RELIAIBLE AIRLINES

DATABASE DESIGN PROJECT

By Adam Sansone

UIN 676475522

TABLE OF CONTENTS

- 3 Project Log
- 4 Cumulative Design and Special Restrictions
- 5 Table Creation
- 7 SQL Table Insert Commands
- 8 Populated Tables
- 11 Reports
- 14 Queries

Project Log						
Date	Time Spent	Activities Completed				
2/10/2020	5pm - 6pm	Reviewed project specs and reviewed DBDL concepts				
2/14/2020	3pm - 6pm	Wrote up user views and started design				
2/17/2020	3pm-7pm	Continued working on user views and DBDL				
2/18/2020	12pm - 2pm	Finished user views and DBDL and created the data structure diagram				
2/24/2020	10am - 12pm 10am -	Added in special conditions and added in available seats to flight table				
3/5/2020	Finalized DBDL and reviewed all user views					
3/18/2020	11am - 12pm	Created a few tables for 2nd part of project in Microsoft Access				
3/20/2020	10am - 12pm	Finished creating tables in MS Access with SQL				
3/23/2020	11am - 1pm	Inserted data into tables using SQL				
3/25/2020	11am - 2pm	Finished creating reports a, b, d, and e in SQL				
4/6/2020	10am - 3pm	Created reports b and c in SQL				
4/8/2020	10am - 2pm	Worked on report c and query a				
4/10/2020	10am - 12pm	Started putting together final writeup				
4/16/2020	12pm - 3pm	Added all tables, some reports and some queries to final writeup				
4/20/2020	10am - 3pm	Finished all reports but one and all queries but one and updated final writeup				
4/24/2020	3pm - 9pm	Finished last report and query and updated final writeup				
4/26/2020	7pm - 9pm	Last minute changes to available flights form				

CUMULATIVE DESIGN

Plane (SerialNumber, TypeCode, ManYear, LastServDate, NextServDate)

FK TypeCode → PlaneType

PlaneType (<u>TypeCode</u>, TypeDesc, PlaneCapacity, PlaneRange)

Pilot (PilotNum, FName, LName, PilotBirthdate, PilotSSN, PilotCellNum, PilotSalary)

AK PilotSSN

PilotCerts (PilotNum, TypeCode, CertDate)

FK PilotNum → Pilot; TypeCode → PlaneType

Flight (<u>FlightNum</u>, OriginCode, DepartTime, DestCode, ArriveTime)

FK OriginCode/DestCode → City

SchFlight (FlightNum, FlightDate, FlightPlane, FlightPilot)

FK FlightNum → Flight; FlightPlane → Plane; FlightPilot → Pilot

City (<u>CityCode</u>, CityName, CityState, AirpotDesc)

SK CityName

Passenger (<u>PassengerID</u>, PassFirstName, PassLastName, PassBirthdate, PassPhoneNum, PassAddress, PassEmail*)

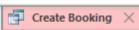
Booking (PassengerID, FlightNum, FlightDate)

FK PassengerID → Passenger; FlightNum, FlightDate → SchFlight

Special Restrictions: Seats available is a derived attribute. Sum of reservations made for a specific scheduled flight is deducted from the capacity of the plane type.

DESIGN STRUCTURE

TABLE CREATION



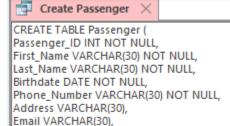
CREATE TABLE Booking (
Passenger_ID INT NOT NULL,
Flight_Num INT NOT NULL,
Flight_Date DATE NOT NULL,
PRIMARY KEY (Passenger_ID, Flight_Num, Flight_Date),
FOREIGN KEY (Passenger_ID) REFERENCES Passenger(Passenger_ID),
FOREIGN KEY (Flight_Num, Flight_Date) REFERENCES Scheduled_Flight (Flight_Num, Flight_Date))

🗗 Create Flight 💢

🗗 Create City 💢

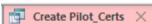
CREATE TABLE City (
City_Code VARCHAR(3) NOT NULL,
City_Name VARCHAR(30) NOT NULL,
City_State VARCHAR(2) NOT NULL,
Airport_Desc VARCHAR(30),
PRIMARY KEY (City_Code))

CREATE TABLE Flight (
Flight_Num INT NOT NULL,
Origin VARCHAR(30) NOT NULL,
Depart_Time TIME NOT NULL,
Destination VARCHAR(30) NOT NULL,
Arrival_Time TIME NOT NULL,
PRIMARY KEY (Flight_Num),
FOREIGN KEY (Origin) REFERENCES City(City_Code),
FOREIGN KEY (Destination) REFERENCES City(City_Code))



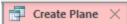
CREATE TABLE PILOT X

CREATE TABLE PILOT (
PILOT_NUM INT NOT NULL,
FIRST_Name VARCHAR(30) NOT NULL,
Last_Name VARCHAR(30) NOT NULL,
Phone_Num VARCHAR(30) NOT NULL,
BIRTHDATE NOT NULL,
SSN VARCHAR(12) UNIQUE NOT NULL,
Salary INT NOT NULL,
PRIMARY KEY (PILOT_Num))



PRIMARY KEY (Passenger_ID))

CREATE TABLE Pilot_Certs (
Pilot_Num INT NOT NULL,
Type_Code VARCHAR(8) NOT NULL,
Cert_Date DATE,
PRIMARY KEY (Pilot_Num, Type_Code),
FOREIGN KEY (Pilot_Num) REFERENCES Pilot(Pilot_Num),
FOREIGN KEY (Type_Code) REFERENCES Plane_Type(Type_Code))



CREATE TABLE Plane (
Serial_Num VARCHAR(12) NOT NULL,
Type_Code VARCHAR(8) NOT NULL,
Man_Year DATE NOT NULL,
Last_Serv DATE,
Next_Serv DATE NOT NULL,
PRIMARY KEY (Serial_Num),
FOREIGN KEY (Type_Code) REFERENCES Plane_Type(Type_Code))

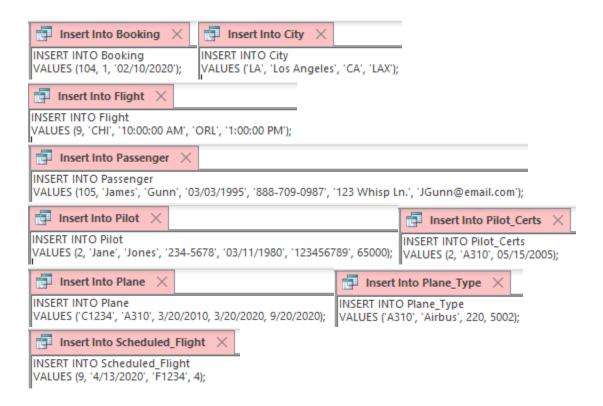


CREATE TABLE Plane_Type (
Type_Code VARCHAR(8) NOT NULL,
Type_Desc VARCHAR(30) NOT NULL,
Plane_Cap INT NOT NULL,
Plane_Range INT NOT NULL,
PRIMARY KEY (Type_Code))

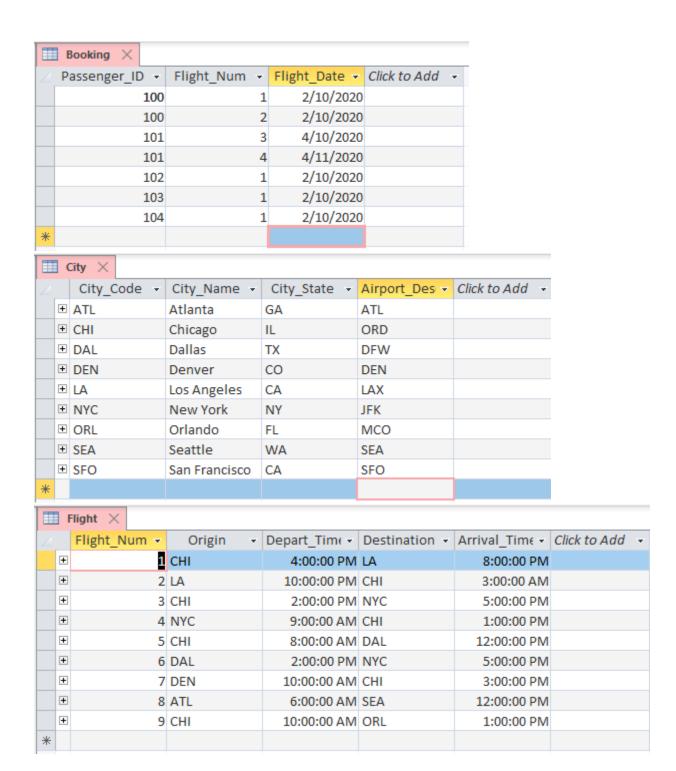
Create Scheduled_Flight

CREATE TABLE Scheduled_Flight (
Flight_Num INT NOT NULL,
Flight_Date DATE NOT NULL,
Plane VARCHAR(12) NOT NULL,
Pilot INT NOT NULL,
PRIMARY KEY (Flight_Num, Flight_Date),
FOREIGN KEY (Flight_Num) REFERENCES Flight(Flight_Num),
FOREIGN KEY (Plane) REFERENCES Plane(Serial_Num),
FOREIGN KEY (Pilot) REFERENCES Pilot(Pilot_Num))

INSERT INTO SQL STATEMENTS



POPULATED TABLES



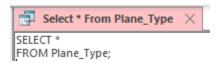
	Passenger ×						
	Passenger_I -	First_Name →	Last_Name -	Birthdate -	Phone_Num -	Address +	Email -
+	100	Bob	Smith	4/12/1990	630-445-7885	123 Fake St.	
+	101	Mary	BoPeep	3/20/1890	630-221-1245	111 Sheep Ln.	
+	102	Joe	John	4/11/1970	630-111-2222	45 W. Heart St.	
+		Jason	Lee		708-123-1414	2 Dot Rd	
+		Abagail	Pun		630-456-4567	12 E. Heart St.	
+	105	James	Gunn	3/3/1995	888-709-0987	123 Whisp Ln.	JGunn@email
	Pilot ×						
	Pilot_Num -	First_Name •	Last_Name +	Phone_Num →	Birthdate +	SSN -	Salary -
+	1	John	Smith	123-4567	3/11/1975	234567891	65000
+	2	Jane	Jones	234-5678	3/11/1980	123456789	6500
+	3	Albert	King	345-6789	6/10/1982	345678901	6500
+	4	Joy	Palace	456-7891	10/10/1979	456789102	7000
	Pilot_Certs X						
	Pilot Num →	Type Code	→ Cert Date	· •			
	_	747	5/10/2				
			5/10/				
	2 767		6/20/2				
	2 A310		6/20/2				
		A320	6/20/2				
	2	A330	6/20/2	2000			
	3	747	4/10/2	2000			
	3	767	4/10/2	2000			
	3 787		4/10/2	2000			
	4	A310	1/10/2				
		A320	1/10/2				
4 A330 1/10/1990 Plane ×							
	Serial Num	→ Type Cod	e - Man Ye	ear - Last	Serv - Nex	t Serv 🕝	
E	± A1234	747	_	0/1970 4	1/6/2019	4/6/2020	

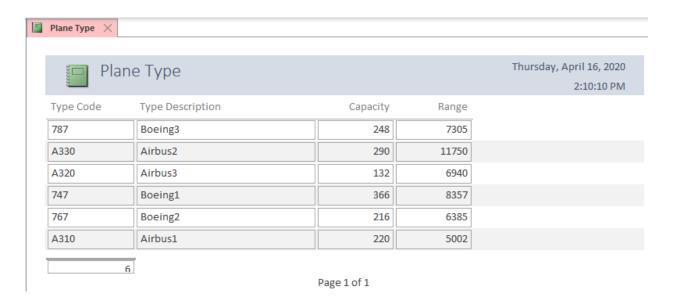
		Plane X								
2	4_		Serial_Num -	Type_Code →	Man_Year -	Last_Serv 🔻	Next_Serv →			
		+	A1234	747	1/10/1970	4/6/2019	4/6/2020			
		+	B1234	767	4/20/1980	1/10/2020	1/10/2021			
		+	C1234	A310	3/25/2020	2/14/2020	2/14/2021			
		+	D1234	787	4/1/2019	4/1/2019	4/1/2020			
		+	E1234	A320	8/15/1980	8/15/2019	8/15/2020			
		+	F1234	A330	11/23/1990	11/23/2019	11/23/2020			

	Plane_Type ×							
4	Type_Code -	Type_Desc -	Plane_Cap ▼	Plane_Range →				
9	747	Boeing1	366	8357				
9	767	Boeing2	216	6385				
9	787	Boeing3	248	7305				
9	A310	Airbus1 220		5002				
9	A320	Airbus3	132	6940				
-	A330	Airbus2	290	11750				
Scheduled_Flight ✓								
	Flight_Num -	Flight_Date -	Plane 🕶	Pilot ▼				
+	1	2/10/2020	A1234	1				
+	2	2/10/2020	A1234	1				
+	3	4/10/2020	C1234	2				
+	4	4/11/2020	B1234	2				
+	5	4/8/2020	D1234	3				
1	6	4/7/2020	D1234	3				
-	7	4/7/2020	E1234	4				
+	9	4/13/2020	F1234	4				

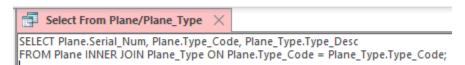
REPORTS

A)





B)





Page 1 of 1

C)

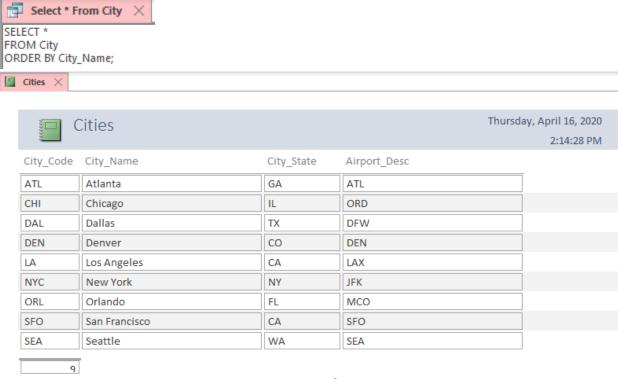


SELECT P.Pilot_Num, P.First_Name, P.Last_Name, T.Type_Code, T.Type_Desc, S.Flight_Num, S.Flight_Date, S.Plane
FROM Pilot AS P, Pilot_Certs AS C, Plane_Type AS T, Scheduled_Flight AS S, Plane AS A
WHERE P.Pilot_Num = S.Pilot
AND S.Plane = A.Serial_Num
AND A.Type_Code = T.Type_Code
AND P.Pilot_Num = C.Pilot_Num
AND C.Type_Code = T.Type_Code;

	Pilot Certific	cation Form X								
	Pilot Certification with Sch. Flights							Friday, April 24, 2020		
	Thor continuation with son. Then is							8:20:19 PM		
-	Pilot ID	First Name	Last Name	Plane Code	Plane Description	Flight#	Flight Date	Plane Serial #		
[:	3	Albert	King	787	Boeing3	6	4/7/2020	D1234		
:	3	Albert	King	787	Boeing3	5	4/8/2020	D1234		
4	4	Joy	Palace	A330	Airbus2	9	4/13/2020	F1234		
4	4	Joy	Palace	A320	Airbus3	7	4/7/2020	E1234		
	1	John	Smith	747	Boeing1	1	2/10/2020	A1234		
:	1	John	Smith	747	Boeing1	2	2/10/2020	A1234		
:	2	Jane	Jones	767	Boeing2	4	4/11/2020	B1234		
	2	Jane	Jones	A310	Airbus1	3	4/10/2020	C1234		
Τ	8									

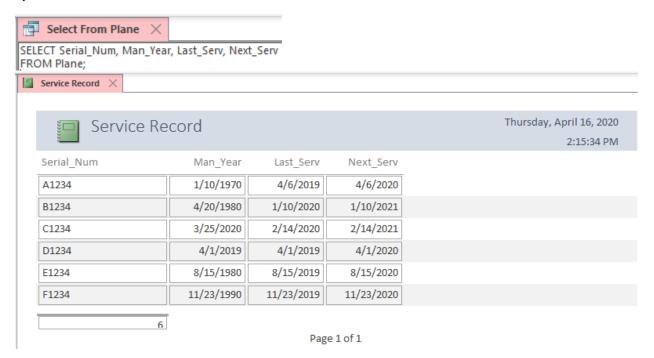
Page 1 of 1

D)



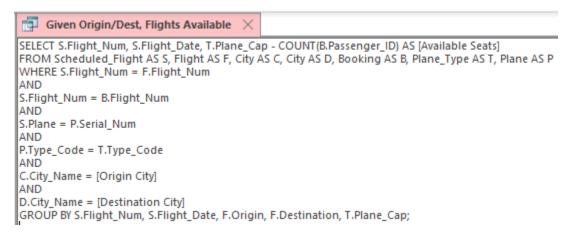
Page 1 of 1

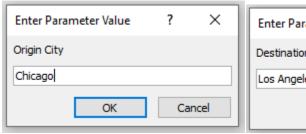
E)

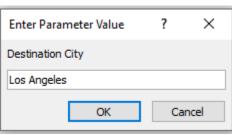


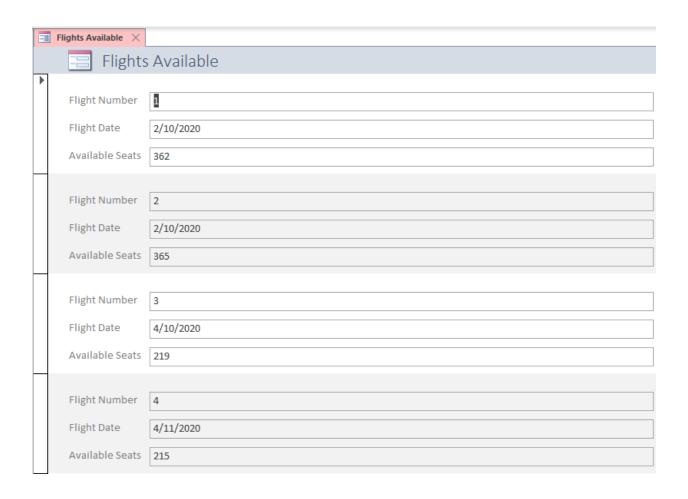
QUERIES

A)

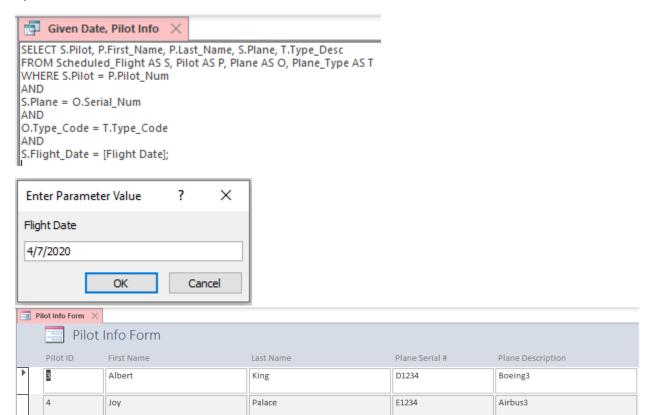








B)



C)

