**Lesson Name**: Einführung in dieDatenbanken

**Teacher Name**: Selçuk Kıran

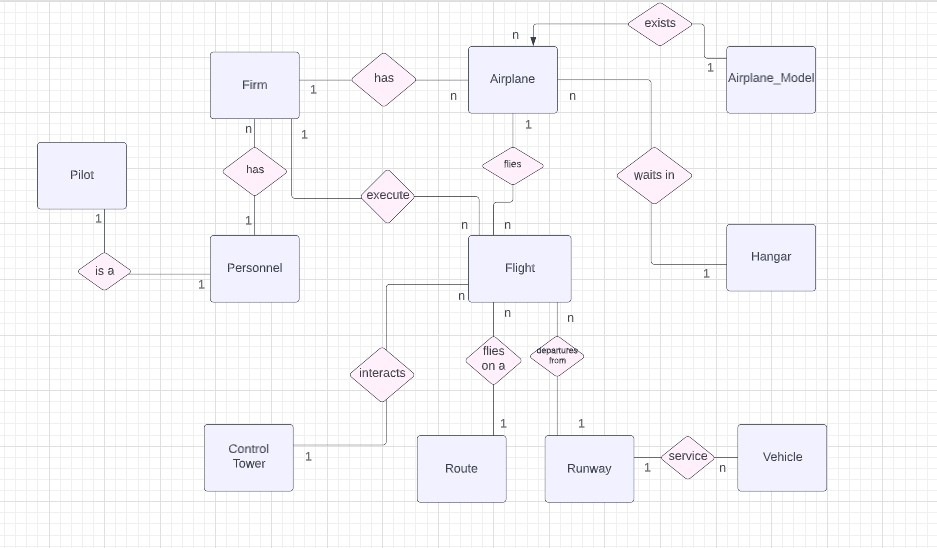
**Subject**: Airport Database

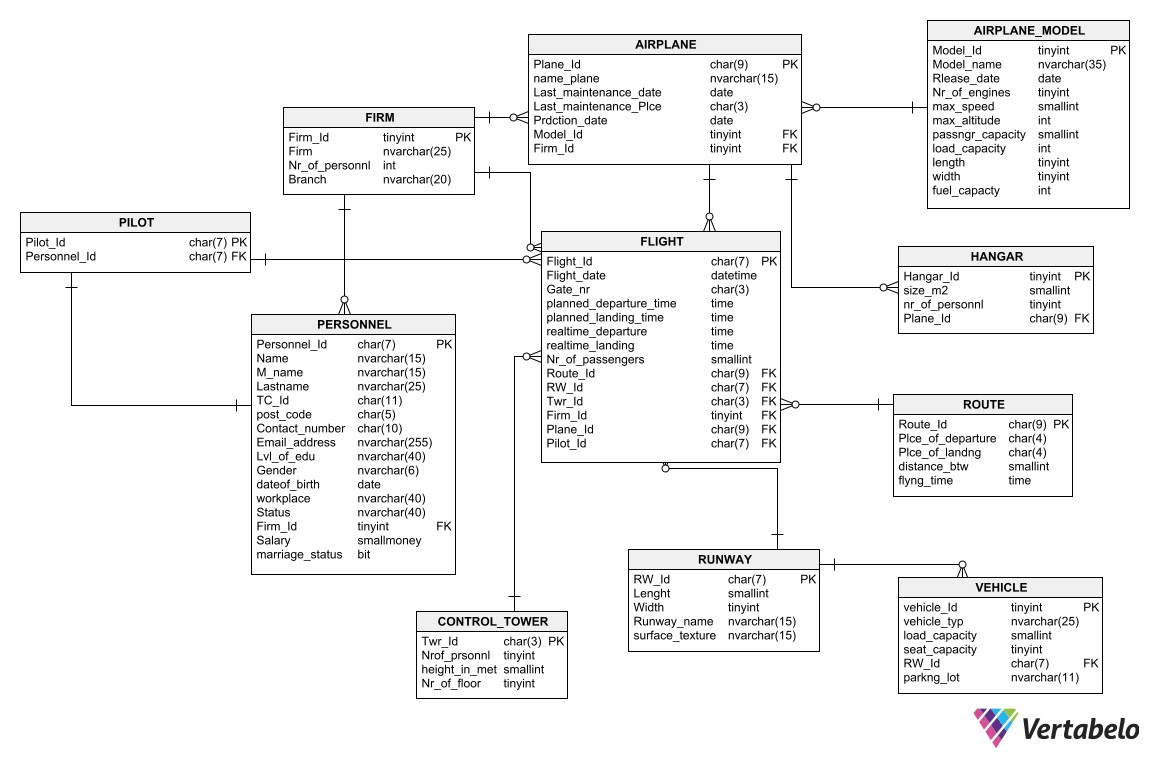
Batuhan Kayaoğlu - 138620036

Cansu Bahtiyar - 138620034

Ecrin Yıldırım - 1638620063

# Baran N. Aktar - 138620012





## Entries

We choose “Airport Database” topic for our homework. We want say something about our reasons for choosing that. While we were observing the homework topic table, we realize that most of our friends chose very similar topics. We took care to choose a topic that would tire us out a bit, that we would enjoy doing and that we would learn new things. And Airport is just for us. We would also like to state that, as Board Members of the newly formed Data Science Club, we take this assignment very seriously and we hope that it will provide us with a multi-faceted benefit.

## About Esenler Airport

Airport Esenler Airport is an establishment serving domestic routes, hosting 20-30 flights a day. We thought that the main interest of an airport was flights. That's why we set our flights table as the center. When we consider the most basic topics related to flights, we add Tower, Runway, Hangar, Plane and Employee Tables in our database. The other tables are derived from the tables we mentioned. There are 4 hangars at Esenler airport that do not provide maintenance services and are only used for aircraft parking. Each hangar is limited to 1 plane capacity. Hangar teams consist up to 5 people. These teams include technicians and engineers. Employees change their workplaces periodically for safety reasons. Employees table contains 2 types of employees. A large part of the employee data consists of employees working in air services such as pilots and cabin crew. Other employees on the sign belong to the companies that the airport works with for ground handling services such as cleaning and security.

### Flight Table

Flight Table gives us information about the route, plane, runway, departure and landing time, number of passengers and gate number. Flight\_Id is the Primary Key of this table and its data type is char(7). For example 2011415 is a flight id. In this id, first 3 digits refeer the firm (201). In this table; Firm\_Id, Twr\_Id, Route\_Id, RW\_Id, Plane\_Id are foreign keys. This table have 2 attributes for departure time and landing time. The difference between planned and realtime attributes is delay time. Airlines pay us a fee for every minute their planes use our runways. In order to calculate how much of an

increase in this fee in case of delay, we show the departure and landing times separately as planned and actual. We also do this so that our data is consistent and accurate.Both of the attributes has datetime as data type. The reason why there are so many foreign keys ithe flights table is that we have put the flights in the center of our database, as we said before.

### Personnel Table

By adding Personnel Table to our database, we aim to save the data from who Works at our airport and pilot in charge. First of all, we add “Personnel\_Id” attribute as a Primary Key. Every Personnel has a different

Personnel ID. We assign Firm\_Id as a Foreign Key because every personnel works at only one company and we want to specify it. Personnel Id’s data type is char(7). Because we want specifically 7 digits as Id. First 3 digits represent the company which personnel works at and last 4 digits represent the number of employees of the company. (For example if personel Id is 2010001, this worker works at the company 201 and he or she is the first employee.) E-mail’s data type is nvarchar(255). Because according to our

research, before the “@” sign we can use up to 64 characters and total maximum number of the character is 255. We saw that most of the databases has the nvarchar(255) like us. We decided to use small Money data type as salary type since the range is between 214,748.3647 and 214,748.3647. Monthly salaries are saved and this range is suitable for our attribute. Lastly we choose using bit type for marriage status because bit type is very useful for saving memory.

### Control Tower Table

We aim to control the number of personnel and flight safety in the towers by including the control tower table in the airport database. Tower crew is not crowded. So we specify the Nr\_prsnel attribute’s data type as tinyint. All aircraft which entering the airspace must contact and follow their instructions which have been given by the control tower. Therefore, the tower must be in a position

and height to see everything that happens at the airport. So we used smallint as the data type by giving the tower height as Height.in.meter in our table. There are multiple towers in the airport. In order to distinguish our towers from each other, we determined Twr\_ID as the Primery Key. Thus, we will be able to record which tower is managed by each flight. We chose char(3) as the data type because we specified a three-digit code system for assigning towers. In addition, we have included the number of floors in the tower with the attribute nr.of.floor and the tinyint data type in our table.

### Vehicle Table

There are special-purpose vehicles on the runways at our airport to be used for various tasks. In order to easily distinguish the vehicles, we assigned them an id as the Primary Key. We chose tinyint as the data type for this attribute in our table with the name vhcl\_id. Since the type (airstairs, belt loaders, apron buses…) load capacity and human capacity of the

vehicles are different, we have shown them with separate attributes. While we used load.capacity for the load and preferred smallint as the datatype, we used the seat\_capacity attribute with tinyint for the human capacity it can carry.

### Airplane Type Table

We have included the types and characteristics of the aircraft that will use in our airport in this table. We do not add this datas at “Airplane Table”. Because each airplane type has same features and we do not want to repeat that in every line at this table. So we create the Airplane Type table to normalize the database. We set an identification number for each aircraft model and used it as the Primary Key. We used tinyint as the data type for this attribute. We only specified the launch date of the airplane as the year.

so we used smallint as data type. We aimed to save

memory. We preferred smallint because the unit used for speed is km/h and it reaches 4-digit numbers. However, since feet are used for height in aviation and the heights expressed in this unit exceed the limits of smallint, so we preferred int. Especially since the amount of fuel required by high-capacity aircraft can reach 6-digit numbers, our choice here was int again. We preferred smallint, because the passenger capacity can exceed 800 people on wide-body aircraft.

## Airplane Table

Airplane Table contains one of the most important datas for flights. This table’s primary key is “Airplane\_Id” and their foreign keys are Model\_Id and Firm\_Id. Airplane\_Id has char(9) data type. An example for that is “TC2010304”. We specifically want 9 digits code. In this example “TC” refeers to the country of the airplane, first 3 number digits (201) remind us airplane’s firm, 6. And 7. Digits (03) are for airplane’s type and last 2 digits (04) are for number of airplanes which belongs same firm and airplane model. A cell with name\_plane attribute can be null. Because we know that some of the airline firms give names to the airplanes but also some of them not. The model\_id, which is used as the primary key in the airplane\_model table, is used as the foreign key in this table. We aimed to include attributes other than the model's name, so we linked the other table via model\_id. Last\_Maintanane\_day gives us the last maintenance date of the mentioned aircraft and Last\_Maintenance\_Plce gives us the last airport where this aircraft was serviced. Last maintence place’s data type is char(4). Because we use the ICAO code system. ICAO codes consist of 4 letters assigned to the airport. The first letter gives the region where the airport is located, and the second letter gives the country where the airport is located. The other 2 letters are usually given in order. For example Esenboğa Airport’s ICAO code is “LTAC”.

### Firm Table

We add firm table because we want to collect the data from companies we work with. Firm\_Id is primary key and its data type is tinyint because of saving memory. The Id’s between 200-255 are airline companies and between 100-200 are other companies. Rest of the attributes are firm name, the personnel number they have and their branch

(security, cleaning, airline…)

### Route Table

We want to give a place to the Route Table in our database because we aim to have all datas which include place of departure and landing, distance and flying time. First of all, we add Route\_Id as Primary key. Route Id’s data type is char(9). Reason of this is using ICAO Code system like we did in Airplane table, in last maintance place attribute. If we try to remind you that what is ICAO

code system, we will say every airport has unique 4 digits code which is the ICAO code. In this attribute first 4 digits refeer to the which airport did the plane take off and last 4 digits refeer to the which airport did the plane landed. For example if the id is “LTAC-LTAI”, it means this route is from Ankara Esenboğa Airport(LTAC) to Antalya Airport(LTAI).

### Hangar Table

Hangar is a simple unit in airport but also one of the main parts too. First, we set the hangar id as the primary key. Hangar Id’s data type is tinyint because it is sequential and ascending in the form like 1- 2-3-4. In order to understand which aircraft are kept in which hangar, we added the airplane id attribute to our Hangar table.

### Pilot Table

In order to add pilot datas to the Flight table, we must create this table. In Personnel Table, a unique Id was identified each employee. However, we can not add this Id as a pilot data. So we prepare Pilot table and add only 2 attributes. First attribute is Pilot\_Id and it is primary key. Its type is char(7)

because we want to remind the firm which pilot works in first 3 digits. Last 4 digits refeers the number of pilots this company works with. Personnel\_Id is Foreign Key. It directs us to Personnel Table so we can find all datas which we required.

### Runway Table

Table We want to give a place to the runways and their information in our database. Because runways are one of the main elements in airport. The most important part in table is Id. We specify the data type as char(7). Because runways are named like "18R/37L" which refeers to the closest radial in compass. According to degree system, 360 is north, 180 is south, 270 is west an 90 is east. The 3 digits number which been showed

in compass, has converted 2- digit form. This number becomes a name of the runway. And letters refeers the landing-take off direction (L for left and R for right). According to our researchs, the longest runways is 5500 metres. So we prefer to define "Length" data type as "smallint". The largest runway in the world is 77 metres. Because of that our "Width" data type is "tinyint."



**Ich schwöre im Namen meiner Ehre und all der Werte, von denen ich geglaubt habe, dass ich bei dieser Prüfung gar nichts von irgendwo kopiert und eingefügt habe.**

**Batuhan Kayaoğlu , Cansu Bahtiyar , Ecrin Yıldırım , Baran N. Aktar**