) VALUES('San Francisco', 'California', 'United States');
INSERT INTO CITY (CNAME, STATE, COUNTRY) VALUES('Frankfurt','Hesse','Germany');
INSERT INTO CITY (CNAME, STATE, COUNTRY) VALUES('Houston','Texas','United States');
INSERT INTO CITY (CNAME, STATE, COUNTRY) VALUES('New York City','New York','United
States'); INSERT INTO CITY (CNAME, STATE, COUNTRY) VALUES('Tampa', 'Florida', 'United
States');
```

Creating AIRPORT Table

```
CREATE TABLE AIRPORT

(AP_NAME VARCHAR(100) NOT NULL,

STATE VARCHAR(15),

COUNTRY VARCHAR(30),

CNAME VARCHAR(15),

PRIMARY KEY(AP_NAME),

FOREIGN KEY(CNAME) REFERENCES CITY(CNAME) ON DELETE CASCADE);


--Inserting values for Table: AIRPORT--


INSERT INTO AIRPORT (AP_NAME, STATE, COUNTRY, CNAME) VALUES('Louisville International Airport','Kentucky','United States','Louisville');

INSERT INTO AIRPORT (AP_NAME, STATE, COUNTRY, CNAME) VALUES('Chandigarh International Airport','Chandigarh','India','Chandigarh');

INSERT INTO AIRPORT (AP_NAME, STATE, COUNTRY, CNAME) VALUES('Dallas/Fort Worth International Airport','Texas','United States','Fort Worth');

INSERT INTO AIRPORT (AP_NAME, STATE, COUNTRY, CNAME) VALUES('Indira GandhiInternational Airport','Delhi','India','Delhi');

INSERT INTO AIRPORT (AP_NAME, STATE, COUNTRY, CNAME) VALUES('Chhatrapati Shivaji International Airport','Maharashtra','India','Mumbai');

INSERT INTO AIRPORT (AP_NAME, STATE, COUNTRY, CNAME) VALUES('San Francisco International Airport','California', 'United States','San Francisco');

INSERT INTO AIRPORT (AP_NAME, STATE, COUNTRY, CNAME) VALUES('Frankfurt Airport','Hesse','Germany','Frankfurt');

INSERT INTO AIRPORT (AP_NAME, STATE, COUNTRY, CNAME) VALUES('George Bush Intercontinental Airport','Texas','United States','Houston');

INSERT INTO AIRPORT (AP_NAME, STATE, COUNTRY, CNAME) VALUES('John F. Kennedy International Airport','New York','United States','New York City');

INSERT INTO AIRPORT (AP_NAME, STATE, COUNTRY, CNAME) VALUES('Tampa International Airport','Florida', 'United States','Tampa');
```

Creating TABLE Airline

```
CREATE TABLE AIRLINE
(AIRLINEID VARCHAR(3) NOT NULL,
AL_NAME VARCHAR(50),
THREE_DIGIT_CODE VARCHAR(3),
PRIMARY KEY(AIRLINEID));


-- Inserting values for Table: AIRLINE --


INSERT INTO AIRLINE (AIRLINEID, AL_NAME, THREE_DIGIT_CODE) VALUES('AA','American Airlines','001');
INSERT INTO AIRLINE (AIRLINEID, AL_NAME, THREE_DIGIT_CODE) VALUES('AI','Air India Limited','098');
INSERT INTO AIRLINE (AIRLINEID, AL_NAME, THREE_DIGIT_CODE) VALUES('LH','Lufthansa', '220');
INSERT INTO AIRLINE (AIRLINEID, AL_NAME, THREE_DIGIT_CODE) VALUES('BA','British Airways','125');
INSERT INTO AIRLINE (AIRLINEID, AL_NAME, THREE_DIGIT_CODE) VALUES('QR','Qatar Airways','157');
INSERT INTO AIRLINE (AIRLINEID, AL_NAME, THREE_DIGIT_CODE) VALUES('9W','Jet Airways','589');
INSERT INTO AIRLINE (AIRLINEID, AL_NAME, THREE_DIGIT_CODE) VALUES('EK','Emirates','176');
INSERT INTO AIRLINE (AIRLINEID, AL_NAME, THREE_DIGIT_CODE) VALUES('EY','Ethiad Airways','607');
```

Creating FLIGHT Table

```
CREATE TABLE FLIGHT
(FLIGHT_CODE VARCHAR(10) NOT NULL,
SOURCE VARCHAR(3),
DESTINATION VARCHAR(3),
ARRIVAL VARCHAR(10),
DEPARTURE VARCHAR(10),
STATUS VARCHAR(10),
DURATION VARCHAR(30),
FLIGHTTYPE VARCHAR(10),
LAYOVER_TIME VARCHAR(30),
NO_OF_STOPS INT,
AIRLINEID VARCHAR(3),
PRIMARY KEY(FLIGHT_CODE),
FOREIGN KEY(AIRLINEID) REFERENCES AIRLINE(AIRLINEID) ON DELETE CASCADE);


INSERT INTO FLIGHT(FLIGHT_CODE, SOURCE, DESTINATION, ARRIVAL, DEPARTURE, STATUS, DURATION, FLIGHTTYPE,
LAYOVER_TIME, NO_OF_STOPS, AIRLINEID)
VALUES('AI2014','BOM','DFW','02:10','03:15','On-time','24hr','Connecting',3,1,'AI');

INSERT INTO FLIGHT(FLIGHT_CODE, SOURCE, DESTINATION, ARRIVAL, DEPARTURE, STATUS, DURATION, FLIGHTTYPE,
LAYOVER_TIME, NO_OF_STOPS, AIRLINEID)
VALUES('QR2305','BOM','DFW','13:00','13:55','Delayed','21hr','Non-stop',0,0,'QR');

INSERT INTO FLIGHT(FLIGHT_CODE, SOURCE, DESTINATION, ARRIVAL, DEPARTURE, STATUS, DURATION, FLIGHTTYPE,
LAYOVER_TIME, NO_OF_STOPS, AIRLINEID)
VALUES('9W2334','IAH','DEL','23:00','13:45','On-time','23hrs','Direct',0,0,'9W');
```

Creating PASSENGER1 Table

```
CREATE TABLE PASSENGER1
(PID INT NOT NULL,
PASSPORTNO VARCHAR(10) NOT NULL,
PRIMARY KEY(PID, PASSPORTNO));

-- Inserting values in table: PASSENGER1--
INSERT INTO PASSENGER1(PID, PASSPORTNO) VALUES(1,'A1234568');
INSERT INTO PASSENGER1(PID, PASSPORTNO) VALUES(2,'B9876541');
INSERT INTO PASSENGER1(PID, PASSPORTNO) VALUES(3,'C2345698');
INSERT INTO PASSENGER1(PID, PASSPORTNO) VALUES(4,'D1002004');
INSERT INTO PASSENGER1(PID, PASSPORTNO) VALUES(5,'X9324666');
INSERT INTO PASSENGER1(PID, PASSPORTNO) VALUES(6,'B8765430');
INSERT INTO PASSENGER1(PID, PASSPORTNO) VALUES(7,'J9801235');
INSERT INTO PASSENGER1(PID, PASSPORTNO) VALUES(8,'A1122334');
INSERT INTO PASSENGER1(PID, PASSPORTNO) VALUES(9,'Q1243567');
INSERT INTO PASSENGER1(PID, PASSPORTNO) VALUES(10,'S1243269');
INSERT INTO PASSENGER1(PID, PASSPORTNO) VALUES(11,'E3277889'); INSERT INTO
PASSENGER1(PID, PASSPORTNO) VALUES(12,'K3212322');
INSERT INTO PASSENGER1(PID, PASSPORTNO) VALUES(13,'P3452390');
INSERT INTO PASSENGER1(PID, PASSPORTNO) VALUES(14,'W7543336');
INSERT INTO PASSENGER1(PID, PASSPORTNO) VALUES(15,'R8990566');
```

Creating PASSENGER2 Table

```
CREATE TABLE PASSENGER2
(PASSPORTNO VARCHAR(10) NOT NULL,
FNAME VARCHAR(20), M
VARCHAR(1),
LNAME VARCHAR(20),
ADDRESS VARCHAR(100),
PHONE INT,
AGE INT,
SEX VARCHAR(1),
PRIMARY KEY(PASSPORTNO));

--INSERTING VALUES IN TABLE: PASSENGER2--
INSERT INTO PASSENGER2(PASSPORTNO,FNAME,M,LNAME,ADDRESS,PHONE,AGE,SEX)
VALUES('A1234568','ALEN','M','SMITH','2230 NORTHSIDE, APT 11, ALBANY, NY',8080367290,30,'M');

INSERT INTO PASSENGER2(PASSPORTNO,FNAME,M,LNAME,ADDRESS,PHONE,AGE,SEX)
VALUES('B9876541','ANKITA','V','AHIR','3456 VIKAS APTS, APT 102,DOMBIVLI, INDIA',8080367280,26,'F');

INSERT INTO PASSENGER2(PASSPORTNO,FNAME,M,LNAME,ADDRESS,PHONE,AGE,SEX)
VALUES('C2345698','KHYATI','A','MISHRA','7820 MCCALLUM COURTS, APT 234, AKRON, OH',8082267280,30,'F');

INSERT INTO PASSENGER2(PASSPORTNO,FNAME,M,LNAME,ADDRESS,PHONE,AGE,SEX)
VALUES('D1002004','ANKITA','S','PATIL','7720 MCCALLUM BLVD, APT 1082, DALLAS, TX',9080367266,23,'F');

INSERT INTO PASSENGER2(PASSPORTNO,FNAME,M,LNAME,ADDRESS,PHONE,AGE,SEX)
VALUES('X9324666','TEJASHREE','B','PANDIT','9082 ESTAES OF RICHARDSON, RICHARDSON, TX',9004360125,28,'F');

INSERT INTO PASSENGER2(PASSPORTNO,FNAME,M,LNAME,ADDRESS,PHONE,AGE,SEX)
VALUES('B8765430','LAKSHMI','P','SHARMA','1110 FIR HILLS, APT 903, AKRON, OH',7666190505,30,'F');
```

Creating EMPLOYEE1 table

```
CREATE TABLE EMPLOYEE1
(SSN INT NOT NULL,
FNAME VARCHAR(20), M
VARCHAR(1),
LNAME VARCHAR(20),
ADDRESS VARCHAR(100),
PHONE INT,
AGE INT,
SEX VARCHAR(1),
JOBTYPE VARCHAR(30),
ASTYPE VARCHAR(30),
ETYPE VARCHAR(30),
SHIFT VARCHAR(20),
POSITION VARCHAR(30),
AP_NAME VARCHAR(100),
PRIMARY KEY(SSN),
FOREIGN KEY(AP_NAME) REFERENCES AIRPORT(AP_NAME) ON DELETE CASCADE);


INSERT INTO EMPLOYEE1(SSN, FNAME, M, LNAME, ADDRESS, PHONE, AGE, SEX, JOBTYPE, ASTYPE, ETYPE, SHIFT, POSITION, AP_NAME)
VALUES(123456789,'LINDA','M','GOODMAN','731 Fondren, Houston, TX',4356789345, 35, 'F','ADMINISTRATIVE
SUPPORT','RECEPTIONIST','','','','Louisville International Airport');


INSERT INTO EMPLOYEE1(SSN, FNAME, M, LNAME, ADDRESS, PHONE, AGE, SEX, JOBTYPE, ASTYPE, ETYPE, SHIFT, POSITION, AP_NAME)
VALUES(333445555,'JOHNY','N','PAUL','638 Voss, Houston, TX',9834561995, 40, 'M','ADMINISTRATIVE
SUPPORT','SECRETARY','','','','Louisville International Airport');
```

Creating EMPLOYEE2 Table

```
CREATE TABLE EMPLOYEE2
(JOBTYPE VARCHAR(30) NOT NULL,
SALARY INT,
PRIMARY KEY(JOBTYPE));

--INSERTING VALUES INTO TABLE: EMPLOYEE2--
INSERT INTO EMPLOYEE2(JOBTYPE, SALARY)VALUES('ADMINISTRATIVE SUPPORT',50000);
INSERT INTO EMPLOYEE2(JOBTYPE, SALARY)VALUES('ENGINEER',70000);
INSERT INTO EMPLOYEE2(JOBTYPE, SALARY)VALUES('TRAFFIC MONITOR',80000);
INSERT INTO EMPLOYEE2(JOBTYPE, SALARY)VALUES('AIRPORT AUTHORITY',90000);
```

Creating TICKET1 Table

```
CREATE TABLE TICKET1

(TICKET_NUMBER INT NOT NULL,

SOURCE VARCHAR(3),

DESTINATION VARCHAR(3),

DATE_OF_BOOKING DATE,

DATE_OF_TRAVEL DATE,

SEATNO VARCHAR(5),

CLASS VARCHAR(15),

DATE_OF_CANCELLATION DATE,

PID INT,

PASSPORTNO VARCHAR(10),

FOREIGN KEY(PID, PASSPORTNO) REFERENCES PASSENGER1(PID, PASSPORTNO) ON DELETE CASCADE);


--INSERTING INTO TABLE: TICKET1--

INSERT INTO TICKET1(TICKET_NUMBER, SOURCE, DESTINATION, DATE_OF_BOOKING, DATE_OF_CANCELLATION,
DATE_OF_TRAVEL, SEATNO, CLASS, PID, PASSPORTNO)

VALUES(0011234111122,'BOM','DFW','11-MAY-16','','15-DEC-16','32A','ECONOMY',1,'A1234568');


INSERT INTO TICKET1(TICKET_NUMBER, SOURCE, DESTINATION, DATE_OF_BOOKING, DATE_OF_CANCELLATION,
DATE_OF_TRAVEL, SEATNO, CLASS, PID, PASSPORTNO)

VALUES(0984567222299,'JFK','BOM','11-JUN-16','10-DEC-16','20-DEC-16','45D','ECONOMY',2,'B9876541');


INSERT INTO TICKET1(TICKET_NUMBER, SOURCE, DESTINATION, DATE_OF_BOOKING, DATE_OF_CANCELLATION,
DATE_OF_TRAVEL, SEATNO, CLASS, PID, PASSPORTNO)

VALUES(1768901333273,'IAH','DEL','21-AUG-16','','25-DEC-16','1A','BUSINESS',3,'C2345698');
```

Creating TICKET2 Table

```
CREATE TABLE TICKET2
(DATE_OF_BOOKING DATE NOT NULL,
SOURCE VARCHAR(3) NOT NULL,
DESTINATION VARCHAR(3) NOT NULL,
CLASS VARCHAR(15) NOT NULL,
PRICE INT,
PRIMARY KEY(DATE_OF_BOOKING, SOURCE, DESTINATION, CLASS));

-- INSERTING VALUES INTO TICKET2--
INSERT INTO TICKET2(DATE_OF_BOOKING, SOURCE, DESTINATION, CLASS, PRICE)
VALUES('11-MAY-16','BOM','DFW','ECONOMY',95000);

INSERT INTO TICKET2(DATE_OF_BOOKING, SOURCE, DESTINATION, CLASS, PRICE)
VALUES('11-JUN-16','JFK','BOM','ECONOMY',100000);

INSERT INTO TICKET2(DATE_OF_BOOKING, SOURCE, DESTINATION, CLASS, PRICE)
VALUES('21-AUG-16','IAH','DEL','BUSINESS',200000);

INSERT INTO TICKET2(DATE_OF_BOOKING, SOURCE, DESTINATION, CLASS, PRICE)
VALUES('10-AUG-16','IXC','IAH','FIRST-CLASS',150000);

INSERT INTO TICKET2(DATE_OF_BOOKING, SOURCE, DESTINATION, CLASS, PRICE)
VALUES('13-JUN-16','JFK','TPA','ECONOMY',98000);

INSERT INTO TICKET2(DATE_OF_BOOKING, SOURCE, DESTINATION, CLASS, PRICE)
VALUES('11-NOV-16','BOM','DFW','ECONOMY',125000);
```

Creating PILOT Table

```
USE AIRPORT_MANAGEMENT_DATABASE
CREATE TABLE PILOT (
        PILOT_ID varchar (30) NOT NULL PRIMARY KEY,
  PNAME varchar(30),
  M varchar(1),
  PLNAME varchar(30),
  REG_NUMBER int,
  ADDRESS varchar(30),
  AGE int,
  SEX varchar(1),
  INCIDENT_NUMBER int,
  AP_NAME varchar(100) FOREIGN KEY REFERENCES AIRPORT(AP_NAME)
);
```

Creating PLANE Table

```
USE AIRPORT_MANAGEMENT_DATABASE
CREATE TABLE PLANE (
   PLANE_ID int NOT NULL PRIMARY KEY,
   MODEL varchar(255),
   CAPACITY int,
   PLANE_REG_DATE DATE,
            PILOT_ID varchar(30) FOREIGN KEY REFERENCES PILOT(PILOT_ID)
);
```

All Queries Which Are Used To Add Values

Adding data into AIRLINE table

```
declare @number int

declare @airId int

select @airId = 1

select @number = 1

while @number <=3000

begin

insert into AIRLINE VALUES (CONVERT(varchar(3),@airId), 'AIName' +
convert(varchar(15),@number),CONVERT(varchar(3),@airId)) --then we change it
into NULL valuse because we will run out of three digits options

select @number = @number + 1

select @airId = @airId + 1

end
```

Adding data into AIRPORT table

```
declare @number int

declare @airName int

select @airName = 1

select @number = 1

while @number <=3000

begin

insert into AIRPORT VALUES ('APNAME'+ convert (varchar(30),
@airName ), 'STATEEX' +
convert(varchar(15),@number),'COUNTRYEX' +
convert(varchar(15),@number), 'CNAME' +
convert(varchar(15),@number))

select @number = @number + 1

select @airName = @airName + 1


end
```

Adding data into CITY table

```
declare @number int

declare @SSN int

select @SSN = 1

select @number = 1

while @number <=3000

begin

insert into CITY VALUES('CNAME' + convert (varchar(30), @SSN),
'STATEEX' + convert (varchar(30), @SSN ) , 'COUNTRYEX' + convert
(varchar(30), @SSN ))

select @number = @number + 1

select @SSN = @SSN + 1

end
```

Adding data into EMPLOYEE1 table

```
declare @number int
declare @SSN int
declare @phone int
declare @number2 int
select @number2 = 1
select @phone = 56087
select @SSN = 5600
select @number = 1
while @number <=3000
begin

                insert into EMPLOYEE1 VALUES (@SSN ,
        'NAME3X' + CONVERT(varchar(15),@SSN),
        'M','LNAMEEX' + CONVERT(varchar(15),@SSN),
        'ADDRESSEX' + CONVERT(varchar(15),@SSN),
        @phone, 26, 'M',
        'JOBTYEX'+ CONVERT(varchar(15),@SSN),
        'ASTYPEX' + CONVERT(varchar(15),@SSN),
        'ETYPEX' + CONVERT(varchar(15),@SSN),
        'SHIFTEX' + CONVERT(varchar(15),@SSN),  'POSITION'+
        CONVERT(varchar(15),@SSN),  'APNAME'+
        CONVERT(varchar(15),@number2))
            select @number = @number + 1
            select @SSN = @SSN + 1
            select @phone = @phone + 55
            select @number2 = @number2 + 1
end
```

Adding data into EMPLOYEE2 table

```
declare @id int
declare @SALARY int
declare @number int
select @number = 1
select @SALARY = 1200
select @id = 5600
while @id <= 3000
begin
   insert into EMPLOYEE2 values('JOBTYPEX' +
CONVERT(varchar(15), @id), @SALARY )
   select @id = @id + 1
select @SALARY = @SALARY + 2
select @number = @number + 1
end
```

Adding data into FLIGHT table

```
declare @SSN int
select @SSN = 1
declare @no int
select @no = 10800
declare @id int
select @id = 1
DECLARE @id2 int
select @id2 = 10900
while @SSN <= 999
begin
        insert into FLIGHT values ('FEX' +
CONVERT(varchar(10),@id2),'B'  +
CONVERT(varchar(2),@no), 'B'  +
CONVERT(varchar(2),@id),'ARREX' +
CONVERT(varchar(17),@no), 'DEX' +
CONVERT(varchar(10),@no), 'ST' +
CONVERT(varchar(10),@no), 'DURATEX' +
CONVERT(varchar(17),@no), 'FTEX' +
CONVERT(varchar(17),@no), 'LAYOVEREX' +
CONVERT(varchar(17),@no), @no,  +'L' +
CONVERT(varchar(17),@id) )
        select @SSN = @SSN + 1
        select @no = @no + 1
        select @id = @id + 1
        SELECT @id2 = @id2 + 1
end
```

Adding data into PASSENGER1 table

```
declare @SSN int
declare @id int
select @id = 1
select @SSN = 1
while @SSN <=3000
begin
        insert into PASSENGER1
values(@id,'PASEX' + CONVERT(varchar(10),
@id))
        select @SSN = @SSN + 1
        select @id = @id + 1
end
```

Adding data into PASSENGER2 table

```
declare @number int
declare @phone int
select @phone = 3536
select @number = 1
while @number <=3000
begin
        insert into PASSENGER2 VALUES ('PASEX'
+ CONVERT(varchar(15),@number), 'FNAMEX'  +
CONVERT(varchar(15),@number),'M','LNAMEX'
+ CONVERT(varchar(15),@number),'ADDRESEX'
+
CONVERT(varchar(15),@number),@phone,29,'
M')
        select @number = @number + 1
        select @phone = @phone + 504
end
```

Adding data into TICKET1 table

```
declare @SSN int

declare @id int

select @id = 186

select @SSN = 1

while @SSN <=3000

begin

        insert into TICKET1 values(@id,'S' +
CONVERT(varchar(2), @id),'D' +
CONVERT(varchar(2), @id), NULL,NULL,'SE' +
CONVERT(varchar(2), @id),
'ECONOMIC',NULL,@id,'PASEX'+
CONVERT(varchar(10), @id))

        select @SSN = @SSN + 1

        select @id = @id + 1

end
```

Adding data into TICKET2 table

```
declare @SSN int
declare @id int
declare @price int
select @price = 2500
select @id = 1
select @SSN = 1
while @SSN <=3000
begin
        insert into TICKET2 values(DATEADD
(month , 1 , 2006 )  ,'S*','D*','ECONOMIC' +
CONVERT(varchar(2), @id),@price)
        select @SSN = @SSN + 1
        selec t @id = @id + 1
        select @price = @price + 1
end
```

Adding data into PILOT table

```
USE AIRPORT_MANAGEMENT_DATABASE

declare @SSN int

declare @id int

declare @regNo int

declare @age int

select @age = 25

select @regNo = 2500

select @id = 1

select @SSN = 1

while @SSN <=3000

begin

insert into PILOT values('PILOTID' + CONVERT(varchar(30),@SSN),'PNAME' +
CONVERT(varchar(30),@SSN), 'M', 'PLNAME' +
CONVERT(varchar(30),@SSN),@regNo,'ADDRESS' +
CONVERT(varchar(30),@SSN), @age, 'M', 2,'APNAME' +
CONVERT(varchar(30),@SSN))

select @SSN = @SSN + 1

select @id = @id + 1

select @regNo = @regNo + 438

select @age = @age + 1

end
```

Adding data into PLANE table

```
use AIRPORT_MANAGEMENT_DATABASE

declare @SSN int

declare @id int

declare @planeID int

declare @capacity int

select @capacity = 12

select @planeID = 500

select @id = 1

select @SSN = 1

while @SSN <=3000

begin

insert into PLANE values(@planeID, 'MODEL' +
CONVERT(varchar(30),@SSN),@capacity, DATEADD(month, 4, '2019-05-30'),
'PILOTID' + CONVERT(varchar(30),@SSN))

select @SSN = @SSN + 1

select @id = @id + 1

select @planeID = @planeID + 432

select @capacity = @capacity + 1

end
```

All Transactions

UPDATE

```
--Let's change every row who goes by the name of Andrew DURHAM
from EMPLOYEE1 table and change it's FNAME
--into MICHEAL, LNAME into BOBO and change it's ADDRESS into
SAN

UPDATE EMPLOYEE1
SET FNAME = 'MICHEAL', LNAME = 'BOBO', ADDRESS = 'SAN'
WHERE FNAME = 'Andrew'and LNAME = 'DURHAM'
```

UPDATE AND DELETE

```
--Let's change the DATE_OF_BOOKING, DATE_OF_TRAVEL and DATE_OF_CANCELLATION columns
into real data value which
--matter of fact they are null right now and delete specific column if the
TICKET_NUMBER is smaller then 500

UPDATE TICKET1
SET DATE_OF_BOOKING = DATEADD(month, 4, '2019-05-30') , DATE_OF_TRAVEL = DATEADD
(month , 5 ,'2019-05-30') , DATE_OF_CANCELLATION  = DATEADD (month , 12 , '2019-
05-30')
WHERE DATE_OF_BOOKING IS NULL and DATE_OF_TRAVEL IS NULL and
DATE_OF_CANCELLATION IS NULL
DELETE FROM TICKET1 WHERE TICKET_NUMBER < 1000
```

Aggregate Procedures

COUNT

```
SELECT COUNT(FNAME)
FROM EMPLOYEE1
WHERE FNAME = 'Jack'
```

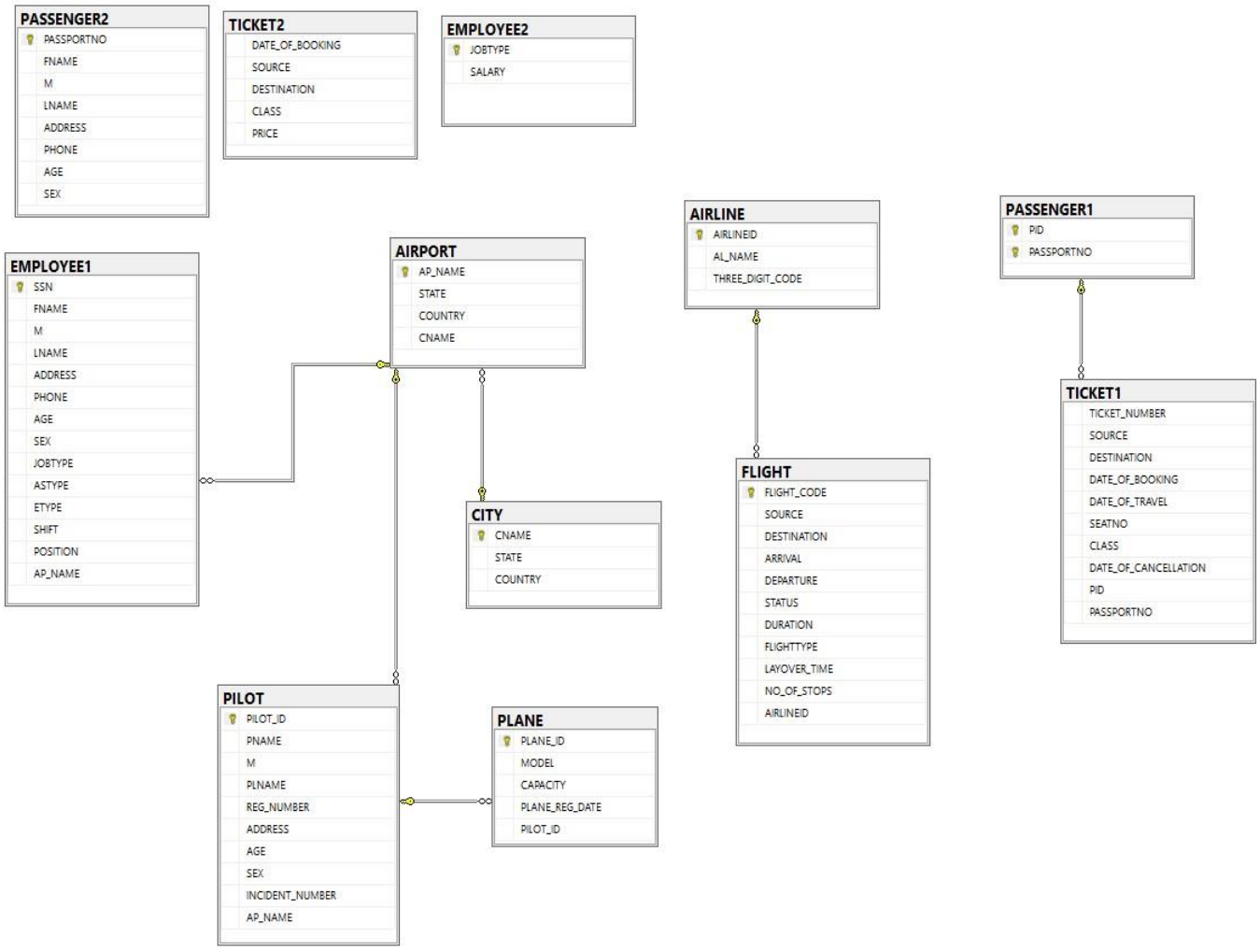
COUNT AND AVG

```
-- Let's find out how many jobs get paid over 2000 and the
average salary they get

SELECT COUNT(JOBTYPE)
FROM EMPLOYEE2
WHERE SALARY > 2000

SELECT AVG(SALARY)
FROM EMPLOYEE2
WHERE SALARY > 2000
```

ER Diagram


PASSENGER2
PASSPORTNO
FNAME
M
LNAME
ADDRESS
PHONE
AGE
SEX
TICKET2
DATE_OF_BOOKING
SOURCE
DESTINATION
CLASS
PRICE
EMPLOYEE2
JOBTYPE
SALARY
AIRLINE
AIRLINEID
AL_NAME
THREE_DIGIT_CODE
PASSENGER1
PID
PASSPORTNO
EMPLOYEE1
SSN
FNAME
M
LNAME
ADDRESS
PHONE
AGE
SEX
JOBTYPE
ASTYPE
ETYPE
SHIFT
POSITION
AP_NAME
AIRPORT
AP_NAME
STATE
COUNTRY
CNAME
TICKET1
TICKET_NUMBER
SOURCE
DESTINATION
DATE_OF_BOOKING
DATE_OF_TRAVEL
SEATNO
CLASS
DATE_OF_CANCELLATION
PID
PASSPORTNO
CITY
CNAME
STATE
COUNTRY
FLIGHT
FLIGHT_CODE
SOURCE
DESTINATION
ARRIVAL
DEPARTURE
STATUS
DURATION
FLIGHTTYPE
LAYOVER_TIME
NO_OF_STOPS
AIRLINEID
PILOT
PILOT_ID
PNAME
M
PLNAME
REG_NUMBER
ADDRESS
AGE
SEX
INCIDENT_NUMBER
AP_NAME
PLANE
PLANE_ID
MODEL
CAPACITY
PLANE_REG_DATE
PILOT_ID

User Statistics Report

User Statistics
[AIRPORT_MANAGEMENT_DATABASE]
on DESKTOP-8P9EPB2\BARAN at 7/2/2021 9:15:44 PM

SQL Server

This report provides details on the activity of all currently connected users within the Database.

Login Name	User Name	# Active Sessions*	# Active Connections*	# Active Requests	# Open Transactions	# Active Cursors	CPU Time (ms.)	Memory Usage (KB)	# Reads	# Writes	Last Request Start Time	Last Request End Time
DESKTOP-8P9EPB2\mbara	mbara	2	2	2	0	0	297.00	64	0	1	7/2/2021 9:15:44 PM	7/2/2021 9:15:44 PM

The values shown in CPU Time, Memory Usage, # Reads and # Writes columns are cumulative values of all the active sessions of that user.

* These are the total sessions/connections associated with the login, and having active requests irrespective of databases.

Table Usage Statistics

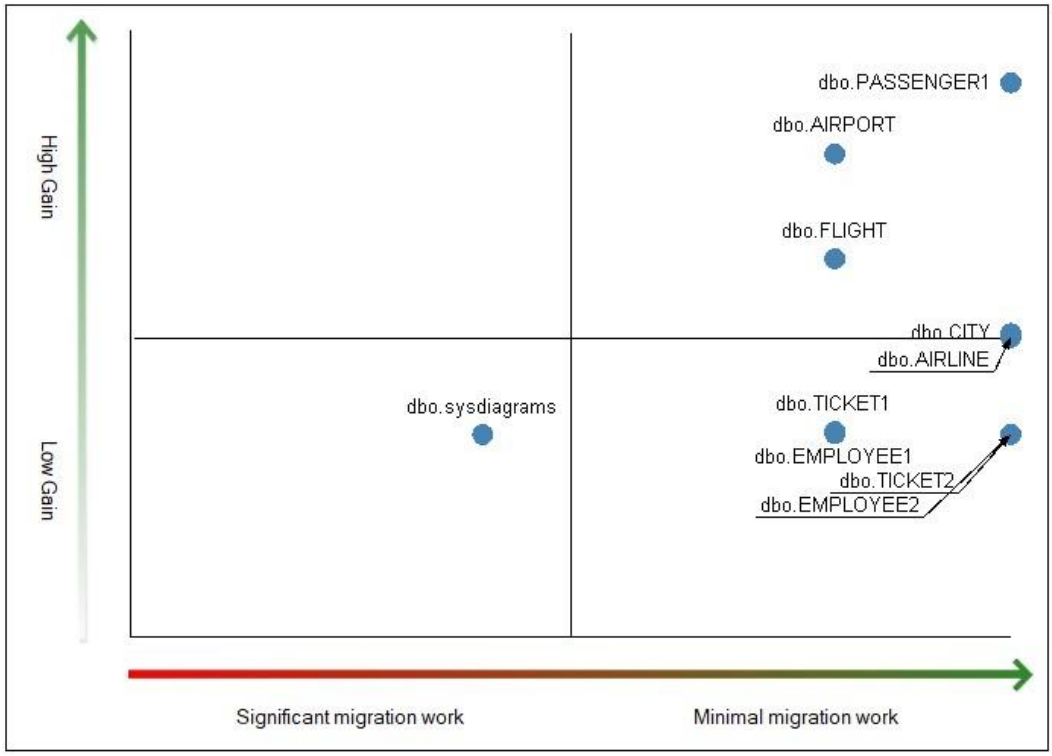
Recommended Tables Based on Usage

[AIRPORT_MANAGEMENT_DATABASE]

on DESKTOP-8P9EPB2\BARAN at 7/2/2021 9:15:05 PM

The following chart contains the top candidate tables for memory optimization based on the access patterns of your workload. The horizontal axis represents decreasing effort of memory optimization, while the vertical axis represents increasing benefits of memory optimization in your workload. You should prioritize the tables in the top right corner of the chart for memory optimization.

Select number of Tables:
5
10
15
20
25
30



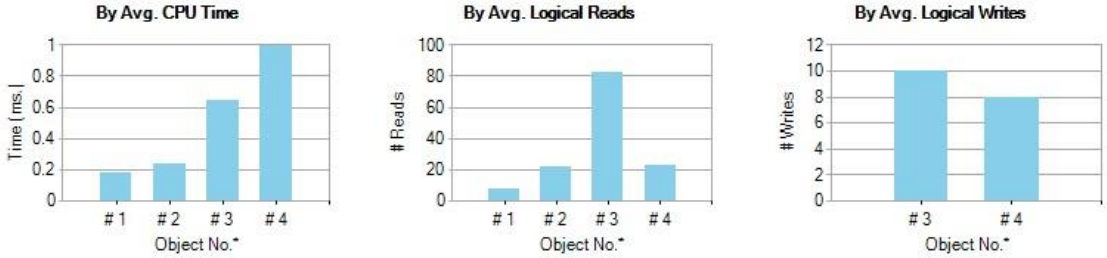
Object Execution Statistics

Object Execution Statistics
[AIRPORT_MANAGEMENT_DATABASE]
SQL Server

on DESKTOP-8P9EPB2\BARAN at 7/4/2021 1:17:28 AM

This report provides detailed historical execution data for all currently cached plans for objects within the Database. This execution data is aggregated over the time during which the plan has been in the cache.

Top Executable Objects



* See column 'Object No.' from table 'All Objects' for value of 'Object No.' It is the unique number assigned to each Object.

All Executable Objects

Shows statement wise execution statistics for all the executable objects.

Object No.*	Object Name	Object Type	Avg. CPU Time (ms.)	Total CPU Time (%)	# Avg. Logical Reads	# Avg. Logical Writes	# Avg. Logical IO	Total Logical IO (%)
1	[dbo]..sp_helpdiagrams	SQL Stored-Procedure	0.18	52.17	7.65	0.00	7.65	35.81
2	[dbo]..sp_helpdiagramdefinition	SQL Stored-Procedure	0.24	20.16	22.00	0.00	22.00	30.30
3	[dbo]..sp_creatediagram	SQL Stored-Procedure	0.65	10.91	82.00	10.00	92.00	25.34
4	[dbo]..sp_alterdiagram	SQL Stored-Procedure	0.99	16.76	23.00	8.00	31.00	8.54

Disk Usage By Top Tables

Disk Usage by Top Tables
[AIRPORT_MANAGEMENT_DATABASE]
SQL Server

on DESKTOP-8P9EPB2\BARAN at 7/4/2021 1:19:54 AM

This report provides detailed data on the utilization of disk space by top 1000 tables within the Database. The report does not provide data for memory optimized tables.

Table Name	# Records	Reserved (KB)	Data (KB)	Indexes (KB)	Unused (KB)
dbo.EMPLOYEE1	3,502	648	584	16	48
dbo.FLIGHT	3,001	584	520	16	48
dbo.PASSENGER2	3,201	456	392	16	48
dbo.PILOT	3,000	456	432	16	8
dbo.TICKET1	5,002	392	384	8	0
dbo.CITY	3,114	328	296	16	16
dbo.AIRPORT	3,048	328	304	16	8
dbo.sysdiagrams	1	216	72	24	120
dbo.PLANE	3,000	200	136	16	48
dbo.AIRLINE	3,153	200	184	16	0
dbo.EMPLOYEE2	4,004	200	128	16	56
dbo.TICKET2	3,015	136	120	8	8
dbo.PASSENGER1	3,015	136	88	16	32