GROUP 12

TEAM MEMBERS:

1. Mohammad SAAD
2. Obed KANI
3. Kenneth Yaw OBENG
4. Naafi Dasana IBRAHIM

\* Equal Contribution from all.

**USED DATABASE: ISEN 142**

BRIEF DESCRIPTION OF DATABASE ENVIRONMENT

Our database environment leverages the robust cloud computing platform offered by IBM Cloud. This selection provides several key benefits:

Specifically, the core of our database environment resides on **IBM Db2 on Cloud**, a managed service that provides a robust Db2 database instance. Db2 is a high-performance relational database management system (DBMS) from IBM, well-suited for handling mission-critical workloads and complex data models. This choice leverages the expertise of IBM in database management while offering the benefits of a cloud-based solution.

**Data Model:**

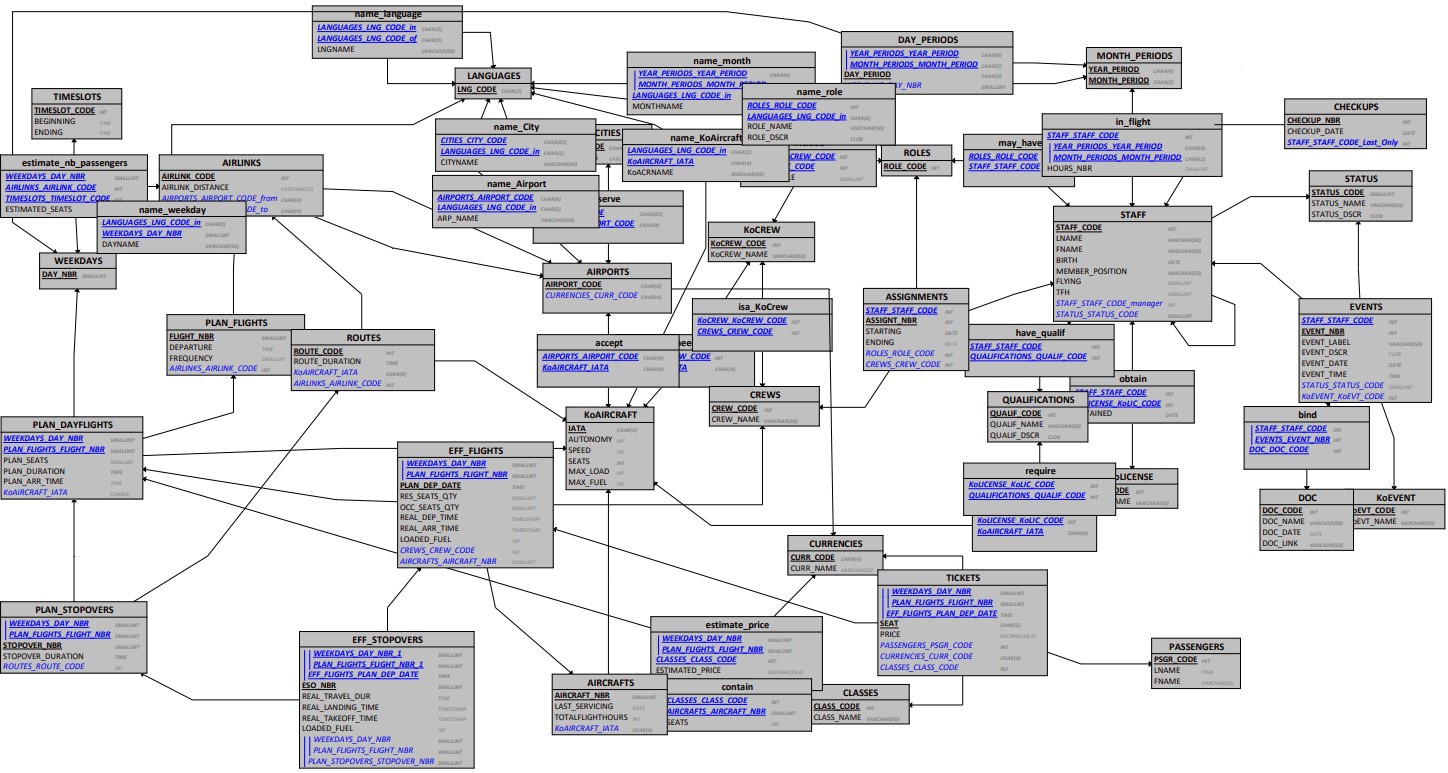
The data model defines the structure and organization of our data. Based on the provided tables (STAFF, KoAIRCRAFT, and allow\_pilot), our data model focuses on managing staff, aircraft, and their flight authorization relationships.

* **Staff Data:** The STAFF table stores information about staff members, including:
  + Staff ID code (primary key)
  + Name
  + Contact information
  + Qualifications (pilot licenses, certifications)
* **Aircraft Data:** The KoAIRCRAFT table stores details about the aircraft fleet, including:
  + Aircraft IATA code (primary key)
  + Aircraft type (e.g., Boeing 737, Airbus A320)
  + Model year
  + Maintenance records (potentially linked to another table)
* **Pilot Permissions:** The allow\_pilot table establishes relationships between staff members and the aircraft they are authorized to fly. This ensures pilots only operate aircraft they are qualified for, enhancing safety and regulatory compliance.

**Collaboration and Access:**

The online nature of IBM Cloud facilitates seamless collaboration among team members working on the project. By sharing login credentials within a secure, controlled team environment, we streamline communication and cooperation. We also make use of Microsoft Teams to stay in the loop and keep ourselves updated with the progress and issues encountered while working on the project. This approach simplified access to the database environment, allowing us members to work together in real-time.

DIAGRAMTIC REPRESENTATION OF DATABASE (DATABASE SCHEMA)Database Schema (Designed with Looping Software)


DATABASE MLD DIAGRAM

SQL STATEMENTS FOR CREATING RELATIONAL OBJECTS (TABLES, SPECIFICALLY)

CREATE TABLE KoCREW(

   KoCREW\_CODE INT NOT NULL,

   KoCREW\_NAME VARCHAR(50) NOT NULL,

   CONSTRAINT PK\_KoCREW PRIMARY KEY(KoCREW\_CODE)

);

-- COMMENT ON TABLE KoCREW IS 'Kinds of Crew';

-- COMMENT ON COLUMN KoCREW.KoCREW\_CODE IS 'Crew Type Identifier';

CREATE TABLE ROLES(

   ROLE\_CODE INT NOT NULL,

   CONSTRAINT PK\_ROLES PRIMARY KEY(ROLE\_CODE)

);

-- COMMENT ON TABLE ROLES IS 'Roles';

-- COMMENT ON COLUMN ROLES.ROLE\_CODE IS 'Role Identifier';

CREATE TABLE CREWS(

   CREW\_CODE INT NOT NULL,

   CREW\_NAME VARCHAR(50) NOT NULL,

   CONSTRAINT PK\_CREWS PRIMARY KEY(CREW\_CODE)

);

-- COMMENT ON TABLE CREWS IS 'Crews';

-- COMMENT ON COLUMN CREWS.CREW\_CODE IS 'Crew Identifier';

CREATE TABLE KoLICENSE(

   KoLIC\_CODE INT NOT NULL,

   KoLIC\_NAME VARCHAR(50) NOT NULL,

   CONSTRAINT PK\_KoLICENSE PRIMARY KEY(KoLIC\_CODE)

);

-- COMMENT ON TABLE KoLICENSE IS 'Kinds of License';

-- COMMENT ON COLUMN KoLICENSE.KoLIC\_CODE IS 'License Type Identifier';

CREATE TABLE QUALIFICATIONS(

   QUALIF\_CODE INT NOT NULL,

   QUALIF\_NAME VARCHAR(50) NOT NULL,

   QUALIF\_DSCR CLOB NOT NULL,

   CONSTRAINT PK\_QUALIFICATIONS PRIMARY KEY(QUALIF\_CODE)

);

-- COMMENT ON TABLE QUALIFICATIONS IS 'Qualifications';

-- COMMENT ON COLUMN QUALIFICATIONS.QUALIF\_CODE IS 'Qualification Identifier';

-- COMMENT ON COLUMN QUALIFICATIONS.QUALIF\_NAME IS 'Qualification Name';

-- COMMENT ON COLUMN QUALIFICATIONS.QUALIF\_DSCR IS 'Qualification Description';

CREATE TABLE STATUS(

   STATUS\_CODE SMALLINT NOT NULL,

   STATUS\_NAME VARCHAR(50) NOT NULL,

   STATUS\_DSCR CLOB NOT NULL,

   CONSTRAINT PK\_STATUS PRIMARY KEY(STATUS\_CODE)

);

-- COMMENT ON TABLE STATUS IS 'Status';

-- COMMENT ON COLUMN STATUS.STATUS\_CODE IS 'Status Identifier';

-- COMMENT ON COLUMN STATUS.STATUS\_NAME IS 'Ex : Embauche en cours, entrée compagnie, sortie compagnie, disponible, indisponible';

-- COMMENT ON COLUMN STATUS.STATUS\_DSCR IS 'Status description';

CREATE TABLE KoEVENT(

   KoEVT\_CODE INT NOT NULL,

   KoEVT\_NAME VARCHAR(50) NOT NULL,

   CONSTRAINT PK\_KoEVENT PRIMARY KEY(KoEVT\_CODE)

);

-- COMMENT ON TABLE KoEVENT IS 'Types of events';

-- COMMENT ON COLUMN KoEVENT.KoEVT\_NAME IS 'entrée compagnie, sortie compagnie, arrêt maladie, reprise maladie, ...';

CREATE TABLE DOC(

   DOC\_CODE INT NOT NULL,

   DOC\_NAME VARCHAR(50) NOT NULL,

   DOC\_DATE DATE,

   DOC\_LINK VARCHAR(50) NOT NULL,

   CONSTRAINT PK\_DOC PRIMARY KEY(DOC\_CODE)

);

-- COMMENT ON TABLE DOC IS 'Documents';

CREATE TABLE CITIES(

   CITY\_CODE CHAR(25) NOT NULL,

   CITY\_GPS VARCHAR(50),

   CONSTRAINT PK\_CITIES PRIMARY KEY(CITY\_CODE)

);

-- COMMENT ON TABLE CITIES IS 'Cities';

CREATE TABLE LANGUAGES(

   LNG\_CODE CHAR(2) NOT NULL,

   CONSTRAINT PK\_LANGUAGES PRIMARY KEY(LNG\_CODE)

);

-- COMMENT ON TABLE LANGUAGES IS 'Languages';

CREATE TABLE KoAIRCRAFT(

   IATA CHAR(4) NOT NULL,

   AUTONOMY INT,

   SPEED INT,

   SEATS INT NOT NULL,

   MAX\_LOAD INT,

   MAX\_FUEL INT,

   CONSTRAINT PK\_KoAIRCRAFT PRIMARY KEY(IATA)

);

-- COMMENT ON TABLE KoAIRCRAFT IS 'Types of aircraft';

-- COMMENT ON COLUMN KoAIRCRAFT.AUTONOMY IS 'Autonomy in kilometers';

-- COMMENT ON COLUMN KoAIRCRAFT.SPEED IS 'Speed in kilometers per hour';

-- COMMENT ON COLUMN KoAIRCRAFT.SEATS IS 'Number of seats';

-- COMMENT ON COLUMN KoAIRCRAFT.MAX\_FUEL IS 'Max fuel in liters';

CREATE TABLE WEEKDAYS(

   DAY\_NBR SMALLINT NOT NULL,

   CONSTRAINT PK\_WEEKDAYS PRIMARY KEY(DAY\_NBR)

);

-- COMMENT ON TABLE WEEKDAYS IS 'Days in Week';

-- COMMENT ON COLUMN WEEKDAYS.DAY\_NBR IS 'From 1:Monday to 7:Sunday';

CREATE TABLE CLASSES(

   CLASS\_CODE INT NOT NULL,

   CLASS\_NAME VARCHAR(50) NOT NULL,

   CONSTRAINT PK\_CLASSES PRIMARY KEY(CLASS\_CODE)

);

-- COMMENT ON TABLE CLASSES IS 'Classes';

CREATE TABLE TIMESLOTS(

   TIMESLOT\_CODE INT NOT NULL,

   BEGINNING TIME,

   ENDING TIME,

   CONSTRAINT PK\_TIMESLOTS PRIMARY KEY(TIMESLOT\_CODE)

);

-- COMMENT ON TABLE TIMESLOTS IS 'Time Slot';

-- COMMENT ON COLUMN TIMESLOTS.TIMESLOT\_CODE IS 'exemple : début de journée, milieu de journée, fin de journée, indifférent';

CREATE TABLE CURRENCIES(

   CURR\_CODE CHAR(4) NOT NULL,

   CURR\_NAME VARCHAR(50),

   CONSTRAINT PK\_CURRENCIES PRIMARY KEY(CURR\_CODE)

);

-- COMMENT ON TABLE CURRENCIES IS 'Currencies';

-- COMMENT ON COLUMN CURRENCIES.CURR\_CODE IS 'EUR, USD, YEN, ...';

CREATE TABLE AIRCRAFTS(

   AIRCRAFT\_NBR SMALLINT NOT NULL,

   LAST\_SERVICING DATE,

   TOTALFLIGHTHOURS INT,

   KoAIRCRAFT\_IATA CHAR(4) NOT NULL,

   CONSTRAINT PK\_AIRCRAFTS PRIMARY KEY(AIRCRAFT\_NBR),

   CONSTRAINT FK\_AIRCRAFTS\_KoAIRCRAFT FOREIGN KEY(KoAIRCRAFT\_IATA) REFERENCES KoAIRCRAFT(IATA)

);

-- COMMENT ON TABLE AIRCRAFTS IS 'Aircrafts';

-- COMMENT ON COLUMN AIRCRAFTS.TOTALFLIGHTHOURS IS 'Total des heures de vol effectué avec appareil';

CREATE TABLE PASSENGERS(

   PSGR\_CODE INT NOT NULL,

   LNAME VARCHAR(50),

   FNAME VARCHAR(50),

   CONSTRAINT PK\_PASSENGERS PRIMARY KEY(PSGR\_CODE)

);

-- COMMENT ON TABLE PASSENGERS IS 'Passengers';

CREATE TABLE STAFF(

   STAFF\_CODE INT NOT NULL,

   LNAME VARCHAR(50) NOT NULL,

   FNAME VARCHAR(50) NOT NULL,

   BIRTH DATE NOT NULL,

   MEMBER\_POSITION VARCHAR(50) NOT NULL,

   FLYING SMALLINT,

   TFH SMALLINT,

   STAFF\_STAFF\_CODE\_manager INT NOT NULL,

   STATUS\_STATUS\_CODE SMALLINT NOT NULL,

   CONSTRAINT PK\_STAFF PRIMARY KEY(STAFF\_CODE),

   CONSTRAINT FK\_STAFF\_STAFF\_manager FOREIGN KEY(STAFF\_STAFF\_CODE\_manager) REFERENCES STAFF(STAFF\_CODE),

   CONSTRAINT FK\_STAFF\_STATUS FOREIGN KEY(STATUS\_STATUS\_CODE) REFERENCES STATUS(STATUS\_CODE)

);

-- COMMENT ON TABLE STAFF IS 'Staff members';

-- COMMENT ON COLUMN STAFF.MEMBER\_POSITION IS '(Hierarchical) position in the entertrise (CEO, COO, CFO, CIO, CMO, ...)';

-- COMMENT ON COLUMN STAFF.TFH IS 'Virtual column - Calculated data : SELECT SUM(Hours\_Numbers) FROM In\_Flight WHERE FK = Staff.Code';

CREATE TABLE MONTH\_PERIODS(

   YEAR\_PERIOD CHAR(4) NOT NULL,

   MONTH\_PERIOD CHAR(2) NOT NULL,

   CONSTRAINT PK\_MONTH\_PERIODS PRIMARY KEY(YEAR\_PERIOD, MONTH\_PERIOD)

);

-- COMMENT ON TABLE MONTH\_PERIODS IS 'Month within Year';

-- COMMENT ON COLUMN MONTH\_PERIODS.MONTH\_PERIOD IS 'Between 1 and 12';

CREATE TABLE DAY\_PERIODS(

   YEAR\_PERIODS\_YEAR\_PERIOD CHAR(4) NOT NULL,

   MONTH\_PERIODS\_MONTH\_PERIOD CHAR(2)  NOT NULL,

   DAY\_PERIOD CHAR(2) NOT NULL,

   WEEKDAYS\_DAY\_NBR SMALLINT NOT NULL,

   CONSTRAINT PK\_DAY\_PERIODS PRIMARY KEY(YEAR\_PERIODS\_YEAR\_PERIOD, MONTH\_PERIODS\_MONTH\_PERIOD, DAY\_PERIOD),

   CONSTRAINT FK\_DAY\_PERIODS\_MONTH\_PERIODS FOREIGN KEY(YEAR\_PERIODS\_YEAR\_PERIOD, MONTH\_PERIODS\_MONTH\_PERIOD) REFERENCES MONTH\_PERIODS(YEAR\_PERIOD, MONTH\_PERIOD),

   CONSTRAINT FK\_DAY\_PERIODS\_WEEKDAYS FOREIGN KEY(WEEKDAYS\_DAY\_NBR) REFERENCES WEEKDAYS(DAY\_NBR)

);

-- COMMENT ON TABLE DAY\_PERIODS IS 'Day within Month';

-- COMMENT ON COLUMN DAY\_PERIODS.DAY\_PERIOD IS 'Between 1 and 31';

CREATE TABLE ASSIGNMENTS(

   STAFF\_STAFF\_CODE INT NOT NULL,

   ASSIGNT\_NBR INT  NOT NULL,

   STARTING DATE NOT NULL,

   ENDING DATE,

   ROLES\_ROLE\_CODE INT NOT NULL,

   CREWS\_CREW\_CODE INT NOT NULL,

   CONSTRAINT PK\_ASSIGNMENTS PRIMARY KEY(STAFF\_STAFF\_CODE, ASSIGNT\_NBR),

   CONSTRAINT FK\_ASSIGNMENTS\_STAFF FOREIGN KEY(STAFF\_STAFF\_CODE) REFERENCES STAFF(STAFF\_CODE),

   CONSTRAINT FK\_ASSIGNMENTS\_ROLES FOREIGN KEY(ROLES\_ROLE\_CODE) REFERENCES ROLES(ROLE\_CODE),

   CONSTRAINT FK\_ASSIGNMENTS\_CREWS FOREIGN KEY(CREWS\_CREW\_CODE) REFERENCES CREWS(CREW\_CODE)

);

-- COMMENT ON TABLE ASSIGNMENTS IS 'Assignment';

CREATE TABLE EVENTS(

   STAFF\_STAFF\_CODE INT NOT NULL,

   EVENT\_NBR INT  NOT NULL,

   EVENT\_LABEL VARCHAR(50) NOT NULL,

   EVENT\_DSCR CLOB,

   EVENT\_DATE DATE NOT NULL,

   EVENT\_TIME TIME NOT NULL,

   STATUS\_STATUS\_CODE SMALLINT,

   KoEVENT\_KoEVT\_CODE INT NOT NULL,

   CONSTRAINT PK\_EVENTS PRIMARY KEY(STAFF\_STAFF\_CODE, EVENT\_NBR),

   CONSTRAINT AK\_EVENTS UNIQUE(STAFF\_STAFF\_CODE),

   CONSTRAINT FK\_EVENTS\_STAFF FOREIGN KEY(STAFF\_STAFF\_CODE) REFERENCES STAFF(STAFF\_CODE),

   CONSTRAINT FK\_EVENTS\_STATUS FOREIGN KEY(STATUS\_STATUS\_CODE) REFERENCES STATUS(STATUS\_CODE),

   CONSTRAINT FK\_EVENTS\_KoEVENT FOREIGN KEY(KoEVENT\_KoEVT\_CODE) REFERENCES KoEVENT(KoEVT\_CODE)

);

-- COMMENT ON TABLE EVENTS IS 'Events';

CREATE TABLE AIRPORTS(

   AIRPORT\_CODE CHAR(4) NOT NULL,

   CURRENCIES\_CURR\_CODE CHAR(4) NOT NULL,

   CONSTRAINT PK\_AIRPORTS PRIMARY KEY(AIRPORT\_CODE),

   CONSTRAINT FK\_AIRPORTS\_CURRENCIES FOREIGN KEY(CURRENCIES\_CURR\_CODE) REFERENCES CURRENCIES(CURR\_CODE)

);

-- COMMENT ON TABLE AIRPORTS IS 'Airports';

CREATE TABLE AIRLINKS(

   AIRLINK\_CODE INT NOT NULL,

   AIRLINK\_DISTANCE VARCHAR(50),

   AIRPORTS\_AIRPORT\_CODE\_from CHAR(4) NOT NULL,

   AIRPORTS\_AIRPORT\_CODE\_to CHAR(4) NOT NULL,

   CONSTRAINT PK\_AIRLINKS PRIMARY KEY(AIRLINK\_CODE),

   CONSTRAINT FK\_AIRLINKS\_AIRPORTS\_\_from FOREIGN KEY(AIRPORTS\_AIRPORT\_CODE\_from) REFERENCES AIRPORTS(AIRPORT\_CODE),

   CONSTRAINT FK\_AIRLINKS\_AIRPORTS\_\_to FOREIGN KEY(AIRPORTS\_AIRPORT\_CODE\_to) REFERENCES AIRPORTS(AIRPORT\_CODE)

);

-- COMMENT ON TABLE AIRLINKS IS 'Airlinks (Airline connections)';

-- COMMENT ON COLUMN AIRLINKS.AIRLINK\_DISTANCE IS 'Flight Distance between Airports in kilometers';

CREATE TABLE ROUTES(

   ROUTE\_CODE INT NOT NULL,

   ROUTE\_DURATION TIME NOT NULL,

   KoAIRCRAFT\_IATA CHAR(4) NOT NULL,

   AIRLINKS\_AIRLINK\_CODE INT NOT NULL,

   CONSTRAINT PK\_ROUTES PRIMARY KEY(ROUTE\_CODE),

   CONSTRAINT FK\_ROUTES\_KoAIRCRAFT FOREIGN KEY(KoAIRCRAFT\_IATA) REFERENCES KoAIRCRAFT(IATA),

   CONSTRAINT FK\_ROUTES\_AIRLINKS FOREIGN KEY(AIRLINKS\_AIRLINK\_CODE) REFERENCES AIRLINKS(AIRLINK\_CODE)

);

-- COMMENT ON TABLE ROUTES IS 'Routes (Airlinks performed with a type of aircraft)';

-- COMMENT ON COLUMN ROUTES.ROUTE\_DURATION IS 'Duration in hours-minutes-seconds';

CREATE TABLE CHECKUPS(

   CHECKUP\_NBR INT NOT NULL,

   CHECKUP\_DATE DATE NOT NULL,

   STAFF\_STAFF\_CODE\_Last\_Only INT NOT NULL,

   CONSTRAINT PK\_CHECKUPS PRIMARY KEY(CHECKUP\_NBR),

   CONSTRAINT AK\_CHECKUPS UNIQUE(STAFF\_STAFF\_CODE\_Last\_Only),

   CONSTRAINT FK\_CHECKUPS\_STAFF\_Last\_Only FOREIGN KEY(STAFF\_STAFF\_CODE\_Last\_Only) REFERENCES STAFF(STAFF\_CODE)

);

-- COMMENT ON TABLE CHECKUPS IS 'Medical Check-up';

CREATE TABLE PLAN\_FLIGHTS(

   FLIGHT\_NBR SMALLINT NOT NULL,

   DEPARTURE TIME,

   FREQUENCY SMALLINT,

   AIRLINKS\_AIRLINK\_CODE INT NOT NULL,

   CONSTRAINT PK\_PLAN\_FLIGHTS PRIMARY KEY(FLIGHT\_NBR),

   CONSTRAINT FK\_PLAN\_FLIGHTS\_AIRLINKS FOREIGN KEY(AIRLINKS\_AIRLINK\_CODE) REFERENCES AIRLINKS(AIRLINK\_CODE)

);

-- COMMENT ON TABLE PLAN\_FLIGHTS IS 'Planned flights';

-- COMMENT ON COLUMN PLAN\_FLIGHTS.FLIGHT\_NBR IS 'Flight Number unique in a day';

-- COMMENT ON COLUMN PLAN\_FLIGHTS.DEPARTURE IS 'Local time of flight departure (UTC)';

-- COMMENT ON COLUMN PLAN\_FLIGHTS.FREQUENCY IS 'Number of flights per week';

CREATE TABLE PLAN\_DAYFLIGHTS(

   WEEKDAYS\_DAY\_NBR SMALLINT NOT NULL,

   PLAN\_FLIGHTS\_FLIGHT\_NBR SMALLINT  NOT NULL,

   PLAN\_SEATS SMALLINT NOT NULL,

   PLAN\_DURATION TIME NOT NULL,

   PLAN\_ARR\_TIME TIME,

   KoAIRCRAFT\_IATA CHAR(4)  NOT NULL,

   CONSTRAINT PK\_PLAN\_DAYFLIGHTS PRIMARY KEY(WEEKDAYS\_DAY\_NBR, PLAN\_FLIGHTS\_FLIGHT\_NBR),

   CONSTRAINT FK\_PLAN\_DAYFLIGHTS\_WEEKDAYS FOREIGN KEY(WEEKDAYS\_DAY\_NBR) REFERENCES WEEKDAYS(DAY\_NBR),

   CONSTRAINT FK\_PLAN\_DAYFLIGHTS\_PLAN\_FLIGHTS FOREIGN KEY(PLAN\_FLIGHTS\_FLIGHT\_NBR) REFERENCES PLAN\_FLIGHTS(FLIGHT\_NBR),

   CONSTRAINT FK\_PLAN\_DAYFLIGHTS\_KoAIRCRAFT FOREIGN KEY(KoAIRCRAFT\_IATA) REFERENCES KoAIRCRAFT(IATA)

);

-- COMMENT ON TABLE PLAN\_DAYFLIGHTS IS 'Planned day flights';

-- COMMENT ON COLUMN PLAN\_DAYFLIGHTS.PLAN\_SEATS IS 'Estimated quantity (number) of passengers for this day in week';

-- COMMENT ON COLUMN PLAN\_DAYFLIGHTS.PLAN\_ARR\_TIME IS 'Estimated local arrival time';

CREATE TABLE PLAN\_STOPOVERS(

   WEEKDAYS\_DAY\_NBR SMALLINT NOT NULL,

   PLAN\_FLIGHTS\_FLIGHT\_NBR SMALLINT  NOT NULL,

   STOPOVER\_NBR SMALLINT  NOT NULL,

   STOPOVER\_DURATION TIME NOT NULL,

   ROUTES\_ROUTE\_CODE INT,

   CONSTRAINT PK\_PLAN\_STOPOVERS PRIMARY KEY(WEEKDAYS\_DAY\_NBR, PLAN\_FLIGHTS\_FLIGHT\_NBR, STOPOVER\_NBR),

   CONSTRAINT FK\_PLAN\_STOPOVERS\_PLAN\_DAYFLIGHTS FOREIGN KEY(WEEKDAYS\_DAY\_NBR, PLAN\_FLIGHTS\_FLIGHT\_NBR) REFERENCES PLAN\_DAYFLIGHTS(WEEKDAYS\_DAY\_NBR, PLAN\_FLIGHTS\_FLIGHT\_NBR),

   CONSTRAINT FK\_PLAN\_STOPOVERS\_ROUTES FOREIGN KEY(ROUTES\_ROUTE\_CODE) REFERENCES ROUTES(ROUTE\_CODE)

);

-- COMMENT ON TABLE PLAN\_STOPOVERS IS 'Planned Stopovers - Arrival airport known thru Route then Airlink';

-- COMMENT ON COLUMN PLAN\_STOPOVERS.STOPOVER\_DURATION IS 'Duration of Stopover on the airport';

CREATE TABLE EFF\_FLIGHTS(

   WEEKDAYS\_DAY\_NBR SMALLINT NOT NULL,

   PLAN\_FLIGHTS\_FLIGHT\_NBR SMALLINT  NOT NULL,

   PLAN\_DEP\_DATE TIME NOT NULL,

   RES\_SEATS\_QTY SMALLINT,

   OCC\_SEATS\_QTY SMALLINT,

   REAL\_DEP\_TIME TIMESTAMP,

   REAL\_ARR\_TIME TIMESTAMP,

   LOADED\_FUEL INT,

   CREWS\_CREW\_CODE INT NOT NULL,

   AIRCRAFTS\_AIRCRAFT\_NBR SMALLINT,

   CONSTRAINT PK\_EFF\_FLIGHTS PRIMARY KEY(WEEKDAYS\_DAY\_NBR, PLAN\_FLIGHTS\_FLIGHT\_NBR, PLAN\_DEP\_DATE),

   CONSTRAINT FK\_EFF\_FLIGHTS\_PLAN\_DAYFLIGHTS FOREIGN KEY(WEEKDAYS\_DAY\_NBR, PLAN\_FLIGHTS\_FLIGHT\_NBR) REFERENCES PLAN\_DAYFLIGHTS(WEEKDAYS\_DAY\_NBR, PLAN\_FLIGHTS\_FLIGHT\_NBR),

   CONSTRAINT FK\_EFF\_FLIGHTS\_CREWS FOREIGN KEY(CREWS\_CREW\_CODE) REFERENCES CREWS(CREW\_CODE),

   CONSTRAINT FK\_EFF\_FLIGHTS\_AIRCRAFTS FOREIGN KEY(AIRCRAFTS\_AIRCRAFT\_NBR) REFERENCES AIRCRAFTS(AIRCRAFT\_NBR)

);

-- COMMENT ON TABLE EFF\_FLIGHTS IS 'Effective flight';

-- COMMENT ON COLUMN EFF\_FLIGHTS.PLAN\_DEP\_DATE IS 'Local Departure date (UTC) - Departure time is obtained with linked Planned Fligtht';

-- COMMENT ON COLUMN EFF\_FLIGHTS.RES\_SEATS\_QTY IS 'Reserved Seats Quantity (Number)';

-- COMMENT ON COLUMN EFF\_FLIGHTS.OCC\_SEATS\_QTY IS 'Occupied Seats Quantity (Number)';

-- COMMENT ON COLUMN EFF\_FLIGHTS.REAL\_DEP\_TIME IS 'Real Departure Time';

-- COMMENT ON COLUMN EFF\_FLIGHTS.REAL\_ARR\_TIME IS 'Real Arriving Time';

-- COMMENT ON COLUMN EFF\_FLIGHTS.LOADED\_FUEL IS 'Loaded Fuel for the complete flight';

CREATE TABLE TICKETS(

   WEEKDAYS\_DAY\_NBR SMALLINT NOT NULL,

   PLAN\_FLIGHTS\_FLIGHT\_NBR SMALLINT  NOT NULL,

   EFF\_FLIGHTS\_PLAN\_DEP\_DATE TIME  NOT NULL,

   SEAT CHAR(5) NOT NULL,

   PRICE DECIMAL(19,4),

   PASSENGERS\_PSGR\_CODE INT NOT NULL,

   CURRENCIES\_CURR\_CODE CHAR(4) NOT NULL,

   CLASSES\_CLASS\_CODE INT NOT NULL,

   CONSTRAINT PK\_TICKETS PRIMARY KEY(WEEKDAYS\_DAY\_NBR, PLAN\_FLIGHTS\_FLIGHT\_NBR, EFF\_FLIGHTS\_PLAN\_DEP\_DATE, SEAT),

   CONSTRAINT FK\_TICKETS\_EFF\_FLIGHTS FOREIGN KEY(WEEKDAYS\_DAY\_NBR, PLAN\_FLIGHTS\_FLIGHT\_NBR, EFF\_FLIGHTS\_PLAN\_DEP\_DATE) REFERENCES EFF\_FLIGHTS(WEEKDAYS\_DAY\_NBR, PLAN\_FLIGHTS\_FLIGHT\_NBR, PLAN\_DEP\_DATE),

   CONSTRAINT FK\_TICKETS\_PASSENGERS FOREIGN KEY(PASSENGERS\_PSGR\_CODE) REFERENCES PASSENGERS(PSGR\_CODE),

   CONSTRAINT FK\_TICKETS\_CURRENCIES FOREIGN KEY(CURRENCIES\_CURR\_CODE) REFERENCES CURRENCIES(CURR\_CODE),

   CONSTRAINT FK\_TICKETS\_CLASSES FOREIGN KEY(CLASSES\_CLASS\_CODE) REFERENCES CLASSES(CLASS\_CODE)

);

-- COMMENT ON TABLE TICKETS IS 'Tickets';

CREATE TABLE EFF\_STOPOVERS(

   WEEKDAYS\_DAY\_NBR\_1 SMALLINT NOT NULL,

   PLAN\_FLIGHTS\_FLIGHT\_NBR\_1 SMALLINT  NOT NULL,

   EFF\_FLIGHTS\_PLAN\_DEP\_DATE TIME NOT NULL,

   ESO\_NBR SMALLINT  NOT NULL,

   REAL\_TRAVEL\_DUR TIME,

   REAL\_LANDING\_TIME TIMESTAMP,

   REAL\_TAKEOFF\_TIME TIMESTAMP,

   LOADED\_FUEL INT,

   WEEKDAYS\_DAY\_NBR SMALLINT NOT NULL,

   PLAN\_FLIGHTS\_FLIGHT\_NBR SMALLINT NOT NULL,

   PLAN\_STOPOVERS\_STOPOVER\_NBR SMALLINT NOT NULL,

   CONSTRAINT PK\_EFF\_STOPOVERS PRIMARY KEY(WEEKDAYS\_DAY\_NBR\_1, PLAN\_FLIGHTS\_FLIGHT\_NBR\_1, EFF\_FLIGHTS\_PLAN\_DEP\_DATE, ESO\_NBR),

   CONSTRAINT FK\_EFF\_STOPOVERS\_EFF\_FLIGHTS\_1 FOREIGN KEY(WEEKDAYS\_DAY\_NBR\_1, PLAN\_FLIGHTS\_FLIGHT\_NBR\_1, EFF\_FLIGHTS\_PLAN\_DEP\_DATE) REFERENCES EFF\_FLIGHTS(WEEKDAYS\_DAY\_NBR, PLAN\_FLIGHTS\_FLIGHT\_NBR, PLAN\_DEP\_DATE),

   CONSTRAINT FK\_EFF\_STOPOVERS\_PLAN\_STOPOVERS FOREIGN KEY(WEEKDAYS\_DAY\_NBR, PLAN\_FLIGHTS\_FLIGHT\_NBR, PLAN\_STOPOVERS\_STOPOVER\_NBR) REFERENCES PLAN\_STOPOVERS(WEEKDAYS\_DAY\_NBR, PLAN\_FLIGHTS\_FLIGHT\_NBR, STOPOVER\_NBR)

);

-- COMMENT ON TABLE EFF\_STOPOVERS IS 'Effective stopovers';

-- COMMENT ON COLUMN EFF\_STOPOVERS.REAL\_TRAVEL\_DUR IS 'Real Travel Duration';

-- COMMENT ON COLUMN EFF\_STOPOVERS.REAL\_LANDING\_TIME IS 'Real Landing Time';

-- COMMENT ON COLUMN EFF\_STOPOVERS.REAL\_TAKEOFF\_TIME IS 'Real Take-off Time';

-- COMMENT ON COLUMN EFF\_STOPOVERS.LOADED\_FUEL IS 'Loaded Fuel during Stopover in liters';

CREATE TABLE include(

   KoCREW\_KoCREW\_CODE INT NOT NULL,

   ROLES\_ROLE\_CODE INT  NOT NULL,

   NBR\_IN\_ROLE SMALLINT,

   CONSTRAINT PK\_include PRIMARY KEY(KoCREW\_KoCREW\_CODE, ROLES\_ROLE\_CODE),

   CONSTRAINT FK\_include\_KoCREW FOREIGN KEY(KoCREW\_KoCREW\_CODE) REFERENCES KoCREW(KoCREW\_CODE),

   CONSTRAINT FK\_include\_ROLES FOREIGN KEY(ROLES\_ROLE\_CODE) REFERENCES ROLES(ROLE\_CODE)

);

-- COMMENT ON TABLE include IS 'Include';

-- COMMENT ON COLUMN include.NBR\_IN\_ROLE IS 'Number of people required in the crew for this role';

CREATE TABLE may\_have(

   ROLES\_ROLE\_CODE INT NOT NULL,

   STAFF\_STAFF\_CODE INT NOT NULL,

   CONSTRAINT PK\_may\_have PRIMARY KEY(ROLES\_ROLE\_CODE, STAFF\_STAFF\_CODE),

   CONSTRAINT FK\_may\_have\_ROLES FOREIGN KEY(ROLES\_ROLE\_CODE) REFERENCES ROLES(ROLE\_CODE),

   CONSTRAINT FK\_may\_have\_STAFF FOREIGN KEY(STAFF\_STAFF\_CODE) REFERENCES STAFF(STAFF\_CODE)

);

-- COMMENT ON TABLE may\_have IS 'May have';

CREATE TABLE need(

   KoCREW\_KoCREW\_CODE INT NOT NULL,

   KoAIRCRAFT\_IATA CHAR(4)  NOT NULL,

   CONSTRAINT PK\_need PRIMARY KEY(KoCREW\_KoCREW\_CODE, KoAIRCRAFT\_IATA),

   CONSTRAINT FK\_need\_KoCREW FOREIGN KEY(KoCREW\_KoCREW\_CODE) REFERENCES KoCREW(KoCREW\_CODE),

   CONSTRAINT FK\_need\_KoAIRCRAFT FOREIGN KEY(KoAIRCRAFT\_IATA) REFERENCES KoAIRCRAFT(IATA)

);

-- COMMENT ON TABLE need IS 'Need';

CREATE TABLE isa\_KoCrew(

   KoCREW\_KoCREW\_CODE INT NOT NULL,

   CREWS\_CREW\_CODE INT  NOT NULL,

   CONSTRAINT PK\_isa\_KoCrew PRIMARY KEY(KoCREW\_KoCREW\_CODE, CREWS\_CREW\_CODE),

   CONSTRAINT FK\_isa\_KoCrew\_KoCREW FOREIGN KEY(KoCREW\_KoCREW\_CODE) REFERENCES KoCREW(KoCREW\_CODE),

   CONSTRAINT FK\_isa\_KoCrew\_CREWS FOREIGN KEY(CREWS\_CREW\_CODE) REFERENCES CREWS(CREW\_CODE)

);

-- COMMENT ON TABLE isa\_KoCrew IS 'Is a Kind of Crew';

CREATE TABLE in\_flight(

   STAFF\_STAFF\_CODE INT NOT NULL,

   YEAR\_PERIODS\_YEAR\_PERIOD CHAR(4)  NOT NULL,

   MONTH\_PERIODS\_MONTH\_PERIOD CHAR(2)   NOT NULL,

   HOURS\_NBR SMALLINT,

   CONSTRAINT PK\_in\_flight PRIMARY KEY(STAFF\_STAFF\_CODE, YEAR\_PERIODS\_YEAR\_PERIOD, MONTH\_PERIODS\_MONTH\_PERIOD),

   CONSTRAINT FK\_in\_flight\_STAFF FOREIGN KEY(STAFF\_STAFF\_CODE) REFERENCES STAFF(STAFF\_CODE),

   CONSTRAINT FK\_in\_flight\_MONTH\_PERIODS FOREIGN KEY(YEAR\_PERIODS\_YEAR\_PERIOD, MONTH\_PERIODS\_MONTH\_PERIOD) REFERENCES MONTH\_PERIODS(YEAR\_PERIOD, MONTH\_PERIOD)

);

-- COMMENT ON TABLE in\_flight IS 'In flight';

CREATE TABLE obtain(

   STAFF\_STAFF\_CODE INT NOT NULL,

   KoLICENSE\_KoLIC\_CODE INT NOT NULL,

   OBTAINED DATE NOT NULL,

   CONSTRAINT PK\_obtain PRIMARY KEY(STAFF\_STAFF\_CODE, KoLICENSE\_KoLIC\_CODE),

   CONSTRAINT FK\_obtain\_STAFF FOREIGN KEY(STAFF\_STAFF\_CODE) REFERENCES STAFF(STAFF\_CODE),

   CONSTRAINT FK\_obtain\_KoLICENSE FOREIGN KEY(KoLICENSE\_KoLIC\_CODE) REFERENCES KoLICENSE(KoLIC\_CODE)

);

-- COMMENT ON TABLE obtain IS 'Obtain';

CREATE TABLE allow\_pilot(

   KoLICENSE\_KoLIC\_CODE INT NOT NULL,

   KoAIRCRAFT\_IATA CHAR(4)  NOT NULL,

   CONSTRAINT PK\_allow\_pilot PRIMARY KEY(KoLICENSE\_KoLIC\_CODE, KoAIRCRAFT\_IATA),

   CONSTRAINT FK\_allow\_pilot\_KoLICENSE FOREIGN KEY(KoLICENSE\_KoLIC\_CODE) REFERENCES KoLICENSE(KoLIC\_CODE),

   CONSTRAINT FK\_allow\_pilot\_KoAIRCRAFT FOREIGN KEY(KoAIRCRAFT\_IATA) REFERENCES KoAIRCRAFT(IATA)

);

-- COMMENT ON TABLE allow\_pilot IS 'Allow pilot';

CREATE TABLE have\_qualif(

   STAFF\_STAFF\_CODE INT NOT NULL,

   QUALIFICATIONS\_QUALIF\_CODE INT  NOT NULL,

   CONSTRAINT PK\_have\_qualif PRIMARY KEY(STAFF\_STAFF\_CODE, QUALIFICATIONS\_QUALIF\_CODE),

   CONSTRAINT FK\_have\_qualif\_STAFF FOREIGN KEY(STAFF\_STAFF\_CODE) REFERENCES STAFF(STAFF\_CODE),

   CONSTRAINT FK\_have\_qualif\_QUALIFICATIONS FOREIGN KEY(QUALIFICATIONS\_QUALIF\_CODE) REFERENCES QUALIFICATIONS(QUALIF\_CODE)

);

-- COMMENT ON TABLE have\_qualif IS 'Have';

CREATE TABLE require(

   KoLICENSE\_KoLIC\_CODE INT NOT NULL,

   QUALIFICATIONS\_QUALIF\_CODE INT  NOT NULL,

   CONSTRAINT PK\_require PRIMARY KEY(KoLICENSE\_KoLIC\_CODE, QUALIFICATIONS\_QUALIF\_CODE),

   CONSTRAINT FK\_require\_KoLICENSE FOREIGN KEY(KoLICENSE\_KoLIC\_CODE) REFERENCES KoLICENSE(KoLIC\_CODE),

   CONSTRAINT FK\_require\_QUALIFICATIONS FOREIGN KEY(QUALIFICATIONS\_QUALIF\_CODE) REFERENCES QUALIFICATIONS(QUALIF\_CODE)

);

-- COMMENT ON TABLE require IS 'Require';

CREATE TABLE bind(

   STAFF\_STAFF\_CODE INT NOT NULL,

   EVENTS\_EVENT\_NBR INT  NOT NULL,

   DOC\_DOC\_CODE INT  NOT NULL,

   CONSTRAINT PK\_bind PRIMARY KEY(STAFF\_STAFF\_CODE, EVENTS\_EVENT\_NBR, DOC\_DOC\_CODE),

   CONSTRAINT FK\_bind\_EVENTS FOREIGN KEY(STAFF\_STAFF\_CODE, EVENTS\_EVENT\_NBR) REFERENCES EVENTS(STAFF\_STAFF\_CODE, EVENT\_NBR),

   CONSTRAINT FK\_bind\_DOC FOREIGN KEY(DOC\_DOC\_CODE) REFERENCES DOC(DOC\_CODE)

);

-- COMMENT ON TABLE bind IS 'Bind';

CREATE TABLE name\_language(

   LANGUAGES\_LNG\_CODE\_in CHAR(2) NOT NULL,

   LANGUAGES\_LNG\_CODE\_of CHAR(2)  NOT NULL,

   LNGNAME VARCHAR(50) NOT NULL,

   CONSTRAINT PK\_name\_language PRIMARY KEY(LANGUAGES\_LNG\_CODE\_in, LANGUAGES\_LNG\_CODE\_of),

   CONSTRAINT FK\_name\_language\_LANGUAGES\_\_in FOREIGN KEY(LANGUAGES\_LNG\_CODE\_in) REFERENCES LANGUAGES(LNG\_CODE),

   CONSTRAINT FK\_name\_language\_LANGUAGES\_\_of FOREIGN KEY(LANGUAGES\_LNG\_CODE\_of) REFERENCES LANGUAGES(LNG\_CODE)

);

-- COMMENT ON TABLE name\_language IS 'Name language';

CREATE TABLE name\_City(

   CITIES\_CITY\_CODE CHAR(25) NOT NULL,

   LANGUAGES\_LNG\_CODE\_in CHAR(2)  NOT NULL,

   CITYNAME VARCHAR(50) NOT NULL,

   CONSTRAINT PK\_name\_City PRIMARY KEY(CITIES\_CITY\_CODE, LANGUAGES\_LNG\_CODE\_in),

   CONSTRAINT FK\_name\_City\_CITIES FOREIGN KEY(CITIES\_CITY\_CODE) REFERENCES CITIES(CITY\_CODE),

   CONSTRAINT FK\_name\_City\_LANGUAGES\_\_in FOREIGN KEY(LANGUAGES\_LNG\_CODE\_in) REFERENCES LANGUAGES(LNG\_CODE)

);

-- COMMENT ON TABLE name\_City IS 'Name city';

-- COMMENT ON COLUMN name\_City.CITYNAME IS 'Name of City';

CREATE TABLE serve(

   CITIES\_CITY\_CODE CHAR(25) NOT NULL,

   AIRPORTS\_AIRPORT\_CODE CHAR(4)  NOT NULL,

   CONSTRAINT PK\_serve PRIMARY KEY(CITIES\_CITY\_CODE, AIRPORTS\_AIRPORT\_CODE),

   CONSTRAINT FK\_serve\_CITIES FOREIGN KEY(CITIES\_CITY\_CODE) REFERENCES CITIES(CITY\_CODE),

   CONSTRAINT FK\_serve\_AIRPORTS FOREIGN KEY(AIRPORTS\_AIRPORT\_CODE) REFERENCES AIRPORTS(AIRPORT\_CODE)

);

-- COMMENT ON TABLE serve IS 'Serve';

CREATE TABLE name\_Airport(

   AIRPORTS\_AIRPORT\_CODE CHAR(4) NOT NULL,

   LANGUAGES\_LNG\_CODE\_in CHAR(2)  NOT NULL,

   ARP\_NAME VARCHAR(50) NOT NULL,

   CONSTRAINT PK\_name\_Airport PRIMARY KEY(AIRPORTS\_AIRPORT\_CODE, LANGUAGES\_LNG\_CODE\_in),

   CONSTRAINT FK\_name\_Airport\_AIRPORTS FOREIGN KEY(AIRPORTS\_AIRPORT\_CODE) REFERENCES AIRPORTS(AIRPORT\_CODE),

   CONSTRAINT FK\_name\_Airport\_LANGUAGES\_\_in FOREIGN KEY(LANGUAGES\_LNG\_CODE\_in) REFERENCES LANGUAGES(LNG\_CODE)

);

-- COMMENT ON TABLE name\_Airport IS 'Name airport';

-- COMMENT ON COLUMN name\_Airport.ARP\_NAME IS 'Name of Airport';

CREATE TABLE accept(

   AIRPORTS\_AIRPORT\_CODE CHAR(4) NOT NULL,

   KoAIRCRAFT\_IATA CHAR(4)  NOT NULL,

   CONSTRAINT PK\_accept PRIMARY KEY(AIRPORTS\_AIRPORT\_CODE, KoAIRCRAFT\_IATA),

   CONSTRAINT FK\_accept\_AIRPORTS FOREIGN KEY(AIRPORTS\_AIRPORT\_CODE) REFERENCES AIRPORTS(AIRPORT\_CODE),

   CONSTRAINT FK\_accept\_KoAIRCRAFT FOREIGN KEY(KoAIRCRAFT\_IATA) REFERENCES KoAIRCRAFT(IATA)

);

-- COMMENT ON TABLE accept IS 'Accept';

CREATE TABLE name\_KoAircraft(

   LANGUAGES\_LNG\_CODE\_in CHAR(2) NOT NULL,

   KoAIRCRAFT\_IATA CHAR(4)  NOT NULL,

   KoACRNAME VARCHAR(50) NOT NULL,

   CONSTRAINT PK\_name\_KoAircraft PRIMARY KEY(LANGUAGES\_LNG\_CODE\_in, KoAIRCRAFT\_IATA),

   CONSTRAINT FK\_name\_KoAircraft\_LANGUAGES\_\_in FOREIGN KEY(LANGUAGES\_LNG\_CODE\_in) REFERENCES LANGUAGES(LNG\_CODE),

   CONSTRAINT FK\_name\_KoAircraft\_KoAIRCRAFT FOREIGN KEY(KoAIRCRAFT\_IATA) REFERENCES KoAIRCRAFT(IATA)

);

-- COMMENT ON TABLE name\_KoAircraft IS 'Kind of aircraft Name';

CREATE TABLE estimate\_price(

   WEEKDAYS\_DAY\_NBR SMALLINT NOT NULL,

   PLAN\_FLIGHTS\_FLIGHT\_NBR SMALLINT  NOT NULL,

   CLASSES\_CLASS\_CODE INT  NOT NULL,

   ESTIMATED\_PRICE DECIMAL(19,4),

   CURRENCIES\_CURR\_CODE CHAR(4) NOT NULL,

   CONSTRAINT PK\_estimate\_price PRIMARY KEY(WEEKDAYS\_DAY\_NBR, PLAN\_FLIGHTS\_FLIGHT\_NBR, CLASSES\_CLASS\_CODE),

   CONSTRAINT FK\_estimate\_price\_PLAN\_DAYFLIGHTS FOREIGN KEY(WEEKDAYS\_DAY\_NBR, PLAN\_FLIGHTS\_FLIGHT\_NBR) REFERENCES PLAN\_DAYFLIGHTS(WEEKDAYS\_DAY\_NBR, PLAN\_FLIGHTS\_FLIGHT\_NBR),

   CONSTRAINT FK\_estimate\_price\_CLASSES FOREIGN KEY(CLASSES\_CLASS\_CODE) REFERENCES CLASSES(CLASS\_CODE),

   CONSTRAINT FK\_estimate\_price\_CURRENCIES FOREIGN KEY(CURRENCIES\_CURR\_CODE) REFERENCES CURRENCIES(CURR\_CODE)

);

-- COMMENT ON TABLE estimate\_price IS 'Estimate ticket price';

CREATE TABLE estimate\_nb\_passengers(

   WEEKDAYS\_DAY\_NBR SMALLINT NOT NULL,

   AIRLINKS\_AIRLINK\_CODE INT  NOT NULL,

   TIMESLOTS\_TIMESLOT\_CODE INT  NOT NULL,

   ESTIMATED\_SEATS SMALLINT NOT NULL,

   CONSTRAINT PK\_estimate\_nb\_passengers PRIMARY KEY(WEEKDAYS\_DAY\_NBR, AIRLINKS\_AIRLINK\_CODE, TIMESLOTS\_TIMESLOT\_CODE),

   CONSTRAINT FK\_estimate\_nb\_passengers\_WEEKDAYS FOREIGN KEY(WEEKDAYS\_DAY\_NBR) REFERENCES WEEKDAYS(DAY\_NBR),

   CONSTRAINT FK\_estimate\_nb\_passengers\_AIRLINKS FOREIGN KEY(AIRLINKS\_AIRLINK\_CODE) REFERENCES AIRLINKS(AIRLINK\_CODE),

   CONSTRAINT FK\_estimate\_nb\_passengers\_TIMESLOTS FOREIGN KEY(TIMESLOTS\_TIMESLOT\_CODE) REFERENCES TIMESLOTS(TIMESLOT\_CODE)

);

-- COMMENT ON TABLE estimate\_nb\_passengers IS 'Estimate number of passengers';

-- COMMENT ON COLUMN estimate\_nb\_passengers.ESTIMATED\_SEATS IS 'Estmated seats quantity (Number)';

CREATE TABLE contain(

   CLASSES\_CLASS\_CODE INT NOT NULL,

   AIRCRAFTS\_AIRCRAFT\_NBR SMALLINT  NOT NULL,

   SEATS INT NOT NULL,

   CONSTRAINT PK\_contain PRIMARY KEY(CLASSES\_CLASS\_CODE, AIRCRAFTS\_AIRCRAFT\_NBR),

   CONSTRAINT FK\_contain\_CLASSES FOREIGN KEY(CLASSES\_CLASS\_CODE) REFERENCES CLASSES(CLASS\_CODE),

   CONSTRAINT FK\_contain\_AIRCRAFTS FOREIGN KEY(AIRCRAFTS\_AIRCRAFT\_NBR) REFERENCES AIRCRAFTS(AIRCRAFT\_NBR)

);

-- COMMENT ON TABLE contain IS 'Contain';

-- COMMENT ON COLUMN contain.SEATS IS 'Nombre de palces disponibles dans la classe';

CREATE TABLE name\_weekday(

   LANGUAGES\_LNG\_CODE\_in CHAR(2) NOT NULL,

   WEEKDAYS\_DAY\_NBR SMALLINT  NOT NULL,

   DAYNAME VARCHAR(50) NOT NULL,

   CONSTRAINT PK\_name\_weekday PRIMARY KEY(LANGUAGES\_LNG\_CODE\_in, WEEKDAYS\_DAY\_NBR),

   CONSTRAINT FK\_name\_weekday\_LANGUAGES\_\_in FOREIGN KEY(LANGUAGES\_LNG\_CODE\_in) REFERENCES LANGUAGES(LNG\_CODE),

   CONSTRAINT FK\_name\_weekday\_WEEKDAYS FOREIGN KEY(WEEKDAYS\_DAY\_NBR) REFERENCES WEEKDAYS(DAY\_NBR)

);

-- COMMENT ON TABLE name\_weekday IS 'Name day';

CREATE TABLE name\_month(

   YEAR\_PERIODS\_YEAR\_PERIOD CHAR(4) NOT NULL,

   MONTH\_PERIODS\_MONTH\_PERIOD CHAR(2)  NOT NULL,

   LANGUAGES\_LNG\_CODE\_in CHAR(2)  NOT NULL,

   MONTHNAME VARCHAR(50) NOT NULL,

   CONSTRAINT PK\_name\_month PRIMARY KEY(YEAR\_PERIODS\_YEAR\_PERIOD, MONTH\_PERIODS\_MONTH\_PERIOD, LANGUAGES\_LNG\_CODE\_in),

   CONSTRAINT FK\_name\_month\_MONTH\_PERIODS FOREIGN KEY(YEAR\_PERIODS\_YEAR\_PERIOD, MONTH\_PERIODS\_MONTH\_PERIOD) REFERENCES MONTH\_PERIODS(YEAR\_PERIOD, MONTH\_PERIOD),

   CONSTRAINT FK\_name\_month\_LANGUAGES\_\_in FOREIGN KEY(LANGUAGES\_LNG\_CODE\_in) REFERENCES LANGUAGES(LNG\_CODE)

);

-- COMMENT ON TABLE name\_month IS 'Name month';

CREATE TABLE name\_role(

   ROLES\_ROLE\_CODE INT NOT NULL,

   LANGUAGES\_LNG\_CODE\_in CHAR(2)  NOT NULL,

   ROLE\_NAME VARCHAR(50) NOT NULL,

   ROLE\_DSCR CLOB NOT NULL,

   CONSTRAINT PK\_name\_role PRIMARY KEY(ROLES\_ROLE\_CODE, LANGUAGES\_LNG\_CODE\_in),

   CONSTRAINT FK\_name\_role\_ROLES FOREIGN KEY(ROLES\_ROLE\_CODE) REFERENCES ROLES(ROLE\_CODE),

   CONSTRAINT FK\_name\_role\_LANGUAGES\_\_in FOREIGN KEY(LANGUAGES\_LNG\_CODE\_in) REFERENCES LANGUAGES(LNG\_CODE)

);

-- COMMENT ON COLUMN name\_role.ROLE\_NAME IS 'Name of role';

SQL STATEMENTS FOR DATA CREATION (SPECIFICALLY INSERT STATEMENTS RELATING TO THE TABLES IN THE DATABASE)

Note: We create and insert synthetic data not based on real individuals into our tables to allow for testing.

INSERT INTO "ISEN142"."IN\_FLIGHT" ("STAFF\_STAFF\_CODE", "YEAR\_PERIODS\_YEAR\_PERIOD", "MONTH\_PERIODS\_MONTH\_PERIOD", "HOURS\_NBR")

VALUES

(1, '2023', '01', 20),

(1, '2023', '02', 15),

(1, '2023', '03', 18),

(1, '2023', '04', 22),

(1, '2023', '05', 25),

(2, '2023', '01', 10),

(2, '2023', '02', 12),

(2, '2023', '03', 8),

(2, '2023', '04', 14),

(2, '2023', '05', 16),

(3, '2023', '01', 18),

(3, '2023', '02', 20),

(3, '2023', '03', 22),

(3, '2023', '04', 24),

(3, '2023', '05', 26),

(4, '2023', '01', 16),

(4, '2023', '02', 14),

(4, '2023', '03', 12),

(4, '2023', '04', 10),

(4, '2023', '05', 8),

(5, '2023', '01', 24),

(5, '2023', '02', 22),

(5, '2023', '03', 20),

(5, '2023', '04', 18),

(5, '2023', '05', 16),

(1, '2023', '06', 28),

(1, '2023', '07', 30),

(1, '2023', '08', 25),

(1, '2023', '09', 22),

(1, '2023', '10', 20),

(1, '2023', '11', 18),

(1, '2023', '12', 16),

(2, '2023', '06', 18),

(2, '2023', '07', 20),

(2, '2023', '08', 22),

(2, '2023', '09', 24),

(2, '2023', '10', 26),

(2, '2023', '11', 28),

(2, '2023', '12', 30),

(3, '2023', '06', 14),

(3, '2023', '07', 16),

(3, '2023', '08', 18),

(3, '2023', '09', 20),

(3, '2023', '10', 22),

(3, '2023', '11', 24),

(3, '2023', '12', 26),

(4, '2023', '06', 20),

(4, '2023', '07', 18),

(4, '2023', '08', 16),

(4, '2023', '09', 14),

(4, '2023', '10', 12),

(4, '2023', '11', 10),

(4, '2023', '12', 8),

(5, '2023', '06', 12),

(5, '2023', '07', 14),

(5, '2023', '08', 16),

(5, '2023', '09', 18),

(5, '2023', '10', 20),

(5, '2023', '11', 22),

(5, '2023', '12', 24);

INSERT INTO "ISEN142"."MONTH\_PERIODS" ("YEAR\_PERIOD", "MONTH\_PERIOD")

VALUES

('2023', '01'),

('2023', '02'),

('2023', '03'),

('2023', '04'),

('2023', '05'),

('2023', '06'),

('2023', '07'),

('2023', '08'),

('2023', '09'),

('2023', '10'),

('2023', '11'),

('2023', '12');

INSERT INTO "ISEN142"."LANGUAGES" ("LNG\_CODE")

VALUES

('EN'),

('FR');

INSERT INTO  "ISEN142"."NAME\_LANGUAGE" ("LANGUAGES\_LNG\_CODE\_IN","LANGUAGES\_LNG\_CODE\_OF","LNGNAME")

VALUES

('FR','EN','Anglais'),

('EN','FR','French');

INSERT INTO "ISEN142"."AIRPORTS" ("AIRPORT\_CODE","CURRENCIES\_CURR\_CODE")

VALUES

('JFK', 'USD'),

('LHR', 'GBP'),

('CDG', 'EUR'),

('BNA', 'USD'),

('CVG', 'GBP'),

('ECP', 'EUR');

INSERT INTO "ISEN142"."NAME\_AIRPORT" ("AIRPORTS\_AIRPORT\_CODE","LANGUAGES\_LNG\_CODE\_IN","ARP\_NAME")

VALUES

  ('JFK', 'EN', 'John F. Kennedy International Airport'),

  ('JFK', 'FR', 'Aéroport international John-F.-Kennedy'),

  ('LHR', 'EN', 'London Heathrow Airport'),

  ('LHR', 'FR', 'Aéroport de Londres Heathrow'),

  ('CDG', 'EN', 'Charles de Gaulle Airport'),

  ('CDG', 'FR', 'Aéroport Charles-de-Gaulle');

INSERT INTO "ISEN142"."NAME\_CITY" ("CITIES\_CITY\_CODE","LANGUAGES\_LNG\_CODE\_IN","CITYNAME")

VALUES

  ('City1', 'EN', 'New York'),

  ('City1', 'FR', 'Nouveau :) York'),

  ('City2', 'EN', 'London'),

  ('City2', 'FR', 'Londres'),

  ('City3', 'EN', 'Paris'),

  ('City3', 'FR', 'Paris');

INSERT INTO "ISEN142"."NAME\_KOAIRCRAFT" ("LANGUAGES\_LNG\_CODE\_IN","KOAIRCRAFT\_IATA","KOACRNAME")

  VALUES

  ('EN', 'AC01', 'Boeing 747'),

  ('FR', 'AC01', 'Boeing 747-FR name'),

  ('EN', 'AC02', 'Airbus A380'),

  ('FR', 'AC02', 'AirbusFR A380-FR name'),

  ('EN', 'AC03', 'Concorde'),

  ('FR', 'AC03', 'Concorde-FR name'),

  ('EN', 'AC04', 'Boeing 777'),

  ('FR', 'AC04', 'Boeing 777-FR name'),

  ('EN', 'AC05', 'Airbus A350'),

  ('FR', 'AC05', 'Airbus A350-FR name');

-- English and French month names for 2023

INSERT INTO "ISEN142"."NAME\_MONTH" ("YEAR\_PERIODS\_YEAR\_PERIOD", "MONTH\_PERIODS\_MONTH\_PERIOD", "LANGUAGES\_LNG\_CODE\_IN", "MONTHNAME")

VALUES ('2023', '01', 'EN', 'January'),

       ('2023', '02', 'EN', 'February'),

       ('2023', '03', 'EN', 'March'),

       ('2023', '04', 'EN', 'April'),

       ('2023', '05', 'EN', 'May'),

       ('2023', '06', 'EN', 'June'),

       ('2023', '07', 'EN', 'July'),

       ('2023', '08', 'EN', 'August'),

       ('2023', '09', 'EN', 'September'),

       ('2023', '10', 'EN', 'October'),

       ('2023', '11', 'EN', 'November'),

       ('2023', '12', 'EN', 'December'),

       ('2023', '01', 'FR', 'Janvier'),

       ('2023', '02', 'FR', 'Février'),

       ('2023', '03', 'FR', 'Mars'),

       ('2023', '04', 'FR', 'Avril'),

       ('2023', '05', 'FR', 'Mai'),

       ('2023', '06', 'FR', 'Juin'),

       ('2023', '07', 'FR', 'Juillet'),

       ('2023', '08', 'FR', 'Août'),

       ('2023', '09', 'FR', 'Septembre'),

       ('2023', '10', 'FR', 'Octobre'),

       ('2023', '11', 'FR', 'Novembre'),

       ('2023', '12', 'FR', 'Décembre'),

        ('2024', '01', 'EN', 'January'),

       ('2024', '02', 'EN', 'February'),

       ('2024', '03', 'EN', 'March'),

       ('2024', '04', 'EN', 'April'),

       ('2024', '05', 'EN', 'May'),

       ('2024', '06', 'EN', 'June'),

       ('2024', '07', 'EN', 'July'),

       ('2024', '08', 'EN', 'August'),

       ('2024', '09', 'EN', 'September'),

       ('2024', '10', 'EN', 'October'),

       ('2024', '11', 'EN', 'November'),

       ('2024', '12', 'EN', 'December'),

       ('2024', '01', 'FR', 'Janvier'),

       ('2024', '02', 'FR', 'Février'),

       ('2024', '03', 'FR', 'Mars'),

       ('2024', '04', 'FR', 'Avril'),

       ('2024', '05', 'FR', 'Mai'),

       ('2024', '06', 'FR', 'Juin'),

       ('2024', '07', 'FR', 'Juillet'),

       ('2024', '08', 'FR', 'Août'),

       ('2024', '09', 'FR', 'Septembre'),

       ('2024', '10', 'FR', 'Octobre'),

       ('2024', '11', 'FR', 'Novembre'),

       ('2024', '12', 'FR', 'Décembre');

-- English and French weekday names

INSERT INTO "ISEN142"."NAME\_WEEKDAY" ("LANGUAGES\_LNG\_CODE\_IN", "WEEKDAYS\_DAY\_NBR", "DAYNAME")

VALUES ('EN', 1, 'Monday'),

       ('EN', 2, 'Tuesday'),

       ('EN', 3, 'Wednesday'),

       ('EN', 4, 'Thursday'),

       ('EN', 5, 'Friday'),

       ('EN', 6, 'Saturday'),

       ('EN', 7, 'Sunday'),

       ('FR', 1, 'Lundi'),

       ('FR', 2, 'Mardi'),

       ('FR', 3, 'Mercredi'),

       ('FR', 4, 'Jeudi'),

       ('FR', 5, 'Vendredi'),

       ('FR', 6, 'Samedi'),

       ('FR', 7, 'Dimanche');

-- English and French airport staff role names and descriptions

INSERT INTO "ISEN142"."NAME\_ROLE" ("ROLES\_ROLE\_CODE", "LANGUAGES\_LNG\_CODE\_IN", "ROLE\_NAME", "ROLE\_DSCR")

VALUES (1, 'EN', 'Airport Security Officer', 'Responsible for ensuring the safety and security of passengers, staff, and airport facilities by monitoring security checkpoints, conducting searches, and implementing security procedures.'),

       (2, 'EN', 'Baggage Handler', 'Responsible for loading and unloading baggage onto and from aircraft, ensuring that luggage is transferred safely and efficiently between flights.'),

       (3, 'EN', 'Flight Attendant', 'Responsible for ensuring the safety and comfort of passengers during flights, providing assistance, serving meals and beverages, and delivering safety demonstrations.'),

       (4, 'EN', 'Airport Operations Manager', 'Responsible for overseeing the day-to-day operations of the airport, including managing staff, coordinating with airlines and ground services, and ensuring compliance with safety and regulatory requirements.'),

       (5, 'EN', 'Customer Service Representative', 'Responsible for assisting passengers with inquiries, ticketing, baggage handling, and other customer service-related tasks to ensure a positive airport experience.'),

       (1, 'FR', 'Agent de sécurité de l aéroport', 'Responsable de garantir la sécurité des passagers, du personnel et des installations aéroportuaires en surveillant les points de contrôle de sécurité, en effectuant des fouilles et en mettant en œuvre des procédures de sécurité.'),

       (2, 'FR', 'Manutentionnaire de bagages', 'Responsable du chargement et du déchargement des bagages dans les avions, en veillant à ce que les bagages soient transférés en toute sécurité et efficacement entre les vols.'),

       (3, 'FR', 'Agent de bord', 'Responsable de garantir la sécurité et le confort des passagers pendant les vols, en fournissant de l aide, en servant des repas et des boissons, et en effectuant des démonstrations de sécurité.'),

       (4, 'FR', 'Gestionnaire des opérations aéroportuaires', 'Responsable de superviser les opérations quotidiennes de l aéroport, y compris la gestion du personnel, la coordination avec les compagnies aériennes et les services au sol, et la garantie du respect des exigences en matière de sécurité et de réglementation.'),

       (5, 'FR', 'Représentant du service clientèle', 'Responsable d aider les passagers avec les demandes de renseignements, l émission de billets, la manipulation des bagages, et d autres tâches liées au service clientèle pour assurer une expérience aéroportuaire positive.');

-- Inserting airport license types

INSERT INTO "ISEN142"."KOLICENSE" ("KOLIC\_CODE", "KOLIC\_NAME")

VALUES

  (1, 'Commercial Airline Pilot License - level 1'),

  (2, 'Commercial Airline Pilot License - level 2'),

  (3, 'Commercial Airline Pilot License - level 3'),

  (4, 'Commercial Airline Pilot License - level 4'),

  (5, 'Commercial Airline Pilot License - level 5');

INSERT INTO "ISEN142"."CREWS" ("CREW\_CODE","CREW\_NAME")

VALUES

(1, 'Crew\_1'),

(2, 'Crew\_2'),

(3, 'Crew\_3'),

(4, 'Crew\_4'),

(5, 'Crew\_5'),

(6, 'Crew\_6'),

(7, 'Crew\_7'),

(8, 'Crew\_8'),

(9, 'Crew\_9'),

(10, 'Crew\_10');

INSERT INTO "ISEN142"."KOCREW" ("KOCREW\_CODE","KOCREW\_NAME")

VALUES

(1, 'Flight Crew'),

(2, 'Ground Crew'),

(3, 'Maintenance Crew'),

(4, 'Security Team'),

(5, 'Cleaning Crew'),

(6, 'Catering Staff'),

(7, 'Airport Operations Crew');

INSERT INTO "ISEN142"."ISA\_KOCREW" ("KOCREW\_KOCREW\_CODE","CREWS\_CREW\_CODE")

VALUES

(1, 2),

(2, 3),

(3, 5),

(4, 7),

(5, 9);

INSERT INTO "ISEN142"."ACCEPT" ("AIRPORTS\_AIRPORT\_CODE","KOAIRCRAFT\_IATA")

VALUES

('JFK', 'AC01'),

('LHR', 'AC01'),

('CDG', 'AC01'),

('BNA', 'AC02'),

('CVG', 'AC02'),

('ECP', 'AC02'),

('JFK', 'AC03'),

('LHR', 'AC03'),

('CDG', 'AC03'),

('BNA', 'AC04'),

('CVG', 'AC04'),

('ECP', 'AC05'),

('JFK', 'AC05'),

('LHR', 'AC06'),

('CDG', 'AC06'),

('BNA', 'AC07'),

('CVG', 'AC07'),

('ECP', 'AC08'),

('JFK', 'AC08'),

('LHR', 'AC09'),

('CDG', 'AC09'),

('BNA', 'AC10'),

('CVG', 'AC10'),

('ECP', 'AC11'),

('JFK', 'AC11'),

('LHR', 'AC12'),

('CDG', 'AC12'),

('BNA', 'AC13'),

('CVG', 'AC13'),

('ECP', 'AC14'),

('JFK', 'AC14'),

('LHR', 'AC15'),

('CDG', 'AC15'),

('BNA', 'AC16'),

('CVG', 'AC16'),

('ECP', 'AC17'),

('JFK', 'AC17'),

('LHR', 'AC18'),

('CDG', 'AC18'),

('BNA', 'AC19'),

('CVG', 'AC19'),

('ECP', 'AC20'),

('JFK', 'AC20');

INSERT INTO "ISEN142"."ASSIGNMENTS" ("STAFF\_STAFF\_CODE","ASSIGNT\_NBR","STARTING","ENDING","ROLES\_ROLE\_CODE","CREWS\_CREW\_CODE")

VALUES

(1, 1, TIMESTAMP '2024-03-19 08:00:00', TIMESTAMP '2024-03-19 16:00:00', 1, 1),

(2, 2, TIMESTAMP '2024-03-20 09:00:00', TIMESTAMP '2024-03-20 17:00:00', 2, 2),

(3, 3, TIMESTAMP '2024-03-21 07:00:00', TIMESTAMP '2024-03-21 15:00:00', 3, 3),

(4, 4, TIMESTAMP '2024-03-22 10:00:00', TIMESTAMP '2024-03-22 18:00:00', 4, 4),

(5, 5, TIMESTAMP '2024-03-23 11:00:00', TIMESTAMP '2024-03-23 19:00:00', 5, 5);

INSERT INTO "ISEN142"."AIRLINKS" ("AIRLINK\_CODE","AIRLINK\_DISTANCE","AIRPORTS\_AIRPORT\_CODE\_FROM","AIRPORTS\_AIRPORT\_CODE\_TO")

VALUES

(1, 5555, 'JFK', 'LHR'),

(2, 5849, 'JFK', 'CDG'),

(3, 1231, 'JFK', 'BNA'),

(4, 947, 'JFK', 'CVG'),

(5, 7128, 'JFK', 'ECP'),

(6, 348, 'LHR', 'CDG'),

(7, 6729, 'LHR', 'BNA'),

(8, 6377, 'LHR', 'CVG'),

(10, 333, 'CDG', 'ECP');

INSERT INTO "ISEN142"."PLAN\_FLIGHTS" ("FLIGHT\_NBR","DEPARTURE","FREQUENCY","AIRLINKS\_AIRLINK\_CODE")

VALUES

(1, '08:00:00', 7, 1),

(2, '09:30:00', 3, 2),

(3, '12:45:00', 5, 3),

(4, '14:20:00', 2, 4),

(5, '17:10:00', 4, 5),

(6, '17:10:00', 4, 10);

INSERT INTO "ISEN142"."PLAN\_DAYFLIGHTS" ("WEEKDAYS\_DAY\_NBR","PLAN\_FLIGHTS\_FLIGHT\_NBR","PLAN\_SEATS","PLAN\_DURATION","PLAN\_ARR\_TIME","KOAIRCRAFT\_IATA")

VALUES

(1, 1, 150, '02:30:00', '10:30:00', 'AC01'),

(2, 2, 200, '03:15:00', '11:45:00', 'AC02'),

(3, 3, 180, '02:45:00', '14:30:00', 'AC03'),

(4, 4, 160, '02:00:00', '16:20:00', 'AC04'),

(5, 5, 220, '04:00:00', '18:45:00', 'AC05'),

(6, 1, 170, '02:30:00', '09:45:00', 'AC06'),

(7, 2, 190, '03:15:00', '12:30:00', 'AC07');

INSERT INTO "ISEN142"."ROUTES" ("ROUTE\_CODE","ROUTE\_DURATION","KOAIRCRAFT\_IATA","AIRLINKS\_AIRLINK\_CODE")

VALUES

(1, '03:30:00', 'AC01', 1),

(2, '04:15:00', 'AC02', 2),

(3, '02:45:00', 'AC03', 3),

(4, '02:30:00', 'AC04', 4),

(5, '03:00:00', 'AC05', 5),

(6, '02:20:00', 'AC06', 6),

(7, '04:00:00', 'AC07', 7),

(8, '03:45:00', 'AC08', 8);

INSERT INTO "ISEN142"."PLAN\_STOPOVERS" ("WEEKDAYS\_DAY\_NBR","PLAN\_FLIGHTS\_FLIGHT\_NBR","STOPOVER\_NBR","STOPOVER\_DURATION","ROUTES\_ROUTE\_CODE")

VALUES

(1, 1, 1, '00:25:00', 1),

(2, 2, 2, '00:30:00', 2),

(3, 3, 3, '00:35:00', 3),

(4, 4, 4, '00:40:00', 1),

(5, 5, 5, '00:30:00', 2),

(6, 1, 1, '00:20:00', 3),

(7, 2, 2, '00:35:00', 1);

INSERT INTO "ISEN142"."EFF\_FLIGHTS" ("WEEKDAYS\_DAY\_NBR","PLAN\_FLIGHTS\_FLIGHT\_NBR","PLAN\_DEP\_DATE","RES\_SEATS\_QTY","OCC\_SEATS\_QTY","REAL\_DEP\_TIME","REAL\_ARR\_TIME","LOADED\_FUEL","CREWS\_CREW\_CODE","AIRCRAFTS\_AIRCRAFT\_NBR")

VALUES

(1, 1, '2024-03-19 08:00:00', 150, 120, '2024-03-19 07:55:00', '2024-03-19 10:35:00', 5000, 1, 1),

(2, 2, '2024-03-20 09:30:00', 180, 160, '2024-03-20 09:32:00', '2024-03-20 11:58:00', 6000, 2, 2),

(3, 3, '2024-03-21 11:00:00', 200, 180, '2024-03-21 10:58:00', '2024-03-21 13:32:00', 7000, 3, 3),

(4, 4, '2024-03-22 13:30:00', 170, 150, '2024-03-22 13:27:00', '2024-03-22 16:02:00', 5500, 4, 4),

(5, 5, '2024-03-23 15:00:00', 190, 170, '2024-03-23 14:57:00', '2024-03-23 17:33:00', 6500, 5, 5),

(6, 1, '2024-03-19 14:45:00', 160, 140, '2024-03-19 14:43:00', '2024-03-19 17:20:00', 5800, 6, 6);

INSERT INTO "ISEN142"."EFF\_STOPOVERS" ("WEEKDAYS\_DAY\_NBR\_1","PLAN\_FLIGHTS\_FLIGHT\_NBR\_1","EFF\_FLIGHTS\_PLAN\_DEP\_DATE","ESO\_NBR","REAL\_TRAVEL\_DUR","REAL\_LANDING\_TIME","REAL\_TAKEOFF\_TIME","LOADED\_FUEL","WEEKDAYS\_DAY\_NBR","PLAN\_FLIGHTS\_FLIGHT\_NBR","PLAN\_STOPOVERS\_STOPOVER\_NBR")

VALUES

(1, 1, '2024-03-19 08:00:00', 1, '00:35:00', '2024-03-19 15:20:00', '2024-03-19 14:45:00', 1000, 1, 1, 1),

(2, 2, '2024-03-20 09:30:00', 2, '00:40:00', '2024-03-20 17:10:00', '2024-03-20 16:30:00', 1100, 2, 2, 2),

(3, 3, '2024-03-21 11:00:00', 3, '00:30:00', '2024-03-21 18:45:00', '2024-03-21 18:15:00', 950, 3, 3, 3);

INSERT INTO "ISEN142"."INCLUDE" ("KOCREW\_KOCREW\_CODE","ROLES\_ROLE\_CODE","NBR\_IN\_ROLE")

VALUES

(1, 1, 1),

(2, 2, 2),

(3, 3, 3),

(4, 4, 4),

(5, 5, 5)

INSERT INTO "ISEN142"."HAVE\_QUALIF" ("STAFF\_STAFF\_CODE","QUALIFICATIONS\_QUALIF\_CODE")

  VALUES

(1, 1),

(2, 2),

(3, 3)

INSERT INTO "ISEN142"."EVENTS" ("STAFF\_STAFF\_CODE","EVENT\_NBR","EVENT\_LABEL","EVENT\_DSCR","EVENT\_DATE","EVENT\_TIME","STATUS\_STATUS\_CODE","KOEVENT\_KOEVT\_CODE")

VALUES

(1, 1, 'Meeting', 'Team meeting to discuss project updates.', TIMESTAMP '2024-03-19 14:30:00', TIME '14:30:00', 1, 1),

(2, 2, 'Training Session', 'Training session on new software tools.', TIMESTAMP '2024-03-20 09:00:00', TIME '09:00:00', 2, 2),

(3, 3, 'Workshop', 'Workshop on agile development methodologies.', TIMESTAMP '2024-03-21 13:00:00', TIME '13:00:00', 3, 3)

INSERT INTO "ISEN142"."CLASSES" ("CLASS\_CODE","CLASS\_NAME")

VALUES

(1, 'Class1'),

(2, 'Class2'),

(3, 'Class3');

INSERT INTO "ISEN142"."ESTIMATE\_PRICE" ("WEEKDAYS\_DAY\_NBR","PLAN\_FLIGHTS\_FLIGHT\_NBR","CLASSES\_CLASS\_CODE","ESTIMATED\_PRICE","CURRENCIES\_CURR\_CODE")

VALUES

(1, 1, 1, 250.00, 'USD'),

(2, 2, 2, 180.50, 'EUR'),

(3, 3, 3, 300.25, 'GBP');

INSERT

INTO  "ISEN142"."CONTAIN" ("CLASSES\_CLASS\_CODE","AIRCRAFTS\_AIRCRAFT\_NBR","SEATS")

VALUES

(1, 1, 300),

(2, 2, 350),

(3, 3, 400);

INSERT

INTO  "ISEN142"."MAY\_HAVE" ("ROLES\_ROLE\_CODE","STAFF\_STAFF\_CODE")

VALUES

(1, 1),

(2, 2),

(3, 3);

INSERT

INTO  "ISEN142"."ALLOW\_PILOT" ("KOLICENSE\_KOLIC\_CODE","KOAIRCRAFT\_IATA")

VALUES

(1,'AC01'),

(2,'AC02'),

(3,'AC03');

INSERT

INTO  "ISEN142"."BIND" ("STAFF\_STAFF\_CODE","EVENTS\_EVENT\_NBR","DOC\_DOC\_CODE")

VALUES

(1,1,1),

(2,2,2),

(3,3,3);

INSERT INTO "ISEN142"."CHECKUPS" ("CHECKUP\_NBR", "CHECKUP\_DATE", "STAFF\_STAFF\_CODE\_LAST\_ONLY")

VALUES

(2, TIMESTAMP '2024-03-19 12:30:00', 2),

(3, TIMESTAMP '2024-03-20 08:45:00', 3);

INSERT INTO "ISEN142"."ESTIMATE\_NB\_PASSENGERS" ("WEEKDAYS\_DAY\_NBR", "AIRLINKS\_AIRLINK\_CODE", "TIMESLOTS\_TIMESLOT\_CODE", "ESTIMATED\_SEATS")

VALUES

(1, 1, 1, 150),

(2, 2, 2, 200),

(3, 3, 3, 180),

(4, 4, 4, 160),

(5, 5, 5, 190),

(6, 6, 6, 170),

(7, 7, 7, 220);

INSERT INTO "ISEN142"."ISA\_KOCREW" ("KOCREW\_KOCREW\_CODE", "CREWS\_CREW\_CODE")

VALUES

(1, 1),

(2, 2),

(3, 3),

(4, 4),

(5, 5);

INSERT INTO "ISEN142"."NEED" ("KOCREW\_KOCREW\_CODE", "KOAIRCRAFT\_IATA")

VALUES

(1, 'AC01'),

(2, 'AC02'),

(3, 'AC03'),

(4, 'AC04'),

(5, 'AC05');

INSERT INTO "ISEN142"."OBTAIN" ("STAFF\_STAFF\_CODE", "KOLICENSE\_KOLIC\_CODE", "OBTAINED")

VALUES

(1, 1, TIMESTAMP '2024-03-19 10:30:00'),

(2, 2, TIMESTAMP '2024-03-19 11:45:00'),

(3, 3, TIMESTAMP '2024-03-19 12:15:00');

INSERT INTO SERVE ("CITIES\_CITY\_CODE","AIRPORTS\_AIRPORT\_CODE")

VALUES

('City1','JFK'),

('City2','LHR'),

('City3','CDG');

INSERT INTO REQUIRE ("KOLICENSE\_KOLIC\_CODE","QUALIFICATIONS\_QUALIF\_CODE")

VALUES

(1,1),

(2,2),

(3,3);

INSERT INTO PASSENGERS ("PSGR\_CODE", "LNAME", "FNAME")

VALUES

(1, 'Smith', 'John'),

(2, 'Doe', 'Jane'),

(3, 'Johnson', 'Michael'),

(4, 'Williams', 'Emily'),

(5, 'Brown', 'David'),

(6, 'Jones', 'Sarah'),

(7, 'Garcia', 'Daniel'),

(8, 'Martinez', 'Maria'),

(9, 'Miller', 'James'),

(10, 'Davis', 'Jessica');

INSERT INTO TICKETS ("WEEKDAYS\_DAY\_NBR", "PLAN\_FLIGHTS\_FLIGHT\_NBR", "EFF\_FLIGHTS\_PLAN\_DEP\_DATE", "SEAT", "PRICE", "PASSENGERS\_PSGR\_CODE", "CURRENCIES\_CURR\_CODE", "CLASSES\_CLASS\_CODE")

VALUES

(1, 1, '12:00:00', 'A1', 150.00, 1, 'USD', 1),

(2, 2, '12:00:00', 'B2', 200.00, 2, 'EUR', 2),

(3, 3, '12:00:00', 'C3', 180.00, 3, 'USD', 3),

(4, 1, '12:00:00', 'D4', 170.00, 4, 'EUR', 1),

(5, 2, '12:00:00', 'E5', 220.00, 5, 'USD', 2),

(6, 3, '12:00:00', 'F6', 190.00, 6, 'EUR', 3),

(7, 1, '12:00:00', 'G7', 160.00, 7, 'USD', 1),

(1, 2, '12:00:00', 'H8', 210.00, 8, 'EUR', 2),

(2, 3, '12:00:00', 'I9', 230.00, 9, 'USD', 3),

(3, 1, '12:00:00', 'J10', 180.00, 10, 'EUR', 1);

QUESTIONS

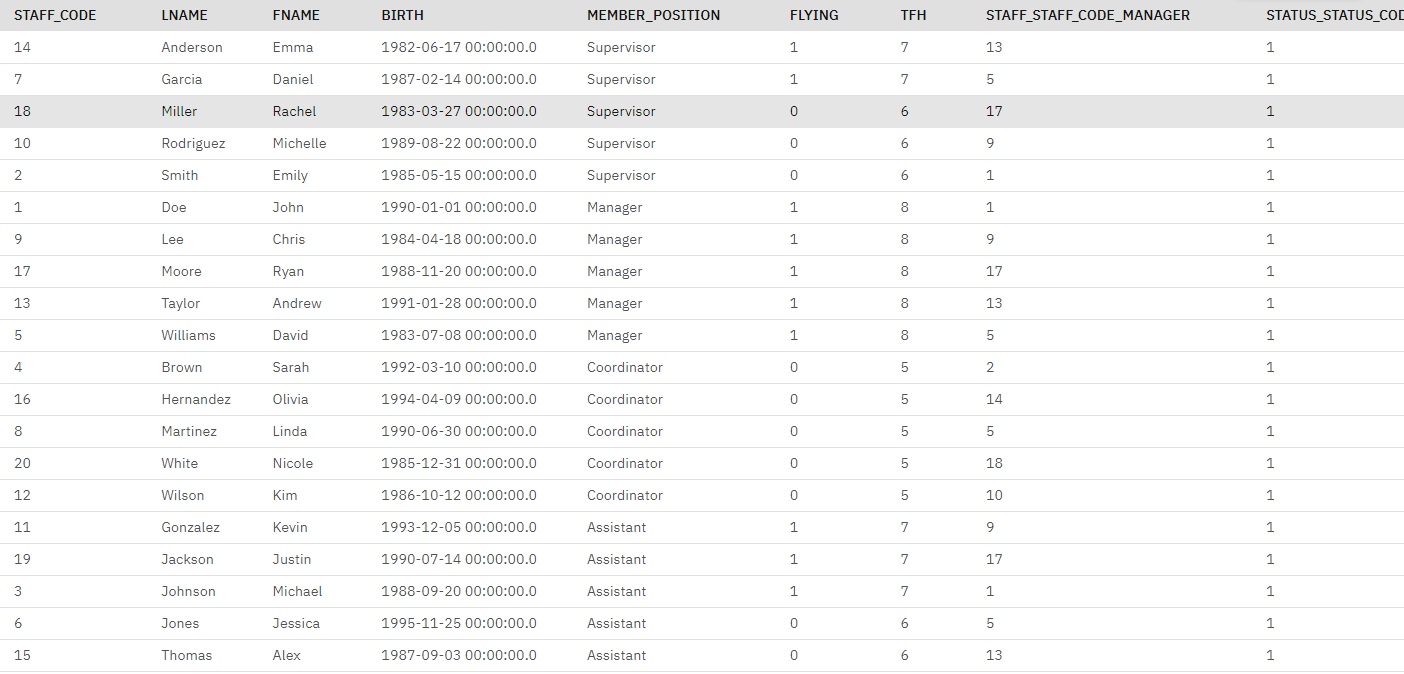
1. Give a list of all staff members sorted by position descending and last name ascending

SELECT \*

FROM STAFF

ORDER BY MEMBER\_POSITION DESC,

         LNAME ASC;



1. For each staff member give total flight hours per month of last year (2023)

SELECT s.STAFF\_CODE,

       s.LNAME,

       s.FNAME,

       YEAR\_PERIOD,

       MONTH\_PERIOD,

       SUM(in\_flight.HOURS\_NBR) AS TOTAL\_MONTHLY\_FLIGHT\_HOURS

FROM STAFF s

INNER JOIN in\_flight ON in\_flight.STAFF\_STAFF\_CODE = s.STAFF\_CODE

INNER JOIN MONTH\_PERIODS ON in\_flight.YEAR\_PERIODS\_YEAR\_PERIOD MONTH\_PERIODS.YEAR\_PERIOD

AND in\_flight.MONTH\_PERIODS\_MONTH\_PERIOD = MONTH\_PERIODS.MONTH\_PERIOD

WHERE YEAR\_PERIOD = '2023'

GROUP BY s.STAFF\_CODE,

         s.LNAME,

         s.FNAME,

         YEAR\_PERIOD,

         MONTH\_PERIOD

ORDER BY s.STAFF\_CODE,

         YEAR\_PERIOD,

         MONTH\_PERIOD;

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STAFF\_CODE | LNAME | FNAME | YEAR\_PERIOD | MONTH\_PERIOD | TOTAL\_MONTHLY\_FLIGHT\_HOURS |
| 1 | Doe | John | 2023 | 1 | 20 |
| 1 | Doe | John | 2023 | 2 | 15 |
| 1 | Doe | John | 2023 | 3 | 18 |
| 1 | Doe | John | 2023 | 4 | 22 |
| 1 | Doe | John | 2023 | 5 | 25 |
| 1 | Doe | John | 2023 | 6 | 28 |
| 1 | Doe | John | 2023 | 7 | 30 |
| 1 | Doe | John | 2023 | 8 | 25 |
| 1 | Doe | John | 2023 | 9 | 22 |
| 1 | Doe | John | 2023 | 10 | 20 |
| 1 | Doe | John | 2023 | 11 | 18 |
| 1 | Doe | John | 2023 | 12 | 16 |
| 2 | Smith | Emily | 2023 | 1 | 10 |
| 2 | Smith | Emily | 2023 | 2 | 12 |
| 2 | Smith | Emily | 2023 | 3 | 8 |
| 2 | Smith | Emily | 2023 | 4 | 14 |
| 2 | Smith | Emily | 2023 | 5 | 16 |
| 2 | Smith | Emily | 2023 | 6 | 18 |
| 2 | Smith | Emily | 2023 | 7 | 20 |
| 2 | Smith | Emily | 2023 | 8 | 22 |
| 2 | Smith | Emily | 2023 | 9 | 24 |
| 2 | Smith | Emily | 2023 | 10 | 26 |
| 2 | Smith | Emily | 2023 | 11 | 28 |
| 2 | Smith | Emily | 2023 | 12 | 30 |
| 3 | Johnson | Michael | 2023 | 1 | 18 |
| 3 | Johnson | Michael | 2023 | 2 | 20 |
| 3 | Johnson | Michael | 2023 | 3 | 22 |
| 3 | Johnson | Michael | 2023 | 4 | 24 |
| 3 | Johnson | Michael | 2023 | 5 | 26 |
| 3 | Johnson | Michael | 2023 | 6 | 14 |
| 3 | Johnson | Michael | 2023 | 7 | 16 |
| 3 | Johnson | Michael | 2023 | 8 | 18 |
| 3 | Johnson | Michael | 2023 | 9 | 20 |
| 3 | Johnson | Michael | 2023 | 10 | 22 |
| 3 | Johnson | Michael | 2023 | 11 | 24 |
| 3 | Johnson | Michael | 2023 | 12 | 26 |
| 4 | Brown | Sarah | 2023 | 1 | 16 |
| 4 | Brown | Sarah | 2023 | 2 | 14 |
| 4 | Brown | Sarah | 2023 | 3 | 12 |
| 4 | Brown | Sarah | 2023 | 4 | 10 |
| 4 | Brown | Sarah | 2023 | 5 | 8 |
| 4 | Brown | Sarah | 2023 | 6 | 20 |
| 4 | Brown | Sarah | 2023 | 7 | 18 |
| 4 | Brown | Sarah | 2023 | 8 | 16 |
| 4 | Brown | Sarah | 2023 | 9 | 14 |
| 4 | Brown | Sarah | 2023 | 10 | 12 |
| 4 | Brown | Sarah | 2023 | 11 | 10 |
| 4 | Brown | Sarah | 2023 | 12 | 8 |
| 5 | Williams | David | 2023 | 1 | 24 |
| 5 | Williams | David | 2023 | 2 | 22 |
| 5 | Williams | David | 2023 | 3 | 20 |
| 5 | Williams | David | 2023 | 4 | 18 |
| 5 | Williams | David | 2023 | 5 | 16 |
| 5 | Williams | David | 2023 | 6 | 12 |
| 5 | Williams | David | 2023 | 7 | 14 |
| 5 | Williams | David | 2023 | 8 | 16 |
| 5 | Williams | David | 2023 | 9 | 18 |
| 5 | Williams | David | 2023 | 10 | 20 |
| 5 | Williams | David | 2023 | 11 | 22 |
| 5 | Williams | David | 2023 | 12 | 24 |

3. For each staff member give total flight hours per year

SELECT STAFF.STAFF\_CODE,

       STAFF.LNAME,

       STAFF.FNAME,

       SUM(in\_flight.HOURS\_NBR) AS TOTAL\_FLIGHT\_HOURS\_2023

FROM STAFF

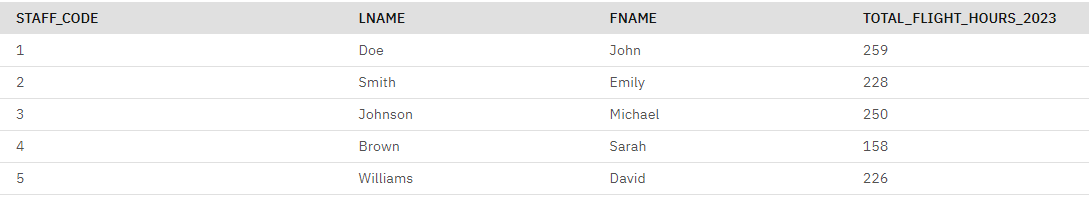
INNER JOIN in\_flight ON STAFF.STAFF\_CODE = in\_flight.STAFF\_STAFF\_CODE

WHERE YEAR\_PERIODS\_YEAR\_PERIOD = '2023'

GROUP BY STAFF.STAFF\_CODE,

         STAFF.LNAME,

         STAFF.FNAME;



4. List all staff members that manage persons and their managed persons

SELECT M.STAFF\_CODE AS Manager\_Code,

       M.LNAME AS Manager\_LastName,

       M.FNAME AS Manager\_FirstName,

       M.BIRTH AS Manager\_BirthDate,

       M.MEMBER\_POSITION AS Manager\_Position,

       M.FLYING AS Manager\_Flying,

       M.TFH AS Manager\_TotalFlightHours,

       S.STAFF\_CODE AS Managed\_Code,

       S.LNAME AS Managed\_LastName,

       S.FNAME AS Managed\_FirstName,

       S.BIRTH AS Managed\_BirthDate,

       S.MEMBER\_POSITION AS Managed\_Position,

       S.FLYING AS Managed\_Flying,

       S.TFH AS Managed\_TotalFlightHours

FROM STAFF M

JOIN STAFF S ON M.STAFF\_CODE = S.STAFF\_STAFF\_CODE\_manager

ORDER BY M.STAFF\_CODE,

         S.STAFF\_CODE;

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| MANAGER\_CODE | MANAGER\_LASTNAME | MANAGER\_FIRSTNAME | MANAGER\_BIRTHDATE | MANAGER\_POSITION | MANAGER\_FLYING | MANAGER\_TOTALFLIGHTHOURS | MANAGED\_CODE | MANAGED\_LASTNAME | MANAGED\_FIRSTNAME | MANAGED\_BIRTHDATE | MANAGED\_POSITION | MANAGED\_FLYING | MANAGED\_TOTALFLIGHTHOURS |
| 1 | Doe | John | 00:00.0 | Manager | 1 | 8 | 1 | Doe | John | 00:00.0 | Manager | 1 | 8 |
| 1 | Doe | John | 00:00.0 | Manager | 1 | 8 | 2 | Smith | Emily | 00:00.0 | Supervisor | 0 | 6 |
| 1 | Doe | John | 00:00.0 | Manager | 1 | 8 | 3 | Johnson | Michael | 00:00.0 | Assistant | 1 | 7 |
| 2 | Smith | Emily | 00:00.0 | Supervisor | 0 | 6 | 4 | Brown | Sarah | 00:00.0 | Coordinator | 0 | 5 |
| 5 | Williams | David | 00:00.0 | Manager | 1 | 8 | 5 | Williams | David | 00:00.0 | Manager | 1 | 8 |
| 5 | Williams | David | 00:00.0 | Manager | 1 | 8 | 6 | Jones | Jessica | 00:00.0 | Assistant | 0 | 6 |
| 5 | Williams | David | 00:00.0 | Manager | 1 | 8 | 7 | Garcia | Daniel | 00:00.0 | Supervisor | 1 | 7 |
| 5 | Williams | David | 00:00.0 | Manager | 1 | 8 | 8 | Martinez | Linda | 00:00.0 | Coordinator | 0 | 5 |
| 9 | Lee | Chris | 00:00.0 | Manager | 1 | 8 | 9 | Lee | Chris | 00:00.0 | Manager | 1 | 8 |
| 9 | Lee | Chris | 00:00.0 | Manager | 1 | 8 | 10 | Rodriguez | Michelle | 00:00.0 | Supervisor | 0 | 6 |
| 9 | Lee | Chris | 00:00.0 | Manager | 1 | 8 | 11 | Gonzalez | Kevin | 00:00.0 | Assistant | 1 | 7 |
| 10 | Rodriguez | Michelle | 00:00.0 | Supervisor | 0 | 6 | 12 | Wilson | Kim | 00:00.0 | Coordinator | 0 | 5 |
| 13 | Taylor | Andrew | 00:00.0 | Manager | 1 | 8 | 13 | Taylor | Andrew | 00:00.0 | Manager | 1 | 8 |
| 13 | Taylor | Andrew | 00:00.0 | Manager | 1 | 8 | 14 | Anderson | Emma | 00:00.0 | Supervisor | 1 | 7 |
| 13 | Taylor | Andrew | 00:00.0 | Manager | 1 | 8 | 15 | Thomas | Alex | 00:00.0 | Assistant | 0 | 6 |
| 14 | Anderson | Emma | 00:00.0 | Supervisor | 1 | 7 | 16 | Hernandez | Olivia | 00:00.0 | Coordinator | 0 | 5 |
| 17 | Moore | Ryan | 00:00.0 | Manager | 1 | 8 | 17 | Moore | Ryan | 00:00.0 | Manager | 1 | 8 |
| 17 | Moore | Ryan | 00:00.0 | Manager | 1 | 8 | 18 | Miller | Rachel | 00:00.0 | Supervisor | 0 | 6 |
| 17 | Moore | Ryan | 00:00.0 | Manager | 1 | 8 | 19 | Jackson | Justin | 00:00.0 | Assistant | 1 | 7 |
| 18 | Miller | Rachel | 00:00.0 | Supervisor | 0 | 6 | 20 | White | Nicole | 00:00.0 | Coordinator | 0 | 5 |

5. List all staff members who actually have been a pilot last year (2023)

SELECT DISTINCT s.STAFF\_ID,

                s.FIRST\_NAME,

                s.LAST\_NAME

FROM STAFF s

JOIN FLIGHTS f ON s.STAFF\_ID = f.PILOT\_ID

WHERE YEAR(f.FLIGHT\_DATE) = 2023;

6. List all staff members without assignments this year (2024)

SELECT STAFF\_CODE,

       LNAME,

       FNAME

FROM STAFF

LEFT JOIN ASSIGNMENTS ON STAFF.STAFF\_CODE = ASSIGNMENTS.STAFF\_STAFF\_CODE

WHERE YEAR(STARTING) != 2024

    OR STARTING IS NULL;



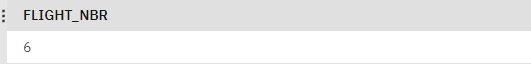
7. What are the flights performed without stopovers

SELECT DISTINCT PLAN\_FLIGHTS.FLIGHT\_NBR

FROM PLAN\_FLIGHTS

LEFT JOIN PLAN\_STOPOVERS ON PLAN\_FLIGHTS.FLIGHT\_NBR = PLAN\_STOPOVERS.PLAN\_FLIGHTS\_FLIGHT\_NBR

WHERE PLAN\_STOPOVERS.PLAN\_FLIGHTS\_FLIGHT\_NBR IS NULL;

small

8. What are the aircrafts not able to perform the longest air link without stopovers

WITH LongestAirLink AS (

SELECT MAX(CAST(AIRLINK\_DISTANCE AS DECIMAL(10,2))) AS LongestDistance

FROM AIRLINKS

),

AircraftsCoveringLongest AS (

SELECT A.IATA, A.AUTONOMY, LA.LongestDistance

FROM KoAIRCRAFT A, LongestAirLink LA

WHERE A.AUTONOMY >= LA.LongestDistance

)

SELECT IATA

FROM AircraftsCoveringLongest

WHERE AUTONOMY < LongestDistance;

9. List of all staff members who worked during a weekend (Saturday or Sunday)

SELECT DISTINCT S.\*

FROM STAFF S

INNER JOIN ASSIGNMENTS A ON S.STAFF\_CODE = A.STAFF\_STAFF\_CODE

INNER JOIN DAY\_PERIODS DP ON A.STARTING = DP.DAY\_PERIOD

AND DP.WEEKDAYS\_DAY\_NBR IN (6,

                            7);

10. Give a list of all flights having an actual number of seat occupied different from expected number of seat occupied

SELECT ef.WEEKDAYS\_DAY\_NBR,

       ef.PLAN\_FLIGHTS\_FLIGHT\_NBR,

       ef.PLAN\_DEP\_DATE,

       ef.RES\_SEATS\_QTY AS Actual\_Occupied\_Seats,

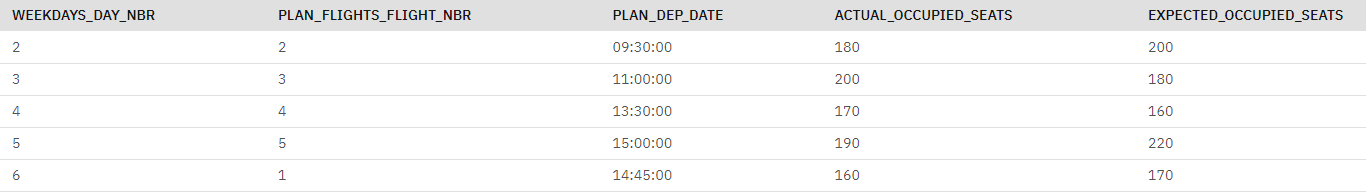
       pdf.PLAN\_SEATS AS Expected\_Occupied\_Seats

FROM EFF\_FLIGHTS ef

JOIN PLAN\_DAYFLIGHTS pdf ON ef.WEEKDAYS\_DAY\_NBR = pdf.WEEKDAYS\_DAY\_NBR

AND ef.PLAN\_FLIGHTS\_FLIGHT\_NBR = pdf.PLAN\_FLIGHTS\_FLIGHT\_NBR

WHERE ef.RES\_SEATS\_QTY <> pdf.PLAN\_SEATS;



11. List of all flight that have been done a weekend (Saturday or Sunday)

SELECT EFF\_FLIGHTS.WEEKDAYS\_DAY\_NBR,

       EFF\_FLIGHTS.PLAN\_FLIGHTS\_FLIGHT\_NBR,

       EFF\_FLIGHTS.PLAN\_DEP\_DATE,

       EFF\_FLIGHTS.REAL\_DEP\_TIME,

       EFF\_FLIGHTS.REAL\_ARR\_TIME

FROM EFF\_FLIGHTS

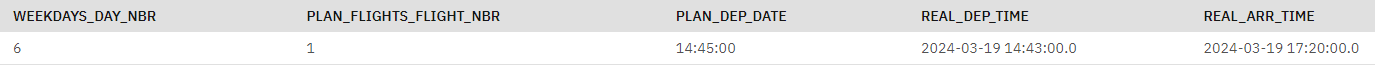
JOIN PLAN\_DAYFLIGHTS ON EFF\_FLIGHTS.WEEKDAYS\_DAY\_NBR = PLAN\_DAYFLIGHTS.WEEKDAYS\_DAY\_NBR

AND EFF\_FLIGHTS.PLAN\_FLIGHTS\_FLIGHT\_NBR = PLAN\_DAYFLIGHTS.PLAN\_FLIGHTS\_FLIGHT\_NBR

JOIN WEEKDAYS ON EFF\_FLIGHTS.WEEKDAYS\_DAY\_NBR = WEEKDAYS.DAY\_NBR

WHERE WEEKDAYS.DAY\_NBR IN (6,

                           7); -- 6 for Saturday, 7 for Sunday



12. For each flight, what is the total amount perceived for tickets compared to the expected one.

SELECT EF.WEEKDAYS\_DAY\_NBR,

       EF.PLAN\_FLIGHTS\_FLIGHT\_NBR,

       SUM(T.PRICE) AS Total\_Perceived\_Amount,

       SUM(EP.ESTIMATED\_PRICE) AS Total\_Expected\_Amount

FROM EFF\_FLIGHTS EF

INNER JOIN TICKETS T ON EF.WEEKDAYS\_DAY\_NBR = T.WEEKDAYS\_DAY\_NBR

AND EF.PLAN\_FLIGHTS\_FLIGHT\_NBR = T.PLAN\_FLIGHTS\_FLIGHT\_NBR

AND EF.PLAN\_DEP\_DATE = T.EFF\_FLIGHTS\_PLAN\_DEP\_DATE

INNER JOIN estimate\_price EP ON EF.WEEKDAYS\_DAY\_NBR = EP.WEEKDAYS\_DAY\_NBR

AND EF.PLAN\_FLIGHTS\_FLIGHT\_NBR = EP.PLAN\_FLIGHTS\_FLIGHT\_NBR

AND T.CLASSES\_CLASS\_CODE = EP.CLASSES\_CLASS\_CODE

WHERE EF.WEEKDAYS\_DAY\_NBR = T.WEEKDAYS\_DAY\_NBR

    AND EF.PLAN\_FLIGHTS\_FLIGHT\_NBR = T.PLAN\_FLIGHTS\_FLIGHT\_NBR

    AND EF.PLAN\_DEP\_DATE = T.EFF\_FLIGHTS\_PLAN\_DEP\_DATE

GROUP BY EF.WEEKDAYS\_DAY\_NBR,

         EF.PLAN\_FLIGHTS\_FLIGHT\_NBR;

13. What is the five most frequently used aircrafts

SELECT KoAIRCRAFT.IATA,

       COUNT(\*) AS departures

FROM EFF\_FLIGHTS

INNER JOIN AIRCRAFTS ON EFF\_FLIGHTS.AIRCRAFTS\_AIRCRAFT\_NBR = AIRCRAFTS.AIRCRAFT\_NBR

INNER JOIN KoAIRCRAFT ON AIRCRAFTS.KoAIRCRAFT\_IATA = KoAIRCRAFT.IATA

GROUP BY KoAIRCRAFT.IATA

ORDER BY departures DESC

LIMIT 5;

