LAU
School of Arts and Sciences
Department of Computer Science & Mathematics

REPORT

DATABASE DESIGN FOR WANAKANDA AIRPOT
BY

AVENGERS DB

Submitted by:

- Heba Rachid
  heba.rachid@lau.edu
- Jehad Oumer
  Jehad.oumer@lau.edu
- Mehiddine Bachache
  mehidine.bachache@lau.edu
- Abdelouahab Elkouadi
  Abdelouahab.elkouadi@lau.edu
- Ahmad Houmani
  ahmad.houmani01@lau.edu

to Dr. Ramzi R. Haraty in partial fulfillment of the requirements for the course “CSC375: Database Management Systems” in Computer Science

Phase IV – 28203 words – 210 pages

November 23,2021

EVALUATION:

Dear Dr Ramzi, thank you for this semester full of knowledge and information. We kindly ask you to fill out this evaluation form to further benefit us letting us know our strong points and improve our weaknesses.

Thank you for filling this form.

- Overall grade:

- ER diagram rating:

- Overall report rating:

- Comments about the project:

LICENSE

Whomever has this report is granted a full access to the ER diagram, the design and all the other byproducts of this project. The user will automatically be welcomed to update, insert, delete, merge or publish any data instances without any restrictions or limitations as long as the following conditions are met:

A copy rights notice should be mentioned in all copies or derivatives extracted from this work.

The authors have full ownership of this product and they are entitled to authorize or refuse any attempt to reference the above work in any way, shape or form.

Any violation of the above conditions might subject the violator to serious judicial measures as per the copy right laws and regulations

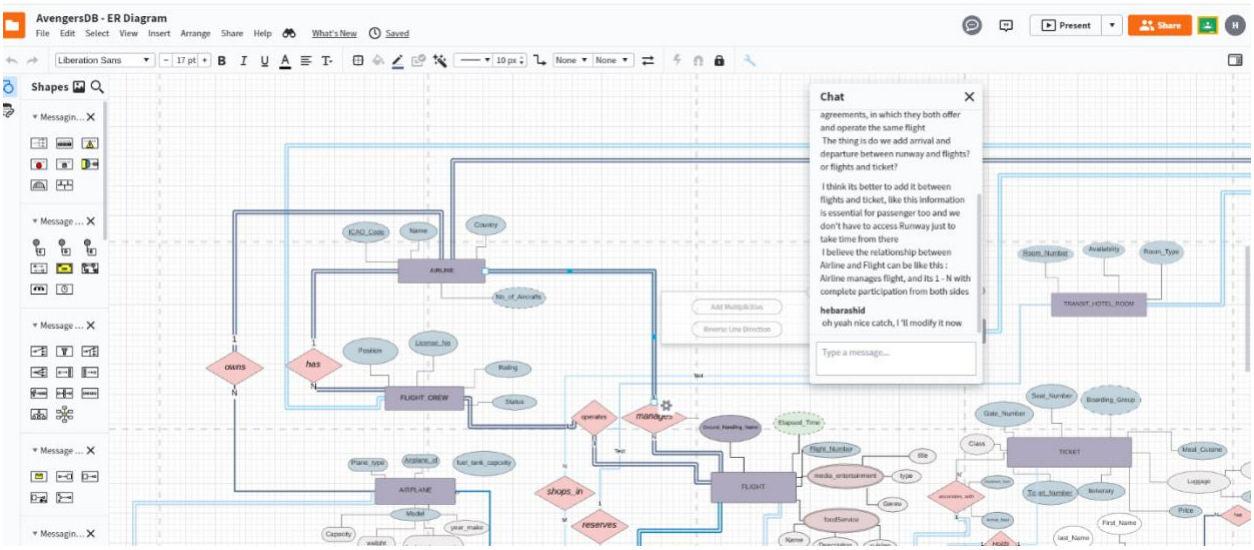
This page intentionally left blank.

Contents

I. Introduction

"Avengers DB "takes the pleasure to submit its report for our project entitled "Birnin Zana International Airport".

Think about any undertaking, from running a very small local store, an insurance company, medical research, the tax collector, pretty much anything. What is one thing that they ALL have in common? They have a need to keep information. Now, 75 years ago, all that information was stored where? Paper. However, since the days of ENIAC, more and more businesses, governments, and individuals, have moved to storing information digitally. This project has provided us with the opportunity to deal with databases at many level, deal with group members, accept new ideas, reject old ones, and lastly design the best version of our selves reflected through the Entity-Relation diagram, schema and more…



We created a WhatsApp group to decide tasks and discuss the project, we also been meeting frequently on zoom to discuss strategies and go over our designs. And lucid chart was our collaborative design platform that we used frequently for discussing our entities and schema.

After all the obstacles we faced, we believe this project has made us gain crucial skills including better thinking process, brainstorming and we had to do extensive researches to understand each entity in a better way and design the most needed and precise attributes. This report will unravel the different viewports of what we have accomplished as a team when designing the airport database.

II. Introduction:

In recent years, Wakanda under the role of King T'Challa "Black Panther" has undergone radical progressive development, with the change of their foreign policies, Wakanda is starting to become visible to the world, now that the United Nations has recognized Wakanda as a sovereign state with the official name of "Kingdom of Wakanda", countries, big cooperation and even representatives from other planets are racing to establish relationships with the kingdom as its becoming the center of advanced technologies and the only source of the rarest element that is stronger than steel and three times lighter than its weight. The current capacities of Wakanda international airport will be soon insufficient to fulfill the massive traffic of people and goods locally and internationally. Thus the Wakandian government is working on implementing their 2025 expansion plan to build the largest airport in the universe in the suburbs of the Golden City "Birnin Zana" the capital of Wakanda, which will serve the needs of the wakandian development.

With the close ties that the King T'Challa "Black Panther" has with the team of world-class superheroes and part time database developers "The Avengers", the latter managed to secure a three-years contract with the Wakandian aviation administration to work on developing and maintaining the database system of the new Birnin Zana International Airport (IATA Code: BZG, ICAO Code: HWVA).

The Avengers have assigned this mission to 5 of their best members namely Jehad Oumer (Spiderman), Heba Rachid (Black Widow), Abdelouahab Elkouadi (Iron Man), Mehidine Bachachi (Thor) and Ahmed Houmani (Quicksilver) to work abroad in the Golden City along with their Wakandian counterparts to develop and maintain the airport's database system.

The Avengers contract began on September 1st, 2020, and they have divided the 3 years' project into smaller milestones, their first milestone is due November 23rd. 2020. Whereby the avengers will have to design the database system and prepare a report for the authorities proposing the structure of the database system, a demo and a progress report.

It is highly appreciated; if you did not watch black panther movie, to watch this 2 min video about wakanda and its beauty https://youtu.be/r8nxdAGg2FY .

III. First Milestone

The avengers had 3 months to design the system, due to lockdown measures the team had to work remotely, this was a hard task for the team, but the avengers have accomplished the missions successfully. They have divided their work into 4 phases.

Phase 1: the team worked on identifying the objects and their characteristics that are involved in the relationship, defining them as entities with attributes that give them meaning, the team then established relationships between the different entities. The end product of this phase was a comprehensive Entities-Relationships (ER) Diagram that visualizes the relational database system design in an abstract manner.

Phase 2: at this phase the team takes the ER diagram and converts it to a relational schema that consists of tables and defined relationships between tables constructed by primary and foreign keys.

Phase 3: after the team have managed to transfer the abstract ER diagram into a schematic blueprint, the team now have all the information they need to start creating the database virtually, Avengers execute SQL commands to create tables, populate them with a demo info, establish relationships, set constraints and design different possible views for the database.

Phase 4: The Avengers always aim to provide optimal and efficient solutions, thus the team is adopting the normalization process in the project, whereby the team re-evaluates the schematic design of the database and explores ways to optimize it.

The Avengers believes that the database design process is an iterative one, there is always a need to re-evaluate, adding additional features and improvements.

IV. System Description and Constraints

Birnin Zana International Airport (IATA Code: BZG, ICAO Code: HWVA) soon will be the new largest airport in the entire galaxy. The airport is a hub for domestic, international and intergalactic flights. This airport is managed by the Wakandian Aviation Authority (WAA) and it hosts nearly 90% of airline and shipping companies in the world. It hosts

The database system of the airport that the Avengers designed considers the main and viable airport objects of operations, including airlines, shipping companies, passengers, luggage, packages, airplanes, and concepts such as flights and tickets.

The database system deals with complex relationships between different entities such as an airline has airplanes, passengers pay tickets and has luggage, flights are on specific airplanes that are maintained by technical service providers. Shipping companies handling different types of packages and departments have hundreds of employees.

To maintain the integrity of the database and ensure consistency, the avengers have set constraints to prevent any violation to the integrity of the database. We have found use of all of the standard constraints, including but not limited to Domain constraints. Tuple Uniqueness constraint. Key constraint. Entity Integrity constraint and Referential Integrity constraint. Those constraints were important to preserve the database integrity, records such flights, airplanes, passengers need to remain distinct and make a unique combination when related to each other. Additionally, we have set some manual constraints such as that 2 flights can't be using the same runway at the same time, passenger visa expiry date should not be earlier than the flight date, flight crew should be made sure that he is "in-duty" before assigning them to a flight. Flight seat numbers should follow a certain entry format; a passenger cannot book a ticket when the number of tickets sold equals the number of seats of that specific airport…etc.

The previous are brief examples of some constraints we used in our database designs, more specific constraints will be presented later in the report.

Avengers have always been a fan oracle products, and they choose to use Oracle Database Express to build the database.

The Avengers will like to acknowledge the effort of their mentor Dr. Ramzi A. Haraty (Doctor Strange) for his guidance throughout the project.

V. ER Diagram Symbols:

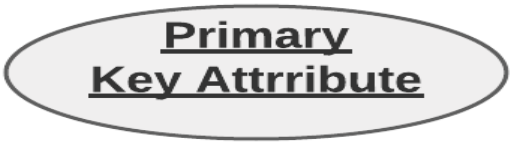
Entity Relationship Diagram represents the relationships between a distinct yet related set of entities that are the core of a database system. ER diagram is a tool used to explain the logical structure of databases and its creation is based on 3 concepts:

Entities, Attributes and relationships.

- Rectangles are used to represent entities
- Ovals are used to define attributes
- Diamond shapes are used to represent relationships

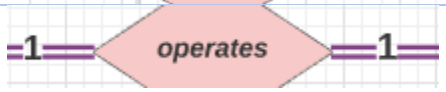
Demonstration:

The purpose of ER diagram is to provide a visual starting point for database design. To simplify the understanding and the grasp of our ER diagram we decided to color each tool used based on the listed criteria below. We are glad with how everything turned out to be, hope you will be too!

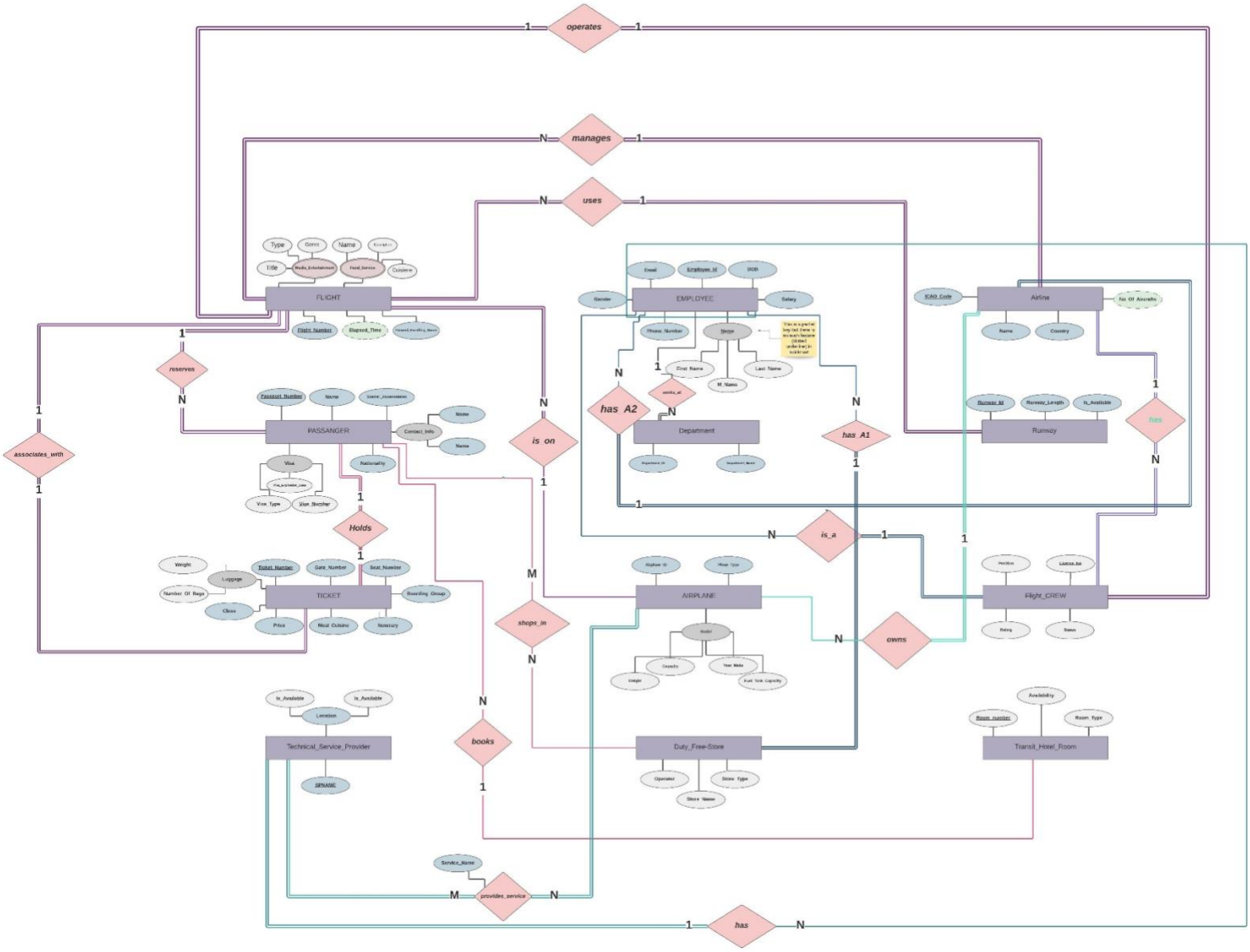
Entity	
Attribute	
Primary Key Attribute	
Multivalued Attribute	

Derived Attribute	
Partial Key	
Composite Attribute	
Relationship	

Cardinality and Participation

Total Participation	
Partial Participation	
Many to Many relations	
One to Many Relation	
One to One Relation	

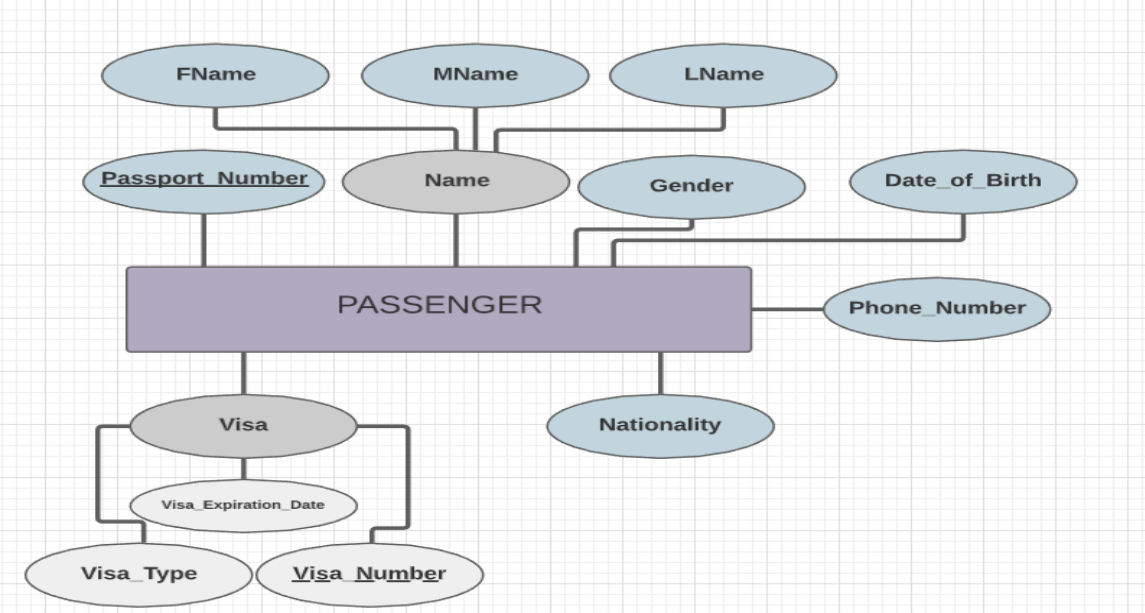
VI. Entity Relation Diagram for Wakanda Airport:



- *If anybody is finding some hardship while zooming and reading the entities. Don't worry. We'll add a lucid chart Link that can be accessed directly and there you can zoom without losing any resolution. Plus, we'll discuss every single entity below.*
  https://lucid.app/invitations/accept/5827c29c-37a5-4cf6-87f3-40007fc456fc

VII. Entity types:

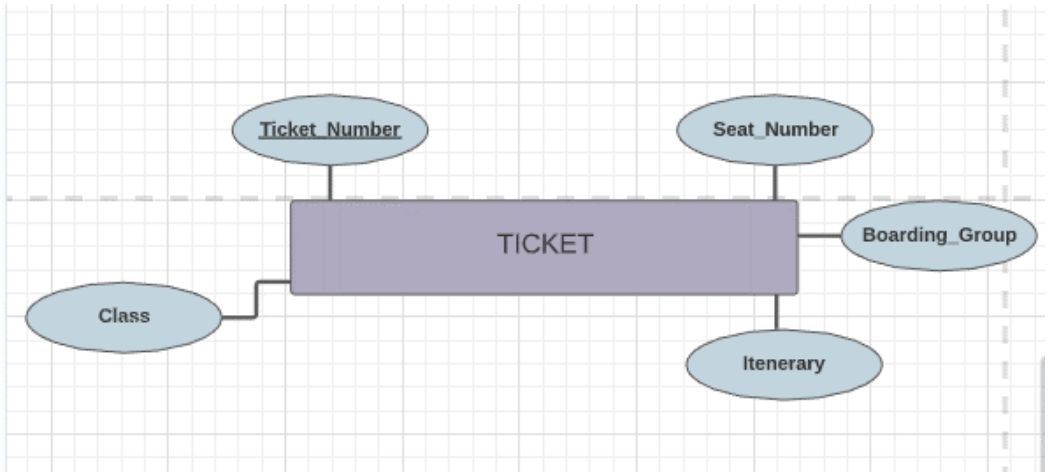
1. Passenger:



Airport systems are designed to accustom passengers as they are the main recipients and they constitute a major entity in the airport's buildup. This entity is directly related to a lot of other key entities since most of the airport facilities and services are directed towards it. In this database, passengers are defined through a set of attributes that differentiate each passenger and give them their own distinguishable identifiers. Each passenger is distinguished by their Passport_Number. It also includes the gender of the passenger as Gender, and the Name of the passenger as a composite attribute that contains 3 simple attributes that designate a passenger's first name as FName, middle Initial as MName, last name as LName along with a passenger's Date_of_Birth, Phone_Number and Nationality. Lastly, this entity includes a composite attribute Visa to assure the validity of a Passenger's visa, this composite attribute contains 3 simple attributes, Visa_Expiration_Date that indicates when a passenger's visa is no longer valid, Visa_Number, and Visa_Type.

- Passport_Number: is of the following format "CC######", CC are the two characters that represent a passenger's city and the following 6 numbers are unique for each passenger.
- Visa_Number: The visa number of each passenger consists of 8 numbers and its unique for each passenger, following this format ########.
- Visa_Type: The vista type of each passenger is either Student, Work, Immigrant, or Tourist.
- Visa_Expiration_Date: The visa expiration date is a date of the following format YYYY-MM-DD.

2. **Ticket:**

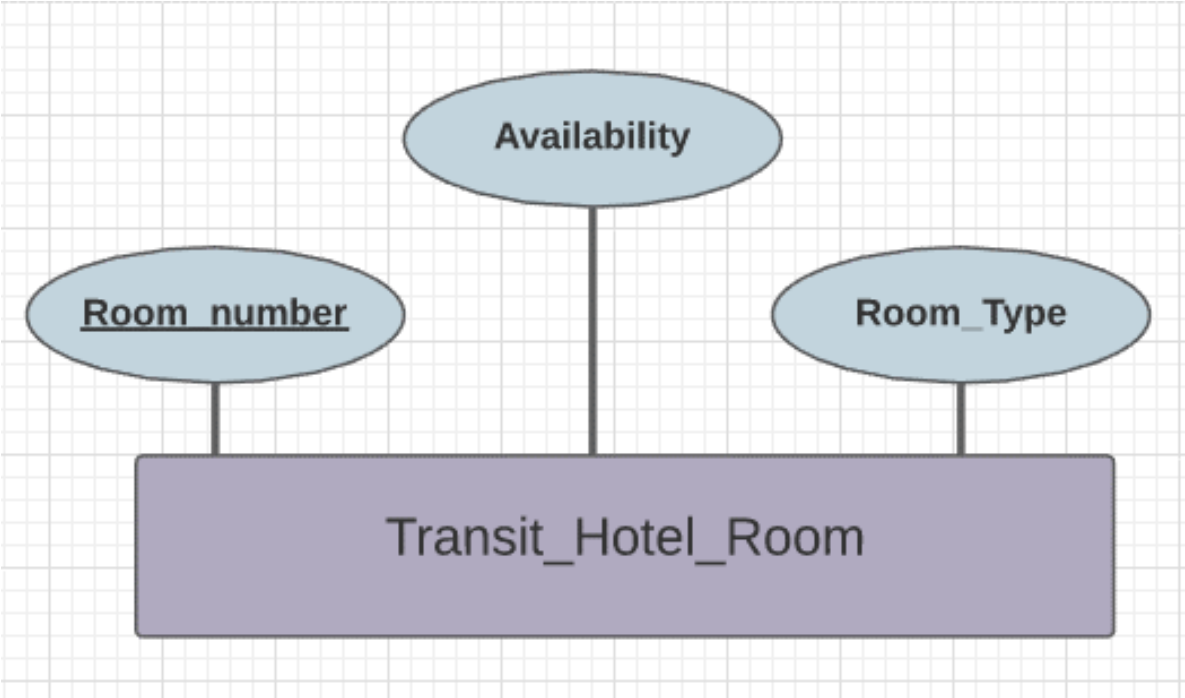


A ticket is the practical mean that enables the passenger to move from location A to B. This entity should have all the details concerning the trip and its different features. In this database, tickets are defined through a set of attributes that differentiate each ticket and give them their own distinguishable identifiers. First,

each ticket is distinguished by its Ticket_Number. In addition, it also has several attributes to easily guide the Passenger to board a plane such as Seat_Number and Boarding_Group that indicate the passenger's seat in a plane and which boarding group the passenger belongs to. Also, this entity type helps a passenger identify his cabin class designated as Class, and the Flight's route as Itenerary.

- Ticket_Number: The ticket number of each passenger is of the following format ########CCC######, The first 8 digits correspond to the date of the flight represented by this ticket following the format YYYY-MM-DD, the next 3 characters followed by the 3 digits CCC### represent the airline's icao code followed by 3 digits to constitute as a whole the specific flight number, and the 3 final digits ### are unique for each passenger.
- Seat_Number: The seat number is of the following format ##C, where the 2 digits ## represent the row and the Character C represents the specific seat in that row.
- Class: The class attribute is either First, Economy, or Business.
- Boarding_Group: The boarding group in each ticket is represented by a single digit #.
- Itenerary: can be either direct or connecting flight.

3. **Transit Hotel Room:**

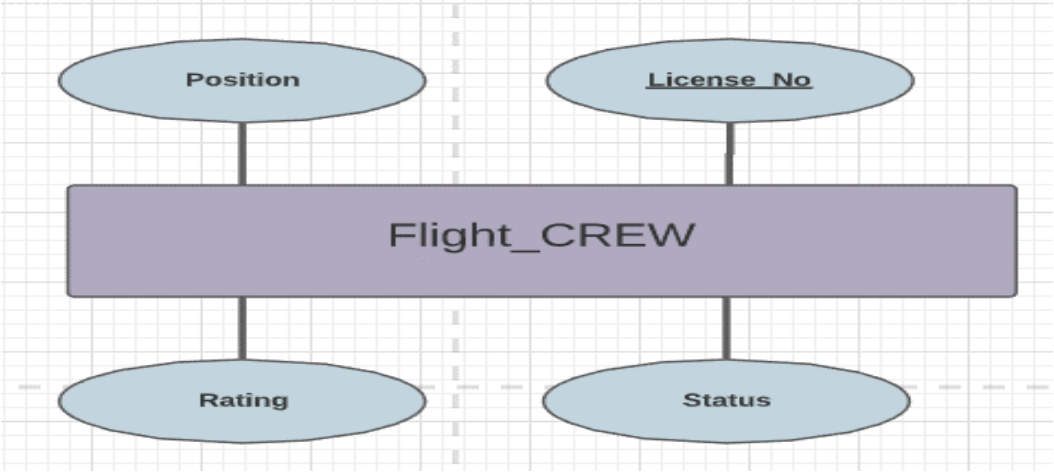


Transit hotel is the place where passengers spend their time in case they have a long transit. It is a necessity in the airport since a lot of connecting flights take place in there. Transit hotel rooms have a number attribute to be easily identified. The **Room_number** is a key attribute for this entity, and it is unique for each room. Information such as the Availability of a room is an attribute that we need to keep track of for managerial purposes. To facilitate the organization of the Transit hotel an additional attribute called Room_Type (single, double, suite…) is added to help better accommodate the passengers.

- **Room_number:** The room number consists of three digits ###.
- Availability: can be represented by a 1 if taken by another passenger(unavailable) and 0 if not taken by another passenger (available).

- Room_Type: The room type can be a single room, double room, deluxe room, or a suite.

4. **Flight Crew**

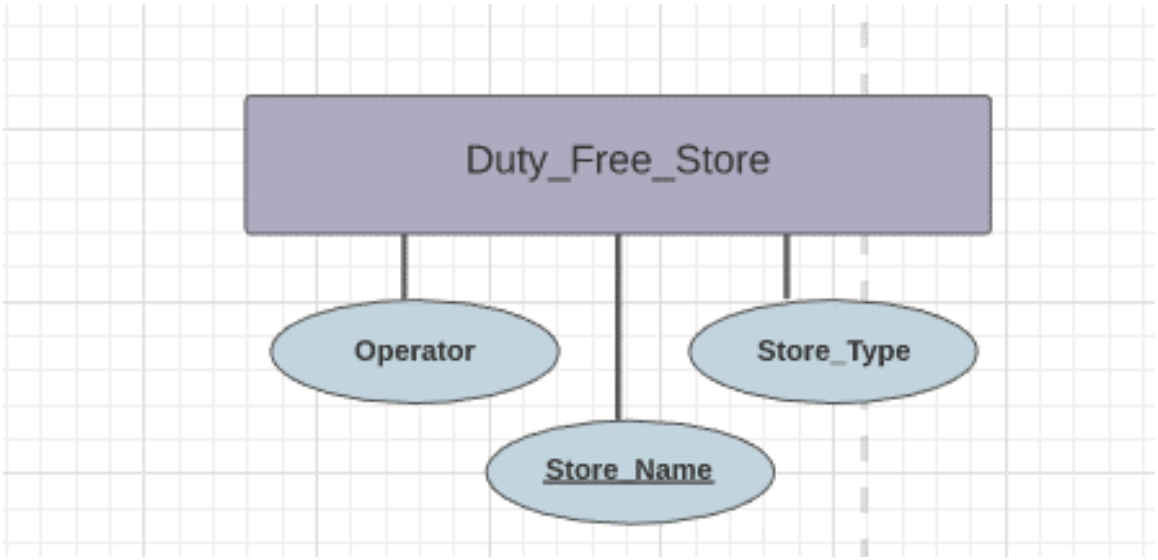


Flight crew personnel are persons who are designated to operate the airplane during the flight. Flight crew personnel are identified by their unique **License_No** and their names are retrieved by the relationship between Flight_CREW and EMPLOYEE entities. Members of flight crew hold different Position; they are either flight attendants - that serve and take care of the passengers during the flight - or cockpit crew (first officer and a second officer for long flights). Moreover, each flight crew personnel have a Rating based on evaluation from their supervisors. Additionally, the Status indicates whether the personnel is in-service or off-service due to reasons such as expiry of their license, retired or on leave.

- **License_No:** The license number consists of a unique combination of 8 digits ########.

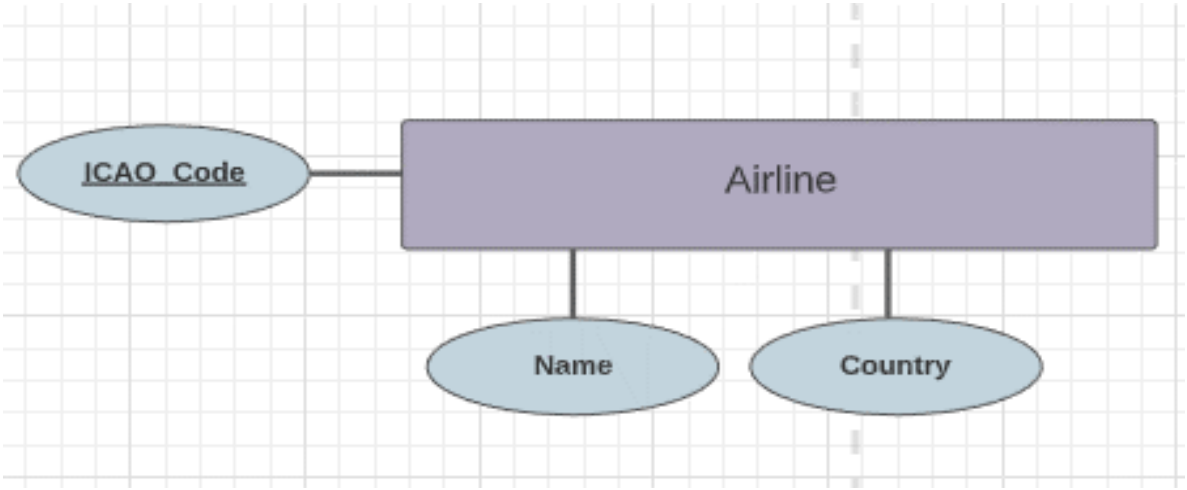
- Position: A flight crew member can either be a Pilot, an Attendant or a First Officer. Each flight crew can consist of 2 attendants and 1 pilot OR 1 attendant, 1 police officer and 1 pilot.
- Rating: The rating of a flight crew member is represented by one digit, ranging from 1 to 5, 1 being the worst rating and 5 is the best rating.
- Status : A flight crew member can have a status of in service, out of service or license expired.

5. Duty Free Store:



Duty free stores are stores that are exempted from paying certain local or international taxes. Passengers can buy products at lower prices in the airport. A Duty_Free_Store exists in the international zones of the airport, stores in the duty-free zones are identified by their unique **Store Name**, each store has an **Operator** which is a company that owns and operates a store or multiple stores, and **Store_type** defines the type of goods they are selling.

6. Airline:

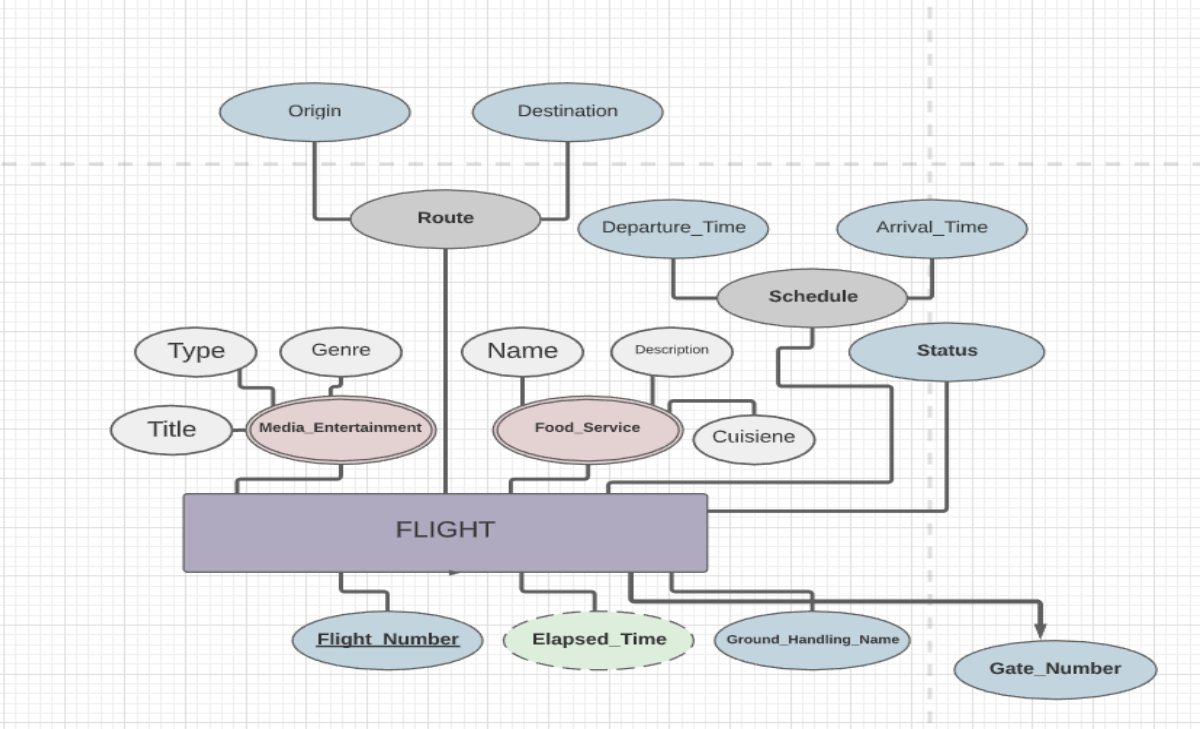


An airline is the company that provides transportation service for passengers through air. An airport hosts many airlines. Airlines are distinguished by their designated **ICAO_Code** (The International Civil Aviation Organization Airlines Designators). Additionally, an airline is attributed by its official business Name along with their Country of legal origin.

Each airline has their own fleet of aircrafts, and the total No_of_Aircrafts is derived from the count of airplanes that are owned by the airline.

- **ICAO_Code:** Each airline has a specific code that consists of three characters CCC that are unique for each airline.

7. Flight:

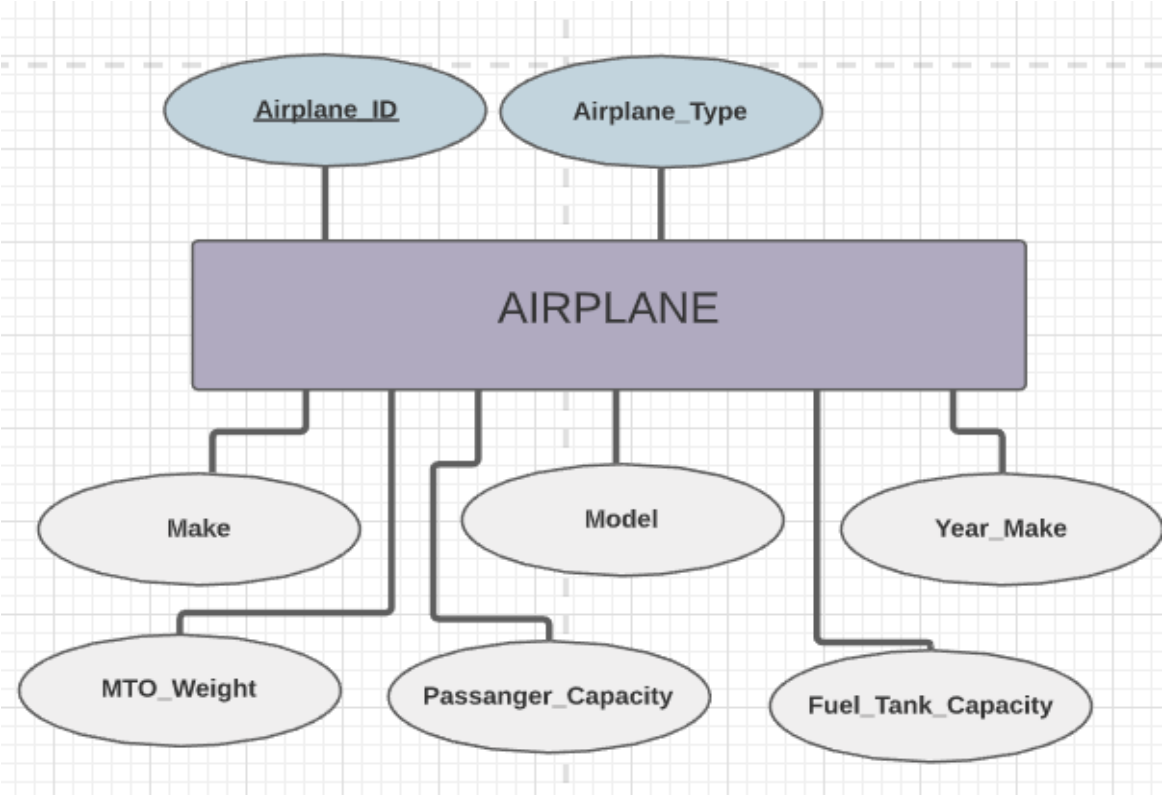


A flight is the physical movement of the plane in air, transporting passengers from one place to another. This entity type should hold all necessary attributes that help us identify the details of the trip. It is referred to by using a unique identifier, **Flight_Number** as a key attribute to avoid confusion between pilots and air traffic control towers. It is important to know when will this flight depart from an airport and when it will arrive to the other, for that purpose it includes a composite attribute Schedule that contains 2 simple attributes Departure_Time and Arrival_Time. Additionally, this entity type includes a composite attribute Route that contains 2 simple attributes, Origin that represents the departure location and Destination that represents the arrival location. It also includes the derived attribute Elapsed_time which specifies the approximate time in air for a passenger while moving from airport A to airport B. in addition to an attribute Ground_Handling_Name that represents the name of the ground handling service that manages this flight. Furthermore, it has a Status attribute to specify the condition of the Flight (cancelled, delayed, on time, boarding). Lastly, this entity type also has 2 multivalued composite attributes, one of them is Media_Entertainment, that contains 3 simple attributes

title, type and genre that includes the different forms of entertainment on that certain flight such as movies, songs, series; Food_Service also has 3 simple attributes when broken down from the composite state Name of the meal**,** Description of the meal and Cuisine of the meal. There is also the Gate_Number attribute that indicate which gate the passengers are using to board on their plane.

- Flight_Number: The flight number is of the following format CCC###, where the first 3 characters represent the airline ICAO code and the 3 digits ### are unique for each flight.
- Departure_Time: The departure time is a time stamp of the following format **YYYY-MM-DD HH24: MI.**
- Arrival_Time: The arrival time is a time stamp of the following format **YYYY-MM-DD HH24: MI.**
- Origin: Consists of 3 characters CCC that represent the airport's specific code.
- Destination: Consists of 3 characters CCC that represent the airport's specific code.
- Ground_Handling_Name: Consists of a set of characters that determines the ground handling services name such as "BBA Aviation plc" "Amadeus IT Holding SA".
- Status: Each flight has a set of characters that determine its status that can be either **Delayed, on time, Cancelled or Boarding**.
- Gate_Number: Each gate number is of the following format **C##C**; this combination is unique for each gate.

8. Airplane:

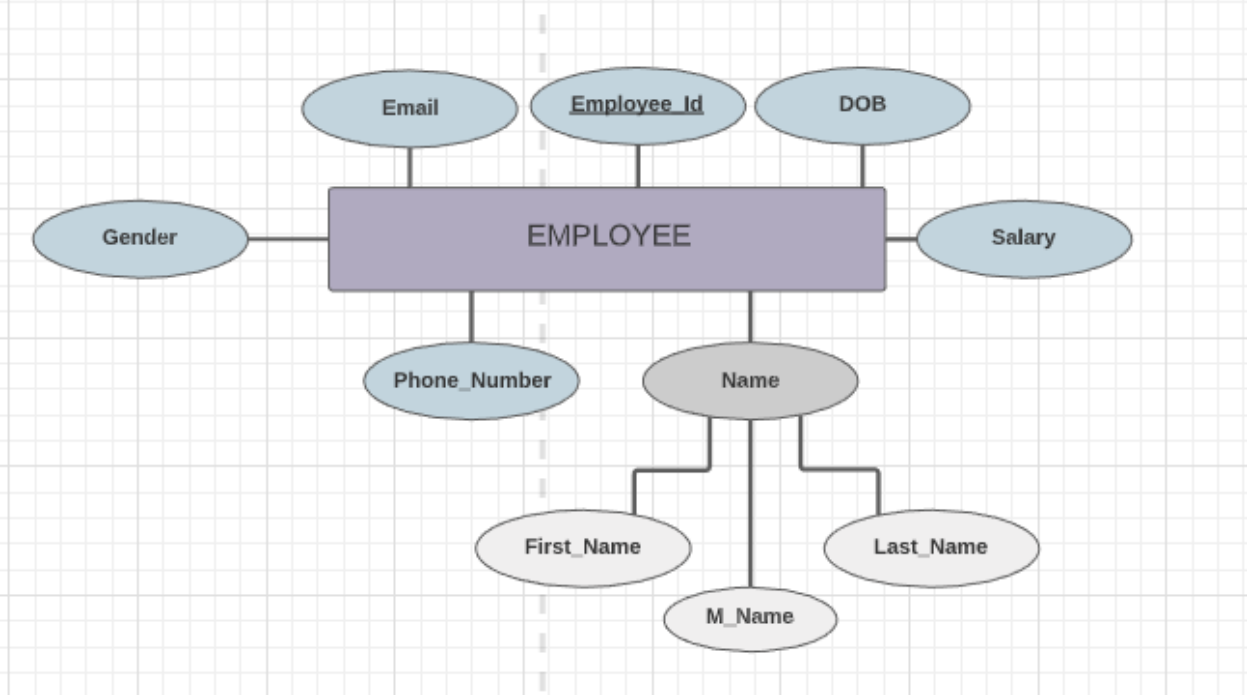


An airplane is the mean of transportation that carries passengers from one place to another in air, this entity type 2 simple attributes, a unique identifier **Airplane_ID** to specifically represent an airplane, Airplane_type to specify if it is a private or commercial airplane. Additionally, it is important to know characteristics of a specific airplane, for that purpose, this entity type holds several attributes such as the manufacturer of the airplane as Make, what model the airplane is as Model, year of make as Year_Make, the maximum number of passengers it can hold as Passenger_Capacity, the maximum weight a plane can takeoff while carrying as MTO_Weight and lastly its Kerosene tank capacity as Fuel_Tank_Capacity.

- **Airplane_ID:** Each airplane has a unique ID that consists of **CCC-###** where the first 3 characters represent the airline that owns the airplane and the 3 digits are unique for each airplane that belongs to that specific airline**.**

- Airplane_type: Each airplane in our airport can either be of type **cargo** or **commercial**.
- Make: A set of characters that determines the manufacturer name of the airplane.
- Model: A set of characters that determines the specific **model** of the airplane.
- Year_Make: is of the format **YYYY**, determines the year of production of the airplane.
- MTO_Weight: Represented by **digits** that specify the maximum takeoff weight in **tonnes**.
- Passenger_Capacity: Digits that represent how many passengers can the plane carry.
- Fuel_Tank_Capacity: Represents the maximum capacity of an airplane's fuel tank in **US GALLONS.**

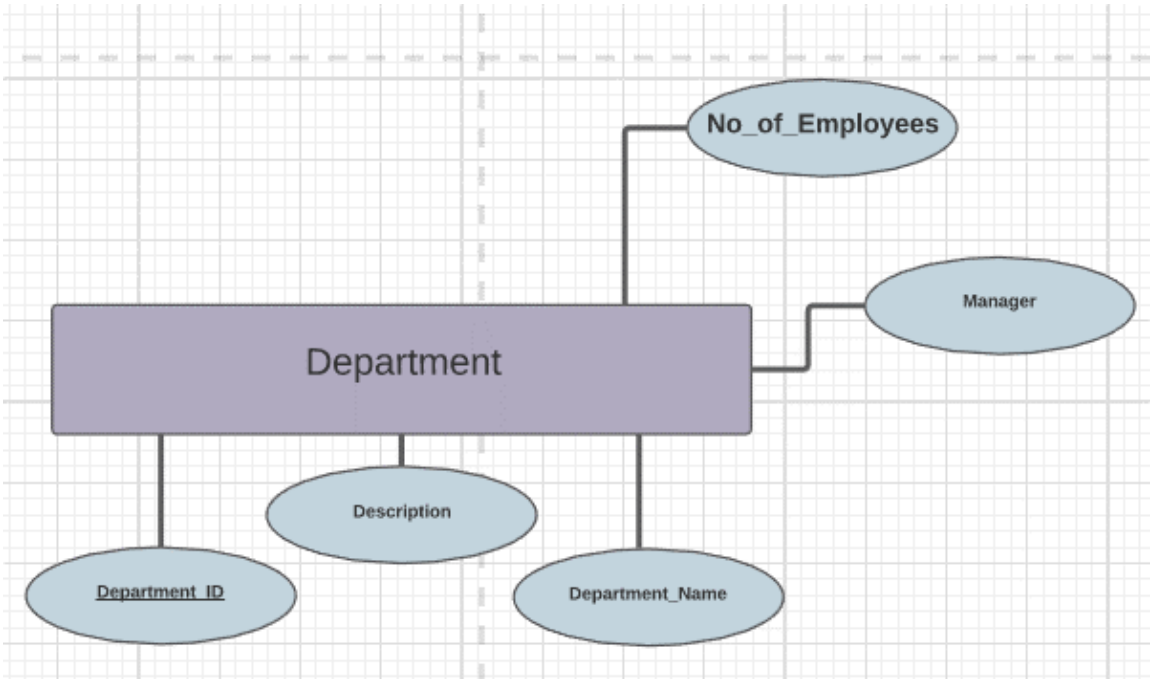
9. Employee



In our database we keep records of employees working on different airport's disciplines such as Airline, Flight crew, technical service provider, Duty free store employees. Employee's primary key is their Employee_ID for it holds a unique identification number of the employee. We Also keep record of other information such as the full name of the employee designated by a composite attribute Name that contains 3 simple attributes to designate an employee's First_Name, middle initial as M_Name, and Last_Name. Additionally, we have many simple attributes such as Salary to keep track of an employee's salary, and we are keeping track of the personal Email and Phone_Number of an employee, the Gender of an employee: male, female or other. Also, this entity type holds DOB attribute to store the date of birth of the employee, it must be strictly in the format: MM-DD-YYYY.

- Employee_ID: Consists of **5 digits** that are unique for each employee.
- Phone_Number: Consists of **a set of digits** starting with 00, followed by the country/area code and the specific phone number of an employee.
- Salary: A **set of digits** that determine an employee's monthly salary in **US DOLLARS**.
- Gender: can either be **M** for male, and **F** for female.

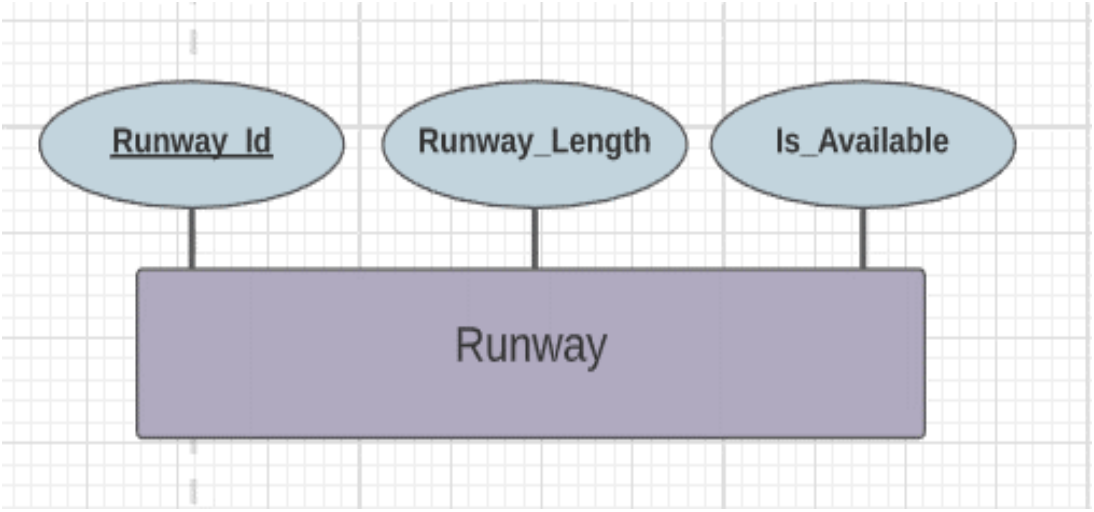
10. Department



A Department is a division of the airport where employees are devoted to a particular organization and discipline. It categorizes the activities that should be done by a certain employee. All the attributes are simple, Department_ID is a primary key that is a unique identifier. Other attributes include the Department_Name, Description of the department and the Manager of a specific department. Lasty, this entity type also holds the current number of employees this department has as No_of_Employees.

- Department_ID: Consists of a single **digit** that represents a specific department.
- Department_ID: A **set of characters** that determine the Department's name.
- Manager: Represents the Manager's **EMPLOYEE_ID.**

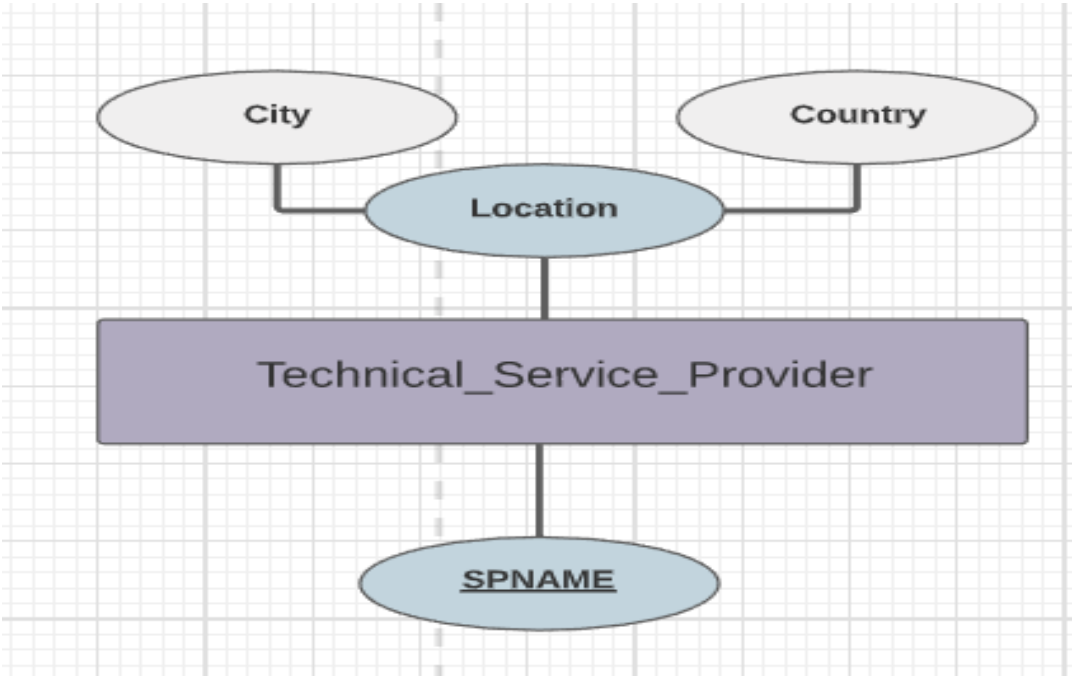
11.Runway



A runway is a big rectangular area in the airport prepared for the landing and takeoff of aircrafts. An airport structure supports multiple Runways; therefore, we used a unique identifier, **Runway_ID** as a key attribute to differentiate between them. Runways inside the same airport have different lengths therefore, each Runway has a Runway_Length, that indicates the length of the runway and the purpose or use of the Runway. An essential attribute called is available indicates the status of the runway (if it Is available for planes or packed with other aircrafts).

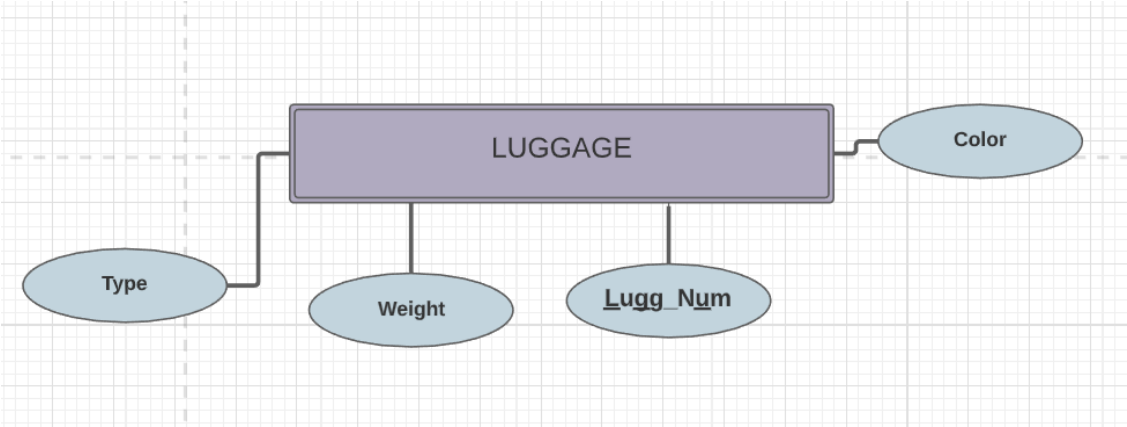
- Runway_ID: Each runway has a unique **ID** consisting of ##C, 2 digits followed by a character.
- Runway_Length: A set of digits that determines the runway's **length in meters**.
- Is_Available: can be represented by a 0 if the runway cannot accommodate any airplane(unavailable) and 1 if its free of airplanes (available).

12. Technical Service Provider



Airports and airlines have a joint interest in making sure that the airport maintains a consistent performance. For this reason, Technical_Service_Provider was made to deliver continuously consistent levels of service. This entity type has a simple key attribute **SPNAME**, which indicates the name of the service provider, and one composite attribute Location that can be broken down into two simple attributes country, and city providing information about which service provider is functioning in which location.

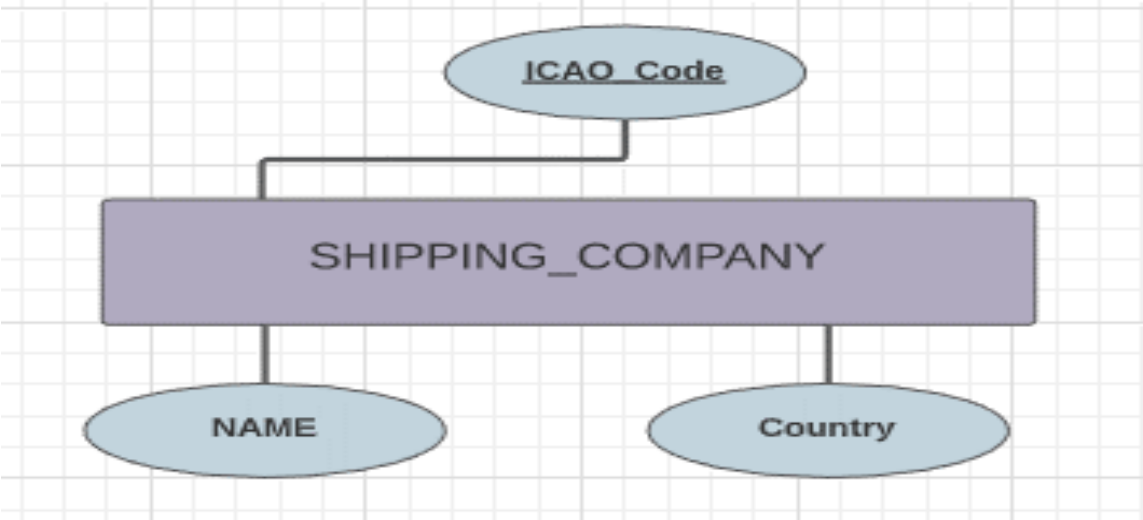
13.LUGGAGE



A suitcase, a carryon bag, or a regular travel bag is one of the most essential items that a passenger carries when travelling. For this reason, LUGGAGE entity type was made to ensure that we keep track of each passenger's luggage. This entity type has a partial key Lugg_Num, which helps identifying a person's luggage, and has 3 simple attributes, color that indicates the color of the luggage, weight to ensure that the luggage weight doesn't exceed the specified limit, type to specify if the luggage is a carry-on bag, a suitcase, a traditional bag or other types of bags.

- Lugg_Num: Consists of **8 digits** to specify a luggage number.
- Type: A **set of characters** to specify the type of the luggage(suitcase, carry on…).
- Weight: Represents the weight of the luggage in **kilograms**.

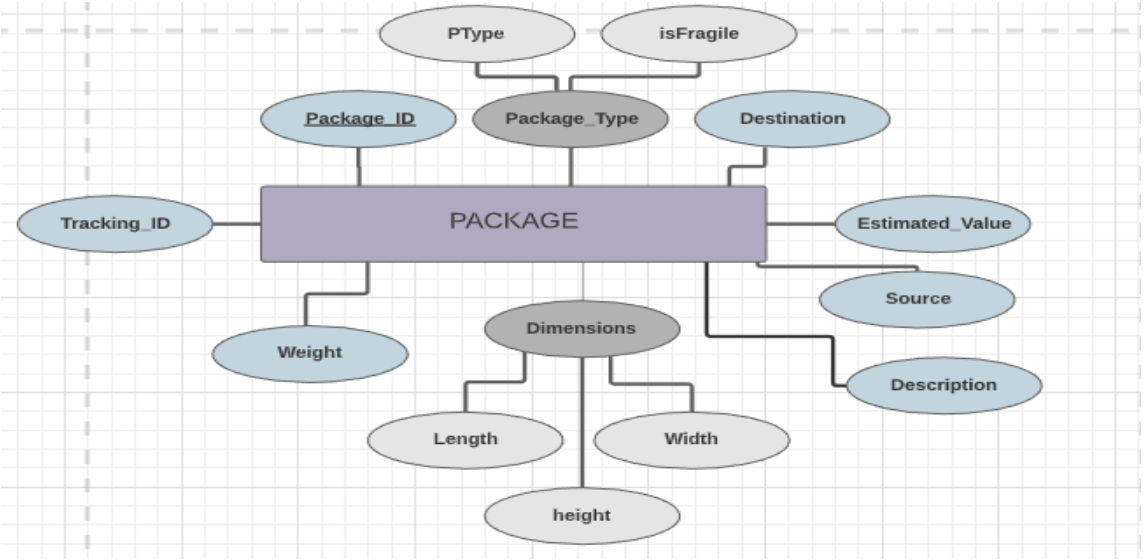
14. SHIPPING_COMPANY:



One of the most important sections in airport is the shipping section. For this reason, SHIPPING_COMPANY entity type was created to ship packages from the airport. Shipping companies are distinguished by their ICAO_Code. Additionally, a SHIPPING_COMPANY is attributed by its official business NAME along with its Country of legal origin.

- ICAO_Code: Consists of **3 characters CCC** that uniquely represent each SHIPPING_COMPANY.

15. PACKAGE:



Shipments are delivered by shipping companies. Thus, PACKAGE entity type was created to designate the shipments. Each PACKAGE is distinguished by its Package_ID. Additionally, a package is attributed by its Weight, Tracking_ID to keep track of the specific shipment's status, in addition to Source and Destination that respectively specify the exact address of which the shipment is shipped from and the exact address it should be delivered to, and the Estimated_Value that a customer will pay as shipping fees. Also, a PACKAGE is also defined by 2 composite attributes, Package_Type that can be broken down to 2 simple attributes, PType that specifies if the type of the shipment (home appliance, glass object, …) and another simple attribute isFragile to indicate if the object should be handled with extra care and caution. The other composite attribute Dimensions can be broken down into 3 simple attributes, Length, Width, and Height to specify the physical measures of the PACKAGE. Finally, a PACKAGE is also attributed by a simple attribute Description that includes extra helpful information about the PACKAGE.

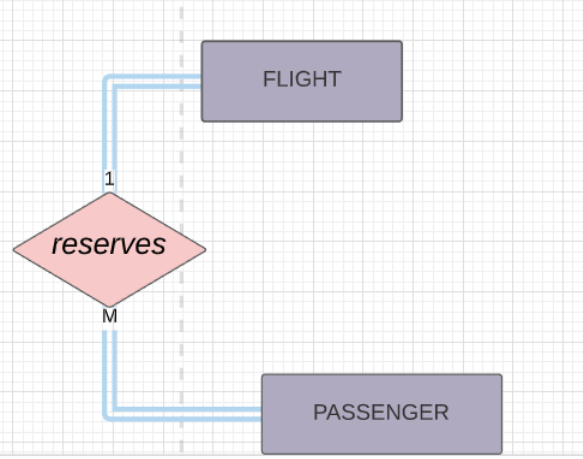
- Package_ID: is of the following format **CCCCCCC######**, the first two characters represent the source, the second two characters represent the destination, followed by the shipping code and the number of the shipment.
- Tracking_ID: is of the following format **CCCC######**, the first two characters represent the source, followed by 2 characters that represent

the current airport code of the shipment and the number of the shipment at the end.

- The dimensions, length, width, and height are all in meters.
- The weight of the package is grams.

VIII. Relationships:

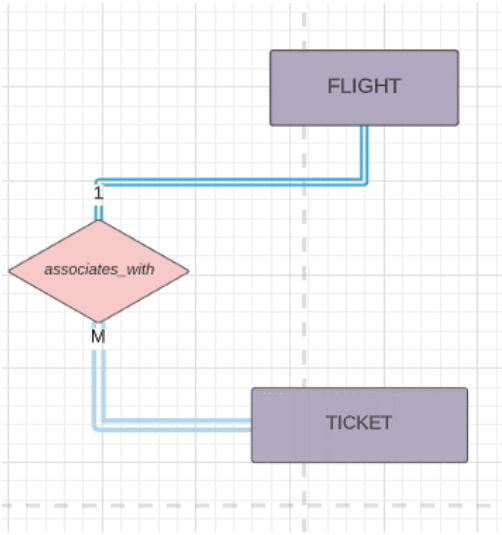
1-Reserves:



For a passenger to take a flight, he must be assigned to a flight. Thus, a reserves relationship must be created between the PASSENGER entity and the FLIGHT entity. The participation is total on both sides since each passenger must reserve a flight while each flight must contain passengers.

A PASSENGER may reserve one flight at a specific time whereas a FLIGHT may hold many PASSENGERs concurrently.

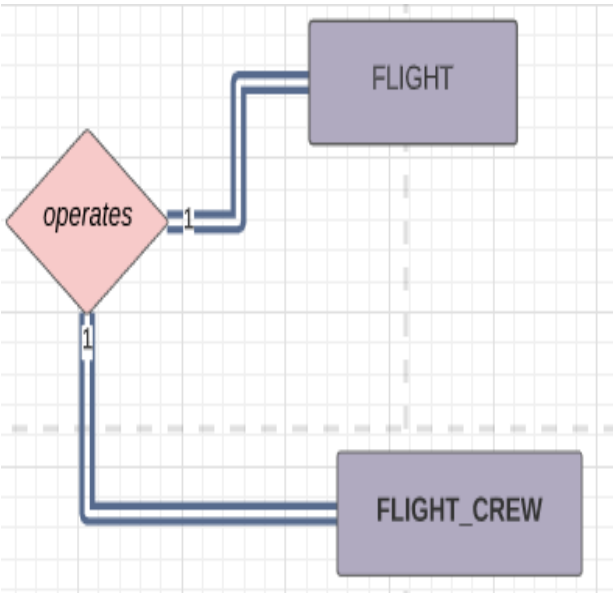
2-Associates_with:



Tickets represent the availability if seats in a given flight. Thus, an associates_with relationship must be created between the FLIGHT entity and the TICKET entity. The participation is total on both sides since each flight needs a ticket to represent it and each ticket must be associated with a flight.

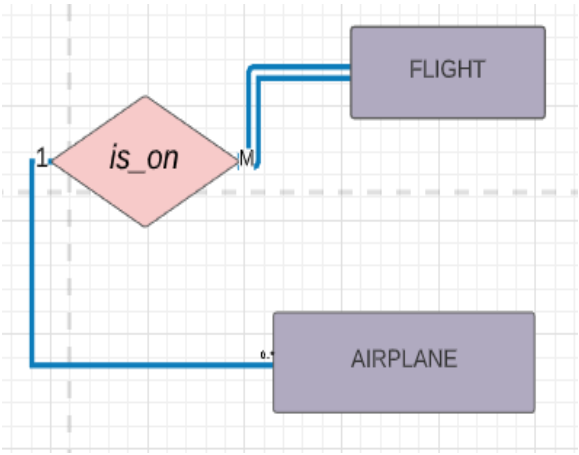
A TICKET associates_with one flight only and a FLIGHT may be represented by many TICKETs.

3-Operates:



A flight needs a flight crew to operate on it while in air. Thus, an operates relationship must be created between the FLIGHT entity and FLIGHT_CREW entity. The participation is total on both sides since each FLIGHT needs a FLIGHT_CREW and each FLIGHT_CREW in service must be associated to a FLIGHT. A FLIGHT_CREW operates on one FLIGHT at a time and a FLIGHT may be operated on by one FLIGHT_CREW only.

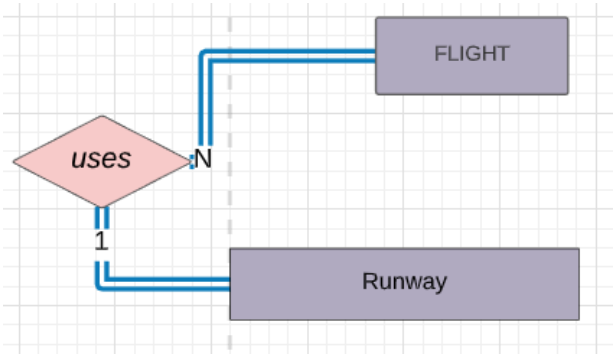
4- Is_on:



Each flight should be assigned to an airplane that carries the passengers. Thus an is_on relationship must be created between the FLIGHT entity and the AIRPLANE entity. The participation is total on the Flight side and partial on airplane side since no flight can take place without being assigned to an airplane, but not every airplane should be assigned to carry passengers (cargo planes…).

A FLIGHT is_on one AIRPLANE at a time, whereas an AIRPLANE can be assigned to many FLIGHTS if they are on different dates.

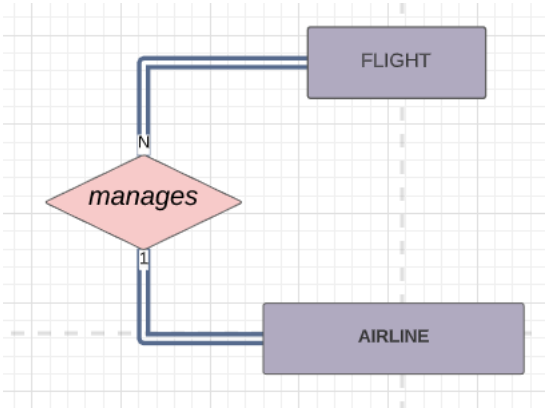
5-Uses:



A flight takes off from a runway in an airport and lands at a runway in an airport. Thus, a uses relationship must be created between FLIGHT entity and RUNWAY entity. The participation is total on both sides each FLIGHT needs a RUNWAY either to takeoff from or to land on, and every RUNWAY must be used by a FLIGHT.

A FLIGHT uses one RUNWAY at a time, whereas many FLIGHTs with different dates can use the same RUNWAY and a RUNWAY is assigned to one flight at a time.

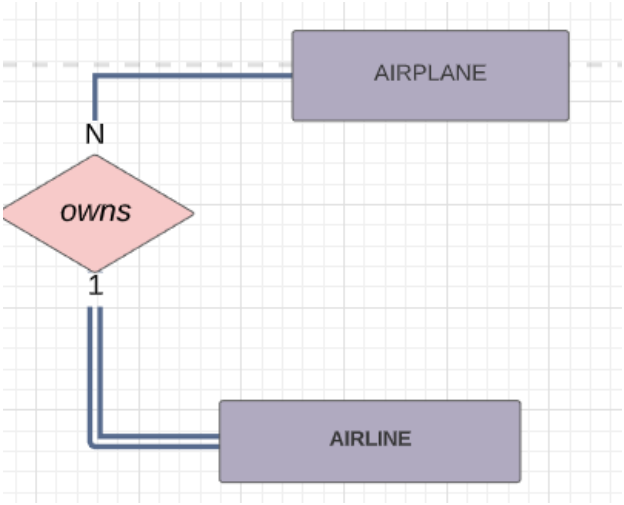
6-Manages:



A FLIGHT belongs to an AIRLINE. Thus, a manages relationship must be created between FLIGHT entity and AIRLINE entity. The participation is total on both sides since each FLIGHT must belong to an AIRLINE, and an AIRLINE must have/manage a FLIGHT.

An AIRLINE manages many FLIGHTs, whereas a FLIGHT may be managed by one AIRLINE.

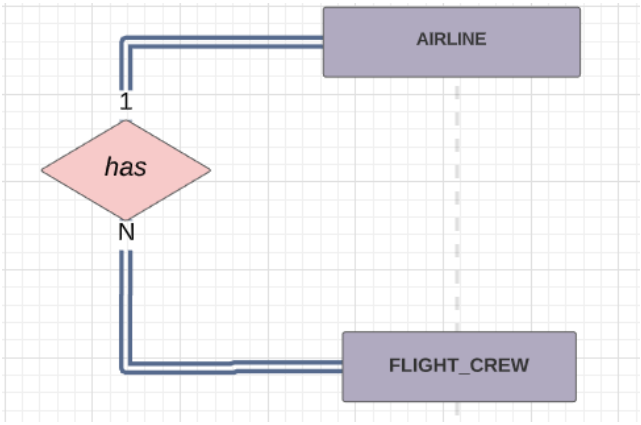
7-Owns:



For an AIRLINE to function, it needs AIRPLANEs. Thus, an owns relationship must be created between AIRPLANE entity and AIRLINE entity. The participation is partial on the AIRPLANE side and total on the AIRLINE side since each AIRLINE needs AIRPLANEs to manage flights, but not every plane belongs to an airline(such as airplanes sold to a private institution, person or governments…).

An AIRLINE owns one or more AIRPLANEs, whereas one airplane may or may not belong to an AIRLINE.

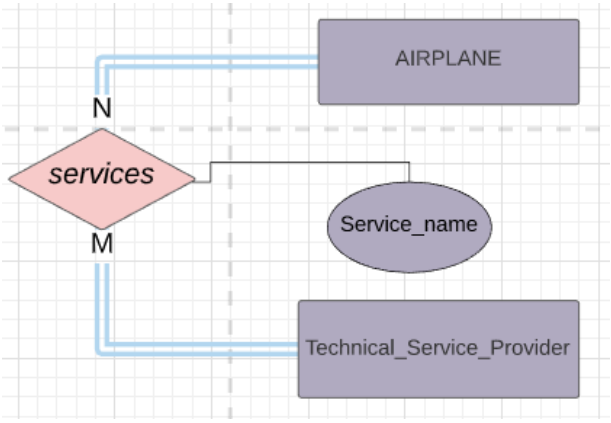
8-Has:



For an AIRLINE to conduct a flight, it must have a FLIGHT_CREW. Thus, a has relationship must be created between AIRLINE and FLIGHT_CREW. The participation is total on both sides since each AIRLINE must have a FLIGHT_CREW and a FLIGHT_CREW must belong to an airline.

An AIRLINE has one or more FLIGHT_CREWs, whereas a FLIGHT_CREW belongs to one AIRLINE.

9-Services:

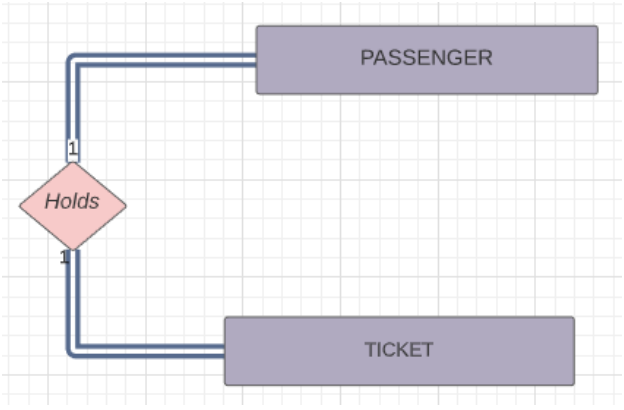


To ensure that an AIRPLANE is in a valid condition to be assigned to a flight, it should be regularly serviced. Thus, a services relationship must be created between AIRPLANE entity and TECHNICAL_SERVICE_PROVIDER entity. The participation is total on both sides since each AIRPLANE must be serviced and repaired and each TECHNICAL_SERVICE_PROVIDER working must repair and service an AIRPLANE.

A TECHNICAL_SERVICE_PROVIDER services one or more AIRPLANEs, whereas an AIRPLANE can be serviced by one or more TECHNICAL_SERVICE_PROVIDERs.

Service_Name: represents the service that the service provider provided for the airplane(refueling, repair...).

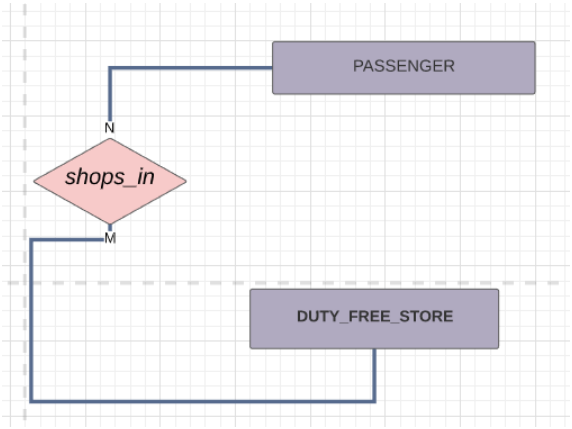
10-Holds:



A PASSENGER must have a TICKET to be assigned to a flight. Thus, a holds relationship must be created between PASSENGER entity and FLIGHT entity. The participation is total on both sides since each PASSENGER must hold a TICKET, and each TICKET is assigned to a PASSENGER.

A PASSENGER holds one unique TICKET, whereas the same TICKET cannot be assigned to 2 distinct PASSENGERS.

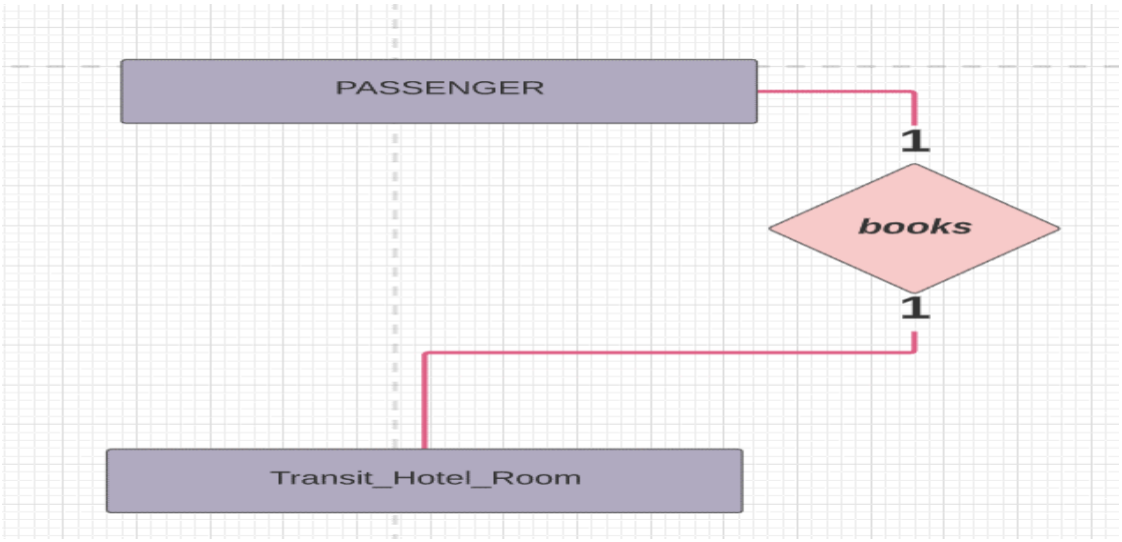
11-Shops_in:



A PASSENGER may buy goods from a DUTY_FREE_STORE in an airport. Thus, a shops_in relationship must be created between PASSENGER entity and DUTY_FREE_STORE entity. The participation is partial on both sides, since a PASSENGER may or may not shop in a DUTY_FREE_STORE.

One or many PASSENGERs shops_in one or many DUTY_FREE_STOREs, whereas one or many DUTY_FREE_STOREs may be visited by one or many PASSENGERS.

12-Has_A:

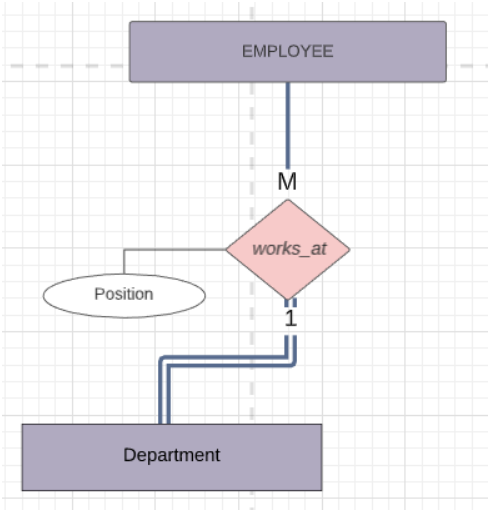


A PASSENGER who has a long transit can book a TRANSIT_HOTEL_ROOM inside the airport until his transit period is over. Thus, a books relationship must be created between PASSENGER entity and TRANSIT_HOTEL_ROOM.

The participation is partial on both sides since it is not obligatory for a PASSENGER to book a TRANSIT_HOTEL_ROOM, and a TRANSIT_HOTEL_ROOM may be booked or not.

A PASSENGER books a TRANSIT_HOTEL_ROOM, whereas a TRANSIT_HOTEL_ROOM may be booked by one specific PASSENGER only.

13-Works_at:

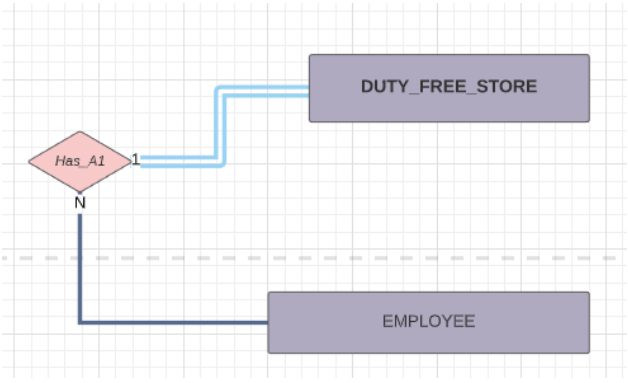


An EMPLOYEE in a particular airport may belong to a DEPARTMENT. Thus, a works_at relationship must be created between EMPLOYEE entity and DEPARTMENT entity. The participation is partial on the EMPLOYEE side and total on the DEPARTMENT side since an EMPLOYEE in an airport belongs to a DEPARTMENT if he's employed by the airport's management, else they belong to the employer who hired them, and ALL DEPARTMENTs must have EMPLOYEEs.

An EMPLOYEE works_at a DEPARTMENT, whereas a DEPARTMENT may have many EMPLOYEEs.

Position: represents the of an EMPLOYEE in their DEPARTMENT.

14-Has_A1:

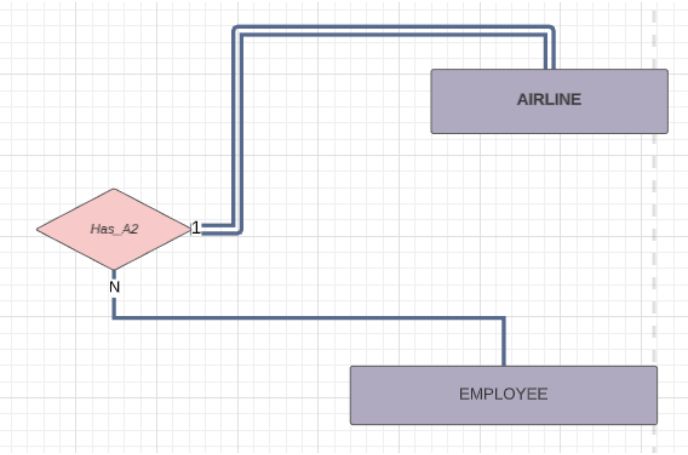


A DUTY_FREE_STORE needs EMPLOYEEs, like any other store.

Thus, a Has_A1 relationship must be created between DUTY_FREE_STORE entity and EMPLOYEE entity. Participation is total on DUTY_FREE_STORE side and partial on EMPLOYEE side since a store MUST have employees and it is not necessary for an EMPLOYEE to work at the DUTY_FREE_STORE.

A DUTY_FREE_STORE Has_A1 one or EMPLOYEEs, whereas an EMPLOYEE can work for one DUTY_FREE_STORE.

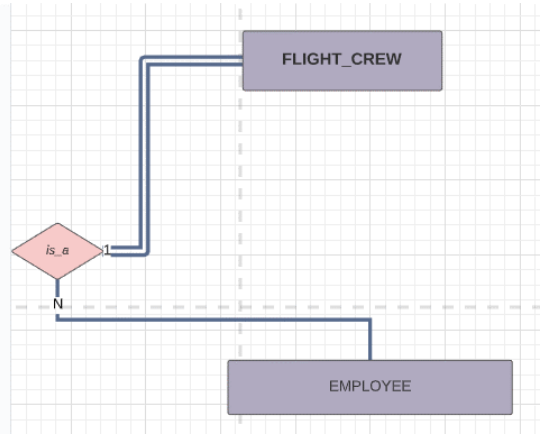
15-Has_A2:



An AIRLINE needs EMPLOYEEs other than FLIGHT CREW members. Thus, a Has_A2 relationship must be created between AIRLINE entity and EMPLOYEE entity . Participation is total on AIRLINE side and partial on EMPLOYEE side since an AIRLINE MUST have employees and it is not necessary for an EMPLOYEE to work at an AIRLINE.

An AIRLINE Has_A2 one or EMPLOYEEs, whereas an EMPLOYEE can work for one AIRLINE.

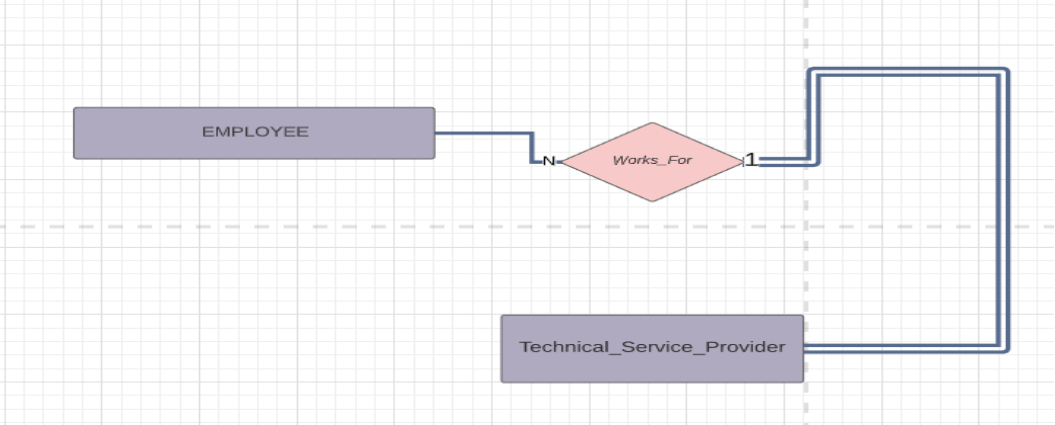
16-Is_a:



A FLIGHT_CREW needs members. Thus, an is_a relationship must be created between EMPLOYEE entity and FLIGHT_CREW entity. The participation is total on FLIGHT_CREW side and partial on EMPLOYEE side since an EMPLOYEE may not necessarily be a FLIGHT_CREW MEMBER but it is obligatory for each FLIGHT_CREW to have EMPLOYEEs.

An EMPLOYEE is_a FLIGHT_CREW member, whereas a FLIGHT_CREW may have many EMPLOYEEs.

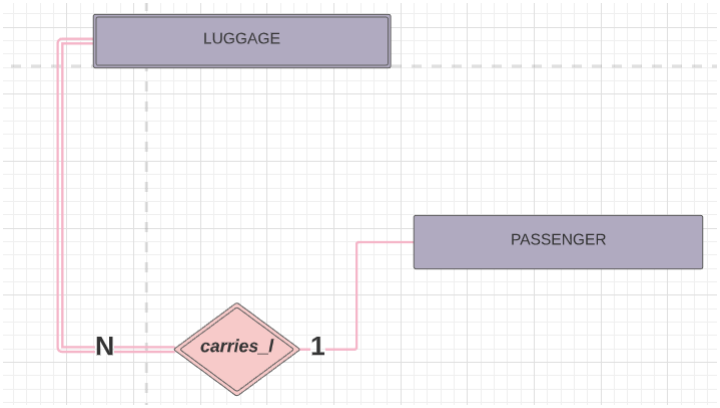
17-Has_A3:



A TECHNICAL_SERVICE_PROVIDER needs EMPLOYEEs. Thus, a Works_For relationship must be created between EMPLOYEE entity and TECHNICAL_SERVICE_PROVIDER entity. The participation is total on TECHNICAL_SERVICE_PROVIDER side and partial on EMPLOYEE side since an EMPLOYEE may not necessarily work for a TECHNICAL_SERVICE_PROVIDER but it is obligatory for each TECHNICAL_SERVICE_PROVIDER to have EMPLOYEEs.

An EMPLOYEE Works_For TECHNICAL_SERVICE_PROVIDER, whereas a TECHNICAL_SERVICE_PROVIDER may have many EMPLOYEEs.

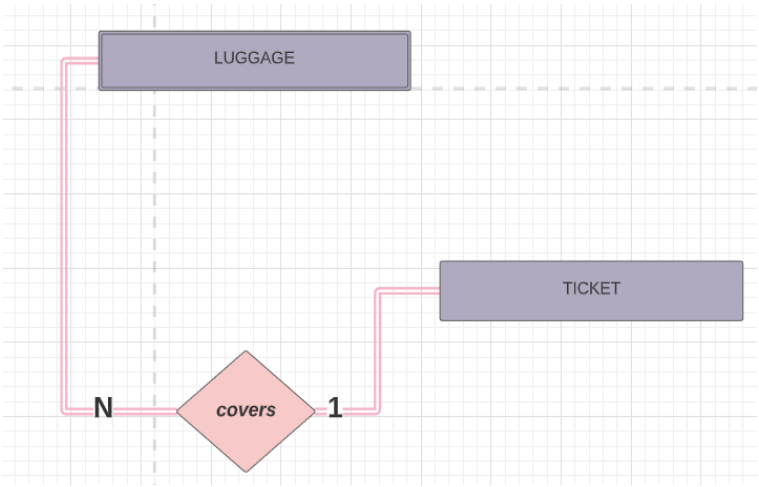
18- Carries_l:



Each passenger can carry luggage while traveling. Thus, a carries_l identifying relationship must be crated between the owner entity PASSENGER and LUGGAGE weak entity. The participation is partial on the passenger side, and total on the luggage side, since a passenger can travel without carrying any type of luggage, and every luggage should belong to a passenger.

Each PASSENGER carries_l one or many LUGGAGE and each LUGGAGE belongs to exactly one passenger.

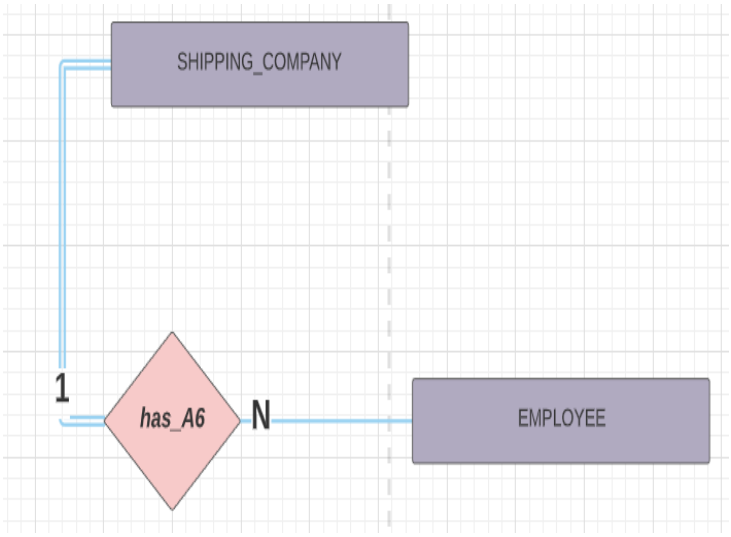
19- covers:



It is important to keep track of each passenger's luggage. Thus, a covers relationship must be created between TICKET entity and LUGGAGE weak entity. Participation is total on both sides since all luggage should be covered by tickets, and all tickets should cover luggage.

Each TICKET covers many LUGGAGE, and every LUGGAGE should belong to one TICKET exactly.

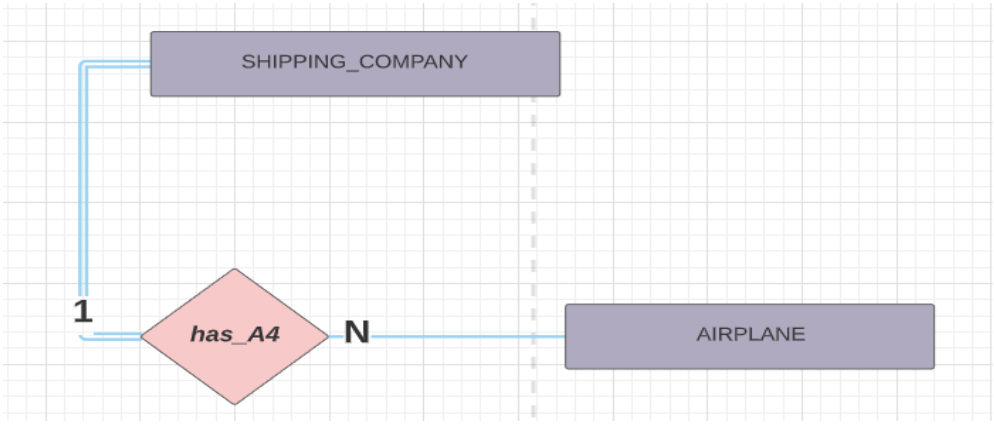
19-Has_A6:



Same as every other company, a SHIPPING_COMPANY needs EMPLOYEEs. Thus, a has_A6 relationship must be created between SHIPPING COMPANY entity and EMPLOYEE entity . Participation is total on SHIPPING_COMPANY side and partial on EMPLOYEE side since a SHIPPING_COMPANY MUST have employees and it is not necessary for every EMPLOYEE to work at a SHIPPING_COMPANY.

A SHIPPING_COMPANY has_A6 one or EMPLOYEEs, whereas an EMPLOYEE can work for one SHIPPING_COMPANY.

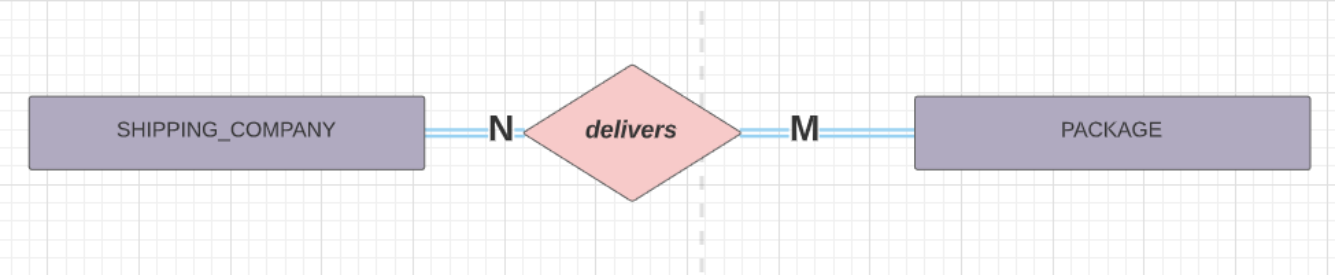
20-Has_A4:



For a SHIPPING_COMPANY to function, it needs AIRPLANEs that deliver its shipments. Thus, a has_A4 relationship must be created between SHIPPING_COMPANY entity and AIRPLANE entity. The participation is partial on the AIRPLANE side and total on the SHIPPING_COMPANY side since each SHIPPING_COMPANY needs AIRPLANEs to deliver shipments, but not every airplane belongs to a SHIPPING_COMPANY.

A SHIPPING_COMPANY has_A4 one or more AIRPLANEs, whereas one AIRPLANE belongs to exactly one SHIPPING_COMPANY.

21-delivers:



The sole purpose of a SHIPPING_COMPANY is to deliver shipments. Thus, a delivers relationship must be created between SHIPPING_COMPANY ENTITY and PACKAGE entity, the participation is total on both sides since each SHIPPING_COMPANY delivers a PACKAGE, and each PACKAGE is delivered by a SHIPPING_COMPANY.

A SHIPPING_COMPANY delivers many packages and many packages are delivered by SHIPPING_COMPANY.

IX. ER to Relational Mapping Algorithm:

After completing the ER diagram for the database, the entities, attributes and the relationships need to be mapped into a lower level relational schema following the steps illustrated in class. The steps go as follow:

step 1: All non-weak entities are mapped into relations in this step. Composite attributes are broken down into their simple components. If we have multiple candidate keys, we need to assign only one to be the primary key.

Step 2: all weak entities are mapped in this step, similar to the previous step we just need to include the primary key of the owning entity in the relation of the weak entity.

Step 3: All binary 1:1 relationship type is mapped into relation schemas. Specifically, in this step, we apply the foreign key approach. (explained later)

Step 4: All binary 1: N relationship types are mapped into relation schemas. again, we apply the foreign key approach. (explained later)

Step 5: All binary M: N relationship types are mapped into relation schemas. New relation is created to represent this relationship. This new relation has both the primary keys of the participating entities as foreign keys.

Step 6: All multivalued attributes are mapped last thing. We create a new relation for them including their sub-attributes and the primary key of the relation on which they exist.

Step 7: Not Applicable in our case.

Step01: Mapping the regular entity types:

The first step includes mapping regular entities (those with primary keys) into relations. Each entity is mirrored via a table relation that contains all its simple attributes and a single chosen primary key that is underlined. Most entities in the airport database are regular entities and they include: PASSANGER, TICKET, AIRLINE, DEPARTEMNT, FLIGHT, AIRPLANE, FLIGHT-CREW, RUNWAY, TECHNICAL-SERVICE-PROVIDER, DUTY-FREE-STORE and EMPLOYEE. We have one weak entity which is TRANSIT-HOTEL-ROOM.

1. PASSANGER

Passport_Number	**Nationality**	**FName**	**LName**	**MName**	**Date_of_Birth**	**Geneder**

Visa_Type	**Visa_Expiration_Date**	**Visa_Number**	**Phone_Number**	**Email**

The **PASSENGER** entity has both simple and composite attributes. In the relation shown above, the composite attributes were broken down to their simple components. The primary key of this relation is **Passport_Number** and the other attributes include the following:

- FName
- MName
- LName

- Date_of_Birth
- Gender
- Nationality
- Phone_Number
- Email
- Visa_Number
- Visa_Type
- Visa_Expiration_Date

2. TICKET

Ticket_Number	**Seat_Number**	**Price**	**Class**	**Boarding-Group**	**Meal_ Cuisine**	**Itinerary**
	Special_Accomodation	**Number_of_bags**				

The **TICKET** entity has both simple and composite attributes. In this relation all the attributes are shown as simple with the primary key **Ticket_Number** underlined. Boarding-grp is a derived attribute that can be obtained from both price and class attributes. Other attributes of this entity include:

- Seat_Number
- Class
- Boarding_Group
- Itinerary
- Meal_Cuisine
- Special_Accomidation
- Price
- Number_of_bags

3. AIRLINE

ICAO-code	Name	country	Num_Of_Aircrafts

The **AIRLINE** entity has only simple attributes. The **ICAO-code** is the primary key in this relation. The **NumOfAircrafts** is a derived attribute in which its value can be obtained from the ICAO-code. Other attributes of this relation include:

- Name
- Country
- Num_of_Aircrafts

4. DEPARTEMNT

Department-ID	Department-name	No_of_Employees	Description

The **DEPARTMENT** entity has two simple attributes**;** **Department-ID** is the primary key. The other attribute includes:

- Department_Name
- No_of_Employees
- Description

5. FLIGHT

Flight_Number	Elapsed-time	Origin	Destination	Departure_Time	Arrival_Time

Gate_Number	Status	Ground_handling_Name

The **FLIGHT** entity has both complex and simple attributes. **Flight_Number** is the primary key attribute of this relation, it also has a derived attribute called Elapsed-time. Other attributes include:

- Gate_Number
- Origin
- Destination
- Departure_Time
- Arrival_Time
- Elapsed_Time
- Ground_Handling_Name
- Status

6. TRANSIT-HOTEL-ROOM

Room_Number	Availability	Room-type

The **TRANSIT-HOTEL-ROOM** entity has three simple attributes from which we chose **Room_Number** to be the primary key in the relation. Other attributes include:

- Room_Type
- Availability
- Passenger

7. AIRPLANE

Airplane_ID	Airplane_Type	Passenger_Capacity	Weight	Year_make	Fuel_Tank_Capacity

MTO_Weight	Model

The **AIRPLANE** entity has a primary key attribute called **Airplane_ID**, a simple attribute and one composite attribute that is broken down into its simple attributes. They all include:

- Airplane_Type
- Make
- Model
- Year_Make
- MTO_Weight
- Passenger_Capacity
- Fuel_Tank_Capacity

8. FLIGHT_CREW

License_No	Position	Rating	Status

The **FLIGHT_CREW** entity has three simple attribute and a key attribute called **License_No.** The other attributes include:

- Position
- Rating
- Statu

9. EMPLOYEE

Employee_ID	Gender	Phone_Number	Date_of_Birth	Salary	FName	Lname	Mname	Email

The **EMPLOYEE** entity has six simple attributes from which **Employee_ID** is a primary key attribute. The entity also has one compound attribute called Name that is broken down to its simple components. Other attributes of this entity are:

- FName
- MName
- LName
- Date_of_Birth
- Gender
- Phone_Number
- Email
- Salary

10.Runway

Runway_ID	Runway_Length	Is_Available

The **Runway** entity has three simple attributes

- Runway_ID
- Runway_Length
- isAvailable

11. Technical Service Provider

SP_Name	City	Country

The **technical Service provider** entity has three simple attributes

- SP_Name: chosen to be the primary key

  The composite attribute location is divided into two simple attributes when mapping

- Country

12. Duty Free Store

Store_Name	Store_Type	Operator

The **Duty-Free Store** entity has also three simple attributes:

- Store Name: which is a unique identifier in this relation
- Operator
- Store_Type

13. Shipping Company

ICAO_Code	Name	Country

The Shipping Company such as DHL, Aramex… have ICAO_Code as a primary entity and simple attributes:

- Name
- Country

14. PACKAGE

Package_ID	Tracking_ID	Package_Type	isFragile	Destination	Source	Length	Width	Height

Description	Weight	Estimated_Value

The Entity PACKAGE has both simple and composite attributes. Package type and dimensions are composite and in the relation all the attributes are shown as simple ones.

- Package_ID: primary Key
- Tracking_ID
- Package_Type
- isFragile
- Length
- Width
- Height
- Weight
- Source
- Destination
- Estimated_Value
- Description

Step02: Mapping weak entity types:

For the weak entity types, only simple attributes are used in mapping the relation. Additionally, weak entity relation has a foreign key attribute which is the primary key of the owning entity. Thus it ca be concluded that the combination of the partial key and the foreign key represent the relation's primary key.

1. **Luggage**

   Luggage is the only weak entity in our ER-diagram is it connected to the owning entities: Passenger.

Luggage_Number	Passport_Number	Luggage_Type	weight	color

The weak entity Luggage has only simple attributes, thus they are included directly in the relation. Plus, we add the primary key is **Passport_Number** for of the owning relation **(Carries)** passport as a foreign key.

- Passport_Number
- Luggage_Number
- Luggage_Type
- Color
- Weight

Step03: Mapping Binary 1:1 Relationship Types:

This step includes mapping the binary 1:1 relationship. To do so, we are applying the foreign key approach in which we select the entity with the total participation side on the relation and we call it S. The other participating relation is names T. After that we simply include the primary key of T as a foreign key in S to represent the 1:1 relationship.

1. Operates

License_Num	**Position**	**Rating**	**Status**	**FFlight_Num**

The "***Operates***" is a binary 1:1 relationship that links the entity **FLIGHT to FLIGHT-CREW.** Each flight must have a flight crew who takes care of passengers during the flight. Each flight

must have only one flight crew, but the same flight crew can operate different flights. The participation on both sides in total so we get to choose which entity to name S and which one is T. We chose **FLIGHT-CREW** to be S and therefore we inserted the primary key of **FLIGHT** (FFlight_Num) as a foreign key in **FLIGHT-CREW**. The foreign key was renamed as FFlight_Num.

2. *Holds*

Ticket_Number	**Seat_Number**	**Price**	**Class**	**Boarding_Group**	**Meal**	**weight**
	Number_of_bags	**Itenerary**	**PPassport_Num**			

The "***Holds***" relationship links the entity **TICKET** to **PASSENGER**. Each passenger needs a valid ticket that enable them to board on the airplane. Each passenger can buy several tickets, but no tickets can be assigned for two different passengers. The participation is total on both sides of the relationship. We decided to consider **TICKET** to be the S entity in which we insert the primary key of **PASSENGER** (PassportNum) as a foreign key in the **TICKET** entity. The foreign key was renamed to as PPassport_Num.

3. *TRANSIT_HOTEL_ROOM (Books)*

Room_Number	**Room_Type**	**Availability**	**PPASSPORT_NUMBER**

The "***Books***" relationship links the entity **PASSENGER** to **TRANSIT-HOTEL-ROOM**. Passengers sometimes find themselves having very long transits for different reasons like delay. In these cases, passengers are offered rooms in the airport hotel in which they can rest while waiting for their flight. One passenger can book several rooms, but one room cannot be booked by several passengers at the same time. The participation is partial on both sides of the relationship we consider **TRANSIT_HOTEL_ROOM** to be the entity in which we insert the primary key of **PASSENGER** (Passport_Number) as a foreign key in the **TRANSIT_HOTEL_ROOM** entity. The foreign key was renamed to as **PPASSPORT_NUMBER.**

Step04: Mapping Binary 1: N Relationship Types:

In this stage, binary one to many relationships are mapped. The entity that has the N on its side is named S while the other participating entity is named T, we then insert the primary key of T as a foreign key in S to represent the 1: N relationship.

1. *FLIGHT (Is-on)*

Flight_Number	Elapsed-time	Origin	Destination	Departure_Time	Arrival_Time
Gate_Number	Status	Ground_handling_Name	AAirplane-ID		

The "***Is-on***" relationship links the entity **AIRPLANE** to **FLIGHT**. The flight as a non-concrete concept takes place within an actual airplane. Airplanes can host several flights from and to different destinations yet, a flight cannot be hosted by several airplanes since it must be associated with one airplane. Since the N is on the **FLIGHT**'s side, we consider **FLIGHT** to be the S entity in which we insert the primary key of **AIRPLANE** (Airplane-ID) as a foreign key in the **FLIGHT** entity. The foreign key was renamed to as **AAirplane-ID**.

2. *FLIGHT (Manages)*

Flight_Number	Elapsed-time	Origin	Destination	Departure_Time	Arrival_Time
Gate_Number	Status	Ground_handling_Name	AICAO_code		

The "***Manages***" relationship links the entity **AIRLINE** to **FLIGHT**. In airports, airlines companies are the one deciding on their flights schedule, crew and other stuff. Each flight is managed by its respective airline to which it belongs, but the same airline can manage multiple flights. Since the N is on the **FLIGHT**'s side, we consider **FLIGHT** to be the S entity in which we insert the primary key of **AIRLINE** (ICAO-code) as a foreign key in the **FLIGHT** entity. The foreign key was renamed to as AICAO-code.

3. FLIGHT(Uses)

Flight_Number	Elapsed-time	Origin	Destination	Departure_Time	Arrival_Time
Gate_Number	Status	Ground_handling_Name	RRunway_ID		

The "***uses***" relationship is a 1: N relationship that links both **RUNWAY** and **FLIGHT.** In the airport, a runway is an area that is prepared for the landing and takeoff of aircraft, to dismiss any conflict that would happen the runway is linked to the flight that (is_on) airplane in this way the runway is able to gather information about the arrival and departure time of any flight from the relationship associates with. Each runway has an **ID** (primary key) and since the N cardinality is on the Flight side, we consider Flight the S entity in which we insert the primary key of Runway as a foreign key in Flight entity. The Foreign key was renamed to Runway_ID.

4. EMPLOYEE(has_A1)

EEmployee_ID	SStore_Name

The "***has_A1***" is a 1: N relationship that links the entity **DUTY_FREE_STORE** to **EMPLOYEE.** The duty-free stores at the airport are run by multiple employees. Every duty-free has multiple employees running it, but a single employee works at only one duty-free. Note that in this relationship we are using the cross-referencing method in which both primary keys of the participating relations (Employee_ID and Store_Name) are inserted as foreign keys in a new relation.

5. EMPLOYEE(has_A2)

EEmployee_ID	AICAO_code

The "***has_A2***" is a 1: N relationship that links the entity **AIRLINE** to **EMPLOYEE.** Each airline in the airport has multiple employees working in it. Several employees can work for the same airline, but one employee can work exclusively at one airline. Note that in this relationship we are using the cross-

referencing method in which both primary keys of the participating relations (Employee_ID and ICAO_code) are inserted as foreign keys in a new relation.

6. **FLIGHT_CREW(has_A3)**

License_NO	Position	Rating	Status	AICAO_code

The "***has_A3***" is a 1: N relationship that links **FLIGHT_CREW** and **AIRLINE**. Flight crew contains only simple attributes. In the airport, Airlines own airplanes (own is considered to be either rented or bought), in our database we keep record of the flight crew that the airline hires such as his rating, status and position. Since Flight_Crew is in the N side we add the primary key of the other entity(Airline) as a foreign key. The foreign key was renamed to AICAO_code.

7. **AIRPLANE (has_A4)**

Airplane_ID	Airplane_Type	Passenger_Capacity	Weight	Year_make	Fuel_Tank_Capacity

MTO_Weight	Model	No_of_Aircrafts	**SICAO_code**

The "***has_A4***" is a 1: N relationship links the entity **AIRPLANE** and **SHIPPING_COMPANY**. Shipping Companies such as aramex, DHL … has (own: either bought or rent). The Airplane is in the N side but Model is a composite attribute so in the relation we include all its simple attributes and we add the primary key of the **SHIPPING_COMPANY** (ICA0_Code) as a

foreign key in the "*has_A4*" relation and we add the attribute in the relation "*has_A4*" which is derived (**No_of_Aircrafts**). The foreign key is renamed as **SICAO_code.**

8. EMPLOYEE (has_A5)

Employee_ID	SSPNAME

The "*has_A5*" is a 1: N relationship links the entity **EMPLOYEE** to **TECHNICAL_SERVICE_PROVIDER.** In the airport, technical service providers take on the responsibility of preparing airplanes and making sure all the technicalities are intact. Each service provider has multiple employees, while multiple employees work for the same service provider. Note that in this relationship we are using the cross-referencing method in which both primary keys of the participating relations (Employee_ID and SPNAME) are inserted as foreign keys in a new relation.

9. *EMPLOYEE (has_A6)*

Employee_ID	SICAO_code

The "*has_A6*" is a 1: N relationship links the entity **EMPLOYEE to SHIPPING_COMPANY.** A lot of international shipment takes place in the airport, shipment companies make sure to deliver the packages. Each shipment company has

multiple employees, while multiple employees work for the same shipping company. Note that in this relationship we are using the cross-referencing method in which both primary keys of the participating relations (Employee_ID and SPNAME) are inserted as foreign keys in a new relation.

10. *EMPLOYEE (is_a)*

Employee_ID	Gender	Phone_Number	Date_of_Birth	Salary	FName	Lname	Mname	Email
			LLisence_No					

The "***is_a***" is a 1: N relationship links the entity **EMPLOYEE** to **FLIGHT_CREW.** An airplane cannot take off without a cabin crew that monitor things while in the air. Each cabin or flight crew has several employees, but several employees work or are part of one flight crew. Since the N is on the **EMPLOYEE**'s side, we consider **EMPLOYEE** to be the S entity in which we insert the primary key of **FLIGHT_CREW** (LisenceNum) as a foreign key in the **EMPLOYEE** entity. The foreign key was renamed to as **LLisence_No**.

11.*AIRPLANE (owns)*

Airplane_ID	Airplane_Type	Passenger_Capacity	Weight	Year_make	Fuel_Tank_Capacity
		MTO_Weight	Model	**AICAO_Code**	

The "***owns***" is a 1: N relationship links the entity **AIRPLANE** to **AIRLINE.** A set of airplanes are usually owned by a given airline. Each airline has several airplanes and multiple airplanes are owned by a single airline. Since the N is on the **AIRPLANE**'s side, we consider **AIRPLANE** to be the S entity in which we insert the primary key of **AIRLINE** (ICAO_Code) as a foreign key in the **EMPLOYEE** entity. The foreign key was renamed to as **AICAO_Code**.

12.*LUGGAGE (covers)*

Luggage_Number	weight	type	Color	TTicket_Number

The "***covers***" is a 1: N relationship links the entity **LUGGAGE** to **TICKET.** For the passenger, every ticket is associated with a luggage. One ticket can have multiple pieces of luggage, but those several pieces belongs to the one ticket. Since the N is on the **LUGGAGE**'s side, we consider **LUGGAGE** to be the S entity in which we insert the primary key of **TICKET** (TicketNum) as a foreign key in the **LUGGAGE** entity. The foreign key was renamed to as **TTicket_Number**.

13.*LUGGAGE (carries)*

LuggNum	weight	type	Color	PPassport_Number

The "***carries***" is a 1: N relationship links the entity **LUGGAGE to PASSENGER.** Each passenger has his/her own suitcases that are names luggage in this database. One passenger supports multiple suitcases, and multiple suitcases can belong to the same passenger. Since the N is on the **LUGGAGE**'s side, we consider **LUGGAGE** to be the S entity in which

we insert the primary key of **PASSENGER** (PassportNum) as a foreign key in the **LUGGAGE** entity. The foreign key was renamed to as **PPassport_Number**.

14. *DEPARTMENT (works_at)*

Department-ID	**Department-name**	**No_of_Employees**	**Description**	**Manager_ID**

The "***works_at***" is a 1: N relationship links the entity **DEPARTMENT to EMPLOYEE.** The airport is divided into several departments, each of those are run my multiple employees. One department supports multiple employees, and multiple employees can work at the same department. Since the N is on the **DEPARTMENT**'s side, we consider **EMPLOYEE** to be the S entity in which we insert the primary key of **EMPLOYEE** (EmployeeID) as a foreign key in the **LUGGAGE** entity. The foreign key was renamed to as **Manager_ID**.

15. *EMPLOYEE(supervises)*

Employee_ID	Gender	Phone_Number	Date_of_Birth	Salary	FName	Lname	Mname	Email	Supervisor_ID

The "***supervises***" is a 1: N self-referencing relationship in the entity **EMPLOYEE.** Employees are divided into simple employees and supervisors who manage the employee and make sure everything is running according to a plan. Since this is a self-referencing relationship, we need to add the primary key of EMPLOYEE (employee_ID) as a foreign key in the same entity. The foreign key was renamed to as **Supervisor_ID**.

16. *PACKAGE (delivers)*

Package_ID	Tracking_ID	Package_Type	isFragile	Destination	Source	Length	Width	Height

Description	Weight	Estimated_Value	**SICAO_code**

The "***works_at***" is a 1: N relationship links the entity **PACKAGE** to **SHIPPING_COMPANY**. Packages are shipped to different parts of the world through a shipment company within the airport. One shipping company can deliver multiple packages, but a single package can be shipped by a single shipping company. Since the N is on the **PACKAGE**'s side, we consider **SHIPPING_COMPANY** to be the S entity in which we insert the primary key of **SHIPPING_COMPANY** (ICAO_code) as a foreign key in the **PACKAGE** entity. The foreign key was renamed to as **SICAO_code.**

Step05: Mapping Binary M: N Relationship Types:

In this step we are mapping many to many binary relationships. For each N:M relationship we must create a new relation that includes the primary keys of all the participating entities as foreign keys. The totality of all the primary keys will be considered as the primary key for the new established relation. If the relation has any attributes, those must as well be added.

1. Shops-in

PPassport_Number	*SStore_Name*

The "***Shops-in***" is a N:M relationship that links the entity **PASSENGER to DUTY_FREE_STORE.** Passengers might shop in different duty-free stores during their travels, and each of these-duty free shops can accommodate several passengers. The relationship is modeled via the creation of a new relation that has the primary keys of both **PASSENGER and DUTY_FREE_STORE** as foreign keys that were renamed to PPassport_Number and SStore_Name respectively. The combination of these two foreign keys makes the primary key of the "***Shops-in***" relation.

2. *Provides_Services*

AAirplane_ID	SSPName	Service_name

The "***Provides_Services***" is a N:M relationship that links the entity **AIRPLANE to TECHNICAL_SERVICE_PROVIDER.** Airplanes need to be technically checked and prepared before taking off. Technical service provider work on many airplanes making sure all their engineering is working, and the same plane can be checked by different technicians depending on the need. The relationship is modeled via the creation of a new relation that has the primary keys of both **AIRPLANE and TECHNICAL_SERVICE_PROVIDER** as foreign keys that were renamed to **AAirplane_ID** and **SSPName** respectively.

X. Normalization Up to The BCNF Normal Form

After establishing all relationships, we should normalize our relational schema to reduce the replication of data, avoid data anomalies (insertion, deletion, deletion and updates), ensure referential integrity, and simplify management of the data.

A. FIRST NORMAL FORM:

Disallows composite attributes, multivalued attributes, and nested relations; attributes whose values for an individual tuple are non-atomic.

1. Single atomic values attributes.

2. Domain of an attribute must only include atomic values and the value of an attribute in a tuple

must be a single value from the domain of that attribute.

3. Disallows having a set of values as an attribute value for a single tuple.

B. SECOND NORMAL FORM:

- The Second normal form is based on the concept of full functional dependency.
- Functional Dependencies: a restriction between two sets of attributes in the database. The values of the A component of the tuple in relation R depend on, or are determined by the values of the B component. We Claim A is functionally dependent on B.
- Prime Attribute: Attribute that is a member of the candidate key in R relation. Anon prime attribute is not a member of a candidate key.
- Full functional dependency : a Functional Dependency Y → Z where removal of any attribute from Y means the Functional Dependency does not hold any more.
- Partial Dependency: A functional dependency X →Z such removal of any attribute A Form X means that the dependency still holds.

A relation R is in the second normal form if:

- Every non-prime attribute in the relation R is fully functionally dependent on every key of R.
- Every non-prime attribute in R is not partially dependent on any key in R.

C. THIRD NORMAL FORM:

- A Relation R should not have a non-key attribute functionally determined by another non-key attribute (or by a set of non-key attributes). There shouldn't exist any transitive dependency of a Non-key attribute on the primary key.
- A functional dependency A→ B in a relation schema R is a transitive dependency if there exists a set of attributes C in R that is neither a candidate key nor a subset of any key of R, and both A→ C and C→ B hold.

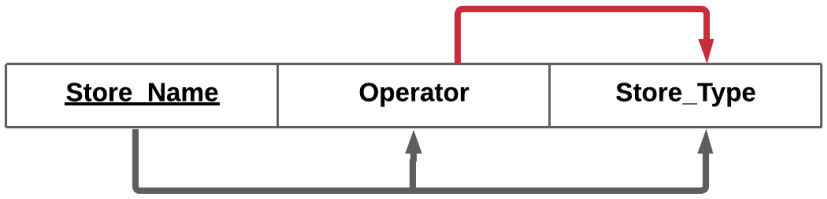
The third normal form or 3NF disallows partial and transitive dependencies on the primary key.

D. BOYCE-CODD NORMAL FORM:

For a relation R to satisfy the Boyce-Cod Normal Form, it should satisfy two conditions:

- It should be in 3NF.
- For any dependency X → Y, X should be a super key. (for a dependency A → B, if B is a prime attribute, A cannot be a non-prime attribute).

1. DUTY_FREE_STORE

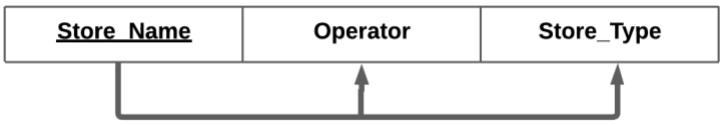


A. **The DUTY_FREE_STORE** relation satisfies all conditions of the 1NF it has neither multivalued attributes nor composite attributes. All attributes are single and atomic.

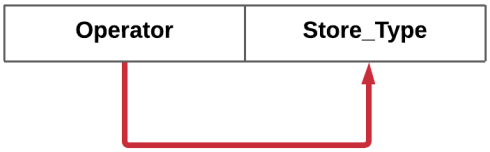
B. **The DUTY_FREE_STORE** relation satisfies all conditions of the 2NF because every non-prime attribute is fully functional dependent on the PK "Store_Name"

C. **The DUTY_FREE_STORE** relation does not satisfy all conditions of the 3NF because the functional dependency represented by Operator →Store_Type is a functional dependency where neither Operator is a super key nor Store_Type is a prime attribute. Thus, further decomposition is needed.

DUTY_FREE_STORE1

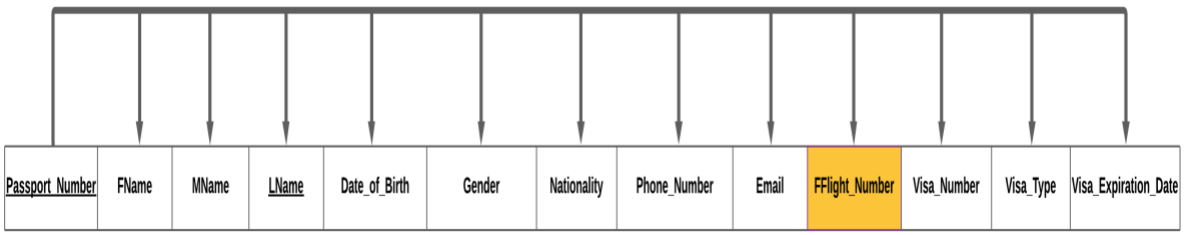


DUTY_FREE_STORE2



The DUTY_FREE_STORE relation schema satisfies all rules of the BOYCE CODD NORMAL FORM (BCNF) because there exists no functional dependency X →A where X is not a super key or A is a prime attribute and X not a super key.

2. PASSENGER



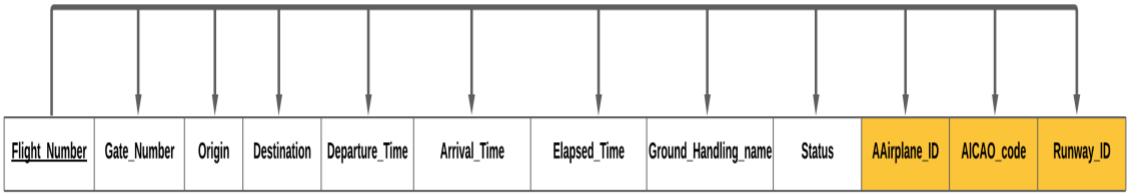
A. **The PASSENGER** relation schema satisfies all conditions of the 1NF it has neither multivalued attributes nor composite attributes. All attributes are single and atomic.

B. **The PASSENGER** relation schema satisfies all conditions of the 2NF because every non-prime attribute is fully functional dependent on the PK "Passport_Number"

C. **The PASSENGER** relation schema satisfies all conditions of the 3NF because it satisfies the 2NF and there are no non-prime attributes that are transitively dependent on the primary key "Passport_Number".

D. **The PASSENGER** relation schema satisfies all rules of the (BCNF) because there exists no functional dependency X →A where X is not a super key or A is a prime attribute and X not a super key.

3. FLIGHT



A. **The FLIGHT** relation schema satisfies all conditions of the 1NF it has neither multivalued attributes nor composite attributes. All attributes are single and atomic.

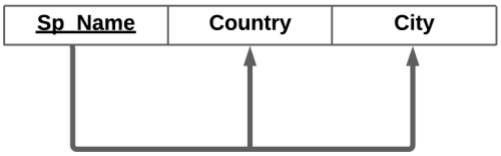
B. **The FLIGHT** relation schema satisfies all conditions of the 2NF because every non-prime attribute is fully functional dependent on the PK "Flight_Number".

C. **The FLIGHT** relation schema satisfies all conditions of the 3NF because it satisfies the 2NF and there are no non-prime attributes that are transitively dependent on the primary key "Flight_Number".

D. **The FLIGHT** relation schema satisfies all rules of the (BCNF) because there exists no functional dependency X →A where X is not a super key or A is a prime attribute and X not a super key.

__departure time and arrival__ are always estimated since they are bound to many conditions of which many are unexpected and could affect the elapsed time of the flight, thus our customer didn't want to be able to know the elapsed time by the departure and arrival time attributes combined

4. TECHNICAL_SERVICE_PROVIDER



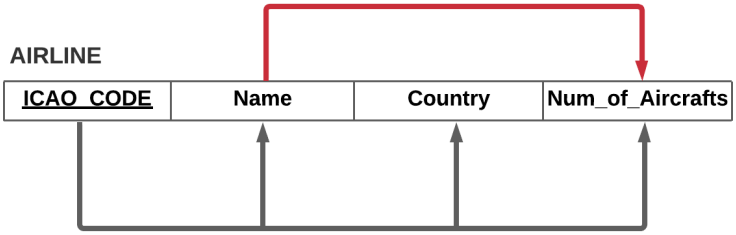
A. **The TECHNICAL_SERVICE_PROVIDER** relation schema satisfies all conditions of the 1NF it has neither multivalued attributes nor composite attributes. All attributes are single and atomic.

B. **The TECHNICAL_SERVICE_PROVIDER** relation schema satisfies all conditions of the 2NF because every non-prime attribute is fully functional dependent on the PK "Sp_Name".

C. **The TECHNICAL_SERVICE_PROVIDER** relation schema satisfies all conditions of the 3NF because it satisfies the 2NF and there are no non-prime attributes that are transitively dependent on the primary key “Sp_Name”.

D. **The TECHNICAL_SERVICE_PROVIDER** relation schema satisfies all rules of the (BCNF) because there exists no functional dependency X →A where X is not a super key or A is a prime attribute and X not a super key.

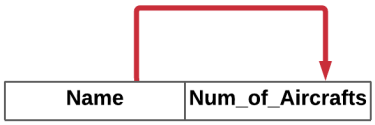
5. AIRLINE



A. **The AIRLINE** relation schema satisfies all conditions of the 1NF it has neither multivalued attributes nor composite attributes. All attributes are single and atomic.

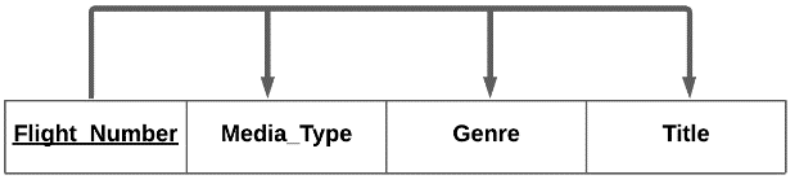
B. **The AIRLINE** relation schema satisfies all conditions of the 2NF because every non-prime attribute is fully functional dependent on the PK “ICAO_CODE”

C. **The AIRLINE** relation schema does not satisfy all conditions of the 3NF because the functional dependency represented by Name →Num_of_Aircrafts is a functional dependency where neither Name is a super key nor Num_of_Aircrafts is a prime attribute. Thus, further decomposition is needed.



A. **The AIRLINE** relation schema satisfies all rules of the (BCNF) because there exists no functional dependency X →A where X is not a super key or A is a prime attribute and X not a super key.

6. MEDIA_ENTERTAINMENT



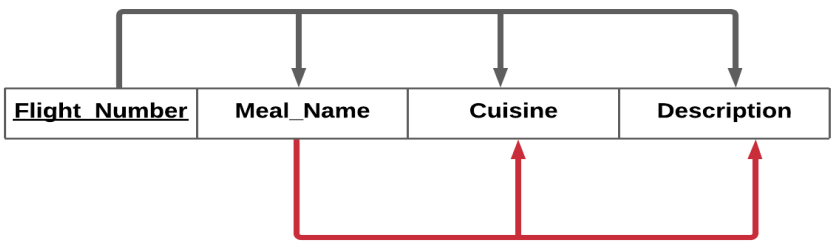
A. **The MEDIA_ENTERTAINMENT** relation schema satisfies all conditions of the 1NF it has neither multivalued attributes nor composite attributes. All attributes are single and atomic.

B. **The MEDIA_ENTERTAINMENT** relation schema satisfies all conditions of the 2NF because every non-prime attribute is fully functional dependent on the PK "Flight_Number".

C. **The MEDIA_ENTERTAINMENT** relation schema satisfies all conditions of the 3NF because it satisfies the 2NF and there are no non-prime attributes that are transitively dependent on the primary key "Flight_Number".

D. **The MEDIA_ENTERTAINMENT** relation schema satisfies all rules of the (BCNF) because there exists no functional dependency X → A where X is not a super key or A is a prime attribute and X not a super key.

7. FOOD_SERVICE

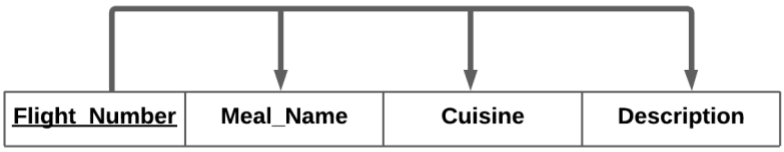


A. **The FOOD_SERVICE** relation schema satisfies all conditions of the 1NF it has neither multivalued attributes nor composite attributes. All attributes are single and atomic.

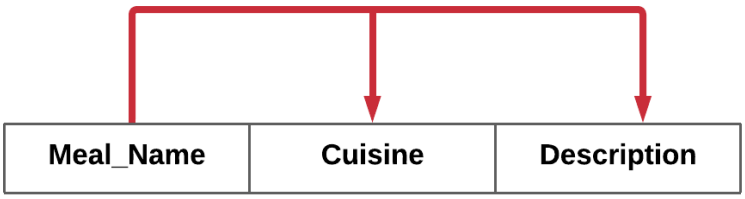
B. **The FOOD_SERVICE** relation schema satisfies all conditions of the 2NF because every non-prime attribute is fully functional dependent on the PK "Flight_Number"

C. **The PASSENGER** relation schema does not satisfy all conditions of the 3NF because the functional dependency represented by Meal_Name→Cuisine, Meal_Name→Description is a functional dependency where neither Meal_Name is a super key nor the other attributes are prime attributes. Thus, further decomposition is needed.

FOOD_SERVICE1

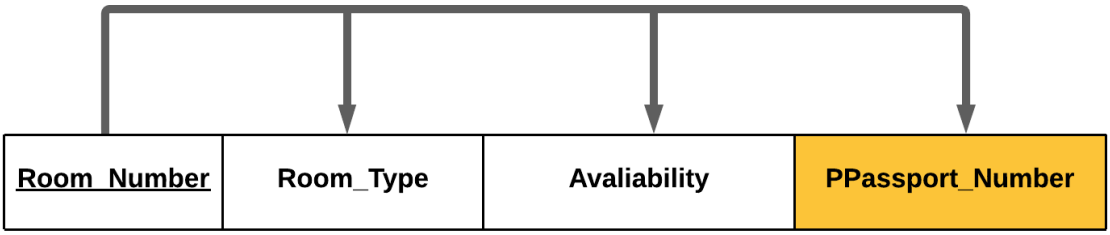


FOOD_SERVICE2



D-**The FOOD_SERVICE** relation schema satisfies all rules of the (BCNF) because there exists no functional dependency X →A where X is not a super key or A is a prime attribute and X not a super key.

8. TRANSIT_HOTEL_ROOM



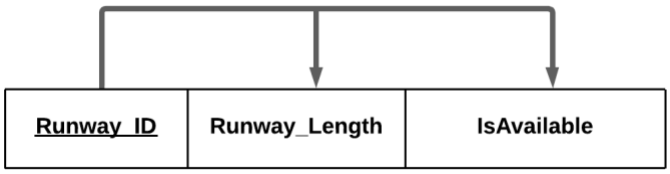
A. **The TRANSIT_HOTEL_ROOM** relation schema satisfies all conditions of the 1NF it has neither multivalued attributes nor composite attributes. All attributes are single and atomic.

B. **The TRANSIT_HOTEL_ROOM** relation schema satisfies all conditions of the 2NF because every non-prime attribute is fully functional dependent on the PK "Room_Number".

C. **The TRANSIT_HOTEL_ROOM** relation schema satisfies all conditions of the 3NF because it satisfies the 2NF and there are no non-prime attributes that are transitively dependent on the primary key "Room_Number".

D. **The TRANSIT_HOTEL_ROOM** relation schema satisfies all rules of the (BCNF) because there exists no functional dependency X → A where X is not a super key or A is a prime attribute and X not a super key.

9. RUNWAY



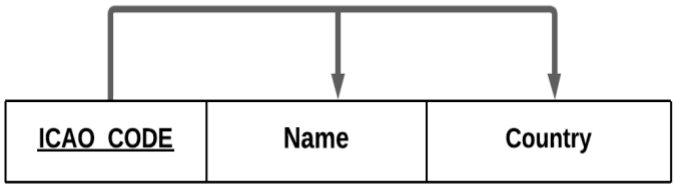
A. **The RUNWAY** relation schema satisfies all conditions of the 1NF it has neither multivalued attributes nor composite attributes. All attributes are single and atomic.

B. **The RUNWAY** relation schema satisfies all conditions of the 2NF because every non-prime attribute is fully functional dependent on the PK "Runway_ID".

C. **The RUNWAY** relation schema satisfies all conditions of the 3NF because it satisfies the 2NF and there are no non-prime attributes that are transitively dependent on the primary key “Runway_ID”.

D. **The RUNWAY** relation schema satisfies all rules of the (BCNF) because there exists no functional dependency X →A where X is not a super key or A is a prime attribute and X not a super key.

10. SHIPPING_COMPANY



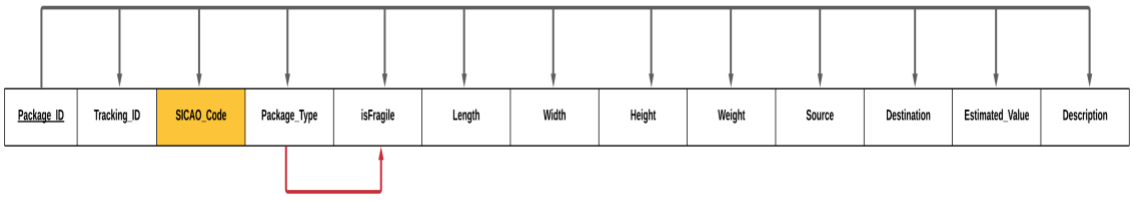
A- **The SHIPPING_COMPANY** relation schema satisfies all conditions of the 1NF it has neither multivalued attributes nor composite attributes. All attributes are single and atomic.

B- **The SHIPPING_COMPANY** relation schema satisfies all conditions of the 2NF because every non-prime attribute is fully functional dependent on the PK “ICAO_CODE”.

C- **The SHIPPING_COMPANY** relation schema satisfies all conditions of the 3NF because it satisfies the 2NF and there are no non-prime attributes that are transitively dependent on the primary key “ICAO_CODE”.

D- **The SHIPPING_COMPANY** relation schema satisfies all rules of the (BCNF) because there exists no functional dependency X →A where X is not a super key or A is a prime attribute and X not a super key.

11. PACKAGE

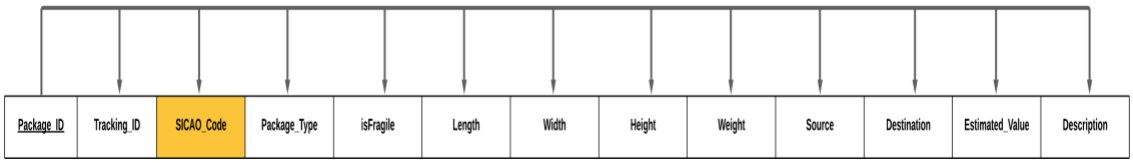


A. **The PACKAGE relation** schema satisfies all conditions of the 1NF it has neither multivalued attributes nor composite attributes. All attributes are single and atomic.

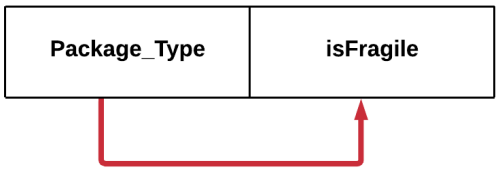
B. **The PACKAGE relation** schema satisfies all conditions of the 2NF because every non-prime attribute is fully functional dependent on the PK “Package_ID”

C. **The PACKAGE relation** schema does not satisfy all conditions of the 3NF because the functional dependency represented by Package_Type→isFragile, is a functional dependency where neither Package_Type is a super key nor isFragile is a prime attributes. Thus, further decomposition is needed.

PACKAGE1

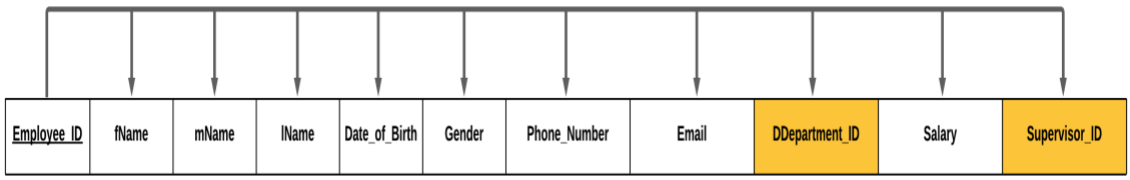


PACKAGE2



D. The PACKAGE relation schema satisfies all rules of the (BCNF) because there exists no functional dependency X →A where X is not a super key or A is a prime attribute and X not a super key.

12. EMPLOYEE



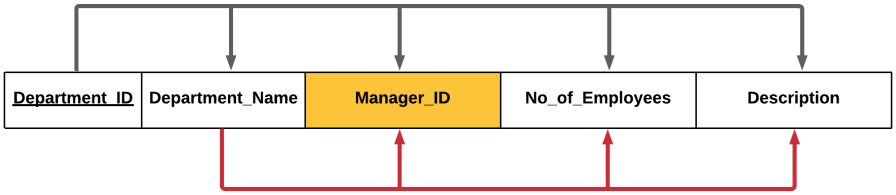
A. **The EMPLOYEE relation** schema satisfies all conditions of the 1NF it has neither multivalued attributes nor composite attributes. All attributes are single and atomic.

B. **The EMPLOYEE relation** schema satisfies all conditions of the 2NF because every non-prime attribute is fully functional dependent on the PK “Employee_ID”.

C. **The EMPLOYEE relation** schema satisfies all conditions of the 3NF because it satisfies the 2NF and there are no non-prime attributes that are transitively dependent on the primary key “Employee_ID”

D. **The EMPLOYEE relation** schema satisfies all rules of the (BCNF) because there exists no functional dependency X →A where X is not a super key or A is a prime attribute and X not a super key.

13. DEPARTMENT

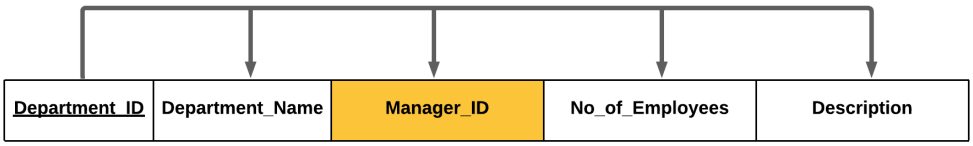


A. **The DEPARTMENT relation** schema satisfies all conditions of the 1NF it has neither multivalued attributes nor composite attributes. All attributes are single and atomic.

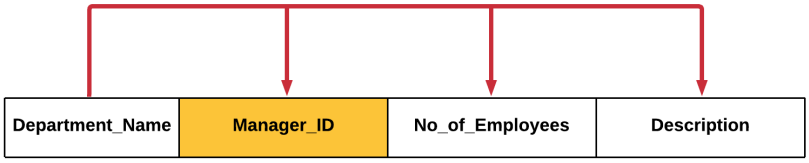
B. **The DEPARTMENT relation** schema satisfies all conditions of the 2NF because every non-prime attribute is fully functional dependent on the PK “Department_ID”

C. **The DEPARTMENT relation** schema does not satisfy all conditions of the 3NF because the functional dependency represented by Department_Name→Manager_ID, Department_Name→No_of_Employees, and Department_Name→Description is a functional dependency where neither Department_Name is a super key nor other attributes are prime attributes. Thus, further decomposition is needed.

DEPARTMENTA

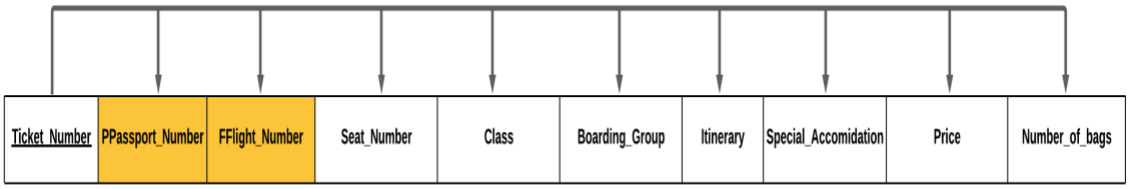


DEPARTMENTB



D. DEPARTMENT relation schema satisfies all rules of the (BCNF) because there exists no functional dependency X →A where X is not a super key or A is a prime attribute and X not a super key.

14.TICKET



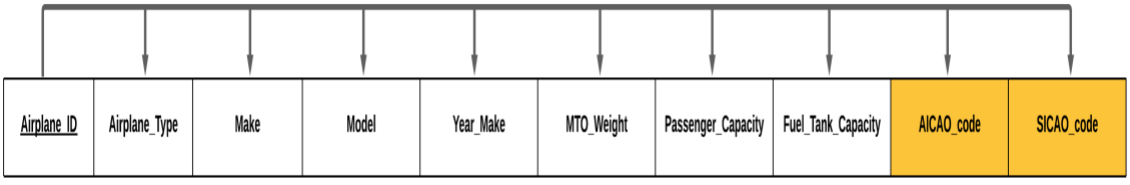
A- **The TICKET relation** schema satisfies all conditions of the 1NF it has neither multivalued attributes nor composite attributes. All attributes are single and atomic.

B- **The TICKET relation** schema satisfies all conditions of the 2NF because every non-prime attribute is fully functional dependent on the PK "Ticket_Number".

C- **The TICKET relation** schema satisfies all conditions of the 3NF because it satisfies the 2NF and there are no non-prime attributes that are transitively dependent on the primary key "Ticket_Number".

D- **The TICKET relation** schema satisfies all rules of the (BCNF) because there exists no functional dependency X →A where X is not a super key or A is a prime attribute and X not a superkey.

15. AIRPLANE



A-**The AIRPLANE relation** schema satisfies all conditions of the 1NF it has neither multivalued attributes nor composite attributes. All attributes are single and atomic.

B-**The AIRPLANE relation** schema satisfies all conditions of the 2NF because every non-prime attribute is fully functional dependent on the PK “Airplane_ID”

C- The AIRPLANE relation schema satisfies all conditions of the 3NF because it satisfies the 2NF and there is no non-prime attributes that are transitively dependent on the primary key “Airplane_ID”.

D- The AIRPLANE relation schema satisfies all rules of the (BCNF) because there exists no functional dependency X →A where X is not a super key or A is a prime attribute and X not a superkey.

16-FLIGHT_CREW

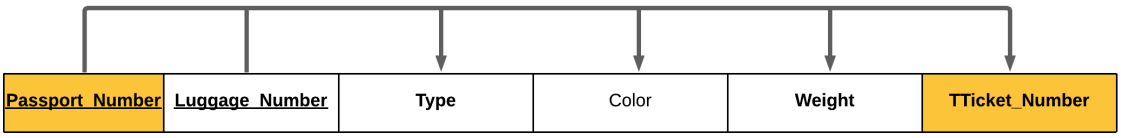
A- The FLIGHT_CREW relation schema satisfies all conditions of the 1NF it has neither multivalued attributes nor composite attributes. All attributes are single and atomic.

B- The FLIGHT_CREW relation schema satisfies all conditions of the 2NF because every non-prime attribute is fully functional dependent on the PK “License_No”.

C- The FLIGHT_CREW relation schema satisfies all conditions of the 3NF because it satisfies the 2NF and there is no non-prime attributes that are transitively dependent on the primary key “License_No”.

D- The FLIGHT_CREW relation schema satisfies all rules of the (BCNF) because there exists no functional dependency X →A where X is not a super key or A is a prime attribute and X not a superkey.

17-LUGGAGE



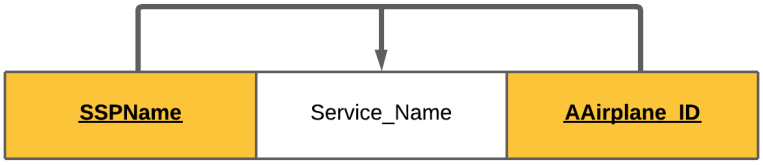
A- The LUGGAGE relation schema satisfies all conditions of the 1NF it has neither multivalued attributes nor composite attributes. All attributes are single and atomic.

B- The LUGGAGE relation schema satisfies all conditions of the 2NF because every non-prime attribute is fully functional dependent on the PK “Passport_Number and Luggage_Number”.

C- The LUGAGGE relation schema satisfies all conditions of the 3NF because it satisfies the 2NF and there are no non-prime attributes that are transitively dependent on the primary key “Passport_Number and Luggage_Number”.

D- The LUGAGGE relation schema satisfies all rules of the (BCNF) because there exists no functional dependency X →A where X is not a super key or A is a prime attribute and X not a superkey.

18-Provides_Services



A- The *Provides_Services* relation schema satisfies all conditions of the 1NF it has neither multivalued attributes nor composite attributes. All attributes are single and atomic.

B- The *Provides_Services* relation schema satisfies all conditions of the 2NF because every non-prime attribute is fully functional dependent on the PK “SSPname and AAirplane_ID”.

C- The *Provides_Services* relation schema satisfies all conditions of the 3NF because it satisfies the 2NF and there is no non-prime attributes that are transitively dependent on the primary key “SSPname and AAirplane_ID”.

D- The *Provides_Services* relation schema satisfies all rules of the (BCNF) because there exists no functional dependency X →A where X is not a super key or A is a prime attribute and X not a superkey.

19- **Employee**_Technical_Service_Provider**(has_A5)**

B- The *has_A5* relation schema satisfies all conditions of the 1NF it has neither multivalued attributes nor composite attributes. All attributes are single and atomic.

B- The *has_A5* relation schema satisfies all conditions of the 2NF because every non-prime attribute is fully functional dependent on the PK "EEmployee_ID".

C- The *has_A5* relation schema satisfies all conditions of the 3NF because it satisfies the 2NF and there is no non-prime attributes that are transitively dependent on the primary key "EEmployee_ID".

D- The *has_A5* relation schema satisfies all rules of the (BCNF) because there exists no functional dependency X →A where X is not a super key or A is a prime attribute and X not a superkey.

Relation Schemas without any non-prime attribute:

Shops_In

SStore_Name	PPassport_Number

Employee_DutyFree (has_A1)

EEmployee_ID	SStore_name

Employee_Airline(has_A2)

EEmployee_ID	AICAO_code

Employee_ShippingCompany(has_A6)

EEmployee_ID	SICAO_code

Step06: Mapping of Multivalued Attributes:

At the stage of the mapping we are dealing with multivalued attributes. Each multivalued attribute is mapped by the creation of a new relation in which we put any associated attributes along with the primary key of the entity on which the multivalued attribute is found. The primary key of the new relation is the combination of all the attributes.

1. **Media_entertainment**

Title	***Media_Type***	***Genre***	***Flight_Number***

The multivalued attribute **Media_entertainment is** found on the **FLIGHT** entity. it is represented as relation in which the combination of all the attributes makes up the primary key. This relation represents the different movies, songs, documentaries that can be watched or listened to during each flight.

2. **Food_service**

Meal_Name	**Description**	**Cuisine**	**FlightNum**

The multivalued attribute **Food_service is** found on the **FLIGHT** entity. it is represented as relation in which the combination of all the attributes makes up the primary key. This relation represents the different meal options that passengers have with further details.

XII - SQL Queries and Oracle Server:

1. **PASSENGER**

```
CREATE TABLE PASSENGER
        (
          Passport_Number    VARCHAR (20)    NOT NULL,
          fName    VARCHAR (15)    NOT NULL,
```

```
mName    VARCHAR (15)    NOT NULL,
lName    VARCHAR (15)    NOT NULL,
Date_of_Birth    DATE    NOT NULL,
Gender    CHAR    NOT NULL,
Nationality    VARCHAR (20)    NOT NULL,
Phone_Number    VARCHAR (20)    NOT NULL,
Email            VARCHAR (20),
Ticket_Number    VARCHAR (20)    NOT NULL,
Flight_Number    VARCHAR (20)    NOT NULL,
Visa_Number    VARCHAR (20),
Visa_Type    VARCHAR (20),
Visa_Expiration_Date    VARCHAR (20),
PRIMARY KEY (Passport_Number)
);
```

//We added foreign keys after creating the tables and inserting all the data

```
ALTER TABLE PASSENGER
ADD FOREIGN KEY (FLIGHT_NUMBER) REFERENCES
FLIGHT (FLIGHT_NUMBER)
ADD FOREIGN KEY (TICKET_NUMBER) REFERENCES
TICKET (TICKET_NUMBER);
```

2. FLIGHT

```
CREATE TABLE FLIGHT
(
Flight_Number    VARCHAR (6)    NOT NULL,
Airline_Code    VARCHAR (3)    NOT NULL,
Airplane_ID    VARCHAR (8)    NOT NULL,
Origin    VARCHAR (4)    NOT NULL,
Destination    VARCHAR (4)    NOT NULL,
Departure_Time    TIMESTAMP    NOT NULL,
Arrival_Time    TIMESTAMP    NOT NULL,
Elapsed_Time    TIMESTAMP    NOT NULL,
Ground_Handling    VARCHAR (20)    NOT NULL,
```

```
    Status    VARCHAR (20)    NOT NULL,
    Gate_Number   VARCHAR (4)    NOT NULL,
    Runway_ID VARCHAR (3) NOT NULL,
    PRIMARY KEY (Flight_Number)
);
//We added foreign keys after creating the tables and inserting all the data
ALTER TABLE FLIGHT
ADD FOREIGN KEY (Airplane_ID) REFERENCES AIRPLANE(Airplane_ID);
ALTER TABLE FLIGHT
ADD FOREIGN KEY(Runway_ID) REFERENCES
RUNWAY(Runway_ID);
ADD FOREIGN KEY(AIRLINE_CODE) REFERENCES
AIRLINE (AIRLINE_CODE);
);
```

3. TECHNICAL_SERVICE_PROVIDER

```
CREATE TABLE TECHNICAL_SERVICE_PROVIDER
(
    SP_Name    VARCHAR (20) NOT NULL,
    Country    VARCHAR (20) NOT NULL,
    City    VARCHAR (20) NOT NULL,
    PRIMARY KEY(SP_Name)
);
```

4. AIRLINE

```
CREATE TABLE AIRLINE
(
    ICAO_CODE    VARCHAR (3) NOT NULL,
    Name    VARCHAR (50) NOT NULL,
    Country    VARCHAR (20) NOT NULL,
    Num_of_Aircrafts    INT  NOT NULL,
    PRIMARY KEY(ICAO_Code)
);
```

5. **MEDIA_ENTERTAINMENT**

```
CREATE TABLE MEDIA_ENTERTAINMENT
  (
    Flight_Number   VARCHAR (6) NOT NULL,
    Media_Type   VARCHAR (10) NOT NULL,
    Genre   VARCHAR (30),
    Title   VARCHAR (80) NOT NULL,
  FOREIGN KEY (Flight_Number) REFERENCES
  FLIGHT(Flight_Number)
  );
```

6. **FOOD_SERVICE**

```
CREATE TABLE FOOD_SERVICE
(
  Flight_Number   VARCHAR (6) NOT NULL,
  Meal_Name   VARCHAR (20) NOT NULL,
  Cuisine   VARCHAR (20),
  Description   VARCHAR (100),
FOREIGN KEY (Flight_Number) REFERENCES FLIGHT(Flight_Number)
);
```

7. **EMPLOYEE**

```
CREATE TABLE EMPLOYEE
(
  Employee_ID   VARCHAR (5)   NOT NULL,
  fName   VARCHAR (15)   NOT NULL,
  mName   VARCHAR (15)   NOT NULL,
  lName   VARCHAR (15)   NOT NULL,
  Date_of_Birth   DATE   NOT NULL,
  Gender   CHAR   NOT NULL,
  Phone_Number   VARCHAR (20)   NOT NULL,
  Email          VARCHAR (20) NOT NULL,
```

```
    Supervisor VARCHAR (5),
    Department_ID    INTEGER    NOT NULL,
    Salary    INTEGER    NOT NULL,
    PRIMARY KEY(Employee_ID)
    FOREIGN KEY (DEPARTMENT_ID) REFERENCES
    DEPARTMENT(DEPARTMENT_ID)
);
```

8. Transit_Hotel_Room

```
 CREATE TABLE Transit_Hotel_Room
(
    Room_Number INTEGER NOT NULL,
    Romm_Type   VARCHAR (10) NOT NULL,
    Availability    INTEGER NOT NULL,
    Passenger    VARCHAR (20),
    PRIMARY KEY (Room_Number),
    FOREIGN KEY (Passenger) REFERENCES
    PASSANGER(Passport_Number)
);
```

9. RUNWAY

```
 CREATE TABLE RUNWAY
(
    Runway_ID    VARCHAR(3)    NOT NULL,
    Runway_Length    INTEGER    NOT NULL,
    IsAvailable    INTEGER NOT NULL,
    PRIMARY KEY (Runway_ID),
);
```

10.DEPARTMENT

```
 CREATE TABLE DEPARTMENT
(
```

```
    Department_ID    INTEGER    NOT NULL,
    Department_Name    VARCHAR (20)    NOT NULL,
    Manager    INTEGER NOT NULL,
    No_of_Employees    INTEGER NOT NULL,
    Description    VARCHAR (100) NOT NULL,
    PRIMARY KEY (Department_ID),
    FOREIGN KEY (Manager) REFERENCES EMPLOYEE(Employee_ID)
);
```

11.AIRPLANE

```
 CREATE TABLE AIRPLANE
(
    Airplane_ID   VARCHAR (8)    NOT NULL,
    Airplane_Type    VARCHAR (10)    NOT NULL,
    Make    VARCHAR (20)    NOT NULL,
    Model    VARCHAR (20)    NOT NULL,
    Year_Make INTEGER     NOT NULL,
    MTO_Weight    INTEGER    NOT NULL,
    Passanger_Capacity    INTEGER      NULL,
    Fuel_Tank_Capacity    INTEGER     NOT NULL,
    AIRLINE_ICAO_CODE  VARCHAR(3)    NOT NULL
    SHIPPING_ICAO_CODE VARCHAR(3)    NOT NULL
    PRIMARY KEY (Airplane_ID)
    FOREIGN KEY (AIRLINE_ICAO_CODE) REFERENCES
    AIRLINE (ICAO_CODE)
);
```

12.FLIGHT_CREW

```
 CREATE TABLE FLIGHT_CREW
(
    License_No VARCHAR (8)    NOT NULL,
    Employee_ID INTEGER    NOT NULL,
    Position    VARCHAR (10) NOT NULL,
    Rating    VARCHAR (10)    NOT NULL,
    Status    VARCHAR (10)    NOT NULL,
   Gate_Number    VARCHAR (4) NOT NULL,
   AIRLINE    VARCHAR (3) NOT NULL
```

```
    Flight_Number    VARCHAR (6),
    PRIMARY KEY(License_No),
    FOREIGN KEY(Employee_ID) REFERENCES    EMPLOYEE(Employee_ID),
    FOREIGN KEY(AIRLINE) REFERENCES AIRLINE(ICAO_Code)
);
```

13.SHIPPING_COMPANY

```
CREATE TABLE SHIPPING_COMPANY
(
    ICAO_CODE    VARCHAR (3) NOT NULL,
    Name    VARCHAR (50) NOT NULL,
    Country    VARCHAR (20) NOT NULL,
    Num_of_Aircrafts    INT NOT NULL,
    PRIMARY KEY(ICAO_Code)
);
```

14.PACKAGE

```
CREATE TABLE PACKAGE
(
    Package_ID    INTEGER NOT NULL,
    Tracking_ID    INTEGER NOT NULL,
    Shipping_Company    VARCHAR (3)    NOT NULL,
    Package_Type    VARCHAR (10)    NOT NULL,
    isFragile    BOOLEAN NOT NULL,
    Length    INTEGER NOT NULL,
    Width    INTEGER NOT NULL,
    Height    INTEGER NOT NULL,
    Weight    INTEGER NOT NULL,
    Source    VARCHAR (50)    NOT NULL,
    Destination    VARCHAR (50)    NOT NULL,
    Estimated_Value    INTEGER NOT NULL,
    Description    VARCHAR (50),
    PRIMARY KEY (Package_ID, Tracking_ID),
```

```
    FOREIGN KEY(Shipping_Company) REFERENCES
SHIPPING_COMPANY(ICAO_Code)
);
```

15.DUTY_FREE_STORE

```
 CREATE TABLE DUTY_FREE_STORE
(
    Store_Name    VARCHAR (20)     NOT NULL,
    Operator    VARCHAR (20) NOT NULL,
    Store_Type    VARCHAR (20) NOT NULL,
    PRIMARY KEY(Store_Name)
);
```

16.TICKET

```
 CREATE TABLE TICKET
(
    Ticket_Number    VARCHAR (17) NOT NULL,
    Passport_Number    VARCHAR (20) NOT NULL,
    Flight_Number    VARCHAR (6) NOT NULL,
    Seat_Number    VARCHAR (3) NOT NULL,
    Class    VARCHAR (10) NOT NULL,
    Boarding_Group    CHAR (1) NOT NULL,
    Itinerary    VARCHAR (10) NOT NULL,
    Special_Accomidation    VARCHAR (10),
    PRIMARY KEY(Ticket_Number),
    FOREIGN KEY(Passport_Number) REFERENCES
PASSANGER(Passport_Number),
    FOREIGN KEY(Flight_Number) REFERENCES FLIGHT(Flight_Number)
);
```

17.LUGGAGE

```
CREATE TABLE LUGGAGE
```

```
(
   Passport_Number VARCHAR (20) NOT NULL,
   Luggage_Number INTEGER    NULL,
   Luggage_Type    VARCHAR (10) NOT NULL,
   Color    VARCHAR (10),
   Weight     INTEGER NULL,
  TICKET_NUMBER VARCHAR (17),
 FOREIGN KEY(Passport_Number) REFERENCES
PASSANGER(Passport_Number),
 FOREIGN KEY (TICKET_NUMBER) REFERENCES TICKET
(TICKET_NUMBER)
);
```

18.PROVIDES_SERVICES

```
CREATE TABLE Provides_Services
(
SP_Name    VARCHAR(20) NOT NULL,
Service_Name VARCHAR(50) NOT NULL,
Airplane_ID    VARCHAR(8) NOT NULL,
FOREIGN KEY(SP_Name) REFERENCES
TECHNICAL_SERVICE_PROVIDER(SP_Name),
FOREIGN KEY(Airplane_ID) REFERENCES AIRPLANE(Airplane_ID)
);
```

19.SHOPS_IN

```
CREATE TABLE Shops_In
(
```

```
    Store_Name VARCHAR(50) NOT NULL,
    Passport_Number VARCHAR(20) NOT NULL,
    FOREIGN KEY(Store_Name) REFERENCES
DUTY_FREE_STORE(Store_Name),
    FOREIGN KEY(Passport_Number) REFERENCES
PASSANGER(Passport_Number)
);
```

20.DUTYFREE_EMPLOYEE

```
CREATE TABLE DutyFree_Employee
(
    Employee_ID VARCHAR(5) NOT NULL,
    Store_Name VARCHAR(50) NOT NULL,
    FOREIGN KEY(Store_Name) REFERENCES
DUTY_FREE_STORE(Store_Name),
    FOREIGN KEY(Employee_ID) REFERENCES EMPLOYEE(Employee_ID)
);
```

21.AIRLINE_EMPLOYEE

```
CREATE TABLE Airline_Employee
(
```

```
    Employee_ID VARCHAR(5) NOT NULL,
    AICAO_Code VARCHAR(3) NOT NULL,
    FOREIGN KEY(AICAO_Code) REFERENCES AIRLINE(ICAO_Code),
    FOREIGN KEY(Employee_ID) REFERENCES
EMPLOYEE(Employee_ID)
);
```

22.SHIPPINGCOMPANY_EMPLOYEE

```
CREATE TABLE ShippingCompany_Employee
(
    Employee_ID VARCHAR(5) NOT NULL,
    SICAO_Code VARCHAR(3) NOT NULL,
FOREIGN KEY(SICAO_Code) REFERENCES
Shipping_Company(ICAO_Code),
FOREIGN KEY(Employee_ID) REFERENCES
EMPLOYEE(Employee_ID)
);
```

23.TECHINCALSP_EMPLOYEE

```
CREATE TABLE TechincalSP_Employee
```

```
(
   Employee_ID VARCHAR(5) NOT NULL,
   TSP_Name VARCHAR(20) NOT NULL,
 FOREIGN KEY(TSP_Name) REFERENCES
Technical_Service_Provider(SP_NAME),
FOREIGN KEY(Employee_ID) REFERENCES
EMPLOYEE(Employee_ID)
);
```

XII - Inserting the data:

1. DUTY_FREE_STORE:

- **INSERT INTO DUTY_FREE_STORE** (Store_Name, Operator, Store_Type)
  VALUES ('Virgin Megastore', 'Virgin', 'Electronics');

- **INSERT INTO DUTY_FREE_STORE** (Store_Name, Operator, Store_Type)
  VALUES ('Jumbo Electronics', 'Sicin ', 'Electronics');

- **INSERT INTO DUTY_FREE_STORE** (Store_Name, Operator, Store_Type)
  VALUES ('Bartartin', 'Aroma', 'Food');

- **INSERT INTO DUTY_FREE_STORE (**Store_Name, Operator, Store_Type)
  VALUES ('Lora ', ' Fantasy', 'Fashion wear');

- **INSERT INTO DUTY_FREE_STORE** (Store_Name, Operator, Store_Type)

VALUES (‘Wakanda Library’, 'Wakanda history', 'books');

- **INSERT INTO DUTY_FREE_STORE** (Store_Name, Operator, Store_Type)
  VALUES (‘flakeshake', 'Snowball ', 'smoothies');

- **INSERT INTO DUTY_FREE_STORE** (Store_Name, Operator, Store_Type)
  VALUES (‘Sephora', 'Sephora', 'makeup');

- **INSERT INTO DUTY_FREE_STORE** (Store_Name, Operator, Store_Type)
  VALUES (‘Vintage', '1765Russia', 'vodka');

- **INSERT INTO DUTY_FREE_STORE** (Store_Name, Operator, Store_Type)
  VALUES (‘Disney Store', 'Disney Store', 'toys');

- **INSERT INTO DUTY_FREE_STORE** (Store_Name, Operator, Store_Type)
  VALUES (‘Hersheys Chocolate World, 'The Hershey Company ', 'Chocolate');

- **INSERT INTO DUTY_FREE_STORE** (Store_Name, Operator, Store_Type)
  VALUES (‘Swarovski', 'Swarovski AG', 'Jewelry');

- **INSERT INTO DUTY_FREE_STORE** (Store_Name, Operator, Store_Type)
  VALUES (‘Invicta Store', 'Invicta Watch Group', 'Fashion wear');

- **INSERT INTO DUTY_FREE_STORE** (Store_Name, Operator, Store_Type)
  VALUES (‘TISSOT ', 'Tissot SA', 'Watches');

- **INSERT INTO DUTY_FREE_STORE** (Store_Name, Operator, Store_Type)
  VALUES (‘Swatch’, 'Swatch', 'Watches');

- **INSERT INTO DUTY_FREE_STORE** (Store_Name, Operator, Store_Type)
  VALUES (‘M&MS World', 'Marc Inc', 'Sweets');

- **INSERT INTO DUTY_FREE_STORE** (Store_Name, Operator, Store_Type)
  VALUES (‘Sunglass Hut ', 'Luxottica Group', 'Sunglasses');

- **INSERT INTO DUTY_FREE_STORE** (Store_Name, Operator, Store_Type)
  VALUES (‘MUJI’ ', 'Ryohin Keikaku Co.', 'Wine');

- **INSERT INTO DUTY_FREE_STORE** (Store_Name, Operator, Store_Type)
  VALUES (‘Levis Store', 'Levi Strauss & Co', 'Fashion’);

- **INSERT INTO DUTY_FREE_STORE** (Store_Name, Operator, Store_Type)
  VALUES (‘Express’, 'Express', 'Sweets’);

- INSERT INTO DUTY_FREE_STORE (Store_Name, Operator, Store_Type)
  VALUES (‘Benihana', 'Benihana', 'Restaurant');

- **INSERT INTO DUTY_FREE_STORE** (Store_Name, Operator, Store_Type)
  VALUES (‘Gap', 'Gap Inc', 'Fashion');

- **INSERT INTO DUTY_FREE_STORE** (Store_Name, Operator, Store_Type)
  VALUES (‘Spiritland', 'Spiritland Productions', 'Music');

- **INSERT INTO DUTY_FREE_STORE** (Store_Name, Operator, Store_Type)
  VALUES (‘Harman', 'Harman', 'Music');

2. PASSENGER:

- **INSERT INTO PASSENGER** (Passport_Number, fName, mName, lName, Date_of_Birth, Gender, Nationality, Phone_Number, Email, Ticket_Number, Flight_Number, Visa_Number, Visa_Type, Visa_Expiration_Date)
  VALUES (‘BR102030', 'Hulk’, ‘Noah', 'Khouja', '2002/11/11', 'M', 'Lebanon', '0096172882551','HulkKhouja11@gmail.com','20201013MEA325001','MEA325','00074715','Student','2025/10/13');

- **INSERT INTO PASSENGER** (Passport_Number, fName, mName, lName, Date_of_Birth, Gender, Nationality, Phone_Number, Email, Ticket_Number, Flight_Number, Visa_Number, Visa_Type, Visa_Expiration_Date)
  VALUES (‘LD102031', 'Iron Man', 'Adam','Saab','1986-11-1','M','United Kingdom','0114402071231234','IronManSaab101@gmail.com','20201013MEA325002','MEA325','19283096','Work','2023-10-15');

- **INSERT INTO PASSENGER** (Passport_Number, fName, mName, lName, Date_of_Birth, Gender, Nationality, Phone_Number, Email, Ticket_Number, Flight_Number, Visa_Number, Visa_Type, Visa_Expiration_Date)

VALUES ('TR764912', 'Thor', 'Monir','Rivier','1976-12-10','M','Lebanon','0096175815769','ThorRivier@gmail.com','20201013MEA325021','MEA325','67332929','Immigrant','2021-03-09');

- **INSERT INTO PASSENGER** (Passport_Number, fName, mName, lName, Date_of_Birth, Gender, Nationality, Phone_Number, Email, Ticket_Number, Flight_Number, Visa_Number, Visa_Type, Visa_Expiration_Date)
  VALUES ('LD424925', 'Black Widow', 'Evelyn','Mosby','1996-01-18','F','United Kingdom','0117274271094975','BlackWidow@gmail.com','20200426LOV690364','LOV690','73967392','Tourist','2020-11-06');

- **INSERT INTO PASSENGER** (Passport_Number, fName, mName, lName, Date_of_Birth, Gender, Nationality, Phone_Number, Email, Ticket_Number, Flight_Number, Visa_Number, Visa_Type, Visa_Expiration_Date)
  VALUES ('PR735932', 'Captain America', 'micheal','Evans','2002/11/11','M','France','33145245282','CaptainAmerica@gmail.com','20200426LOV69038','LOV690','92597825','Work','2023-11-16');

- **INSERT INTO PASSENGER** (Passport_Number, fName, mName, lName, Date_of_Birth, Gender, Nationality, Phone_Number, Email, Ticket_Number, Flight_Number, Visa_Number, Visa_Type, Visa_Expiration_Date)
  VALUES ('ZH771661', 'Clint Barton', 'khaled','evian','1989-11-21','M','Lebanon','0096103010256','ClintEvian@gmail.com','20200426-LOV-690369','LOV690','00012394','Immigrant','2025-11-29');

- **INSERT INTO PASSENGER** (Passport_Number, fName, mName, lName, Date_of_Birth, Gender, Nationality, Phone_Number, Email, Ticket_Number, Flight_Number, Visa_Number, Visa_Type, Visa_Expiration_Date)
  VALUES (‘NY926392', 'Loki', 'anthony','nestle','1995-03-01','M','United States','0015417543010','LokiNestle@gmail.com','20200609AAL29852','AAL298','02135614','Student','2022-10-12');

- **INSERT INTO PASSENGER** (Passport_Number, fName, mName, lName, Date_of_Birth, Gender, Nationality, Phone_Number, Email, Ticket_Number, Flight_Number, Visa_Number, Visa_Type, Visa_Expiration_Date)
  VALUES (‘TX866394', 'Pepper potts', 'carlos','johnson','1998-04-23','F','United States','001213-3243692','PepperJohnson@gmail.com','20200609AAL29853','AAL298','12426551','Student','2030-10-10');

- **INSERT INTO PASSENGER** (Passport_Number, fName, mName, lName, Date_of_Birth, Gender, Nationality, Phone_Number, Email, Ticket_Number, Flight_Number, Visa_Number, Visa_Type, Visa_Expiration_Date)
  VALUES (‘TR437918', 'Jarvis', 'Tony','Bettany','2001-11-23','F','Libya','00244-124-532','JarvisTony@gmail.com','20200609AAL29862','AAL298','16853179','Student','2030-05-09');

- **INSERT INTO PASSENGER** (Passport_Number, fName, mName, lName, Date_of_Birth, Gender, Nationality, Phone_Number, Email, Ticket_Number, Flight_Number, Visa_Number, Visa_Type, Visa_Expiration_Date)

VALUES (‘OR484363', 'Jasper Sitwell', 'Lee','Mathers','1982-04-13','M','Algeria','00244146901','JasperMathers@gmail.com','20200128LUT948104','LUT948','56318279','Work','2022-09-15');

- **INSERT INTO PASSENGER** (Passport_Number, fName, mName, lName, Date_of_Birth, Gender, Nationality, Phone_Number, Email, Ticket_Number, Flight_Number, Visa_Number, Visa_Type, Visa_Expiration_Date)
  VALUES (‘AB972922', 'Jocasta', 'mandarin','Alabman','1999-11-02','F','Cote Divoire','00244483789','JocastaAlabman@gmail.com','20200128LUT948156','LUT948','22646247','Tourist','2026-11-17’);

- **INSERT INTO PASSENGER** (Passport_Number, fName, mName, lName, Date_of_Birth, Gender, Nationality, Phone_Number, Email, Ticket_Number, Flight_Number, Visa_Number, Visa_Type, Visa_Expiration_Date)
  VALUES (‘MM425012', 'Jerry Mercenary', 'Colin','Sixtos','1990-12-27','M','India','0091024628023657','JerryMRC@gmail.com','20200128LUT948204','LUT948','91842637','Work','2025-04-21');

- **INSERT INTO PASSENGER** (Passport_Number, fName, mName, lName, Date_of_Birth, Gender, Nationality, Phone_Number, Email, Ticket_Number, Flight_Number, Visa_Number, Visa_Type, Visa_Expiration_Date)
  VALUES (‘BE484363', 'Alexis Denisof', 'Marshall','Jackson','1976-09-22','M','India','009102228043918','AlexisDenisof@gmail.com','20170510LBR931891','LBR931','25724301','Immigrant','2022-11-15');

- **INSERT INTO PASSENGER** (Passport_Number, fName, mName, lName, Date_of_Birth, Gender, Nationality, Phone_Number, Email, Ticket_Number, Flight_Number, Visa_Number, Visa_Type, Visa_Expiration_Date)
  VALUES (‘MM289371', 'Captain Marvel', 'Mia','Errickson','1979-12-27','F','India','0091389729471201','CaptainMarval@gmail.com','20170510LBR931893','LBR931','83765219','Work','2023-06-23');

- **INSERT INTO PASSENGER** (Passport_Number, fName, mName, lName, Date_of_Birth, Gender, Nationality, Phone_Number, Email, Ticket_Number, Flight_Number, Visa_Number, Visa_Type, Visa_Expiration_Date)
  VALUES (‘BE484368', 'Scarlet Witch', 'Ranvir','kapoor','1975-11-13','F','India','0091192830827235','ScarletWitch@gmail.com','20170510LBR931862','LBR931','98639823','Tourist','2021-12-16');

- **INSERT INTO PASSENGER** (Passport_Number, fName, mName, lName, Date_of_Birth, Gender, Nationality, Phone_Number, Email, Ticket_Number, Flight_Number, Visa_Number, Visa_Type, Visa_Expiration_Date)
  VALUES (‘MO436262', 'Rashmi Rustagi', 'Sergi','Boris','1985-03-06','F','Russia','007-9535553026','RashmiBoris@gmail.com','20161112AFL666196','AFL666','01397203','Immigrant','2022-06-27');

- **INSERT INTO PASSENGER** (Passport_Number, fName, mName, lName, Date_of_Birth, Gender, Nationality, Phone_Number, Email, Ticket_Number, Flight_Number, Visa_Number, Visa_Type, Visa_Expiration_Date)

VALUES ('SP924053', 'Ashley Johnson', 'Dimitri','Abramov','1999-12-10','F','Russia','007-852093632','AshelyAbramov@gmail.com','20161112AFL66637','AFL666','72160293','Student','2025-11-2');

- **INSERT INTO PASSENGER** (Passport_Number, fName, mName, lName, Date_of_Birth, Gender, Nationality, Phone_Number, Email, Ticket_Number, Flight_Number, Visa_Number, Visa_Type, Visa_Expiration_Date)

  VALUES ('KN732421', 'Colin Strause', 'Igor','Agafnov','2000-11-03','M','Russia','007-9065558118','ColinAGAF@gmail.com','20191203TAP717109','TAP717','12219283','Student','2022-05-20');

- **INSERT INTO PASSENGER** (Passport_Number, fName, mName, lName, Date_of_Birth, Gender, Nationality, Phone_Number, Email, Ticket_Number, Flight_Number, Visa_Number, Visa_Type, Visa_Expiration_Date)

  VALUES ('PO777612', 'Damion Politier', 'Ricardo','Fonte','1988-02-26','M','PORTUGUESE','00351-800180449','DAMFONTE@gmail.com','20191203TAP717709','TAP717','05730243','Work','2025-11-10');

3. **LUGGAGE:**

INSERT ALL

- **INTO LUGGAGE** (PASSPORT_NUMBER, LUGGAGE_NUMBER, LUGGAGE_TYPE, COLOR, WEIGHT, TICKET_NUMBER)

VALUES ('PO777612', NULL, 'Suitcase', 'Blue', NULL, '20191203TAP717709')

- **INTO LUGGAGE** (PASSPORT_NUMBER, LUGGAGE_NUMBER, LUGGAGE_TYPE, COLOR, WEIGHT, TICKET_NUMBER)

VALUES ('KN732421', 20645100, 'Carry-on', 'Black', 8, '20191203TAP717109')

- **INTO LUGGAGE** (PASSPORT_NUMBER, LUGGAGE_NUMBER, LUGGAGE_TYPE, COLOR, WEIGHT, TICKET_NUMBER)

VALUES ('SP924053', 20441075, 'Suitcase', 'Black', 20, '20161112AFL66637')

- **INTO LUGGAGE** (PASSPORT_NUMBER, LUGGAGE_NUMBER, LUGGAGE_TYPE, COLOR, WEIGHT, TICKET_NUMBER)
- VALUES ('MO436262', NULL, 'Personal item', 'Green', Null, '20161112AFL666196')

- **INTO LUGGAGE** (PASSPORT_NUMBER, LUGGAGE_NUMBER, LUGGAGE_TYPE, COLOR, WEIGHT, TICKET_NUMBER)

VALUES ('BE484363', 20411856, 'Suitcase', 'Blue', 23, '20190110LBR931891')

- **INTO LUGGAGE** (PASSPORT_NUMBER, LUGGAGE_NUMBER, LUGGAGE_TYPE, COLOR, WEIGHT, TICKET_NUMBER)

VALUES ('MM289371', 20854410, 'Suitcase', 'Red', 17, '20190412LBR931893')

- **INTO LUGGAGE** (PASSPORT_NUMBER, LUGGAGE_NUMBER, LUGGAGE_TYPE, COLOR, WEIGHT, TICKET_NUMBER)

VALUES ('MM425012', NULL, 'Suitcase', 'Black', NULL, '20200128LUT948204')

- **INTO LUGGAGE** (PASSPORT_NUMBER, LUGGAGE_NUMBER, LUGGAGE_TYPE, COLOR, WEIGHT, TICKET_NUMBER)

VALUES ('AB972922', 20449930, 'Carry-on', 'Red', 9, '20200128LUT948156 ')

- **INTO LUGGAGE** (PASSPORT_NUMBER, LUGGAGE_NUMBER, LUGGAGE_TYPE, COLOR, WEIGHT, TICKET_NUMBER)

VALUES ('OR484363', 20954834, 'Suitcase', 'Grey', 35, '20200128LUT948104');

- **INTO LUGGAGE** (PASSPORT_NUMBER, LUGGAGE_NUMBER, LUGGAGE_TYPE, COLOR, WEIGHT, TICKET_NUMBER)

VALUES ('OR484363', 20954833, 'Suitcase’, 'Brown', 13, '20200128LUT948104')

- **INTO LUGGAGE** (PASSPORT_NUMBER, LUGGAGE_NUMBER, LUGGAGE_TYPE, COLOR, WEIGHT, TICKET_NUMBER)

VALUES ('TR437918', 20403000, 'Personal item’, ‘White', 5, '20200609AAL29862')

- **INTO LUGGAGE** (PASSPORT_NUMBER, LUGGAGE_NUMBER, LUGGAGE_TYPE, COLOR, WEIGHT, TICKET_NUMBER)

VALUES ('TX866394', 20408410, 'Suitcase’, 'Grey', 36, '20200919AAL27164')

- **INTO LUGGAGE** (PASSPORT_NUMBER, LUGGAGE_NUMBER, LUGGAGE_TYPE, COLOR, WEIGHT, TICKET_NUMBER)

VALUES ('NY926392', 20111020, 'Carry-on’, 'Brown', 10, ‘20200609AAL29852')

- **INTO LUGGAGE** (PASSPORT_NUMBER, LUGGAGE_NUMBER, LUGGAGE_TYPE, COLOR, WEIGHT, TICKET_NUMBER)

VALUES ('ZH771661', 20403111, 'Suitcase’, 'Blue', 29, '20200420MEA869823')

- **INTO LUGGAGE** (PASSPORT_NUMBER, LUGGAGE_NUMBER, LUGGAGE_TYPE, COLOR, WEIGHT, TICKET_NUMBER)

VALUES ('PR735932', 20000011, 'Suitcase’, 'White', 14, '20200901ORY741482')

- **INTO LUGGAGE** (PASSPORT_NUMBER, LUGGAGE_NUMBER, LUGGAGE_TYPE, COLOR, WEIGHT, TICKET_NUMBER)

VALUES ('LD424925', 20478401, 'Suitcase’, 'Black’, 33, '20200426LOV690364')

- **INTO LUGGAGE** (PASSPORT_NUMBER, LUGGAGE_NUMBER, LUGGAGE_TYPE, COLOR, WEIGHT, TICKET_NUMBER)

VALUES ('LD102031', 20200541, 'Carry-on’, 'Blue', 6, '20201015LHR763019')

- **INTO LUGGAGE** (PASSPORT_NUMBER, LUGGAGE_NUMBER, LUGGAGE_TYPE, COLOR, WEIGHT, TICKET_NUMBER)

VALUES ('TR764912', 21111574, 'Personal item', 'Grey', 4, '20200309MEA864008')

- **INTO LUGGAGE** (PASSPORT_NUMBER, LUGGAGE_NUMBER, LUGGAGE_TYPE, COLOR, WEIGHT, TICKET_NUMBER)

VALUES ('BR102030', NULL, 'Carry-on', 'Black', NULL, '20201013MEA325001')

SELECT*from dual;

4. **RUNWAY:**

INSERT ALL

INTO RUNWAY (RUNWAY_ID, RUNWAY_LENGTH, ISAVAILABLE)

VALUES ('36C', 2500,1)

INTO RUNWAY (RUNWAY_ID, RUNWAY_LENGTH, ISAVAILABLE)

VALUES ('36L', 3500,0)

INTO RUNWAY (RUNWAY_ID, RUNWAY_LENGTH, ISAVAILABLE)

VALUES ('36R', 2000,1)

INTO RUNWAY (RUNWAY_ID, RUNWAY_LENGTH, ISAVAILABLE)

VALUES ('18C', 4000,1)

INTO RUNWAY (RUNWAY_ID, RUNWAY_LENGTH, ISAVAILABLE)

VALUES ('18L', 1800,1)

INTO RUNWAY (RUNWAY_ID, RUNWAY_LENGTH, ISAVAILABLE)

VALUES ('18R', 2800,1)

INTO RUNWAY (RUNWAY_ID, RUNWAY_LENGTH, ISAVAILABLE)

VALUES ('09C', 3000,1)

INTO RUNWAY (RUNWAY_ID, RUNWAY_LENGTH, ISAVAILABLE)

VALUES ('09L', 3500,1)

INTO RUNWAY (RUNWAY_ID, RUNWAY_LENGTH, ISAVAILABLE)

VALUES ('09R', 2300,1)

INTO RUNWAY (RUNWAY_ID, RUNWAY_LENGTH, ISAVAILABLE)

VALUES ('27C', 2800,1)

INTO RUNWAY (RUNWAY_ID, RUNWAY_LENGTH, ISAVAILABLE)

VALUES ('27L', 1500,1)

INTO RUNWAY (RUNWAY_ID, RUNWAY_LENGTH, ISAVAILABLE)

VALUES ('27R', 2800,1)

5. **SHIPPING_COMPANY:**

INSERT ALL

- **INTO SHIPPING_COMPANY** (ICAO_CODE, NAME, COUNTRY, NUM_OF_AIRCRAFTS)

VALUES ('FDX','FEDEx Express','USA',692)

- **INTO SHIPPING_COMPANY** (ICAO_CODE, NAME, COUNTRY, NUM_OF_AIRCRAFTS)

VALUES ('UPS','United Parcel Service','USA', 274)

- **INTO SHIPPING_COMPANY** (ICAO_CODE, NAME, COUNTRY, NUM_OF_AIRCRAFTS)

VALUES ('DHL','DHL Aviation','USA',250)

- I**NTO SHIPPING_COMPANY** (ICAO_CODE, NAME, COUNTRY, NUM_OF_AIRCRAFTS)

VALUES ('QTR','Qatar Airways Cargo','Qatar', 27)

- **INTO SHIPPING_COMPANY** (ICAO_CODE, NAME, COUNTRY, NUM_OF_AIRCRAFTS)

VALUES ('UAE','Emirates SkyCarg','UAE', 11)

- **INTO SHIPPING_COMPANY** (ICAO_CODE, NAME, COUNTRY, NUM_OF_AIRCRAFTS)

VALUES ('GEC','Lufthansa Cargo','Germany', 15)

- **INTO SHIPPING_COMPANY** (ICAO_CODE, NAME, COUNTRY, NUM_OF_AIRCRAFTS)

VALUES ('AZG','Azgard Cargo','Azgard', 800)

6. **PACKAGE:**

INSERT ALL

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('LBFRDHL152485', 'LBWA152485', 'DHL', 'Glass object', 1, 0.85, 0.40, 0.12, 700, 'BEY', 'BZG', 120, 'A glass vase, flower print on it')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('DZUSUPS841254', 'DZWA841254', 'UPS', 'Paper', 0, 0.30, 0.30, 0.0001, 250, 'ALG', 'BZG', 100, 'An blue envelope with papers inside')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('NOUKAZG121496', 'WAUK121496', 'AZG', 'Electronic device', 0, 0.2, 0.1, 0.01, 200, 'BZG', 'LCY', 1800, 'A white iPhone 12 box')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('ARBSDHL005816', 'ARWA005816', 'DHL ','Instrument', 1, 1.2, 0.8, 1.1, 105, 'BAI', 'BZG', 4000, 'A piano assembling parts')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('CYDKQTR000086', 'WADK000086', 'QTR', 'Electronic device', 1, 0.49, 0.25, 0.06, 3000, 'BZG', 'STK', 2300, 'An brown Hp computer corton')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('DOEGUAE667810', 'DOWA667810', 'UAE', 'Ceiled Box', 1, 0.80, 0.65, 0.60, 65000, 'JLK', 'BZG', 250, 'A box with christmas decoration')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('FJGEGEC140297', 'WAGE140297', 'GEC', 'Machine', 1, 1.2, 0.92, 1.0, 75000, 'BZG', 'NUE', 600, 'An LG washing machine box')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('HUISFDX119512', 'HUWA119512', 'FDX', 'Medical supplies' , 0, 1, 1, 1, 92000, 'LQA', 'BZG', 320 , 'A box of blue medical masks')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('JOJPFDX991570', 'WAJP991570', 'FDX', 'Ceiled box', 0, 1.5, 0.9, 0.8, 2500, 'BZG', 'TOK', 85, 'A Bicycle parts before assembling')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('HKGUGEC119703', 'WAGU119703', 'GEC', 'Medical supplies', 1, 1.5, 1.5, 1.5, 12000, 'BZG','NAK', 100000, 'A first ship of COVID-19 vaccine')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('BRCAQTR318420', 'WACA318420', 'QTR', 'Document', 0, 0.4, 0.12, 0.01, 0.75, 'BZG', 'GTY', 1000, 'A yellow stamped envelope')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('TFLBAZG158412', 'TFWA158412', 'AZG', 'Electronic device', 1, 0.8, 0.75, 0.06, 4000, 'TRO', 'BZG', 2500, 'An Apple iPod air box')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('TFLBUPS158002', 'WALB158002', 'UPS', 'Electronic device', 1, 0.8, 0.75, 0.06, 4000, 'BZG','ROM', 2500, 'An Apple iPod air 2 box')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('FRBGUAE441590', 'FRWA441590', 'UAE', 'Glass object', 1, 0.8, 0.75, 0.006, 9000, 'CDG','BZG', 1680, 'A big decorative mirror with golden frame')

SELECT * FROM dual;

7. **MEDIA_ENTERTAINMENT**

INSERT ALL

- **INTO MEDIA_ENTERTAINMENT** (Flight_ Number, Media_Type, Genre, Title)
VALUES ('MEA325','Music','Alternative Rock', 'Live to Rise')

- **INTO MEDIA_ENTERTAINMENT** (Flight_ Number, Media_Type, Genre, Title)
VALUES ('MEA325','Music','Rock','Im Alive')

- **INTO MEDIA_ENTERTAINMENT** (Flight_ Number, Media_Type, Genre, Title)
VALUES ('MEA325','Music','Classic','Dirt and Roses')

- **INTO MEDIA_ENTERTAINMENT** (Flight_ Number, Media_Type, Genre, Title)
VALUES ('MEA325','Movie','Action','Guardians of the Galaxy')
- **INTO MEDIA_ENTERTAINMENT** (Flight_ Number, Media_Type, Genre, Title)
VALUES ('LOV690','Music','Rock','Live to Rise')

- **INTO MEDIA_ENTERTAINMENT** (Flight_ Number, Media_Type, Genre, Title)
VALUES ('LOV690','Music','Rock','Im Alive')

- **INTO MEDIA_ENTERTAINMENT** (Flight_ Number, Media_Type, Genre, Title)
VALUES ('LOV690','Music','Heavy Metal', 'Even If I Could')

- **INTO MEDIA_ENTERTAINMENT** (Flight_ Number, Media_Type, Genre, Title)
VALUES ('LOV690','Movie','Mystery','The Eternals')

- **INTO MEDIA_ENTERTAINMENT** (Flight_ Number, Media_Type, Genre, Title)
VALUES ('AAL298','Music','Alternative rock', 'Live to Rise')

- **INTO MEDIA_ENTERTAINMENT** (Flight_ Number, Media_Type, Genre, Title)
VALUES ('AAL298','Music','Opera','Red Ledger')

- **INTO MEDIA_ENTERTAINMENT** (Flight_ Number, Media_Type, Genre, Title)

VALUES ('AAL298','Music','Jazz','They Called It')

INTO MEDIA_ENTERTAINMENT (Flight_ Number, Media_Type, Genre, Title)
VALUES ('AAL298','Movie','Suspense','Avengers: Endgame')

INTO MEDIA_ENTERTAINMENT (Flight_ Number, Media_Type, Genre, Title)
VALUES ('LUT948','Movie','Thriller','Thor: The Dark World')

INTO MEDIA_ENTERTAINMENT (Flight_ Number, Media_Type, Genre, Title)
VALUES ('LUT948','Movie','Action','Guardians of the Galaxy')

INTO MEDIA_ENTERTAINMENT (Flight_ Number, Media_Type, Genre, Title)
VALUES ('LUT948','Movie','Suspense','Avengers: Endgame')

INTO MEDIA_ENTERTAINMENT (Flight_ Number, Media_Type, Genre, Title)
VALUES ('LUT948','Music','Opera','Red Ledger')

INTO MEDIA_ENTERTAINMENT (Flight_ Number, Media_Type, Genre, Title)
VALUES ('LBR931','Movie','Drama','Octor Strange')

INTO MEDIA_ENTERTAINMENT (Flight_ Number, Media_Type, Genre, Title)
VALUES ('LBR931','Movie','Action','Guardians of the Galaxy')

INTO MEDIA_ENTERTAINMENT (Flight_ Number, Media_Type, Genre, Title)
VALUES ('LBR931','Movie','Dark Comedy', 'Shang-Chi and the Legend of the Ten Rings')

INTO MEDIA_ENTERTAINMENT (Flight_ Number, Media_Type, Genre, Title)
VALUES ('AFL666','Movie','Fiction','Doctor Strange in the Multiverse of Madness')

INTO MEDIA_ENTERTAINMENT (Flight_ Number, Media_Type, Genre, Title)
VALUES ('AFL666','Movie','Action','Guardians of the Galaxy')

INTO MEDIA_ENTERTAINMENT (Flight_ Number, Media_Type, Genre, Title)

VALUES ('AFL666','Movie','Mystery','The Eternals')

- **INTO MEDIA_ENTERTAINMENT** (Flight_ Number, Media_Type, Genre, Title)
VALUES ('AFL666','Music','Rock','Im Alive')

- **INTO MEDIA_ENTERTAINMENT** (Flight_ Number, Media_Type, Genre, Title)
VALUES ('TAP717','Movie','Fiction','Doctor Strange in the Multiverse of Madness')

- **INTO MEDIA_ENTERTAINMENT** (Flight_ Number, Media_Type, Genre, Title)
VALUES ('TAP717','Movie','Action','Guardians of the Galaxy')

- **INTO MEDIA_ENTERTAINMENT** (Flight_ Number, Media_Type, Genre, Title)
VALUES ('TAP717','Movie','Mystery','The Eternals')

- **INTO MEDIA_ENTERTAINMENT** (Flight_ Number, Media_Type, Genre, Title)
VALUES ('TAP717','Music','Alternative Rock', 'Live to Rise')

SELECT * FROM dual;

8. **FOOD_SERVICE**

INSERT ALL

- **INTO FOOD_SERVICE** (Flight_Number, Meal_Name, Cuisine, Description)
VALUES ('MEA325','Rice Pilaf', 'Lebanon', 'Rice cooked with a little flavorful oil and chopped onion, and then simmered with broth ')

- **INTO FOOD_SERVICE** (Flight_Number, Meal_Name, Cuisine, Description)
VALUES ('LOV690','Bouillabaisse','France','Seafood stew made with fresh bony fish, shellfish, mussels, vegetables, and herbs ')

- **INTO FOOD_SERVICE** (Flight_Number, Meal_Name, Cuisine, Description)
VALUES ('AAL298','drop biscuits and sausage gravy', 'United States', 'soft dough biscuits covered in meat gravy')

- **INTO FOOD_SERVICE** (Flight_Number, Meal_Name, Cuisine, Description)

VALUES ('LUT948','kapenta with sadza', 'Cote Divoire ',' Small freshwater fish paired with maize porridge ')

- **INTO FOOD_SERVICE** (Flight_Number, Meal_Name, Cuisine, Description)

VALUES ('LBR931','Chicken tikka masala', 'India', 'chunks of roasted marinated chicken in a spiced curry. Rice cooked with a little flavorful oil and chopped onion, and then simmered with broth ')

- **INTO FOOD_SERVICE** (Flight_Number, Meal_Name, Cuisine, Description)

VALUES ('AFL666','Pelmeni','Russia','pastry dumplings filled with minced meat and wrapped in a dough')

INTO FOOD_SERVICE (Flight_Number, Meal_Name, Cuisine, Description)

VALUES ('TAP717','Acorda','Potuguese','Siced bread with garlic, chopped coriander, and poached eggs.')

SELECT * FROM dual;

9. **SHOPS_IN**

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)

VALUES ('Vintage','PR735932')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)

VALUES ('Vintage','LD424925')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)

VALUES ('Vintage','TR764912')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)

VALUES ('Disney Store','NY926392')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
VALUES ('Vintage','NY926392')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
VALUES ('Hersheys Chocolate World','NY926392')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
VALUES ('Swarovski','TX866394')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
VALUES ('Swarovski','BR102030')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
VALUES ('Hersheys Chocolate World','BR102030')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
VALUES ('Invicta Store','TX866394')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
VALUES ('Invicta Store','TR437918')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
VALUES ('TISSOT','TR437918')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
VALUES ('TISSOT','TX866394')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
VALUES ('TISSOT','TR437918')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
VALUES ('TISSOT','OR484363')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
VALUES ('TISSOT','BR102030')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
VALUES ('TISSOT','BR102030')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
VALUES ('Swatch','OR484363')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
VALUES ('Swatch','AB972922')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
VALUES ('Swatch','BR102030')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
VALUES ('Swatch','OR484363')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
VALUES ('M&MS World','MM425012')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
VALUES ('MUJI','BE484363')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
VALUES ('Levis Store','MM289371')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
VALUES ('Express','BE484368')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
VALUES ('Express','BE484363')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
  VALUES ('Express','MO436262')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
  VALUES ('Benihana','BE484368')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
  VALUES ('Gap','BE484363')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
  VALUES ('Spiritland','MO436262')

- **INSERT INTO SHOPS_IN** (STORE_NAME, PASSPORT_NUMBER)
  VALUES ('Harman','MO436262')

10. TECHNICAL_SERVICE_PROVIDER

INSERT ALL
- **INTO TECHNICAL_SERVICE_PROVIDER** (SP_NAME, COUNTRY, CITY)
VALUES ('Synairgia', 'Bulgaria', 'Sofia')

- **INTO TECHNICAL_SERVICE_PROVIDER** (SP_NAME, COUNTRY, CITY)
VALUES ('Purvis Brothers', 'USA', 'Valencia RD')

- **INTO TECHNICAL_SERVICE_PROVIDER** (SP_NAME, COUNTRY, CITY)
VALUES ('Jett Pro Line', 'USA', 'Oakland')

- **INTO TECHNICAL_SERVICE_PROVIDER** (SP_NAME, COUNTRY, CITY)
VALUES ('AVtech', 'USA', 'Denver')

- **INTO TECHNICAL_SERVICE_PROVIDER (**SP_NAME, COUNTRY, CITY)
VALUES ('Badger Aero', 'USA', 'Port Washington')

- **INTO TECHNICAL_SERVICE_PROVIDER** (SP_NAME, COUNTRY, CITY)

VALUES ('J&B', 'USA', 'Palmetto')

- **INTO TECHNICAL_SERVICE_PROVIDER** (SP_NAME, COUNTRY, CITY)

VALUES ('Island aviation', 'Maldives', 'Male')

- **INTO TECHNICAL_SERVICE_PROVIDER** (SP_NAME, COUNTRY, CITY)

VALUES ('ScandinavianAVS', 'Sweden', 'Stockholm')

SELECT * FROM dual;

11. EMPLOYEE

INSERT ALL

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SUPERVISOR, DEPARTMENT_ID, SALARY)

VALUES ('01255', 'Saul', 'Declan', 'Bloom', DATE '1980-10-15', 'M', '00255655752470', 'S.Bloom@WAirport.wa','01050',1,5000)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SUPERVISOR, DEPARTMENT_ID, SALARY)

VALUES ('01370', 'Ronald', 'Charley', 'Hodgson', DATE '1985-12-18', 'M', '00255470062327', 'R.Hodgson@WAirport.wa','01050',1,3000)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SUPERVISOR, DEPARTMENT_ID, SALARY)

VALUES ('01050', 'Eilidh', 'Mikayla', 'Tang', DATE '1991-06-22', 'F', '00255699095155', 'E.Tangn@WAirport.wa','01030',1,10000)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SUPERVISOR, DEPARTMENT_ID, SALARY)

VALUES ('01030', 'Ruqayyah', 'Selin', 'Wicks', DATE '1993-04-18', 'F', '00255287310459', 'R.Wicks@WAirport.wa', NULL,1,12000)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SUPERVISOR, DEPARTMENT_ID, SALARY) VALUES ('02870', 'Tyson', 'Dani', 'Childs', DATE '1985-07-20', 'M', '00255156951614', 'T.Childs@WAirport.wa','02550',2,3500)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SUPERVISOR, DEPARTMENT_ID, SALARY) VALUES ('02880', 'Savanna', 'Lula', 'Martinez', DATE '1985-12-18', 'F', '00255857775127', 'S.Martinez@WAirport.wa','02550',2,7500)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SUPERVISOR, DEPARTMENT_ID, SALARY) VALUES ('02550', 'Rhona', 'Vickie', 'Adamson', DATE '1994-08-05', 'F', '00255972549338', 'R.Adamson@WAirport.wa','03244',2,15000)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SUPERVISOR, DEPARTMENT_ID, SALARY) VALUES ('03244', 'Ellenor', 'Gurpreet','Andrew', DATE '1982-04-9', 'F', '00255829667051', 'E.Andrew@WAirport.wa', NULL,2,20000)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SUPERVISOR, DEPARTMENT_ID, SALARY) VALUES ('03470', 'Anoushka', 'Timur', 'Francis', DATE '1996-01-12', 'M', '00255248822756', 'A.Francis@WAirport.wa','03111',3,4500)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SUPERVISOR, DEPARTMENT_ID, SALARY) VALUES ('03111', 'Burhan', 'Annie', 'Kent', DATE '1980-03-12', 'F', ' 00255103911777', 'B.Kent@WAirport.wa','03120',3,10000)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SUPERVISOR, DEPARTMENT_ID, SALARY) VALUES ('03120', 'Danyaal', 'Timur', 'Robins', DATE '1992-03-20', 'M', '00255351343249', 'D.Robins@WAirport.wa', NULL,3,10000)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SUPERVISOR, DEPARTMENT_ID, SALARY) VALUES ('11120', 'Rhodri', 'Penelope', 'Rios', DATE '1975-04-22', 'M', '00255889789600', 'R.Rios@MEA.lb',10000)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SUPERVISOR, DEPARTMENT_ID, SALARY) VALUES ('11330', 'Thor', 'Alexander', 'Odin', DATE '2000-12-15', 'M', '00255658003182', 'T.Odin@MEA.lb',3000)
- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SUPERVISOR, DEPARTMENT_ID, SALARY) VALUES ('12133', 'Anisah', 'Kain', 'Ramos', DATE '1988-04-30', 'F', '00255319770450', 'A.Ramos@LOV.uk',13000)
- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SUPERVISOR, DEPARTMENT_ID, SALARY) VALUES ('12122', 'Cerys', 'Beatric', 'Villalobos', DATE '1993-10-15', 'M', '00255317000648', 'C.Villalobos@LOV.uk',3300)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SUPERVISOR, DEPARTMENT_ID, SALARY) VALUES ('20133', 'Gillian', 'Joni', 'Barker', DATE '1990-03-25', 'M', '00255341411788', 'D.Robins@LUT.de',35000)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SUPERVISOR, DEPARTMENT_ID, SALARY) VALUES ('20223', 'Rebekah', 'Erica', 'Collins', DATE '1994-04-25', 'F', '00255299313675', 'R.Collins@LUT.de',25000)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SUPERVISOR, DEPARTMENT_ID, SALARY) VALUES ('11332', 'Garry', 'Storm', 'Higgs', DATE '1990-03-25', 'M', '00255341411788', 'G.Higgs@QTR.qa',20000)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SUPERVISOR, DEPARTMENT_ID, SALARY

VALUES ('13133', 'Tamara', 'Malia', 'Bender', DATE '1973-03-16', 'F', '00255321788165', 'T.Bender@QTR.qa',33000)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SUPERVISOR, DEPARTMENT_ID, SALARY)

VALUES ('14133', 'Mahir', NULL, 'Usman', DATE '1990-05-25', 'M', '00255393514258', 'M.Usman@UPS.us',34500)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SUPERVISOR, DEPARTMENT_ID, SALARY)

VALUES ('14123', 'Polly', 'Giovann', 'Farrow', DATE '1973-06-16', 'F', '00255226299123', 'P.Farrow@UPS.us',43200)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SUPERVISOR, DEPARTMENT_ID, SALARY)

VALUES ('22133', 'Vernon', 'Todd', 'McDaniel', DATE '1986-12-25', 'M', '00255357810125', 'V.Mcdaniel@DHL.us',42120)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SUPERVISOR, DEPARTMENT_ID, SALARY)

VALUES ('22123', 'Elena', 'Luella', 'Walls', DATE '1973-11-16', 'F', '00255532367394', 'E.Walls@DHL.us',31230)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('15122', 'Argog', 'Arko', 'Aegir', DATE '1300-12-25', 'M', '00255421063129', 'A.Aegir@AZG.az',12340)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('15332', 'Angerboda', 'Bestla', 'Brynhildr', DATE '1222-11-22', 'F', '002555374118150', 'A.Brynhildr@AZG.az',32120)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('20122', 'Argog', 'Arko', 'Aegir', DATE '1300-12-25', 'M', '00255421063129', 'A.Aegir@AZG.az',12340)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('20332', 'Angerboda', 'Bestla', 'Brynhildr', DATE ’1222-11-22', 'F', '002555374118150’, ‘A.Brynhildr@AZG.az',32120)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('19122', 'Argog', 'Arko', 'Aegir', DATE '1300-12-25', 'M', '00255421063129', 'A.Aegir@AZG.az',33340)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('19332', 'Angerboda', 'Bestla', 'Brynhildr', DATE ’1222-11-22', 'F', '002555374118150’, ‘A.Brynhildr@AZG.az',23450)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('16992', 'Jaeden', 'Jorden', 'Woodcock', DATE ’1887-07-17', 'M', '00255278976176’, ‘J.Woodcock@GEC.de',23450)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('16832', 'Lexie', 'Rhiana', 'Knight', DATE '1988-09-10', 'F', '00255243669600', 'L.Knight@GEC.de',23450)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('17532', 'Jaeden', 'Jorden', 'Woodcock', DATE ’1987-07-17', 'M', '00255278976176', 'J.Woodcock@GEC.de',23450)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('17121', 'Lexie', 'Rhiana', 'Knight', DATE '1988-09-10', 'F', '00255243669600', 'L.Knight@GEC.de',23450)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('18742', 'Todd', 'Marshall', 'Guzman', DATE '1985-02-17', 'M', '00255956063962', 'T.Guzman@UAE.ae',23450)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('18852', 'Nial', 'Vernon', 'Maddox', DATE '1994-07-19', 'F', '00255355757150', 'N.Maddox@UAE.ae',23450)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('26055', 'Ewen ', 'Ray',' Olson', DATE '1984-11-13', 'F', '00255701934484', 'E.Olson@AAL.us',750)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('26232', 'Nour ', 'Jay', 'Marvin', DATE '1975-01-11', 'F', '00255871672535', 'N.Marvin@AAL.us',1750)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('26845', 'Kamran', 'Alex', 'Mason', DATE '1979-12-07', 'M', '00255796385009', 'K.Mason@AAL.us',860)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('28099', 'Fionnuala', 'Issac', 'Malone', DATE '1974-11-13', 'M', '00255701934484', 'F.Malone@LBR.az',1400)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('28100', 'Nyla', 'Greg', 'Perez', DATE '1974-07-23', 'F', '00255349811429', 'N.Greg@LBR.az',1000)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('28543', 'Edna', 'Elle', 'Hope', DATE '1991-02-01', 'M', '00255213204953', 'E.Hope@LBR.az',2300)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('28082', 'Brody', 'Issac', 'Hunter', DATE '1971-11-13', 'M', '00255909806956', 'B.Hunter@AFL.ru',1350)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('28102', 'Dominick', 'Rudd', 'Shaun', DATE '1971-04-27', 'M', '00255602080433', 'D.Rudd@AFL.ru',2650)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('28356', 'Tony', 'Tarik', 'Laing', DATE '1989-04-09', 'M', '00255628582115', 'T.Laing@AFL.ru',950)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('29084', 'Huxley', 'Eugene', 'Herman', DATE '1970-05-11', 'F', '00255590724505', 'H.Herman@TAP.pt',1650)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('29321', 'Zakary', 'Sanjay', 'Harvey', DATE '1989-08-17', 'F', '002556897658901', 'Z.Sanjar@TAP.pt',950)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('29004', 'Kain', 'Fleur', 'Singh', DATE '1970-05-11', 'M', '00255314498005', 'K.Singh@TAP.pt',1250)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('24375', 'William', 'Calum','Stanley', DATE '1987-11-22', 'M', '00255771161631', 'W.MStanley@MEA.lb',32320)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('24022', 'Alaya', NULL,' Neale', DATE '1984-08-12', 'F', '00255523097593', 'A.MNeale@MEA.lb',2320)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('24033', 'Nelly', 'Raees', 'Britton', DATE '1989-12-11', 'M', '00255252213337', 'N.Britton@MEA.lb',15120)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('25213', 'Laith', 'Reon', 'Dolan', DATE '1980-09-15', 'M', '00255140520656', 'Laith.Dolan@LOV.it',32120)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('25105', 'Bear', 'Maci', 'Bright', DATE '1986-11-28', 'M', '00255132586448', 'B.Bright@LOV.it',43120)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('25432', 'Gillian', 'Poppy ','Christie', DATE '1986-10-12', 'M', '00255371387752', 'G.Christie@LOV.it',21130)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('31032', 'Jaidon', 'Cinar', 'Faulkner', DATE '1992-06-21', 'M', '00255439830733', 'J.Faulkner@Synairgia.com',23020)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('31045', 'Mehdi', NULL, 'Mustafa', DATE '1992-09-12', 'M', '00255860864615', 'M.Mustafa@Synairgia.com',32400)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('38045', 'Firat', 'Ziggy', 'Truonga', DATE '1990-03-06', 'M', '00255320620027', 'F.Truonga@PurvisB.com',23820)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('38012', ' Haleemah', NULL,' Hendricks', DATE '1990-12-12', 'F', '00255475011924',' H.Hendricks@PurvisB.com',32420)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('32045', 'Anis', 'Orson', 'Hirst', DATE '1977-08-12', 'M', '00255716203317', 'A.Hirst@JetLine.com',42130)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('32012', 'Rhiann', NULL, 'Wagstaff', DATE '1980-08-17', 'F', '00255113414898', 'W.Wagstaff@JetLine.com',42620)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('33065', 'Anis', 'Orson', 'Hirst’, DATE '1977-08-12', 'M', '00255716203317', 'A.Hirst@AVtech.com', 42130)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('33204', 'Rhiann', NULL, 'Wagstaff', DATE '1980-08-17', 'F', '00255113414898', 'R.Wagstaff@AVtech.com', 42620)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('34075', 'Murtaza', 'Dane', 'Newton', DATE '1979-09-30', 'M', '00255756091468', 'M.Dane@BadgerAero.com', 42130)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('34205', 'Jamila', NULL, 'Neville', DATE '1983-09-11', 'F', '00255636038252', 'J.Neville@BadgerAero.com', 42620)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('35022', 'Ayub', 'Iylah', 'Monaghan', DATE '1973-07-23', 'M', '00255587020118', 'A. Monaghan@J&B.com',52380)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('35086', 'Asiya', NULL,'Goddard', DATE '1996-07-12', 'F', '00255219262951','A. Goddard@J&B.com',32340)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('36098', 'Barney', 'Allan', 'Rigby', DATE '1978-02-13', 'M', '00255607153673',' B.Rigby@IslandA.com', 32460)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('36022', 'Alfie', 'Alfie', 'Goddard', DATE '1992-03-17', 'M', '00255944079085', 'A.Goddard@IslandA.com', 12350)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('37054', 'Tariq', NULL, 'Day', DATE '1991-03-16', 'M', '00255688044454', 'T.Day@ScandinavianAVS.com',20340)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('37086', 'Aleisha', 'Gordon', 'Allan', DATE '1993-02-10', 'F', '00255831529187', 'A.Allan@ScandinavianAVS.com',15080)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('41023', 'Kenzie', 'Berat', 'George', DATE '1978-11-12', 'M', '00255734469543',' K.George@Bartartin.com',2380)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('42145', 'Faiza', 'Harrell', 'Steadman', DATE '1978-01-21', 'F', '00255526675063',' F.Steadman@Benihana.com',3450)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('43202', 'Benjamin', 'Kenzie', 'Pham', DATE '1994-12-11', 'M', '00255981968660',' B.Pham@DisneyStore.com',3450)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('44231', 'Lori', 'Berat', 'Zavala', DATE '1998-09-22', 'F', '00255483969147', 'L.Zavala@Express.com',1450)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('45061', 'Alton', 'Lennox', 'Leech', DATE '1996-12-21', 'M', '00255756091468',' A.Leech@Gap.com',2150)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('46082', 'Domonic', 'Kiara', 'Huber', DATE '1996-06-20', 'F', '00255944079085', 'D.Huber@Harman.com',3450)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('47092', 'Zavier', null, 'Murat', DATE '1996-01-21','M', '00255636038252', 'Z.Murat@Hersheys.com',2340)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('48265', 'Dania', null, 'Mccabe', DATE '1991-02-01','F', '00255248460481', 'Z.Murat@Invicta.com',1250)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('49056', 'Etienne', 'Dolcie','Iles', DATE '1998-01-02','M', '00255430502245', 'E.Iles@Jumbo.com',2450)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('50021', 'Joe', 'Matias', 'Beaumont', DATE '1995-12-15','M', '00255639887023', 'J.Beaumont@Levis.com',1780)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('51025', 'Laylah', 'Rueben', 'York', DATE '1995-09-23','F', '00255617661135', 'L.York@Lora.com',2350)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('52034', 'Fabien', 'Franklin', 'Reid', DATE '1998-05-21','M', '00255734106505', 'F.Reid@M&MS.com',2340)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('53124', 'Tasnia', 'Eesha', 'Bowden', DATE '1996-07-07','F', '00255672910749', 'T.Bowdend@MUJI.com',3240)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('54032', 'Sylvia', 'Morin', 'Person', DATE '1995-08-12','F', '00255160863343', 'T.Bowdend@Sephora.com',1790)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('55012', 'Riaan', 'Zuniga', 'York', DATE '1996-07-07','F', '00255558601718', 'R.York@Spiritland.com',1890)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('56090', 'Chace', NULL, 'Peacock', DATE '1996-01-12','F', '00255794819336', 'C.Peacock@SunglassHut.com',1230)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('57101', 'Meerab', 'Byers', 'Healy', DATE '1997-05-12','M', '00255525658203', 'M.Healy@Swarovski.com',2550)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('58120', 'Carina', 'Tara', 'Macgregor', DATE '1996-12-02','F', '00255993564972', 'C.Macgregor@Swatch.com',2760)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('59090', 'Faizaan', NULL, 'Timms', DATE '1997-11-12','M', '00255554572672', 'F.Timms@TISSOT.com',2430)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('60020', 'Syeda', NULL, 'Crawford', DATE '1997-10-25','F', '00255714854367', 'S.Crawford@Vintage.com',3450)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('61250', 'Solomon', 'Alford', 'Cornish', DATE '1990-03-12','M', '00255714854367', 'S.Cornish@VirginMegastore.com',2340)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('62045', 'Aida', 'Viola', 'Walton', DATE '1997-11-12','F', '00255262101418', 'A.Walton@WakandaLibrary.com',5340)

- **INTO EMPLOYEE** (EMPLOYEE_ID, FNAME, MNAME, LNAME, DATE_OF_BIRTH, GENDER, PHONE_NUMBER, EMAIL, SALARY)

VALUES ('63021', 'Chandler', 'Kirstin', 'Frye', DATE '1991-09-12','M', '002555121067644', 'C.Frye@flakeshake.com',1230)

SELECT * FROM dual;

12. AIRPLANE

SHIPPING + COMMERCIAL

INSERT ALL

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, PASSENGER_CAPACITY, FUEL_TANK_CAPACITY, AIRLINE_ICAO_CODE)

VALUES ('MEA-843', 'Commercial', 'Boeing', 'A220-300', 2012, 68, 160, 5681, 'MEA')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, PASSENGER_CAPACITY, FUEL_TANK_CAPACITY, AIRLINE_ICAO_CODE)

VALUES ('MEA-905', 'Commercial', 'Boeing', 'A319', 1996, 76, 156, 7980, 'MEA')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, PASSENGER_CAPACITY, FUEL_TANK_CAPACITY, AIRLINE_ICAO_CODE)

VALUES ('MEA-985', 'Commercial', 'Boeing', 'A330', 1992, 233, 440, 36743, 'MEA')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, PASSENGER_CAPACITY, FUEL_TANK_CAPACITY, AIRLINE_ICAO_CODE)

VALUES ('LOV-343', 'Commercial', 'Antonov', 'An-158', 2010, 43, 99, 4302, 'LOV')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, PASSENGER_CAPACITY, FUEL_TANK_CAPACITY, AIRLINE_ICAO_CODE)

VALUES ('LOV-745', 'Commercial', 'Antonov', 'An-148', 2004, 44, 85, 3929, 'LOV')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, PASSENGER_CAPACITY, FUEL_TANK_CAPACITY, AIRLINE_ICAO_CODE)

VALUES ('LOV-645', 'Commercial', 'Boeing', '777-200LR', 1993, 317, 347, 47894, 'LOV')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, PASSENGER_CAPACITY, FUEL_TANK_CAPACITY, AIRLINE_ICAO_CODE)

VALUES ('AAL-735', 'Commercial', 'Boeing', '737-100', 1967, 49, 124, 6875, 'AAL')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, PASSENGER_CAPACITY, FUEL_TANK_CAPACITY, AIRLINE_ICAO_CODE)

VALUES ('AAL-654', 'Commercial', 'Boeing', '747-400', 1988, 396, 416, 63032, 'AAL')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, PASSENGER_CAPACITY, FUEL_TANK_CAPACITY, AIRLINE_ICAO_CODE)

VALUES ('AAL-494', 'Commercial', 'Bombardier', 'CRJ705', 1999, 38, 75, 2898, 'AAL')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, PASSENGER_CAPACITY, FUEL_TANK_CAPACITY, AIRLINE_ICAO_CODE)

VALUES ('LUT-166', 'Commercial', 'Comac', 'ARJ21', 2007, 43, 90, 4832, 'LUT')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, PASSENGER_CAPACITY, FUEL_TANK_CAPACITY, AIRLINE_ICAO_CODE)

VALUES ('LUT-167', 'Commercial', 'Embraer', 'ERJ-135', 1999, 19, 30, 1711, 'LUT')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, PASSENGER_CAPACITY, FUEL_TANK_CAPACITY, AIRLINE_ICAO_CODE)

VALUES ('LUT-168', 'Commercial', 'Embraer', 'ERJ-145', 1992, 22, 42, 1359, 'LUT')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, PASSENGER_CAPACITY, FUEL_TANK_CAPACITY, AIRLINE_ICAO_CODE)

VALUES ('LBR-900', 'Commercial', 'Ilyushin', 'Il-96-300', 1992, 250, 300, 40317, 'LBR')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, PASSENGER_CAPACITY, FUEL_TANK_CAPACITY, AIRLINE_ICAO_CODE)

VALUES ('LBR-901', 'Commercial', 'Ilyushin', 'Il-114Il', 1992, 23, 64, 8360, 'LBR')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, PASSENGER_CAPACITY, FUEL_TANK_CAPACITY, AIRLINE_ICAO_CODE)

VALUES ('LBR-902', 'Commercial', 'Embraer', 'E-175', 2002, 38, 78, 3044, 'LBR')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, PASSENGER_CAPACITY, FUEL_TANK_CAPACITY, AIRLINE_ICAO_CODE)

VALUES ('AFL-888', 'Commercial', 'Sukhoi', 'SSJ-100', 2000, 45, 98, 6932, 'AFL')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, PASSENGER_CAPACITY, FUEL_TANK_CAPACITY, AIRLINE_ICAO_CODE)

VALUES ('AFL-878', 'Commercial', 'Tupolev', 'Tu-204', 1990, 103, 210, 13371, 'AFL')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, PASSENGER_CAPACITY, FUEL_TANK_CAPACITY, AIRLINE_ICAO_CODE)

VALUES ('AFL-868', 'Commercial', 'Boeing', '787', 2009, 253, 248, 333340, 'AFL')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, PASSENGER_CAPACITY, FUEL_TANK_CAPACITY, AIRLINE_ICAO_CODE)

VALUES ('TAP-302', 'Commercial', 'Antonov', 'An-148', 2004, 44, 85, 3929, 'TAP')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, PASSENGER_CAPACITY, FUEL_TANK_CAPACITY, AIRLINE_ICAO_CODE)

VALUES ('TAP-305', 'Commercial', 'Boeing', '747-400', 1988, 396, 416, 63032, 'TAP')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, PASSENGER_CAPACITY, FUEL_TANK_CAPACITY, AIRLINE_ICAO_CODE)

VALUES ('TAP-952', 'Commercial', 'Embraer', 'ERJ-135', 1999, 19, 30, 1711, 'TAP')

SELECT * FROM dual;

//Shipping airplanes

INSERT ALL

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, FUEL_TANK_CAPACITY, SHIPPING_ICAO_CODE)

VALUES ('FDX-019', 'CARGO','Antonov','AN-225',1988, 640, 31748, 'FDX')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, FUEL_TANK_CAPACITY, SHIPPING_ICAO_CODE)

VALUES ('FDX-020', 'CARGO','Airbus','A330-200F',2007, 230, 23775, 'FDX')

//////////////////////UPS

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, FUEL_TANK_CAPACITY, SHIPPING_ICAO_CODE)

VALUES ('UPS-221', 'CARGO','Airbus','A330-700',2016, 242, 36743, 'UPS')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, FUEL_TANK_CAPACITY, SHIPPING_ICAO_CODE)

VALUES ('UPS-209', 'CARGO','Airbus','A330-P2F',2012, 233, 36750, 'UPS')

//////////////////////////DHL

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, FUEL_TANK_CAPACITY, SHIPPING_ICAO_CODE)

VALUES ('DHL-925', 'CARGO','Boeing','747-8F',2011, 442, 63027, 'DHL')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, FUEL_TANK_CAPACITY, SHIPPING_ICAO_CODE)

VALUES ('DHL-672', 'CARGO','Ilyushin','Il-96-400T',1988, 270, 37929, 'DHL')

//////////////////////////QTR

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, FUEL_TANK_CAPACITY, SHIPPING_ICAO_CODE)

VALUES ('QTR-284', 'CARGO','Ilyushin','Il-76TD',1971, 190, 28926, 'QTR')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, FUEL_TANK_CAPACITY, SHIPPING_ICAO_CODE)

VALUES ('QTR-268', 'CARGO','Ilyushin','Il-114T',1988, 23, 2319, 'QTR')
////////////////////////////UAE

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, FUEL_TANK_CAPACITY, SHIPPING_ICAO_CODE)

VALUES ('UAE-361', 'CARGO','Boeing','777F',2009, 347, 47890, 'UAE')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, FUEL_TANK_CAPACITY, SHIPPING_ICAO_CODE)

VALUES ('UAE-420', 'CARGO','Boeing','757-200SF',2001, 93, 11274, 'UAE')
///////////////GEC

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, FUEL_TANK_CAPACITY, SHIPPING_ICAO_CODE)

VALUES ('GEC-010', 'CARGO', 'Avengers INC','AV-2173',3700, 500000, 92385671, 'GEC')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, FUEL_TANK_CAPACITY, SHIPPING_ICAO_CODE)

VALUES ('UAE-011', 'CARGO', 'Avengers INC','AV-2913',5200, 600000, 73628463, 'GEC')
////////////////////AZG

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, FUEL_TANK_CAPACITY, SHIPPING_ICAO_CODE)

VALUES ('AZG-616', 'CARGO', 'Odin INDUSTRIES','OD-7326',2800, 42470, 832749249, 'AZG')

- **INTO AIRPLANE** (AIRPLANE_ID, AIRPLANE_TYPE, MAKE, MODEL, YEAR_MAKE, MTO_WEIGHT, FUEL_TANK_CAPACITY, SHIPPING_ICAO_CODE)

VALUES ('AZG-674', 'CARGO', 'Odin INDUSTRIES','OD-9263',8900, 6962530, 937438211, 'AZG')

13. Package

INSERT ALL

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('LBWADHL152485', 'LBWA152485', 'DHL', 'Glass object', 1, 0.85, 0.40, 0.12, 700, 'BEY', 'BZG', 120, 'A glass vase, flower print on it')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('WAUSUPS841254', 'DZUS841254', 'UPS', 'Paper', 0, 0.30, 0.30, 0.0001, 250, 'BZG', 'LAX', 100, 'An blue envelope with papers inside')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('WALYAZG121496', 'NOUK121496', 'AZG', 'Electronic device', 0, 0.2, 0.1, 0.01, 200, 'BZG', 'TIP', 1800, 'A white iPhone 12 box')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('ARWADHL005816', 'ARBS005816', 'DHL', 'Instrument', 1, 1.2, 0.8, 1.1, 105, 'BAI', 'BZG', 4000, 'A piano assembling parts')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('WACAQTR318420', 'BRCA318420', 'QTR', 'Document', 0, 0.4, 0.12, 0.01, 108,'BZG','GTY', 1000, 'A yellow stamped envelope')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('WAITAZG158412', 'TFLB158412', 'AZG', 'Electronic device', 1, 0.8, 0.75, 0.06,4000, 'BZG','ROM', 2500, 'An Apple iPod air box')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('TFWAUPS158002', 'TFLB158002', 'UPS', 'Electronic device', 1, 0.8, 0.75, 0.06,4000, 'TRO','BZG', 2500, 'An Apple iPod air 2 box')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('FRWAUAE441590', 'FRBG441590', 'UAE', 'Glass object', 1, 0.8, 0.75, 0.006,9000, 'CDG','BZG', 1680, 'A big decorative mirror with golden frame')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('WADKQTR000086', 'CYDK000086', 'QTR', 'Electronic device', 1, 0.49, 0.25, 0.06, 3000, 'BZG', 'STK', 2300, 'An brown Hp computer corton')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('DOWAUAE667810', 'DOEG667810', 'UAE', 'Ceiled Box', 1, 0.80, 0.65, 0.60, 65000,'JLK', 'BZG', 250, 'A box with christmas decoration')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('WAGEGEC140297', 'FJGE140297', 'GEC', 'Machine', 1, 1.2, 0.92, 1.0, 75000, 'BZG', 'NUE', 600, 'An LG washing machine box')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('HUWAFDX119512', 'HUIS119512', 'FDX', 'Medical supplies', 0, 1, 1, 1, 92000, 'LQA', 'BZG', 320, 'A box of blue medical masks')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('WAJPFDX991570', 'JOJP991570', 'FDX', 'Ceiled box', 0, 1.5, 0.9, 0.8, 2500, 'BZG', 'TOK', 85, 'A Bicycle parts before assembling')

- **INTO PACKAGE** (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)

VALUES ('HKWAGEC119703', 'HKGU119703', 'GEC', 'Medical supplies', 1, 1.5, 1.5, 1.5, 12000, 'HON','BZG', 100000, 'A first ship of COVID-19 vaccine')

SELECT * FROM dual;

14. PROVIDES_SERVICES

INSERT ALL

INTO PROVIDES_SERVICES (SP_NAME, SERVICE_NAME, AIRPLANE_ID)

VALUES('Synairgia','Refueling','FDX-020')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Purvis Brothers','Refueling','UPS-209')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Jett Pro Line', 'Mechanical maintenance','DHL-672')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('ScandinavianAVS', 'Cleaning', 'DHL-672')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Badger Aero', 'Audio and visual maintenance','QTR-268')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('J&B', 'Entertainment and communications','UAE-420')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('AVtech', 'Electrical maintenance','UAE-420')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('ScandinavianAVS', 'Cleaning','GEC-011')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Jett Pro Line', 'Mechanical maintenance','AZG-674')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('ScandinavianAVS', 'Cleaning', 'UPS-209')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Badger Aero', 'Audio and visual maintenance','FDX-019')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('J&B', 'Entertainment and communications','FDX-019')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('AVtech', 'Electrical maintenance','DHL-925')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('ScandinavianAVS', 'Cleaning','UAE-361')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Jett Pro Line', 'Mechanical maintenance','AZG-674')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Island aviation', 'Seats repair','AZG-674')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Island aviation', 'Seats repair','AZG-674')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Island aviation', 'Seats repair','QTR-268')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Island aviation', 'Seats repair','UAE-361')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Island aviation', 'Seats repair','GEC-011')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Island aviation', 'Seats repair','AZG-616')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Island aviation', 'Seats repair','UPS-221')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)

VALUES ('Island aviation', 'Seats repair','UPS-209')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Island aviation', 'Seats repair','DHL-925')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Island aviation', 'Seats repair','DHL-672')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES('Synairgia','Refueling','MEA-843')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Purvis Brothers','Refueling','MEA-985')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Jett Pro Line', 'Mechanical maintenance','LOV-745')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('ScandinavianAVS', 'Cleaning', 'AAL-654')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Badger Aero', 'Audio and visual maintenance','AAL-494')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('J&B', 'Entertainment and communications','LUT-166')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('AVtech', 'Electrical maintenance','LUT-166')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('ScandinavianAVS', 'Cleaning','LBR-902')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Jett Pro Line', 'Mechanical maintenance','LBR-901')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('ScandinavianAVS', 'Cleaning', 'AFL-888')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Badger Aero', 'Audio and visual maintenance','AFL-868')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('J&B', 'Entertainment and communications','LBR-902')

INTO PROVIDES_SERVICES (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('AVtech', 'Electrical maintenance','TAP-305')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('ScandinavianAVS', 'Cleaning','TAP-952')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES('Jett Pro Line', 'Mechanical maintenance','FDX-020')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Badger Aero', 'Audio and visual maintenance','LOV-343')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('J&B', 'Entertainment and communications','LUT-167')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('AVtech', 'Electrical maintenance','AAL-494')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('ScandinavianAVS', 'Cleaning','LBR-901')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Jett Pro Line', 'Mechanical maintenance','LUT-167')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Island aviation', 'Seats repair','MEA-843')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Island aviation', 'Seats repair','LOV-745')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Island aviation', 'Seats repair','AAL-494')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Island aviation', 'Seats repair','LUT-167')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Island aviation', 'Seats repair','LBR-900')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Island aviation', 'Seats repair','LBR-902')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Island aviation', 'Seats repair','AFL-868')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Island aviation', 'Seats repair','TAP-305')

- **INTO PROVIDES_SERVICES** (SP_NAME, SERVICE_NAME, AIRPLANE_ID)
VALUES ('Island aviation', 'Seats repair','TAP-952')

SELECT * FROM dual;

15. SHIPPINGCOMPANY_EMPLOYEE

INSERT ALL

- **INTO SHIPPINGCOMPANY_EMPLOYEE** (EMPLOYEE_ID, SICAO_CODE) VALUES('10065','FDX')
- **INTO SHIPPINGCOMPANY_EMPLOYEE** (EMPLOYEE_ID, SICAO_CODE) VALUES('10087','FDX')
- **INTO SHIPPINGCOMPANY_EMPLOYEE** (EMPLOYEE_ID, SICAO_CODE) VALUES('14133','UPS')
- **INTO SHIPPINGCOMPANY_EMPLOYEE** (EMPLOYEE_ID, SICAO_CODE) VALUES('14123','UPS')
- **INTO SHIPPINGCOMPANY_EMPLOYEE** (EMPLOYEE_ID, SICAO_CODE) VALUES('22133','DHL')
- **INTO SHIPPINGCOMPANY_EMPLOYEE** (EMPLOYEE_ID, SICAO_CODE) VALUES('22123','DHL')
- **INTO SHIPPINGCOMPANY_EMPLOYEE** (EMPLOYEE_ID, SICAO_CODE) VALUES('11332','QTR')
- **INTO SHIPPINGCOMPANY_EMPLOYEE** (EMPLOYEE_ID, SICAO_CODE) VALUES('13133','QTR')
- **INTO SHIPPINGCOMPANY_EMPLOYEE** (EMPLOYEE_ID, SICAO_CODE) VALUES('18742','UAE')
- **INTO SHIPPINGCOMPANY_EMPLOYEE** (EMPLOYEE_ID, SICAO_CODE) VALUES('18852','UAE')

- **INTO SHIPPINGCOMPANY_EMPLOYEE** (EMPLOYEE_ID, SICAO_CODE) VALUES('16992','GEC')

- **INTO SHIPPINGCOMPANY_EMPLOYEE** (EMPLOYEE_ID, SICAO_CODE) VALUES('16832','GEC')

- **INTO SHIPPINGCOMPANY_EMPLOYEE** (EMPLOYEE_ID, SICAO_CODE) VALUES('17532','GEC')

- **INTO SHIPPINGCOMPANY_EMPLOYEE** (EMPLOYEE_ID, SICAO_CODE) VALUES('17121','GEC')

- **INTO SHIPPINGCOMPANY_EMPLOYEE** (EMPLOYEE_ID, SICAO_CODE) VALUES('15122','AZG')
- **INTO SHIPPINGCOMPANY_EMPLOYEE** (EMPLOYEE_ID, SICAO_CODE) VALUES('15332','AZG')

- **INTO SHIPPINGCOMPANY_EMPLOYEE** (EMPLOYEE_ID, SICAO_CODE) VALUES('20122','AZG')

- **INTO SHIPPINGCOMPANY_EMPLOYEE** (EMPLOYEE_ID, SICAO_CODE) VALUES('20332','AZG')

- **INTO SHIPPINGCOMPANY_EMPLOYEE** (EMPLOYEE_ID, SICAO_CODE) VALUES('19122','AZG')

- **INTO SHIPPINGCOMPANY_EMPLOYEE** (EMPLOYEE_ID, SICAO_CODE) VALUES('19332','AZG')

SELECT*FROM dual;

16. TECHINCALSP_EMPLOYEE

INSERT ALL

- **INTO TECHINCALSP_EMPLOYEE** (EMPLOYEE_ID, TSP_NAME)
  VALUES('31032','Synairgia')

- **INTO TECHINCALSP_EMPLOYEE** (EMPLOYEE_ID, TSP_NAME)
  VALUES('31045','Synairgia')

- **INTO TECHINCALSP_EMPLOYEE** (EMPLOYEE_ID, TSP_NAME)
VALUES ('32045','Purvis Brothers')

- **INTO TECHINCALSP_EMPLOYEE** (EMPLOYEE_ID, TSP_NAME)
VALUES ('32125','Purvis Brothers')

- **INTO TECHINCALSP_EMPLOYEE** (EMPLOYEE_ID, TSP_NAME)
VALUES ('38045','Jett Pro Line')

- **INTO TECHINCALSP_EMPLOYEE** (EMPLOYEE_ID, TSP_NAME)
VALUES ('32012','Jett Pro Line')

- **INTO TECHINCALSP_EMPLOYEE** (EMPLOYEE_ID, TSP_NAME)
VALUES('33065','AVtech')

- **INTO TECHINCALSP_EMPLOYEE** (EMPLOYEE_ID, TSP_NAME)
VALUES('33204','AVtech')

- **INTO TECHINCALSP_EMPLOYEE** (EMPLOYEE_ID, TSP_NAME)
VALUES ('34075','Badger Aero')

- **INTO TECHINCALSP_EMPLOYEE** (EMPLOYEE_ID, TSP_NAME)
VALUES ('34205','Badger Aero')

- **INTO TECHINCALSP_EMPLOYEE** (EMPLOYEE_ID, TSP_NAME)
VALUES('35022','J&B')

- **INTO TECHINCALSP_EMPLOYEE** (EMPLOYEE_ID, TSP_NAME)
VALUES('35086','J&B')

- **INTO TECHINCALSP_EMPLOYEE** (EMPLOYEE_ID, TSP_NAME)
VALUES ('36098','Island aviation')

- **INTO TECHINCALSP_EMPLOYEE** (EMPLOYEE_ID, TSP_NAME)
VALUES ('36022','Island aviation')

- **INTO TECHINCALSP_EMPLOYEE** (EMPLOYEE_ID, TSP_NAME)
VALUES('37054','ScandinavianAVS')

- **INTO TECHINCALSP_EMPLOYEE** (EMPLOYEE_ID, TSP_NAME)
VALUES('37086','ScandinavianAVS')

SELECT*FROM dual;

17. AIRLINE_EMPLOYEE

INSERT ALL

- **INTO AIRLINE_EMPLOYEE** (EMPLOYEE_ID, AICAO_CODE)
VALUES('27022','LUT')

- **INTO AIRLINE_EMPLOYEE** (EMPLOYEE_ID, AICAO_CODE)
VALUES('27039','LUT')

- **INTO AIRLINE_EMPLOYEE** (EMPLOYEE_ID, AICAO_CODE)
VALUES('27105','LUT')

- **INTO AIRLINE_EMPLOYEE** (EMPLOYEE_ID, AICAO_CODE)
VALUES('26055','AAL')

- **INTO AIRLINE_EMPLOYEE** (EMPLOYEE_ID, AICAO_CODE)
VALUES('26232','AAL')

- **INTO AIRLINE_EMPLOYEE** (EMPLOYEE_ID, AICAO_CODE)
VALUES('27105','AAL')

- **INTO AIRLINE_EMPLOYEE** (EMPLOYEE_ID, AICAO_CODE)
VALUES('28099','LBR')

- **INTO AIRLINE_EMPLOYEE** (EMPLOYEE_ID, AICAO_CODE)

VALUES('28100','LBR')

- **INTO AIRLINE_EMPLOYEE** (EMPLOYEE_ID, AICAO_CODE)
VALUES('28543','LBR')

- **INTO AIRLINE_EMPLOYEE** (EMPLOYEE_ID, AICAO_CODE)
VALUES('28082','AFL')

- **INTO AIRLINE_EMPLOYEE** (EMPLOYEE_ID, AICAO_CODE)
VALUES('28102','AFL')

- **INTO AIRLINE_EMPLOYEE** (EMPLOYEE_ID, AICAO_CODE)
VALUES('28356','AFL')

- **INTO AIRLINE_EMPLOYEE** (EMPLOYEE_ID, AICAO_CODE)
VALUES('29084','TAP')

- **INTO AIRLINE_EMPLOYEE** (EMPLOYEE_ID, AICAO_CODE)
VALUES('29321','TAP')

- **INTO AIRLINE_EMPLOYEE** (EMPLOYEE_ID, AICAO_CODE)
VALUES('29004','TAP')

- **INTO AIRLINE_EMPLOYEE** (EMPLOYEE_ID, AICAO_CODE)
VALUES('25213','LOV')

- **INTO AIRLINE_EMPLOYEE** (EMPLOYEE_ID, AICAO_CODE)
VALUES('25105','LOV')

- **INTO AIRLINE_EMPLOYEE** (EMPLOYEE_ID, AICAO_CODE)
VALUES('25432','LOV')

- **INTO AIRLINE_EMPLOYEE** (EMPLOYEE_ID, AICAO_CODE)
VALUES('24375','MEA')

- **INTO AIRLINE_EMPLOYEE** (EMPLOYEE_ID, AICAO_CODE)
VALUES('24022','MEA')

- **INTO AIRLINE_EMPLOYEE** (EMPLOYEE_ID, AICAO_CODE)
VALUES('24033','MEA')

SELECT*FROM dual;

18. **FLIGHT_CREW:**

INSERT ALL

- **INTO FLIGHT_CREW** (LICENSE_NO, EMPLOYEE_ID, POSITION, RATING, STATUS, AIRLINE, FLIGHT_NUMBER)

VALUES ('04291946','24375','PILOT','4','in service','MEA','MEA325')

- **INTO FLIGHT_CREW** (LICENSE_NO, EMPLOYEE_ID, POSITION, RATING, STATUS, AIRLINE, FLIGHT_NUMBER)

VALUES ('49286834','24022','ATTENDANT','3','license expired','MEA','MEA325')

- **INTO FLIGHT_CREW** (LICENSE_NO, EMPLOYEE_ID, POSITION, RATING, STATUS, AIRLINE, FLIGHT_NUMBER)

VALUES ('76215542','24033','FIRST OFFICER','4','out of service','MEA','MEA325')

//LOV690

- **INTO FLIGHT_CREW** (LICENSE_NO, EMPLOYEE_ID, POSITION, RATING, STATUS, AIRLINE, FLIGHT_NUMBER)

VALUES ('77702712','25213','PILOT','5','out of service','LOV','LOV690')

- **INTO FLIGHT_CREW** (LICENSE_NO, EMPLOYEE_ID, POSITION, RATING, STATUS, AIRLINE, FLIGHT_NUMBER)

VALUES ('464201072','25105','ATTENDANT','2','license expired','LOV','LOV690')

- **INTO FLIGHT_CREW** (LICENSE_NO, EMPLOYEE_ID, POSITION, RATING, STATUS, AIRLINE, FLIGHT_NUMBER)

VALUES ('76003443','25432','ATTENDANT','5','in service','LOV','LOV690')

//AAL298

- **INTO FLIGHT_CREW** (LICENSE_NO, EMPLOYEE_ID, POSITION, RATING, STATUS, AIRLINE, FLIGHT_NUMBER)

VALUES ('33200044','26055','PILOT','1','out of service','AAL','AAL298')

- **INTO FLIGHT_CREW** (LICENSE_NO, EMPLOYEE_ID, POSITION, RATING, STATUS, AIRLINE, FLIGHT_NUMBER)

VALUES ('90467765','26232','FIRST OFFICER','4','in service','AAL','AAL298')

- **INTO FLIGHT_CREW** (LICENSE_NO, EMPLOYEE_ID, POSITION, RATING, STATUS, AIRLINE, FLIGHT_NUMBER)

VALUES ('33484681','26845','ATTENDANT','5','in service','AAL','AAL298')
//LUT984

- **INTO FLIGHT_CREW** (LICENSE_NO, EMPLOYEE_ID, POSITION, RATING, STATUS, AIRLINE, FLIGHT_NUMBER)

VALUES ('98730760','27022','PILOT','5','in service','LUT','LUT984')

- **INTO FLIGHT_CREW** (LICENSE_NO, EMPLOYEE_ID, POSITION, RATING, STATUS, AIRLINE, FLIGHT_NUMBER)

VALUES ('69115802','27039','ATTENDANT','1','out of service','LUT','LUT984')

- **INTO FLIGHT_CREW** (LICENSE_NO, EMPLOYEE_ID, POSITION, RATING, STATUS, AIRLINE, FLIGHT_NUMBER)

VALUES ('95540394','27105','ATTENDANT','5','in service','LUT','LUT984')
//LBR931

- **INTO FLIGHT_CREW** (LICENSE_NO, EMPLOYEE_ID, POSITION, RATING, STATUS, AIRLINE, FLIGHT_NUMBER)

VALUES ('00855885','28099','PILOT','2','out of service','LBR','LBR931')

- **INTO FLIGHT_CREW** (LICENSE_NO, EMPLOYEE_ID, POSITION, RATING, STATUS, AIRLINE, FLIGHT_NUMBER)

VALUES ('34853400','28100','FIRST OFFICER','4','in service','LBR','LBR931')

- **INTO FLIGHT_CREW** (LICENSE_NO, EMPLOYEE_ID, POSITION, RATING, STATUS, AIRLINE, FLIGHT_NUMBER)

VALUES ('05542622','28543','ATTENDANT','2','license expired','LBR','LBR931')
//AFL666

- **INTO FLIGHT_CREW** (LICENSE_NO, EMPLOYEE_ID, POSITION, RATING, STATUS, AIRLINE, FLIGHT_NUMBER)

VALUES ('46215532','28082','PILOT','1','license expired','AFL','AFL666')

- **INTO FLIGHT_CREW** (LICENSE_NO, EMPLOYEE_ID, POSITION, RATING, STATUS, AIRLINE, FLIGHT_NUMBER)

VALUES ('41740508','28102','ATTENDANT','4','in service','AFL','AFL666')

- **INTO FLIGHT_CREW** (LICENSE_NO, EMPLOYEE_ID, POSITION, RATING, STATUS, AIRLINE, FLIGHT_NUMBER)

VALUES ('47548061','28356','ATTENDANT','5','in service','AFL','AFL666')

//TAP717

- **INTO FLIGHT_CREW** (LICENSE_NO, EMPLOYEE_ID, POSITION, RATING, STATUS, AIRLINE, FLIGHT_NUMBER)

VALUES ('94729188','29084','PILOT','5','in service','TAP','TAP717')

- **INTO FLIGHT_CREW** (LICENSE_NO, EMPLOYEE_ID, POSITION, RATING, STATUS, AIRLINE, FLIGHT_NUMBER)

VALUES (**'36218563','29321','FIRST OFFICER','3','license expired','TAP','TAP717'**)

- **INTO FLIGHT_CREW** (LICENSE_NO, EMPLOYEE_ID, POSITION, RATING, STATUS, AIRLINE, FLIGHT_NUMBER)

VALUES ('07649008','29004','ATTENDANT','5','in service','TAP','TAP717')

SELECT*FROM dual;

19. TRANSIT_HOTEL_ROOM

INSERT ALL

- **INTO TRANSIT_HOTEL_ROOM** (ROOM_NUMBER, ROOM_TYPE, AVAILABILITY, PASSENGER)

VALUES (101, 'Single Room', 1, 'LD424925')

- **INTO TRANSIT_HOTEL_ROOM** (ROOM_NUMBER, ROOM_TYPE, AVAILABILITY, PASSENGER)

VALUES (102, 'Twin Room', 0, NULL)

- **INTO TRANSIT_HOTEL_ROOM** (ROOM_NUMBER, ROOM_TYPE, AVAILABILITY, PASSENGER)

VALUES (103, 'Double Room', 1,'MO436262’)

- **INTO TRANSIT_HOTEL_ROOM** (ROOM_NUMBER, ROOM_TYPE, AVAILABILITY, PASSENGER)

VALUES (104, 'Single', 0, NULL)

- **INTO TRANSIT_HOTEL_ROOM** (ROOM_NUMBER, ROOM_TYPE, AVAILABILITY, PASSENGER)

VALUES (201, 'Suite', 0, NULL)

- **INTO TRANSIT_HOTEL_ROOM** (ROOM_NUMBER, ROOM_TYPE, AVAILABILITY, PASSENGER)

VALUES (202, 'Single Room', 1,'MM289371’)

- **INTO TRANSIT_HOTEL_ROOM** (ROOM_NUMBER, ROOM_TYPE, AVAILABILITY, PASSENGER)

VALUES (203, 'Suite', 1, 'BE484368’)

- **INTO TRANSIT_HOTEL_ROOM** (ROOM_NUMBER, ROOM_TYPE, AVAILABILITY, PASSENGER)

VALUES (204, 'Deluxe Room', 0, NULL)

- **INTO TRANSIT_HOTEL_ROOM** (ROOM_NUMBER, ROOM_TYPE, AVAILABILITY, PASSENGER)

VALUES (301, 'Twin Room', 1, 'BR102030')

- **INTO TRANSIT_HOTEL_ROOM** (ROOM_NUMBER, ROOM_TYPE, AVAILABILITY, PASSENGER)

VALUES (302, 'Single', 1, 'LD102031')

- **INTO TRANSIT_HOTEL_ROOM** (ROOM_NUMBER, ROOM_TYPE, AVAILABILITY, PASSENGER)

VALUES (303, 'Double Room', 0, NULL)

- **INTO TRANSIT_HOTEL_ROOM** (ROOM_NUMBER, ROOM_TYPE, AVAILABILITY, PASSENGER)

VALUES (304, 'Single', 1, 'ZH771661')

- **INTO TRANSIT_HOTEL_ROOM** (ROOM_NUMBER, ROOM_TYPE, AVAILABILITY, PASSENGER)

VALUES (401, 'Deluxe Room', 1, 'PR735932')

- **INTO TRANSIT_HOTEL_ROOM** (ROOM_NUMBER, ROOM_TYPE, AVAILABILITY, PASSENGER)

VALUES (402, 'Single', 1, 'TX866394')

- **INTO TRANSIT_HOTEL_ROOM** (ROOM_NUMBER, ROOM_TYPE, AVAILABILITY, PASSENGER)

VALUES (403, 'Double Room', 0, NULL)

- **INTO TRANSIT_HOTEL_ROOM** (ROOM_NUMBER, ROOM_TYPE, AVAILABILITY, PASSENGER)

VALUES (404, 'Single', 1, 'TR437918')

SELECT * FROM dual;

20. FLIGHT

INSERT ALL

- **INTO FLIGHT** (FLIGHT_NUMBER, AIRLINE_CODE, AIRPLANE_ID, ORIGIN, DESTINATION, DEPARTURE_TIME, ARRIVAL_TIME, GROUND_HANDLING, STATUS, GATE_NUMBER, RUNWAY_ID)

VALUES ('MEA325', 'MEA','MEA-905', 'BEY', 'BZG', TO_TIMESTAMP ('2020-12-20 23:45', 'YYYY-MM-DD HH24:MI'), TO_TIMESTAMP ('2020-12-21 05:00', 'YYYY-MM-DD HH24:MI'), 'Aviapartner Group', 'On Time', 'A11B', '36C')

- **INTO FLIGHT** (FLIGHT_NUMBER, AIRLINE_CODE, AIRPLANE_ID, ORIGIN, DESTINATION, DEPARTURE_TIME, ARRIVAL_TIME, GROUND_HANDLING, STATUS, GATE_NUMBER, RUNWAY_ID)

VALUES ('LOV923','LOV','LOV-645', 'BZG','TOP', TO_TIMESTAMP ('2020-11-27 05:30', 'YYYY-MM-DD HH24:MI'), TO_TIMESTAMP ('2020-11-27 11:45', 'YYYY-MM-DD HH24:MI'), 'Dnata', 'On Time', 'H20B', '09R')

- **INTO FLIGHT** (FLIGHT_NUMBER, AIRLINE_CODE, AIRPLANE_ID, ORIGIN, DESTINATION, DEPARTURE_TIME, ARRIVAL_TIME, GROUND_HANDLING, STATUS, GATE_NUMBER, RUNWAY_ID)

VALUES ('LOV690', 'LOV','LOV-343', 'BZG', 'TIP', TO_TIMESTAMP ('2020-11-23 16:25', 'YYYY-MM-DD HH24:MI'), TO_TIMESTAMP ('2020-11-23 22:25', 'YYYY-MM-DD HH24:MI'), 'BBA Aviation plc', 'Delayed', 'B02C', '36L')

- **INTO FLIGHT** (FLIGHT_NUMBER, AIRLINE_CODE, AIRPLANE_ID, ORIGIN, DESTINATION, DEPARTURE_TIME, ARRIVAL_TIME, GROUND_HANDLING, STATUS, GATE_NUMBER, RUNWAY_ID)

VALUES ('AAL298','AAL','AAL-735', 'BZG','GTY', TO_TIMESTAMP ('2020-11-23 14:20', 'YYYY-MM-DD HH24:MI'), TO_TIMESTAMP ('2020-11-23 20:30', 'YYYY-MM-DD HH24:MI'), 'Dnata', 'Cancelled', 'C17A', '27R')

- **INTO FLIGHT** (FLIGHT_NUMBER, AIRLINE_CODE, AIRPLANE_ID, ORIGIN, DESTINATION, DEPARTURE_TIME, ARRIVAL_TIME, GROUND_HANDLING, STATUS, GATE_NUMBER, RUNWAY_ID)

VALUES ('LUT948','LUT','LUT-166', 'BZG','LAX', TO_TIMESTAMP ('2020-11-24 19:30', 'YYYY-MM-DD HH24:MI'), TO_TIMESTAMP ('2020-11-24 23:20', 'YYYY-MM-DD HH24:MI'), 'BBA Aviation plc', 'On Time', 'D18A', '18L')

- **INTO FLIGHT** (FLIGHT_NUMBER, AIRLINE_CODE, AIRPLANE_ID, ORIGIN, DESTINATION, DEPARTURE_TIME, ARRIVAL_TIME, GROUND_HANDLING, STATUS, GATE_NUMBER, RUNWAY_ID)

VALUES ('LBR931','LBR','LBR-900', 'ROM','BZG', TO_TIMESTAMP ('2020-12-26 09:20', 'YYYY-MM-DD HH24:MI'), TO_TIMESTAMP ('2020-12-26 14:35', 'YYYY-MM-DD HH24:MI'), 'Amadeus IT Holding SA', 'On Time', 'B07B', '09L')

- **INTO FLIGHT** (FLIGHT_NUMBER, AIRLINE_CODE, AIRPLANE_ID, ORIGIN, DESTINATION, DEPARTURE_TIME, ARRIVAL_TIME, GROUND_HANDLING, STATUS, GATE_NUMBER, RUNWAY_ID)

VALUES ('AFL666','AFL','AFL-888', 'BZG','BAI', TO_TIMESTAMP ('2020-12-01 20:25', 'YYYY-MM-DD HH24:MI'), TO_TIMESTAMP ('2020-12-01 23:55', 'YYYY-MM-DD HH24:MI'), 'BBA Aviation plc', 'On Time', 'D18A', '27C')

- INTO FLIGHT (FLIGHT_NUMBER, AIRLINE_CODE, AIRPLANE_ID, ORIGIN, DESTINATION, DEPARTURE_TIME, ARRIVAL_TIME, GROUND_HANDLING, STATUS, GATE_NUMBER, RUNWAY_ID)

VALUES ('TAP717','TAP','TAP-302', 'ALG','BZG', TO_TIMESTAMP ('2020-11-28 13:15', 'YYYY-MM-DD HH24:MI'), TO_TIMESTAMP ('2020-11-28 20:45', 'YYYY-MM-DD HH24:MI'), 'BBA Aviation plc', 'Delayed', 'A17A', '09R')

SELECT * FROM dual;

21. AIRLINE

INSERT ALL

INTO AIRLINE (ICAO_CODE, NAME, COUNTRY, NUM_OF_AIRCRAFTS)
VALUES ('AFL', 'Avengers Fly Limited', 'Russia', 940)
INTO AIRLINE (ICAO_CODE, NAME, COUNTRY, NUM_OF_AIRCRAFTS)
VALUES ('LBR', 'Liberty Beacon Airlines', 'Asgard', 1030)
INTO AIRLINE (ICAO_CODE, NAME, COUNTRY, NUM_OF_AIRCRAFTS)
VALUES ('LUT', 'Luft Airlines', 'Atlantis', 150)
INTO AIRLINE (ICAO_CODE, NAME, COUNTRY, NUM_OF_AIRCRAFTS)
VALUES ('AAL', 'American Airlines', 'USA', 863)
INTO AIRLINE (ICAO_CODE, NAME, COUNTRY, NUM_OF_AIRCRAFTS)
VALUES ('LOV','LOVE Airlines', 'Italy', 354)
INTO AIRLINE (ICAO_CODE, NAME, COUNTRY, NUM_OF_AIRCRAFTS)
VALUES ('MEA', 'Middle East Airlines', 'Lebanon', 21)

INTO AIRLINE (ICAO_CODE, NAME, COUNTRY, NUM_OF_AIRCRAFTS)
VALUES ('TAP','TAP Air Portugal', 'Portugal', 87)
SELECT * FROM dual;

22.DEPARTMENT

INSERT ALL

- **INTO DEPARTMENT** (Department_ID, Department_Name, Manager, No_of_Employees, Description)

VALUES (1,'Medical Center',01030,20,'Basic medical care and specialized medicine')

- **INTO DEPARTMENT** (Department_ID, Department_Name, Manager, No_of_Employees, Description)

VALUES (2,'Airport Security',02244,100,'Prepare financial reports, P&L, Balance sheets and budgets; Also Financial Controls to avoid errors, fraud and theft')

- **INTO DEPARTMENT** (Department_ID, Department_Name, Manager, No_of_Employees, Description)

VALUES (3,'Search and Rescue',03120,50,'Search and rescue services are provided to the survivors of aircraft accidents ')

- **INTO DEPARTMENT** (Department_ID, Department_Name, Manager, No_of_Employees, Description)

VALUES (4,'Labor and Wages',04356,10,'The department of labor and wages issues promotions and to foster, promote, and develop the welfare of the wage earners ')

- **INTO DEPARTMENT** (Department_ID, Department_Name, Manager, No_of_Employees, Description)

VALUES (5,'Aerodrome Services',05721,108,'Support airports, aviation authorities, airfield lighting suppliers, airfield consultants ‘)

- **INTO DEPARTMENT** (Department_ID, Department_Name, Manager, No_of_Employees, Description)

VALUES (6,'Legal department',06231,209,'Legal department is responsible for litigation, investigations, compliance, mergers and acquisitions.')

- **INTO DEPARTMENT** (Department_ID, Department_Name, Manager, No_of_Employees, Description)

VALUES (7,'Planning and Economic Department',07010,29,'monitors macroeconomic developments relevant to the planning environment')

- **INTO DEPARTMENT** (Department_ID, Department_Name, Manager, No_of_Employees, Description)

VALUES (8,'Special Transport Services',08105,302,'The Special Transport Services provide an airport bus, or airport shuttle ‘)

- **INTO DEPARTMENT** (Department_ID, Department_Name, Manager, No_of_Employees, Description)

VALUES (9,'Central operations and air control Service',09045,13,'Prevent collisions, organize and expedite the flow of air traffic ‘)

SELECT * FROM dual;

23. TICKET

INSERT ALL

- **INTO TICKET** (Ticket_Number, Passport_Number, Flight_Number, Seat_Number, class, Boarding_Group, Itinerary, Special_Accomidation)

VALUES ('20201013MEA325001','BR102030','MEA325','01A','First',1,'direct’, NULL)

- **INTO TICKET** (Ticket_Number, Passport_Number, Flight_Number, Seat_Number, class, Boarding_Group, Itinerary, Special_Accomidation)

VALUES ('20201013MEA325002','LD102031','MEA325','19D','Business',2 ,'direct’, NULL)

- **INTO TICKET** (Ticket_Number, Passport_Number, Flight_Number, Seat_Number, class, Boarding_Group, Itinerary, Special_Accomidation)

VALUES ('20201013MEA3250021','TR764912','MEA325','04B','First',1 ,'direct’, NULL)

- **INTO TICKET** (Ticket_Number, Passport_Number, Flight_Number, Seat_Number, class, Boarding_Group, Itinerary, Special_Accomidation)

VALUES ('20200426LOV690364','LD424925','LOV690','01A','First',1,'direct', NULL)

- **INTO TICKET** (Ticket_Number, Passport_Number, Flight_Number, Seat_Number, class, Boarding_Group, Itinerary, Special_Accomidation)

VALUES ('20200426LOV690368','PR735932','LOV690','11H','Economy',3,'direct', NULL)

- **INTO TICKET** (Ticket_Number, Passport_Number, Flight_Number, Seat_Number, class, Boarding_Group, Itinerary, Special_Accomidation)

VALUES ('20200426LOV690369','ZH771661','LOV690','03D','Business',2, 'direct', NULL)

- **INTO TICKET** (Ticket_Number, Passport_Number, Flight_Number, Seat_Number, class, Boarding_Group, Itinerary, Special_Accomidation)

VALUES ('20200609AAL29852','NY926392','AAL298','09A','First',1,'connecting','wheelchair')

- **INTO TICKET** (Ticket_Number, Passport_Number, Flight_Number, Seat_Number, class, Boarding_Group, Itinerary, Special_Accomidation)

VALUES ('20200609AAL29853','TX866394','AAL298','11K','Economy',3, 'connecting', NULL)

- **INTO TICKET** (Ticket_Number, Passport_Number, Flight_Number, Seat_Number, class, Boarding_Group, Itinerary, Special_Accomidation)

VALUES ('20200609AAL29862','NY926392','AAL298','09A','First',1,'connecting', NULL)

- **INTO TICKET** (Ticket_Number, Passport_Number, Flight_Number, Seat_Number, class, Boarding_Group, Itinerary, Special_Accomidation)

VALUES ('20200128LUT948104', 'OR484363', 'LUT948', '13H', 'Economy',3, 'direct', 'Bulkhead')

- **INTO TICKET** (Ticket_Number, Passport_Number, Flight_Number, Seat_Number, class, Boarding_Group, Itinerary, Special_Accomidation)

VALUES ('20200128LUT948156','AB972922','LUT948','19E','Business',2 ,'direct', NULL)

- **INTO TICKET** (Ticket_Number, Passport_Number, Flight_Number, Seat_Number, class, Boarding_Group, Itinerary, Special_Accomidation)

VALUES ('20200128LUT948204','MM425012','LUT948','01A','First',1,'direct', NULL)

- **INTO TICKET** (Ticket_Number, Passport_Number, Flight_Number, Seat_Number, class, Boarding_Group, Itinerary, Special_Accomidation)

VALUES ('20170510LBR931891', 'BE484363','LBR931', '13H','Economy',3, 'connecting', NULL)

- **INTO TICKET** (Ticket_Number, Passport_Number, Flight_Number, Seat_Number, class, Boarding_Group, Itinerary, Special_Accomidation)

VALUES ('20170510LBR931893', 'MM289371', 'LBR931', '16C', 'First',1,'connecting' ,NULL)

- **INTO TICKET** (Ticket_Number, Passport_Number, Flight_Number, Seat_Number, class, Boarding_Group, Itinerary, Special_Accomidation)

VALUES ('20170510LBR931862', 'BE484364','LBR931','17A','Fisrt',1, 'connecting', NULL)

- **INTO TICKET** (Ticket_Number, Passport_Number, Flight_Number, Seat_Number, class, Boarding_Group, Itinerary, Special_Accomidation)

VALUES ('20161112AFL666196', 'MO436262', 'AFL666', '11F', 'Business',2, 'direct', 'assistance')

- **INTO TICKET** (Ticket_Number, Passport_Number, Flight_Number, Seat_Number, class, Boarding_Group, Itinerary, Special_Accomidation)

VALUES ('20161112AFL666637','SP924053','AFL666','17E','Business',2, ,'direct', NULL)

- **INTO TICKET** (Ticket_Number, Passport_Number, Flight_Number, Seat_Number, class, Boarding_Group, Itinerary, Special_Accomidation)

VALUES('20191203TAP717109','KN732421','TAP717','11F','Business',2,'connecting','wheelchair')

- **INTO TICKET** (Ticket_Number, Passport_Number, Flight_Number, Seat_Number, class, Boarding_Group, Itinerary, Special_Accomidation)

VALUES ('20191203TAP717709','PO777612','TAP717','10F','Business',2,'connecting', NULL)

SELECT * FROM dual;

Final Tables State:

NOTE: Some tables have many entries, thus we split the table into two parts rather that making the layout landscape

1. AIRLINE

ICAO_CODE	NAME	COUNTRY	NUM_OF_AIRCRAFTS
AFL	Avengers Fly Limited	Russia	940
LBR	Liberty Beacon Airlines	Asgard	1030
LUT	Luft Airlines	Atlantis	150
AAL	American Airlines	USA	863
LOV	LOVE Airlines	Italy	354
MEA	Middle East Airlines	Lebanon	21
TAP	TAP Air Portugal	Portugal	87

2. AIRLINE_EMPLOYEE

EMPLOYEE_ID	AICAO_CODE
27022	LUT
27039	LUT
27105	LUT
26055	AAL
26232	AAL

27105	AAL
28099	LBR
28100	LBR
28543	LBR
28082	AFL
28102	AFL
28356	AFL
29084	TAP
29321	TAP
29004	TAP
25213	LOV
25105	LOV
25432	LOV
24375	MEA
24022	MEA
24033	MEA

3. AIRPLANE

First half:

AIRPLANE_ID	AIRPLANE_TYPE	MAKE	MODEL	YEAR_MAKE	MTO_WEIGHT
FDX-019	CARGO	Antonov	AN-225	1988	640
FDX-020	CARGO	Airbus	A330-200F	2007	230
UPS-221	CARGO	Airbus	A330-700	2016	242
UPS-209	CARGO	Airbus	A330-P2F	2012	233
DHL-925	CARGO	Boeing	747-8F	2011	442
DHL-672	CARGO	Ilyushin	Il-96-400T	1988	270
QTR-284	CARGO	Ilyushin	Il-76TD	1971	190
QTR-268	CARGO	Ilyushin	Il-114T	1988	23
UAE-361	CARGO	Boeing	777F	2009	347
UAE-420	CARGO	Boeing	757-200SF	2001	93
GEC-010	CARGO	Avengers INC	AV-2173	3700	500000
GEC-011	CARGO	Avengers INC	AV-2913	5200	600000

AZG-616	CARGO	Odin INDUSTRIES	OD-7326	2800	42470
AZG-674	CARGO	Odin INDUSTRIES	OD-9263	8900	6962530
MEA-843	Commercial	Boeing	A220-300	2012	68
MEA-905	Commercial	Boeing	A319	1996	76
MEA-985	Commercial	Boeing	A330	1992	233
LOV-343	Commercial	Antonov	An-158	2010	43
LOV-745	Commercial	Antonov	An-148	2004	44
LOV-645	Commercial	Boeing	777-200LR	1993	317
AAL-735	Commercial	Boeing	737-100	1967	49
AAL-654	Commercial	Boeing	747-400	1988	396
AAL-494	Commercial	Bombardier	CRJ705	1999	38
LUT-166	Commercial	Comac	ARJ21	2007	43
LUT-167	Commercial	Embraer	ERJ-135	1999	19
LUT-168	Commercial	Embraer	ERJ-145	1992	22
LBR-900	Commercial	Ilyushin	Il-96-300	1992	250
LBR-901	Commercial	Ilyushin	Il-114ll	1992	23
LBR-902	Commercial	Embraer	E-175	2002	38
AFL-888	Commercial	Sukhoi	SSJ-100	2000	45
AFL-878	Commercial	Tupolev	Tu-204	1990	103
AFL-868	Commercial	Boeing	787	2009	253
TAP-302	Commercial	Antonov	An-148	2004	44
TAP-305	Commercial	Boeing	747-400	1988	396
TAP-952	Commercial	Embraer	ERJ-135	1999	19

Second half:

PASSENGER_CAPACITY	FUEL_TANK_CAPACITY	AIRLINE_ICAO_CODE	SHIPPING_ICAO_CODE
	31748		FDX
	23775		FDX
	36743		UPS
	36750		UPS
	63027		DHL
	37929		DHL
	28926		QTR
	2319		QTR
	47890		UAE
	11274		UAE
	92385671		GEC
	73628463		GEC

	832749249		AZG
	937438211		AZG
160	5681	MEA	
156	7980	MEA	
440	36743	MEA	
99	4302	LOV	
85	3929	LOV	
347	47894	LOV	
124	6875	AAL	
416	63032	AAL	
75	2898	AAL	
90	4832	LUT	
30	1711	LUT	
42	1359	LUT	
300	40317	LBR	
64	8360	LBR	
78	3044	LBR	
98	6932	AFL	
210	13371	AFL	
248	333340	AFL	
85	3929	TAP	
416	63032	TAP	
30	1711	TAP	

4. DEPARTMENT

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER	NO_OF_EMPLOYEES	DESCRIPTION
1	Medical Center	1030	20	Basic medical care and specialised medicine
2	Airport Security	2244	100	Prepare financial reports, P&L, Balance sheets and budgets; Also Financial Controls to avoid errors, fraud and theft
3	Search and Rescue	3120	50	Search and rescue services are provided to the survivors of aircraft accidents
4	Labor and Wages	4356	10	The department of labor and wages issues promotions and to foster, promote, and develop the welfare of the wage earners

5	Aerodrome Services	5721	108	Support airports, aviation authorities, airfield lighting suppliers, airfield consultants
6	Legal department	6231	209	Legal department is resopnsible for litigation, investigations, compliance, mergers and acquisitions.
7	Planning and Economic Department	7010	29	monitors macroeconomic developments relevant to the planning environment
8	Special Transport Services	8105	302	The Special Transport Services provide an airport bus, or airport shuttle
9	Central operations and air control Service	9045	13	Prevent collisions, organize and expedite the flow of air traffic

5. DUTY_FREE_EMPLOYEE

EMPLOYEE_ID	STORE_NAME
41023	Bartartin
42145	Benihana
43202	Disney Store
44231	Express
45061	Gap
46082	Harman
47092	Hersheys Chocolate World
48265	Invicta Store
49056	Jumbo Electronics
50021	Levis Store
51025	Lora
52034	M&MS World
51025	Lora
53124	MUJI
54032	Sephora
55012	Spiritland
56090	Sunglass Hut
57101	Swarovski
58120	Swatch
59090	TISSOT
60020	Vintage
61250	Virgin Megastore

62045	Wakanda Library
63021	flakeshake

6. DUTY_FREE_STORE

STORE_NAME	OPERATOR	STORE_TYPE
flakeshake	Snowball	smoothies
Vintage	1765Russia	vodka
TISSOT	Tissot SA	Watches
Swatch	Swatch	Watches
M&MS World	Mars Inc	Sweets
Gap	Gap Inc	Fashion
Sunglass Hut	Luxottica Group	Sunglasses
MUJI	Ryohin Keikaku Co.	Wine
Virgin Megastore	Virgin	Electronics
Bartartin	Aroma	Food
Wakanda Library	Wakanda history	books
Sephora	Sephora	makeup
Disney Store	Disney Store	toys
Hersheys Chocolate World	The Hershey Company	Chocolate
Swarovski	Swarovski AG	Jewelry
Levis Store	Levi Strauss & Co	Fashion
Express	Express	Sweets
Benihana	Benihana	Restaurant
Lora	Fantasy	Fashion wear
Invicta Store	Invicta Watch Group	Fashion wear
Spiritland	Spiritland Productions	Music
Harman	Harman	Music
Jumbo Electronics	Sicin	Electronics

7. EMPLOYEE

EMPLOYEE_ID	FNAME	MNAME	LNAME	DATE_OF_BIRTH	GENDER	PHONE_NUMBER	EMAIL	SUPERVISOR	DEPARTMENT_ID	SALARY
11120	Rhodri	Penelope	Rios	22/04/1975	M	2558897896	R.Rios@MEA.lb			100000
11330	Thor	Alexander	Odin	15/12/2000	M	255658003182	T.Odin@MEA.lb			30000
12133	Anisah	Kain	Ramos	30/04/1988	F	255319770450	A.Ramos@LOV.uk			130000
12122	Cerys	Beatric	Villalobos	15/10/1993	M	255317000648	C.Villalobos@LOV.uk			33000
20133	Gillian	Joni	Barker	25/03/1990	M	255341411788	D.Robins@LUT.de			350000
20223	Rebekah	Erica	Collins	25/04/1994	F	255299313675	R.Collins@LUT.de			250000
11332	Garry	Storm	Higgs	25/03/1990	M	255341411788	G.Higgs@QTR.qa			200000
13133	Tamara	Malia	Bender	16/03/1973	F	255321788165	T.Bender@QTR.qa			330000
14133	Mahir		Usman	25/05/1990	M	255393514258	M.Usman@UPS.us			345000
14123	Polly	Giovann	Farrow	16/06/1973	F	255226299123	P.Farrow@UPS.us			432000
22133	Vernon	Todd	Mcdaniel	25/12/1986	M	255357810125	V.Mcdaniel@DHL.us			421200
22123	Elena	Luella	Walls	16/11/1973	F	255532367394	E.Walls@DHL.us			312300
15122	Argog	Arko	Aegir	25/12/1300	M	255421063129	A.Aegir@AZG.az			123400
15332	Angerboda	Bestla	Brynhildr	22/11/1222	F	255537411815 0	A.Brynhildr@AZG.az			321200
20122	Argog	Arko	Aegir	25/12/1300	M	255421063129	A.Aegir@AZG.az			123400
20332	Angerboda	Bestla	Brynhildr	22/11/1222	F	255537411815 0	A.Brynhildr@AZG.az			321200

19122	Argog	Arko	Aegir	25/12/1300	M	255421063129	A.Aegir@AZG.az			33340
19332	Angerboda	Bestla	Brynhildr	22/11/1222	F	2555374118150	A.Brynhildr@AZG.az			23450
16992	Jaeden	Jorden	Woodcock	17/07/1887	M	255278976176	J.Woodcock@GEC.de			23450
16832	Lexie	Rhiana	Knight	10/9/1988	F	255243669600	L.Knight@GEC.de			23450
17532	Jaeden	Jorden	Woodcock	17/07/1987	M	255278976176	J.Woodcock@GEC.de			23450
17121	Lexie	Rhiana	Knight	10/9/1988	F	255243669600	L.Knight@GEC.de			23450
18742	Todd	Marshall	Guzman	17/02/1985	M	255956063962	T.Guzman@UAE.ae			23450
18852	Nial	Vernon	Maddox	19/07/1994	F	255355757150	N.Maddox@UAE.ae			23450
24375	William	Calum	Stanley	22/11/1987	M	255771161631	W.MStanley@MEA.lb			32320
24022	Alaya		Neale	12/8/1984	F	255523097593	A.MNeale@MEA.lb			2320
24033	Nelly	Raees	Britton	11/12/1989	M	255252213337	N.Britton@MEA.lb			15120
25213	Laith	Reon	Dolan	15/09/1980	M	255140520656	Laith.Dolan@LOV.it			32120
25105	Bear	Maci	Bright	28/11/1986	M	255132586448	B.Bright@LOV.it			43120
25432	Gillian	Poppy	Christie	12/10/1986	M	255371387752	G.Christie@LOV.it			21130
31032	Jaidon	Cinar	Faulkner	21/06/1992	M	255439830733	J.Faulkner@Synairgia.com			23020
31045	Mehdi		Mustafa	12/9/1992	M	255860864615	M.Mustafa@Synairgia.com			32400

38045	Firat	Ziggy	Truonga	6/3/1990	M	255320620027	F.Truonga@PurvisB.com			23820
38012	Haleemah		Hendricks	12/12/1990	F	255475011924	H.Hendricks@PurvisB.com			32420
32045	Anis	Orson	Hirst	12/8/1977	M	255716203317	A.Hirst@JetLine.com			42130
32012	Rhiann		Wagstaff	17/08/1980	F	255113414898	W.Wagstaff@JetLine.com			42620
33065	Anis	Orson	Hirst	12/8/1977	M	255716203317	A.Hirst@AVtech.com			42130
33204	Rhiann		Wagstaff	17/08/1980	F	255113414898	R.Wagstaff@AVtech.com			42620
34075	Murtaza	Dane	Newton	30/09/1979	M	255756091468	M.Dane@BadgerAero.com			42130
34205	Jamila		Neville	11/9/1983	F	255636038252	J.Neville@BadgerAero.com			42620
35022	Ayub	Iylah	Monaghan	23/07/1973	M	255587020118	A.Monaghan@J&B.com			52380
35086	Asiya		Goddard	12/7/1996	F	255219262951	A.Goddard@J&B.com			32340
36098	Barney	Allan	Rigby	13/02/1978	M	255607153673	B.Rigby@IslandA.com			32460
36022	Alfie	Alfie	Goddard	17/03/1992	M	255944079085	A.Goddard@IslandA.com			12350
37054	Tariq		Day	16/03/1991	M	255688044454	T.Day@ScandinavianAVS.com			20340
37086	Aleisha	Gordon	Allan	10/2/1993	F	255831529187	A.Allan@ScandinavianAVS.com			15080
1255	Saul	Declan	Bloom	15/10/1980	M	255655752470	S.Bloom@WAirport.wa	1050	1	5000
1370	Ronald	Charley	Hodgson	18/12/1985	M	255470062327	R.Hodgson@WAirport.wa	1050	1	3000

1050	Eilidh	Mikayla	Tang	22/06/1991	F	255699095155	E.Tangn@WAirport.wa	1030	1	10000
1030	Ruqayyah	Selin	Wicks	18/04/1993	F	255287310459	R.Wicks@WAirport.wa		1	12000
2870	Tyson	Dani	Childs	20/07/1985	M	255156951614	T.Childs@WAirport.wa	2550	2	3500
2880	Savanna	Lula	Martinez	18/12/1985	F	255857775127	S.Martinez@WAirport.wa	2550	2	7500
2550	Rhona	Vickie	Adamson	5/8/1994	F	255972549338	R.Adamson@WAirport.wa	3244	2	15000
3244	Ellenor	Gurpreet	Andrew	9/4/1982	F	255829667051	E.Andrew@WAirport.wa		2	20000
3470	Anoushka	Timur	Francis	12/1/1996	M	255248822756	A.Francis@WAirport.wa	3111	3	4500
3111	Burhan	Annie	Kent	12/3/1980	F	255103911777	B.Kent@WAirport.wa	3120	3	10000
3120	Danyaal	Timur	Robins	20/03/1992	M	255351343249	D.Robins@WAirport.wa		3	10000
29084	Huxley	Eugene	Herman	11/5/1970	F	255590724505	H.Herman@TAP.pt			1650
29004	Kain	Fleur	Singh	11/5/1970	M	255314498005	K.Singh@TAP.pt			1250
41023	Kenzie	Berat	George	12/11/1978	M	255734469543	K.George@Bartartin.com			2380
42145	Faiza	Harrell	Steadman	21/01/1978	F	255526675063	F.Steadman@Benihana.com			3450
43202	Benjamin	Kenzie	Pham	11/12/1994	M	255981968660	B.Pham@DisneyStore.com			3450
44231	Lori	Berat	Zavala	22/09/1998	F	255483969147	L.Zavala@Express.com			1450
45061	Alton	Lennox	Leech	21/12/1996	M	255756091468	A.Leech@Gap.com			2150
46082	Domonic	Kiara	Huber	20/06/1996	F	255944079085	D.Huber@Harman.com			3450
47092	Zavier		Murat	21/01/1996	M	255636038252	Z.Murat@Hersheys.com			2340
48265	Dania		Mccabe	1/2/1991	F	255248460481	Z.Murat@Invicta.com			1250
49056	Etienne	Dolcie	Iles	2/1/1998	M	255430502245	E.Iles@Jumbo.com			2450

50021	Joe	Matias	Beaumont	15/12/1995	M	255639887023	J.Beaumont@Levis.com			1780
51025	Layla h	Rueben	York	23/09/1995	F	255617661135	L.York@Lora.com			2350
52034	Fabien	Franklin	Reid	21/05/1998	M	255734106505	F.Reid@M&MS.com			2340
53124	Tasnia	Eesha	Bowden	7/7/1996	F	255672910749	T.Bowdend@MUJI.com			3240
54032	Sylvia	Morin	Person	12/8/1995	F	255160863343	T.Bowdend@Sephora.com			1790
55012	Riaan	Zuniga	York	7/7/1996	F	255558601718	R.York@Spiritland.com			1890
56090	Chace		Peacock	12/1/1996	F	255794819336	C.Peacock@SunglassHut.com			1230
57101	Meerab	Byers	Healy	12/5/1997	M	255525658203	M.Healy@Swarovski.com			2550
58120	Carina	Tara	Macgregor	2/12/1996	F	255993564972	C.Macgregor@Swatch.com			2760
59090	Faizaan		Timms	12/11/1997	M	255554572672	F.Timms@TISSOT.com			2430
60020	Syeda		Crawford	25/10/1997	F	255714854367	S.Crawford@Vintage.com			3450
61250	Solomon	Alford	Cornish	12/3/1990	M	255714854367	S.Cornish@VirginMegastore.com			2340
62045	Aida	Viola	Walton	12/11/1997	F	255262101418	A.Walton@WakandaLibrary.com			5340
63021	Chandler	Kristin	Frye	12/9/1991	M	255512106764 4	C.Frye@flakeshake.com			1230
26232	Nour	Jay	Marvin	11/1/1975	F	255871672535	N.Marvin@AAL.us			1750
26055	Ewen	Ray	Olson	13/11/1984	F	255701934484	E.Olson@AAL.us			750
26845	Kamran	Alex	Mason	7/12/1979	M	255796385009	K.Mason@AAL.us			860
28099	Fionnuala	Issac	Malone	13/11/1974	M	255701934484	F.Malone@LBR.az			1400
28100	Nyla	Greg	Perez	23/07/1974	F	255349811429	N.Greg@LBR.az			1000
28543	Edna	Elle	Hope	1/2/1991	M	255213204953	E.Hope@LBR.az			2300
28082	Brody	Issac	Hunter	13/11/1971	M	255909806956	B.Hunter@LBR.az			1350
28102	Dominick	Rudd	Shaun	27/04/1971	M	255602080433	D.Rudd@LBR.az			2650
28356	Tony	Tarik	Laing	9/4/1989	M	255628582115	T.Laing@AFL.ru			950

29321	Zakary	Sanjay	Harvey	17/08/1989	F	2556897658901	Z.Sanjar@TAP.pt			950

8. FLIGHT

First half:

FLIGHT_NUMBER	AIRLINE_CODE	AIRPLANE_ID	ORIGIN	DESTINATION	DEPARTURE_TIME
MEA325	MEA	MEA-905	BEY	BZG	20-DEC-20 11.45.00PM
LOV690	LOV	LOV-343	BZG	TIP	23-NOV-20 04.25.00PM
LUT948	LUT	LUT-166	BZG	LAX	24-NOV-20 07.30.00PM
LBR931	LBR	LBR-900	ROM	BZG	26-DEC-20 09.20.00AM
AFL666	AFL	AFL-888	BZG	BAI	01-DEC-20 08.25.00PM
TAP717	TAP	TAP-302	ALG	BZG	28-NOV-20 01.15.00 PM
AAL298	AAL	AAL-735	BZG	GTY	23-NOV-20 02.20.00 PM
LOV923	LOV	LOV-645	BZG	TOP	27-NOV-20 05.30.00AM

Second half:

ARRIVAL_TIME	GROUND_HANDLING	STATUS	GATE_NUMBER	RUNWAY_ID
21-DEC-20 05.00.00 AM	Aviapartner Group	ON_TIME	A11B	36C
23-NOV-20 10.25.00 PM	BBA Aviation plc	Delayed	B02C	36L
24-NOV-20 11.20.00 PM	BBA Aviation plc	On Time	D18A	18L
26-DEC-20 02.35.00 PM	Amadeus IT Holding SA	On Time	B07B	09L
01-DEC-20 11.55.00PM	BBA Aviation plc	On Time	D18A	27C
28-NOV-20 08.45.00PM	BBA Aviation plc	Delayed	A17A	09R
23-NOV-20 08.30.00PM	Dnata	Cancelled	C17A	27R
27-NOV-20 11.45.00AM	Dnata	On Time	H20B	09R

9. FLIGHT_CREW

LICENSE_NO	EMPLOYEE_ID	POSITION	RATING	STATUS	AIRLINE	FLIGHT_NUMBER
4291946	24375	PILOT	4	in service	MEA	MEA325
49286834	24022	ATTENDANT	3	license expired	MEA	MEA325
76215542	24033	FIRST OFFICER	4	out of service	MEA	MEA325
77702712	25213	PILOT	5	out of service	LOV	LOV690
46401072	25105	ATTENDANT	2	license expired	LOV	LOV690
76003443	25432	ATTENDANT	5	in service	LOV	LOV690
33200044	26055	PILOT	1	out of service	AAL	AAL298
90467765	26232	FIRST OFFICER	4	in service	AAL	AAL298
33484681	26845	ATTENDANT	5	in service	AAL	AAL298
98730760	27022	PILOT	5	in service	LUT	LUT984
69115802	27039	ATTENDANT	1	out of service	LUT	LUT984

95540394	27105	ATTENDANT	5	in service	LUT	LUT984
855885	28099	PILOT	2	out of service	LBR	LBR931
34853400	28100	FIRST OFFICER	4	in service	LBR	LBR931
5542622	28543	ATTENDANT	2	license expired	LBR	LBR931
46215532	28082	PILOT	1	license expired	AFL	AFL666
41740508	28102	ATTENDANT	4	in service	AFL	AFL666
47548061	28356	ATTENDANT	5	in service	AFL	AFL666
94729188	29084	PILOT	5	in service	TAP	TAP717
36218563	29321	FIRST OFFICER	3	license expired	TAP	TAP717
7649008	29004	ATTENDANT	5	in service	TAP	TAP717

10.FOOD_SERVICE

FLIGHT_NUMBER	MEAL_NAME	CUISINE	DESCRIPTION
MEA325	Rice Pilaf	Lebanon	Rice cooked with a little flavorful oil and chopped onion, and then simmered with broth
LOV690	Bouillabaisse	France	Seafood stew made with fresh bony fish, shellfish, mussels, vegetables,and herbs
AAL298	drop biscuitand sausage gravy	United States	soft dough biscuits covered in meat gravy
LUT948	kapenta with sadza	Cote Divoire	Small freshwater fish paired with maize porridge
LBR931	Chicken tikka masala	India	chunks of roasted marinated chicken in a spiced curry
AFL666	Pelmeni	Russia	pastry dumplings filled with minced meat and wrapped in a dough
TAP717	Acorda	Potuguese	Siced bread with garlic,chopped coriander,and poached eggs.

11.LUGGAGE

PASSPORT_NUMBER	LUGGAGE_NUMBER	LUGGAGE_TYPE	COLOR	WEIGHT	TICKET_NUMBER
OR484363	20954834	Suitcase	Grey	35	20200128LUT948104
PO777612		Suitcase	Blue	10	20191203TAP717709
KN732421	20645100	Carry-on	Black	8	20191203TAP717109
SP924053	20441075	Suitcase	Black	20	20161112AFL66637

MO436262		Personal item	Green	12	20161112AFL666196
BE484363	20411856	Suitcase	Blue	23	20170510LBR931891
MM289371	20854410	Suitcase	Red	17	20170510LBR931893
MM425012		Suitcase	Black	30	20200128LUT948204
AB972922	20449930	Carry-on	Red	9	20200128LUT948156
OR484363	20954833	Suitcase	Blue	13	20200128LUT948104
TR437918	20403000	Personal item	White	5	20200609AAL29862
TX866394	20408410	Suitcase	Grey	36	20200919AAL27164
NY926392	20111020	Carry-on	Brown	10	20200609AAL29852
ZH771661	20403111	Suitcase	Blue	29	20200420MEA869823
PR735932	20000011	Suitcase	White	14	20200901ORY741482
LD424925	20478401	Suitcase	Black	33	20200426LOV690364
LD102031	20200541	Carry-on	Blue	6	20201015LHR763019
TR764912	21111574	Personal item	Grey	4	20200309MEA864008
BR102030		Carry-on	Black	22	20201013MEA325001

12.MEDIA_ENTERTAINMENT

FLIGHT_NUMBER	MEDIA_TYPE	GENRE	TITLE
MEA325	Music	Alternative Rock	Live to Rise
MEA325	Music	Rock	Im Alive
MEA325	Music	Classic	Dirt and Roses
MEA325	Movie	Action	Guardians of the Galaxy
LOV690	Music	Rock	Live to Rise
LOV690	Music	Rock	Im Alive
LOV690	Music	Heavy Metal	Even If I Could
LOV690	Movie	Mystery	The Eternals
AAL298	Music	Alternative rock	Live to Rise
AAL298	Music	Opera	Red Ledger
AAL298	Music	Jazz	They Called It
AAL298	Movie	Suspense	Avengers: Endgame
LUT948	Movie	Thriller	Thor: The Dark World
LUT948	Movie	Action	Guardians of the Galaxy
LUT948	Movie	Suspense	Avengers: Endgame
LUT948	Music	Opera	Red Ledger
LBR931	Movie	Drama	Octor Strange
LBR931	Movie	Action	Guardians of the Galaxy
LBR931	Movie	Dark Comedy	Shang-Chi and the Legend of the Ten Rings
AFL666	Movie	Fiction	Doctor Strange in the Multiverse of Madness
AFL666	Movie	Action	Guardians of the Galaxy
AFL666	Movie	Mystery	The Eternals
AFL666	Music	Rock	Im Alive
TAP717	Movie	Fiction	Doctor Strange in the Multiverse of Madness
TAP717	Movie	Action	Guardians of the Galaxy
TAP717	Movie	Mystery	The Eternals
TAP717	Music	Alternative Rock	Live to Rise

13.PACKAGE

PACKAGE_ID	TRACKING_ID	SHIPPING_COMPANY	PACKAGE_TYPE	ISFRAGILE	LENGTH
LBWADHL152485	LBWA152485	DHL	Glass object	1	0.85
DZWAUPS841254	DZWA841254	UPS	Paper	0	0.3
WAUKAZG121496	WAUK121496	AZG	Electronic device	0	0.2
ARWADHL005816	ARWA005816	DHL	Instrument	1	1.2
WADKQTR000086	WADK000086	QTR	Electronic device	1	0.49
DOWAUAE667810	DOWA667810	UAE	Ceiled Box	1	0.8
WAGEGEC140297	WAGE140297	GEC	Machine	1	1.2
HUWAFDX119512	HUWA119512	FDX	Medical supplies	0	1
WAJPFDX991570	WAJP991570	FDX	Ceiled box	0	1.5
WAGUGEC119703	WAGU119703	GEC	Medical supplies	1	1.5
WACAQTR318420	WACA318420	QTR	Document	0	0.4
TFWAZG158412	TFWA158412	AZG	Electronic device	1	0.8
WALBUPS158002	WALB158002	UPS	Electronic device	1	0.8
FRWAUAE441590	FRWA441590	UAE	Glass object	1	0.8
WABAADHL21496	WABA293021	DHL	Electronic device	0	0.2

WIDTH	HEIGHT	WEIGHT	SOURCE	DESTINATION	ESTIMATED_VALUE	DESCRIPTION
0.4	0.12	700	BEY	BZG	120	A glass vase, flower print on it
0.3	0.0001	250	ALG	BZG	100	An blue envelope with papers inside
0.1	0.01	200	BZG	LCY	1800	A white iPhone 12 box
0.8	1.1	105	BAI	BZG	4000	A piano assembling parts
0.25	0.06	3000	BZG	STK	2300	An brown Hp computer corton
0.65	0.6	65000	JLK	BZG	250	A box with christmas decoration
0.92	1	75000	BZG	NUE	600	An LG washing machine box
1	1	92000	LQA	BZG	320	A box of blue medical masks
0.9	0.8	2500	BZG	TOK	85	A Bicycle parts before assembling

1.5	1.5	12000	BZG	NAK	100000	A first ship of COVID-19 vaccine
0.12	0.01	0.75	BZG	GTY	1000	A yellow stamped envelope
0.75	0.06	4000	TRO	BZG	2500	An Apple iPod air box
0.75	0.06	4000	BZG	ROM	2500	An Apple iPod air 2 box
0.75	0.006	9000	CDG	BZG	1680	A big decorative mirror with golden frame
0.2	0.2	360	BZG	BAI	700	A lost Iphone 11 pro from passenger

14.PASSENGER

First half:

DATE_OF_BIRTH	NATIONALITY	PHONE_NUMBER	EMAIL	TICKET_NUMBER
13/10/1996	United Kingdom	117274271094975	BlackWidow@gmail.com	2020101 3MEA325020
22/09/1976	India	9102228043918	AlexisDenisof@gmail.com	2020101 3MEA325022
27/12/1979	India	913897294712001	CaptainMarval@gmail.com	2017051 0LBR931893
12/11/1985	Russia	007-9535553026	OliviaBoris@gmail.com	2016111 2AFL66616
10/12/1999	Russia	007-852093632	AshelyAbramov@gmail.com	2016111 2AFL666637
10/12/1976	Lebanon	96175815769	ThorRivier@gmail.com	2020101 3MEA325021
13/11/1975	India	911928308272325	ScarletWitch@gmail.com	2017051 0LBR931862
11/11/2002	Lebanon	96172882551	HulkKhouja11@gmail.com	2020101 3MEA325001
1/11/1986	United Kingdom	114402071231234	IronManSaab101@gmail.com	2020101 3MEA325002
11/11/2002	France	33145245282	CaptainAmerica@gmail.com	2020042 6LOV690308
21/11/1989	Lebanon	96103010256	ClintEvian@gmail.com	2020042 6LOV690369
1/3/1995	United States	15417543010	LokiNestle@gmail.com	2020101 3MEA325023

23/04/1998	United States	001213-324-3692	PepperJohnson@gmail.com	'20200609AAL29853
23/11/2001	Libya	00244-124-532	JarvisTony@gmail.com	20200609AAL29862
13/04/1982	Algeria	244146901	JasperMathers@gmail.com	20200128LUT948104
2/11/1999	Cote Divoire	244483789	JocastaAlabman@gmail.com	20200128LUT948104
27/12/1990	India	91024628023657	JerryMRC@gmail.com	20200128LUT948204
3/11/2000	Russia	007-9065558118	ColinAGAF@gmail.com	20191203TAP717709
26/02/1988	PORTUGUESE	00351-800180449	DAMFONTE@gmail.com	20191203TAP71789

Second half:

FLIGHT_NUMBER	VISA_NUMBER	VISA_TYPE	VISA_EXPIRATION_DATE	GENDER
MEA325	73967392	Tourist	11/6/2020	F
MEA325	25724301	Immigrant	11/15/2022	M
LBR931	83765219	Work	6/23/2023	F
AFL666	1397203	Immigrant	6/27/2022	F
AFL666	72160293	Student	11/2/2025	F
MEA325	67332929	Immigrant	3/9/2021	M
LBR931	98639823	Tourist	12/16/2021	F
MEA325	74715	Student	10/13/2025	M
MEA325	19283096	Work	10/15/2023	M
LOV690	92597825	Work	11/16/2023	M
LOV690	12394	Immigrant	11/29/2025	M
MEA325	2135614	Student	10/12/2022	M
AAL298	12426551	Student	10/10/2030	F
AAL298	16853179	Student	5/9/2030	F
LUT948	56318279	Work	9/15/2022	M
LUT948	22646247	Tourist	11/1/2026	F
LUT948	91842637	Work	4/21/2025	M
TAP717	12219283	Student	5/20/2022	M
TAP717	5730243	Work	11/10/2025	M

15.PROVIDES_SERVICES

SP_NAME	SERVICE_NAME	AIRPLANE_ID
Synairgia	Refueling	FDX-020

Purvis Brothers	Refueling	UPS-209
Jett Pro Line	Mechanical maintenance	DHL-672
ScandinavianAVS	Cleaning	DHL-672
Badger Aero	Audio and visual maintenance	QTR-268
J&B	Entertainment and communications	UAE-420
AVtech	Electrical maintenance	UAE-420
ScandinavianAVS	Cleaning	GEC-011
Jett Pro Line	Mechanical maintenance	AZG-674
ScandinavianAVS	Cleaning	UPS-209
Badger Aero	Audio and visual maintenance	FDX-019
J&B	Entertainment and communications	FDX-019
AVtech	Electrical maintenance	DHL-925
ScandinavianAVS	Cleaning	UAE-361
Jett Pro Line	Mechanical maintenance	AZG-674
Island aviation	Seats repair	AZG-674
Island aviation	Seats repair	AZG-674
Island aviation	Seats repair	QTR-268
Island aviation	Seats repair	UAE-361
Island aviation	Seats repair	GEC-011
Island aviation	Seats repair	AZG-616
Island aviation	Seats repair	UPS-221
Island aviation	Seats repair	UPS-209
Island aviation	Seats repair	DHL-925
Island aviation	Seats repair	DHL-672
Synairgia	Refueling	MEA-843
Purvis Brothers	Refueling	MEA-985
Jett Pro Line	Mechanical maintenance	LOV-745
ScandinavianAVS	Cleaning	AAL-654
Badger Aero	Audio and visual maintenance	AAL-494
J&B	Entertainment and communications	LUT-166
AVtech	Electrical maintenance	LUT-166
ScandinavianAVS	Cleaning	LBR-902
Jett Pro Line	Mechanical maintenance	LBR-901
ScandinavianAVS	Cleaning	AFL-888
Badger Aero	Audio and visual maintenance	AFL-868
J&B	Entertainment and communications	LBR-902
AVtech	Electrical maintenance	TAP-305
ScandinavianAVS	Cleaning	TAP-952

Jett Pro Line	Mechanical maintenance	FDX-020
Badger Aero	Audio and visual maintenance	LOV-343
J&B	Entertainment and communications	LUT-167
AVtech	Electrical maintenance	AAL-494
ScandinavianAVS	Cleaning	LBR-901
Jett Pro Line	Mechanical maintenance	LUT-167
Island aviation	Seats repair	MEA-843
Island aviation	Seats repair	LOV-745
Island aviation	Seats repair	AAL-494
Island aviation	Seats repair	LUT-167
Island aviation	Seats repair	LBR-900
Island aviation	Seats repair	LBR-902
Island aviation	Seats repair	AFL-868
Island aviation	Seats repair	TAP-305
Island aviation	Seats repair	TAP-952

16.RUNWAY

RUNWAY_ID	RUNWAY_LENGTH	ISAVAILABLE
36C	2500	1
36L	3500	1
36R	2000	1
18C	4000	1
18L	1800	1
09C	3000	1
09L	3500	1
09R	2300	1
27C	2800	1
27L	1500	1
27R	2800	1

17.SHIPPINGCOMPANY_EMPLOYEE

EMPLOYEE_ID	SICAO_CODE
10065	FDX

10087	FDX
14133	UPS
14123	UPS
22133	DHL
22123	DHL
11332	QTR
13133	QTR
18742	UAE
18852	UAE
16992	GEC
16832	GEC
17532	GEC
17121	GEC
15122	AZG
15332	AZG
20122	AZG
20332	AZG
19122	AZG
19332	AZG

18.SHIPPING_COMPANY

ICAO_CODE	NAME	COUNTRY	NUM_OF_AIRCRAFTS
FDX	FEDEx Express	USA	692
UPS	United Parcel Service	USA	274
DHL	DHL Aviation	USA	250
QTR	Qatar Airways Cargo	Qatar	27
UAE	Emirates SkyCarg	UAE	11
GEC	Lufthansa Cargo	Germany	15
AZG	Azgard Cargo	Azgard	800

19.SHOPS_IN

STORE_NAME	PASSPORT_NUMBER

Vintage	LD424925
Vintage	TR764912
Disney Store	NY926392
Invicta Store	TX866394
TISSOT	TX866394
TISSOT	OR484363
TISSOT	BR102030
Swatch	OR484363
M&MS World	MM425012
MUJI	BE484363
Express	MO436262
Benihana	BE484368
Gap	BE484363
Swarovski	TX866394
Invicta Store	TR437918
Swatch	AB972922
Swatch	BR102030
Express	BE484368
Spiritland	MO436262
Vintage	PR735932
Vintage	PR735932
Vintage	NY926392
Hersheys Chocolate World	NY926392
Hersheys Chocolate World	BR102030
Swarovski	BR102030
TISSOT	TR437918
TISSOT	TR437918
TISSOT	BR102030
Swatch	OR484363
Levis Store	MM289371
Express	BE484363
Harman	MO436262

20.TECHINCALSP_EMPLOYEE

EMPLOYEE_ID	TSP_NAME
31032	Synairgia
31045	Synairgia

32045	Purvis Brothers
32125	Purvis Brothers
38045	Jett Pro Line
38012	Jett Pro Line
33065	AVtech
33204	AVtech
34075	Badger Aero
34205	Badger Aero
35022	J&B
35086	J&B
36098	Island aviation
36022	Island aviation
37054	ScandinavianAVS
37086	ScandinavianAVS

21.TECHNICAL_SERVICE_PROVIDER

SP_NAME	COUNTRY	CITY
Synairgia	Bulgaria	Sofia
Purvis Brothers	USA	Valencia RD
Jett Pro Line	USA	Oakland
AVtech	USA	Denver
Badger Aero	USA	Port Washington
J&B	USA	Palmetto
Island aviation	Maldives	Male
ScandinavianAVS	Sweden	Stockholm

22.TICKET

First half:

TICKET_NUMBER	PASSPORT_NUMBER	FLIGHT_NUMBER	SEAT_NUMBER
20201013MEA325001	BR102030	MEA325	01A
20201013MEA325002	LD102031	MEA325	19D
20201013MEA325021	TR764912	MEA325	04B
20201013MEA325020	LD424925	MEA325	01A

20200426LOV690368	PR735932	LOV690	11H
20200426LOV690369	ZH771661	LOV690	03D
20201013MEA325023	NY926392	MEA325	09A
20200609AAL29853	TX866394	AAL298	11K
20200609AAL29862	TR437918	AAL298	09A
20200128LUT948104	OR484363	LUT948	13H
20200128LUT948156	AB972922	LUT948	19E
20200128LUT948204	MM425012	LUT948	01A
20201013MEA325022	BE484363	MEA325	13H
20170510LBR931893	MM289371	LBR931	16C
20170510LBR931862	BE484364	LBR931	17A
20161112AFL666196	MO436262	AFL666	11F
20161112AFL666637	SP924053	AFL666	17E
20191203TAP717109	KN732421	TAP717	11F
20191203TAP717709	PO777612	TAP717	10F

Second half:

CLASS	BOARDING_GROUP	ITINERARY	SPECIAL_ACCOMIDATION
First	1	direct	
Business	2	direct	
First	1	direct	
First	1	direct	
Economy	3	direct	
Business	2	direct	
First	1	connecting	wheelchair
Economy	3	connecting	
First	1	connecting	
Economy	3	direct	Bulkhead
Business	2	direct	
First	1	direct	
Economy	3	connecting	
First	1	connecting	
First	1	connecting	
Business	2	direct	assistance
Business	2	direct	
Business	2	connecting	wheelchair
Business	2	connecting	

XI. Sample Transactions:

QUERY 1: INEFFICIENT AIRPLANES.

1. PROBLEM

In the recent years, airplanes have evolved and airlines have used fuel efficient airplanes that can maximize their profit by being fuel efficient, have a high maximum takeoff weight, and carry a large number of passengers. For this purpose, the airlines that operate in WAKANDA AIRPORT, have decided to remove the airplanes with a larger maximum takeoff weight, low passenger capacity and a low fuel tank size from their fleet. This way, all the airlines at WAKANDA AIRPORT can operate efficiently on the best planes they have, while waiting on the new planes to be delivered. As the database administrators at WAKANDA AIRPORT, we were asked to initiate a query to locate and delete all airplanes with passenger capacity less than or equal to 150, have a maximum takeoff weight larger or equal to 20 and a fuel tank capacity less than or equal to 3200.

2. QUERY

```
///for testing
SELECT *
FROM
    AIRPLANE
WHERE
        PASSENGER_CAPACITY <= 150
        AND MTO_WEIGHT >=20
```

```
AND FUEL_TANK_CAPACITY <= 3200;
```

```
//Deleting

DELETE
 FROM
AIRPLANE
WHERE
PASSENGER_CAPACITY <= 150
AND MTO_WEIGHT >=20
AND FUEL_TANK_CAPACITY <= 3200;
```

3. OUTPUT OF FIRST QUERY:

AIRPLANE_ID	AIRPLANE_TYPE	MAKE	MODEL	YEAR_MAKE	MTO_WEIGHT	PASSENGER_CAPACITY	FUEL_TANK_CAPACITY	AIRLINE_ICAO_CODE
AAL-494	Commercial	Bombardier	CRJ705	1999	38	75	2898	AAL
LUT-168	Commercial	Embraer	ERJ-145	1992	22	42	1359	LUT
LBR-902	Commercial	Embraer	E-175	2002	38	78	3044	LBR

QUERY 2: CYBER ATTACK AND TICKETS LEAK

1. PROBLEM

A cyber-attack was done on WAKANDA AIRPORT database by a DC heroes group, including wonder woman, Batman, Flash and Aqua man. This cyberattack led to an information leak to an unknown source concerning one of the flights, an unknown source was able to have possession on a number of tickets concerning a

certain flight. This source posted these specific tickets for sale for a lower price on the dark web. These tickets belong to the flight MEA325. The security department at WAKANDA airport called the authorities and they initiated an investigation concerning the attack and the buyers of these tickets. To start the investigation, the authorities asked us as administrators to retrieve details about buyers of these tickets. Specifically, the authorities wanted us to locate the full names, passport numbers, ticket numbers of the passengers who bought the tickets belonging to the MEA325 flight and their numbers ranging between 020 and 023 based on that our Tickets number for the corresponding flight are as of the following format "20201013MEA325###" with the last 3 digits being unique to each passenger.

2. QUERY

```
SELECT
TICKET.TICKET_NUMBER,
TICKET.PASSPORT_NUMBER, FNAME,
MNAME, LNAME,
TICKET.FLIGHT_NUMBER
FROM
ticket, passenger
WHERE
 TICKET.PASSPORT_NUMBER=PASSENGER.PASSPORT_NUMBER
AND TICKET.TICKET_NUMBER LIKE '20201013MEA325%'
AND SUBSTR (TICKET.TICKET_NUMBER, 15) BETWEEN '020'AND '023';
```

3-OUTPUT

TICKET_NUMBER	PASSPORT_NUMBER	FNAME	MNAME	LNAME	FLIGHT_NUMBER
20201013MEA325020	LD424925	Black Widow	Evelyn	Mosby	MEA325
20201013MEA325021	TR764912	Thor	Monir	Rivier	MEA325
20201013MEA325022	BE484363	Alexis Denisof	Marshall	Jackson	MEA325
20201013MEA325023	NY926392	Loki	anthony	nestle	MEA325

QUERY 3: CORONA VIRUS VACCINE SHIPPING

1-PROBLEM

With all the technological advancement that Wakanda has, Wkandian scientists made the first ever to exist, a COVID-19 vaccine. The Wkandian government feels obligated to rescue the rest of humanity via the immediate distribution of the vaccines all across the globe. To pursue this endeavor, high officials have decided that the fastest way is to provide earth with the cure is through Wakanda's new airport. One of the shipping companies is loaded with the vaccine. We need to retrieve all the flight information of that shipment, the weight of the package and its dimensions…

2-QUERY

```
SELECT *
FROM PACKAGE
WHERE
DESCRIPTION LIKE '%COVID-19%';
```

3-OUTPUT

PACKAGE_ID	TRACKING_ID	SHIPPING_COMPANY	PACKAGE_TYPE	ISFRAGILE	LENGTH	WIDTH	HEIGHT	WEIGHT	SOURCE	DESTINATION	ESTIMATED_VALUE	DESCRIPTION
HKGUGEC119703	WAGU119703	GEC	Medical supplies	1	2	2	2	12000	BZG	NAK	100000	A first ship of COVID-19 vaccine

QUERY 4: LOST EXTRA WEIGHT PAYMENT RECEIPT

1. PROBLEM

A passenger lost his original extra weight payment receipt and he needs a backup copy to submit to the company he works at for reimbursement purposes. The person was asked to provide his passport number to check the flight he took. It turned out that the person had 2 suitcases during his last flight and he does not remember which luggage he actually paid for, all he remembers is that he paid for one suitcase and that he had a grey and blue suitcases. We need to use the passenger's passport number to check the two suitcases he checked-in in the airport during his last flight. Once we find the two suitcases associated with his passport number we select the one that weighs more than 30kg (as the airport does not charge for suitcases that weigh less than 30kg), we take that ticket number and use it to get the luggage number. Once we obtain the luggage number, we can print a backup receipt.

2. QUERY

```
SELECT
 LUGGAGE_NUMBER,
PASSPORT_NUMBER,
WEIGHT,
COLOR
FROM
LUGGAGE
```

```
WHERE
TICKET_NUMBER = (   SELECT
                    TICKET_NUMBER
                    FROM
                    TICKET
                    WHERE
                    PASSPORT_NUMBER='OR484363')
 AND WEIGHT >30;
```

3-OUTPUT

LUGGAGE_NUMBER	PASSPORT_NUMBER	WEIGHT	COLOR
20954834	OR484363	35	Grey

QUERY 5: LOST PHONE

1. **PROBLEM**

While the cleaning lady was cleaning a passenger's room, she found a phone that the passenger forgot. She took it back to the lost and found office in the airport and they want to check who's the owner of the phone and how to send it back to him/her. As a database manager, you need to check and extract the passport number of the passenger that reserved that room using its room number. Once the passport number is found, we obtain the contact information of the passenger and we send a notification email saying that we are shipping his/her phone to them. Using the passport number, we obtained the flight number and we checked the destination the passenger headed to. After that we make a new package instance with the type 'lost object' and we ship it to the destination the passenger went to.

2. **QUERY**

```
DECLARE l_selected_destination VARCHAR (3);
```

```
BEGIN

   SELECT DESTINATION
   INTO l_selected_destination
   FROM
   TICKET, FLIGHT
   WHERE
   PASSPORT_NUMBER = (
                                    SELECT passenger
                                    FROM TRANSIT_HOTEL_ROOM
                                    WHERE room_number= 103)

   AND FLIGHT.FLIGHT_NUMBER=TICKET.FLIGHT_NUMBER;
INSERT
 INTO
PACKAGE (PACKAGE_ID, TRACKING_ID, SHIPPING_COMPANY,
PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH, HEIGHT, WEIGHT,
SOURCE, DESTINATION, ESTIMATED_VALUE, DESCRIPTION)
VALUES ('WABAADHL21496', 'WABA293021', 'DHL', 'Electronic device', 0,
0.2, 0.2, 0.2, 360, 'BZG', l_selected_destination, 700, 'A lost Iphone 11 pro from
passenger');
END;
```

Or more compact version

```
INSERT INTO PACKAGE (PACKAGE_ID, TRACKING_ID,
SHIPPING_COMPANY, PACKAGE_TYPE, ISFRAGILE, LENGTH, WIDTH,
```

```
HEIGHT, WEIGHT, SOURCE, DESTINATION, ESTIMATED_VALUE,
DESCRIPTION)
VALUES ('WABAADHL21496', 'WABA293021', 'DHL', 'Electronic device', 0,
0.2, 0.2, 0.2, 360, 'BZG', (
        SELECT DESTINATION
        FROM TICKET, FLIGHT
        WHERE PASSPORT_NUMBER = (
                                        SELECT passenger
                                        FROM TRANSIT_HOTEL_ROOM
                                        WHERE room_number= 103)
AND FLIGHT.FLIGHT_NUMBER = TICKET.FLIGHT_NUMBER), 700, 'A lost
Iphone 11 pro from passenger');
SELECT *
FROM PACKAGE
WHERE PACKAGE_ID='WABAADHL21496';
```

3. OUTPUT

PACKAGE_ID	TRACKING_ID	SHIPPING_COMPANY	PACKAGE_TYPE	ISFRAGILE	LENGTH	WIDTH	HEIGHT	WEIGHT	SOURCE	DESTINATION	ESTIMATED_VALUE	DESCRIPTION
WABAADHL21496	WABA293021	DHL	Electronic device	0	0.2	0.2	0.2	360	BZG	BAI	700	A lost Iphone 11 pro from passenger

QUERY 6: SECURITY CHECKS

1. PROBLEM

"One single vulnerability is all an attacker needs ", quoted by Window Synder. It is hard to build a completely secure system, but what you can do is lower the chances of cybersecurity breaches, that's why (airport name) hired, you Erick, as a professional white-hat hacker, to test their system. Erick is so clever when it comes to hacking, his first step is called Reconnaissance in which is information gathering, and Erick knows that a database breach is all he has to do to hack into the real server to inject a backdoor shell, and he is also aware of that only employees with high authority will have more permissions on the system. A major threat to a database can be SQL injection, which Erick will surely use to help him in. Erick is mainly interested in assessing the security of airlines systems. The first step that Erick did is to retrieve all airlines' employee's emails. Unfortunately, the email is encrypted, but he had an intel from an insider saying that the default password of the employees is a composite of the first character of the employee's first name + the first character of the employee's last name + employee's ID reversed. So the next step for Erick is to retrieve all employees that work in airlines and generate a list of their default passwords.

2. QUERY

```
SELECT
EMPLOYEE.EMPLOYEE_ID,
FNAME,
LNAME,
EMAIL,
(SUBSTR(FNAME,1,1)
 || SUBSTR(LNAME,1,1)
```

```
 || REVERSE(EMPLOYEE.EMPLOYEE_ID))
AS DEFAULT_PASSWORD
FROM
EMPLOYEE,
AIRLINE_EMPLOYEE
WHERE
EMPLOYEE.EMPLOYEE_ID = AIRLINE_EMPLOYEE.EMPLOYEE_ID;
```

3. OUTPUT

EMPLOYEE_ID	FNAME	LNAME	EMAIL	DEFAULT_PASSWORD
24375	William	Stanley	W.MStanley@MEA.lb	WS57342
24022	Alaya	Neale	A.MNeale@MEA.lb	AN22042
24033	Nelly	Britton	N.Britton@MEA.lb	NB33042
25213	Laith	Dolan	Laith.Dolan@LOV.it	LD31252
25105	Bear	Bright	B.Bright@LOV.it	BB50152
25432	Gillian	Christie	G.Christie@LOV.it	GC23452
29084	Huxley	Herman	H.Herman@TAP.pt	HH48092
29004	Kain	Singh	K.Singh@TAP.pt	KS40092
26232	Nour	Marvin	N.Marvin@AAL.us	NM23262
26055	Ewen	Olson	E.Olson@AAL.us	EO55062
28099	Fionnuala	Malone	F.Malone@LBR.az	FM99082
28100	Nyla	Perez	N.Greg@LBR.az	NP00182
28543	Edna	Hope	E.Hope@LBR.az	EH34582
28082	Brody	Hunter	B.Hunter@LBR.az	BH28082
28102	Dominick	Shaun	D.Rudd@LBR.az	DS20182
28356	Tony	Laing	T.Laing@AFL.ru	TL65382
29321	Zakary	Harvey	Z.Sanjar@TAP.pt	ZH12392

QUERY 7: DATA RETRIEVAL AFTE SERVER DAMAGE

1. **PROBLEM**

No doubt that servers are a better way to save information. We rarely see companies' information written on papers. That is why (airport name) counts on local storage to store everything. These servers are safely found under the airport ground. Unfortunately, one day a storm took place, and water came down like waves on these servers which caused them to malfunction, causing data corruption on some servers. After a few days of hard work, the IT department was able to retrieve these data. Not to fall into the same mistake again, they thought of saving the data in a more secured place called the cloud. The airport directly called Nokia Networks to help them with this issue. As soon as Nokia signed the contract, they asked for complete authentication on their database system. Nokia's employees start checking the amount of saved data on the airport servers. The data was too big, especially the passenger's data, so as not to cause any error rather than retrieving all passenger's info at once, they asked you as the database manager, to retrieve only 25% of these records.

2. **QUERY**

```
SELECT *
FROM
PASSENGER
WHERE
 ROWNUM <=0.25* (
SELECT COUNT (*)
FROM PASSENGER)
```

3. **OUTPUT**

PASSPORT_NUMBER	FNAME	MNAME	LNAME	FLIGHT_NUMBER	TICKET_NUMBER
LD424925	Black Widow	Evelyn	Mosby	MEA325	20201013MEA325020
BE484363	Alexis Denisof	Marshall	Jackson	MEA325	20201013MEA325022
MM289371	Captain Marvel	Mia	Errickson	LBR931	20170510LBR931893
MO436262	Rashmi Rustagi	Sergi	Boris	AFL666	20161112AFL66616

QUERY 8: GIFTING PASSENGERS

1. PROBLEM

Gifts can make anyone happier. This strategy was used by (airport name), in which if the passenger's date of birth is the same as his flight, he will get a gift from the airport. Giving the same gift to all lucky passengers is not so fun, that is why they have set some rules. If the passenger's name starts with a vowel, he gets a gift to have a 25% discount on his ticket. On the other hand, if the first character is not a vowel, then the passenger enjoys an open luggage weight without paying extra fees. Now we do not know who are the lucky passengers, that is why we asked you the database manager to retrieve the name of the passengers whose date of birth is the same as the current date as for month and day.

Another three tasks are to check between these lucky passengers if the first character is a vowel or not, so if it is a vowel, retrieve the ticket price with passenger Visa_Number, else retrieve luggage Weight.

2. QUERY

SELECT

```
PASSPORT_NUMBER,
FNAME,
MNAME,
LNAME,
DATE_OF_BIRTH,
TICKET_NUMBER,
(case
 WHEN substr(fname,1,1) ='A' THEN '25% ticket discount'
 WHEN substr(fname,1,1) ='E' THEN '25% ticket discount'
 WHEN substr(fname,1,1) ='I' THEN '25% ticket discount'
 WHEN substr(fname,1,1) ='O' THEN '25% ticket discount'
 WHEN substr(fname,1,1) ='Y' THEN '25% ticket discount'
ELSE 'Free overweight luggage' END) AS OFFER
FROM PASSENGER
WHERE
substr(TICKET_NUMBER,5,4) =
 TO_CHAR (DATE_OF_BIRTH, 'MMDD');
select substr('hola',2,4);
```

3-OUTPUT

PASSPORT_NUMBER	FNAME	MNAME	LNAME	DATE_OF_BIRTH	TICKET_NUMBER	OFFER
LD424925	Black Widow	Evelyn	Mosby	10/13/1996	20201013MEA325020	Free overweight luggage
MO436262	Olivia	Sergi	Boris	11/12/1985	20161112AFL666162	25% ticket discount

Query 9:

1. **Problem**

Due to the recent corona virus(COVID19) widespread and its effects on economies, people's jobs and salaries. The airport started to get less and less passengers especially with some countries not allowing their citizens to exit the country nor anyone to enter the country. The airport's management are looking for ideas and strategies to lower the expenses of the airport and maximize profits. For this purpose, they proposed to lower the salaries of each employee

who has a salary of 3000$ or more. As database administrators, we were asked to locate all the employees with salaries Greater than or equal to 3000$ and reduce them to their salary by 20%:

2. Query

```
SELECT EMPLOYEE_ID, SALARY, (SALARY*0.75) AS SALARY_AFTER_CUT
FROM EMPLOYEE
WHERE SALARY>=3000;
```

```
UPDATE EMPLOYEE
SET SALARY = SALARY*0.75
WHERE SALARY>=3000;
```

62 row(s) updated.

//a sample from the 62 rows

EMPLOYEE_ID	SALARY	SALARY_AFTER_CUT
11120	7500	5625
12133	9750	7312.5
20133	26250	19687.5
20223	18750	14062.5
11332	15000	11250
13133	24750	18562.5
14133	25875	19406.25
14123	32400	24300
22133	31590	23692.5

22123	23423	17567.25
15122	9255	6941.25
15332	24090	18067.5
20122	9255	6941.25
20332	24090	18067.5
19122	25005	18753.75

Query 10:

1. Problem:

Flight scheduling is a critical thing to consider in an airport, but at the same time, it can be hard to figure it all as a human being. Programmers are known for their problem-solving ability. One of the ways they use is called Dynamic programming, in which they start taking a problem and dividing it into many sub problems, hence by solving one sub problem they will be able to implement the same technique on all other sub problems. This time the problem they are facing is that that runways 09L and 09R will be going off service for maintenance and flights that were supposed to use those runways now will be hosted on runway 27L. First of all, the programmer needs to change the runway for every flight using runways 09L and 09 to be on 27L Instead. Next the programmers need to create a view that combines all flights (arriving and departing) that will be using runway 27L and their time slot. Lastly the programmer needs to make sure there are no conflicting time slots and that all flights are using the runway at different times.

2. Query

```
UPDATE FLIGHT
SET RUNWAY_ID = '27L'
WHERE RUNWAY_ID ='09L' OR RUNWAY_ID ='09R';

CREATE VIEW RUNWAY_27L AS
SELECT FLIGHT_NUMBER, (CASE
```

```
        WHEN ORIGIN='BZG'
        THEN DEPARTURE_TIME
        ELSE ARRIVAL_TIME END) AS TIME_SLOT
    FROM FLIGHT
    WHERE RUNWAY_ID = '27L';


SELECT FLIGHT_NUMBER, TIME_SLOT
FROM RUNWAY_27L
GROUP BY FLIGHT_NUMBER, TIME_SLOT
HAVING
COUNT(TIME_SLOT) > 1;
```

FLIGHT_NUMBER	TIME_SLOT
LBR931	26-DEC-20 02.35.00PM
TAP717	28-NOV-20 08.45.00 PM
LOV923	27-NOV-20 05.30.00 AM

Query 11:

1. **Problem**

One of the measures that are taken to control the COVID-19 situation in Wakanda is to monitor the health status of passengers visiting Wakanda, passengers provide airlines with their personal information including flight number and seat number in the flight. A covid-19 case was reported from an expat that arrived Wakanda recently, health authorities asked the airport to provide them with the passengers who were sitting next to the positive case. The database manager has to retrieve the seat number of the positive case (given his passport number) and retrieve contact information of the passengers that sat on the same row during the flight.

2. **Query**

```
select seat_number
from ticket
where ticket_number = (
        select ticket_number
        from passenger
        where passport_number = 'AB972922;
```

Through the previous select statement, we got the seat number of the passenger that was tested positive

PASSPORT_NUMBER	SEAT_NUMBER
AB972922	01B

Using his seat number, we will retrieve the passengers that are sitting in the same row as the targeted passenger.

```
SELECT passenger.passport_number, seat_number, fname,mname, lname, NATIONALITY,
passenger.flight_number, phone_number, email
from passenger, ticket
where substr(seat_number,1,2) = substr (
        (
        select seat_number
        from ticket
        where ticket_number = (
                select ticket_number
                from passenger
                where passport_number = 'AB972922')),1,2)
         and substr(seat_number,3,1) != substr(
        (
        select seat_number
        from ticket
        where ticket_number = (
```

```
        select ticket_number
        from passenger
        where passport_number = 'AB972922')),3,1)
and ticket.flight_number = (
        select flight_number
        from ticket
        where passport_number ='AB972922')
and passenger.passport_number=ticket.passport_number;
```

3. Output

PASSPORT_NUMBER	SEAT_NUMBER	FNAME	MNAME	LNAME	NATIONALITY	FLIGHT_NUMBER	PHONE_NUMBER	EMAIL
OR484363	01C	Jasper Sitwell	Lee	Mathers	Algeria	LUT948	00245641469 01	JasperMathers@gmail.com
MM425012	01A	Jerry Mercenary	Colin	Sixtos	India	LUT948	0012910246 28	JerryMRC@gmail.com

XII. Conclusion:

A database is a must to assure the proper functioning and management of an enterprise that consists of many departments, employees, and many services. An airport that accommodates thousands of passengers that travel from and to the airport daily, many shipments and cargo being shipped from and to the airport through shipping companies. Passengers buy tickets that are assigned to specific flight, each flight is also assigned to a specific airplane that belongs to a specific airline, and each airline services its planes through many technical service providers. Upon waiting for a flight, the passenger might visit the duty free area that contains several stores, where can he/she can buy goods ranging from food to beauty products and electronics, if a passenger has to wait between flight, they can also book a room to stay in through the airport's transit hotel. The airport, as any other enterprise, also must ensure that its facilities and personnel are managed and fully operation. The airplanes used by airlines and shipping companies must be efficient, safe, and well serviced to avoid any problems and loss of customers. Therefore, an airport must be monitored and managed by a database to ensure the efficient functioning of all its departments and services.

To keep track and document our journey of correctly designing, implementing, populating, and applying transactions on our database, we built a report consisting of the different phases of building this airport database. Our first step (PHASE 1) was to create and design entity types that belong to an airport and assure their functionality through relationships between these entity types. This conceptual design was put into an ER diagram using Lucid chart web software. Phase 2 was to map these entity types and the relationships to relational schemas. Phase 3 consisted of building and populating the database using Structured Query

Language (SQL). Finally, after our design and data were correctly put together, we ensured the quality of our database by normalization of the database.

XIII. Requirements

PHASE 1

Team of 5: met.
Choose a name for the group: AVENGERSDB.
Real life database: met.
Minimum of 10 different entities: met.
One weak entity: met.
At least one multivalued attribute: met.
Describe in plain English: met.
Minimum of 10 essential pages: met.
Use Times New Roman font: met.
Font size of 11 and spacing of 1.5: met.
Build an ER model with attributes on relationships: met.

PHASE 2

Correct phase 1 based on received feedback: met.

Translate the ER diagram to relational schemas following the explained 7 steps: met.

PHASE 3

Correct phase 2 .
Build the database on the oracle server by creating tables: met.
Populating our database: met.
Come up with 10 transaction queries: met.
Express the queries in SQL: met.
Execute the SQL queries against the database using oracle: met.
Print the results of our queries: met.

PHASE 4

Normalize the database up to BCNF normal form: met.
Hand in the project: met.