

```
CREATE TABLE [dbo].[aircraftDatabase](
    [icao24] [varchar](6) NULL,
    [registration] [varchar](13) NULL,
    [manufacturericao] [varchar](500) NULL,
    [manufacturername] [varchar](500) NULL,
    [model] [varchar](101) NULL,
    [typecode] [varchar](50) NULL,
    [serialnumber] [varchar](28) NULL,
    [linenumber] [varchar](9) NULL,
    [icaoaircrafttype] [varchar](10) NULL,
    [operator] [varchar](500) NULL,
    [operatorcallsign] [varchar](500) NULL,
    [operatoricao] [varchar](5) NULL,
    [operatoriata] [varchar](5) NULL,
    [owner] [varchar](500) NULL,
    [testreg] [varchar](6) NULL,
    [registered] [varchar](50) NULL,
    [reguntil] [varchar](50) NULL,
    [status] [varchar](max) NULL,
    [built] [datetime] NULL,
    [firstflightdate] [datetime] NULL,
    [seatconfiguration] [varchar](max) NULL,
    [engines] [varchar](500) NULL,
    [modes] [bit] NULL,
    [adsb] [bit] NULL,
    [acars] [bit] NULL,
    [notes] [varchar](500) NULL,
    [categoryDescription] [varchar](500) NULL,
    [vyrazeno] [bit] NULL
)
```

```
CREATE TABLE [dbo].[AOC](
    [Country] [varchar](2) NULL,
    [OPERATOR NAME] [nvarchar](255) NULL,
    [ICAO CODE] [nvarchar](255) NULL,
    [ICAO TYPE] [nvarchar](255) NULL,
    [REGNO] [nvarchar](255) NULL,
    [MODE S (HEX)] [nvarchar](255) NULL,
    [APP DATE] [datetime] NULL,
    [LAST MES] [datetime] NULL,
    [icao24] [varchar](6) NULL
)
```

```
CREATE TABLE [dbo].[airports](
    [ident] [nvarchar](50) NOT NULL,
    [type] [nvarchar](50) NULL,
    [name] [nvarchar](100) NULL,
    [elevation_ft] [int] NULL,
    [continent] [nvarchar](50) NULL,
    [iso_country] [nvarchar](50) NULL,
    [iso_region] [nvarchar](50) NULL,
    [municipality] [nvarchar](50) NULL,
    [gps_code] [nvarchar](50) NULL,
    [iata_code] [nvarchar](50) NULL,
    [local_code] [nvarchar](50) NULL,
    [coordinates] [nvarchar](50) NULL,
    [lat] [float] NULL,
    [lon] [float] NULL,
    [airport] [bit] NULL
)
```

```
CREATE TABLE [dbo].[Events](
    [icao24] [varchar](6) NOT NULL,
    [from] [datetime] NOT NULL,
    [till] [datetime] NOT NULL,
    [lat] [decimal](24, 20) NOT NULL,
    [lon] [decimal](24, 20) NOT NULL,
    [latmin] [decimal](24, 20) NULL,
    [latmax] [decimal](24, 20) NULL,
    [lonmin] [decimal](24, 20) NULL,
    [lonmax] [decimal](24, 20) NULL,
    [onground] [bit] NOT NULL,
    [callsign] [varchar](8) NULL,
    [ident] [varchar](8) NULL,
    [latfirst] [decimal](24, 20) NULL,
    [lonfirst] [decimal](24, 20) NULL,
    [latlast] [decimal](24, 20) NULL,
    [lonlast] [decimal](24, 20) NULL,
    [departure] [varchar](8) NULL,
    [arrival] [varchar](8) NULL,
    [byday] [int] NULL,
    [bynight] [int] NULL,
    [bymorning] [int] NULL,
    [byafternoon] [int] NULL,
    [distance] [decimal](24, 10) NULL
)
```

```

CREATE PROCEDURE dbo.UpdateAirports
/*
    Podle GPS udalosti doplni identifikatory letist
*/
AS
BEGIN

    -- do sloupce Events.Ident doplni identifikator letiste pobytu na zemi
    UPDATE [Events]
    SET Ident = (
        SELECT TOP 1 ident
        FROM airports
        WHERE airport=1 AND [Events].lat - 1 < lat AND [Events].lat + 1 > lat AND [Events].lon - 1 < lon AND [Events].lon + 1 > lon
        ORDER BY geography::Point(lat, lon, 4326).STDistance(geography::Point([Events].lat, [Events].lon, 4326))
    )
    WHERE OnGround = 1

    -- do sloupce Events.Departure doplni identifikator letiste odletu
    UPDATE [Events]
    SET Departure = (
        SELECT TOP 1 ident
        FROM airports
        WHERE airport=1 AND [Events].latfirst - 1 < lat AND [Events].latfirst + 1 > lat AND [Events].lonfirst - 1 < lon AND [Events].lonfirst + 1 > lon
        ORDER BY geography::Point(lat, lon, 4326).STDistance(geography::Point([Events].latfirst, [Events].lonfirst, 4326))
    )
    WHERE OnGround = 0

    -- do sloupce Events.Arrival doplni identifikator letiste priletu
    UPDATE [Events]
    SET Arrival = (
        SELECT TOP 1 ident
        FROM airports
        WHERE airport=1 AND [Events].latlast - 1 < lat AND [Events].latlast + 1 > lat AND [Events].lonlast - 1 < lon AND [Events].lonlast + 1 > lon
        ORDER BY geography::Point(lat, lon, 4326).STDistance(geography::Point([Events].latlast, [Events].lonlast, 4326))
    )
    WHERE OnGround = 0

END

```

```

CREATE VIEW [dbo].[vw_flight]
/*
    zakladni pohled, který na tabulku udalosti navazuje detaily o letištích, letadlech a AOC
    - omezuje výběr podle icao24, výrobce a modelu
    - do výpisu doplňuje počítané sloupce s binární hodnotou pro ML
*/
AS
SELECT TOP (100) PERCENT
    dbo.Events.icao24, -- identifikátor stroje
    dbo.Events.[from] AS firstSeen, -- čas vzniku udalosti
    dbo.Events.till AS lastSeen, -- čas konce udalosti
    dbo.Events.onground, -- příznak pozice stroje: 0 = ve vzduchu, 1 = na zemi
    dbo.Events.callsign, -- označení letu - dostupné je u některých záznamů
    dbo.Events.ident AS OnGroundAirport, -- identifikátor letiště když je stroj na zemi
    dbo.Events.departure AS DepartureAirport, -- identifikátor letiště vzletu
    dbo.Events.arrival AS ArrivalAirport, -- identifikátor letiště přistání
    dbo.aircraftDatabase.model,
    dbo.aircraftDatabase.operator,
    dbo.aircraftDatabase.owner,
    dbo.aircraftDatabase.registered,
    dbo.aircraftDatabase.modes,
    dbo.aircraftDatabase.manufacturericao,
    dbo.aircraftDatabase.manufacturername,
    dbo.aircraftDatabase.registration,
    dbo.aircraftDatabase.typecode,
    dbo.aircraftDatabase.serialnumber,
    dbo.aircraftDatabase.linenummer,
    dbo.aircraftDatabase.icaoaircrafttype,
    dbo.aircraftDatabase.operatorcallsign,
    dbo.aircraftDatabase.operatoricao,
    dbo.aircraftDatabase.operatoriata,
    dbo.aircraftDatabase.testreg,
    dbo.aircraftDatabase.reguntil,
    dbo.aircraftDatabase.status,
    dbo.aircraftDatabase.built,
    dbo.aircraftDatabase.firstflightdate,
    dbo.aircraftDatabase.seatconfiguration,
    dbo.aircraftDatabase.adsb,
    dbo.aircraftDatabase.acars,
    dbo.aircraftDatabase.notes,
    dbo.aircraftDatabase.categoryDescription,
    a1.[continent] AS ArrivalContinent, -- označení kontinentu přistání
    a1.[iso_country] AS ArrivalCountry, -- označení země přistání
    a2.[continent] AS DepartureContinent, -- označení kontinentu vzletu
    a2.[iso_country] AS DepartureCountry, -- označení země vzletu
    a3.[continent] AS OnGroundContinent, -- označení kontinentu když je na zemi
    a3.[iso_country] AS OnGroundCountry, -- označení země, když je na zemi
    dbo.Events.distance AS Distance, -- překonaná vzdálenost, je dostupná i pro pohyby na zemi
    dbo.Events.byday AS ByDay, -- počet intervalů, po které probíhala udalost za dne místního času
    dbo.Events.bynight AS ByNight, -- počet intervalů, po které probíhala udalost v noci místního času
    dbo.Events.bymorning AS ByMorning, -- počet intervalů, po které probíhala udalost dopoledne místního času
    dbo.Events.byafternoon AS ByAfternoon, -- počet intervalů, po které probíhala udalost odpoledne místního času
    CASE WHEN dbo.Events.onground = 0 AND dbo.Events.bynight > 0 AND dbo.Events.byday = 0
        THEN 1 ELSE 0 END AS NightFlight, -- příznak nočního letu: 1 = let probíhal alespoň částečně v noci místního času
    CASE WHEN dbo.Events.onground = 0 AND dbo.Events.bynight = 0 AND dbo.Events.byday > 0

```

```

        THEN 1 ELSE 0 END AS DayFlight, -- priznak denniho letu: 1 = let probihal alespon castecne za dne mistiho casu
CASE WHEN dbo.Events.onground = 0 AND a2.[iso_country] <> a1.[iso_country]
        THEN 1 ELSE 0 END AS InternationalFlight, -- priznak kezinarnodniho letu: 1 = zeme vzletu a pristani je jina
CASE WHEN dbo.Events.onground = 0 AND dbo.Events.distance > 140000
        THEN 1 ELSE 0 END AS FlightAbove140km, -- priznak delky letu: 1 = delka letu je vetsi nez 140 km
CASE WHEN dbo.Events.onground = 0 AND dbo.Events.distance > 250000
        THEN 1 ELSE 0 END AS FlightAbove250km, -- priznak delky letu: 1 = delka letu je vetsi nez 250 km
CASE WHEN dbo.Events.onground = 1 AND DATEDIFF(HOUR, dbo.Events.[from], dbo.Events.[till]) > 2
        THEN 1 ELSE 0 END AS OnGround2Hours, -- priznak trvani pobytu na zemi: 1 = doba na zemi je vetsi nez 2 hodiny
CASE WHEN dbo.Events.onground = 1 AND DATEDIFF(HOUR, dbo.Events.[from], dbo.Events.[till]) > 4
        THEN 1 ELSE 0 END AS OnGround4Hours, -- priznak trvani pobytu na zemi: 1 = doba na zemi je vetsi nez 4 hodiny
CASE WHEN dbo.Events.onground = 0 AND dbo.Events.departure = dbo.Events.arrival
        THEN 1 ELSE 0 END AS SameAirport, -- priznak okruzniho letu: 1 = stejne letiste vzletu i pristani
CASE WHEN dbo.Events.onground = 0 AND DATEDIFF(MINUTE, dbo.Events.[from], dbo.Events.[till]) > 30
        THEN 1 ELSE 0 END AS FlightOver30Minutes -- priznak trvani doby letu: 1 = doba letu je vetsi nez 30 minut
FROM dbo.Events
LEFT JOIN dbo.aircraftDatabase -- tabulka letadel
        ON dbo.Events.icao24 = dbo.aircraftDatabase.icao24
LEFT JOIN [dbo].[airports] a1 -- tabulka letist
        ON dbo.Events.arrival = a1.[ident]
LEFT JOIN [dbo].[airports] a2
        ON dbo.Events.departure = a2.[ident]
LEFT JOIN [dbo].[airports] a3
        ON dbo.Events.ident = a3.[ident]
LEFT JOIN [dbo].[AOC] -- tabulka AOC
        ON dbo.Events.icao24 = dbo.AOC.icao24
WHERE
-- filtr podle icao24
(dbo.Events.icao24 LIKE '49D%'
OR dbo.Events.icao24 LIKE '440%'
OR dbo.Events.icao24 LIKE '44%'
OR dbo.Events.icao24 LIKE '50%'
OR dbo.Events.icao24 LIKE '39%'
OR dbo.Events.icao24 LIKE '3E%'
OR dbo.Events.icao24 LIKE '3C%'
OR dbo.Events.icao24 LIKE '3D%'
OR dbo.Events.icao24 LIKE '300%'
OR dbo.Events.icao24 LIKE '48%'
OR dbo.Events.icao24 LIKE '505%'
OR dbo.Events.icao24 LIKE '4B%'
OR dbo.Events.icao24 LIKE '40%'
)
AND
-- filtr podle vyrobce
(dbo.aircraftDatabase.manufacturername LIKE '%Pilatus%'
OR dbo.aircraftDatabase.manufacturername LIKE '%Cessna%'
OR dbo.aircraftDatabase.manufacturername LIKE '%Piper%'
OR dbo.aircraftDatabase.manufacturername LIKE '%Beech%'
OR (
        -- filtr podle typu
        dbo.aircraftDatabase.manufacturername LIKE '%Embraer%'
        AND dbo.aircraftDatabase.model <> '195'
        AND dbo.aircraftDatabase.model <> '170'
        AND dbo.aircraftDatabase.model <> '190'
        AND dbo.aircraftDatabase.model <> 'A-1'

```

```

AND dbo.aircraftDatabase.model <> 'C-99'
AND dbo.aircraftDatabase.model <> 'Lineage 1000'
AND dbo.aircraftDatabase.model <> 'ERJ-190 Lineage 1000'
AND dbo.aircraftDatabase.model <> '175 (long wing)'
AND dbo.aircraftDatabase.model <> '175 (short wing)'
AND NOT dbo.aircraftDatabase.model LIKE '%ERJ%'
AND NOT dbo.aircraftDatabase.model LIKE '%EMB%'
AND NOT dbo.aircraftDatabase.model LIKE '%CRJ%'
)
OR dbo.aircraftDatabase.manufacturername LIKE '%Gates learjet%'
OR dbo.aircraftDatabase.manufacturername LIKE '%Dassault%'
OR dbo.aircraftDatabase.manufacturername LIKE '%Sea Hawker%'
OR dbo.aircraftDatabase.manufacturername LIKE '%Hamilton%'
OR dbo.aircraftDatabase.manufacturername LIKE '%Gulfstream%'
OR (
  -- filtr podle typu
  dbo.aircraftDatabase.manufacturername LIKE '%Bombardier%'
  AND dbo.aircraftDatabase.model <> 'BD-500 CSeries CS100'
  AND dbo.aircraftDatabase.model <> 'BD-500 CSeries CS300'
  AND dbo.aircraftDatabase.model <> 'Regional Jet CRJ-1000'
  AND dbo.aircraftDatabase.model <> 'CL-600 Regional Jet CRJ-1000'
  AND dbo.aircraftDatabase.model <> 'E-11'
)
OR dbo.aircraftDatabase.manufacturername LIKE '%Moravan%'
OR dbo.aircraftDatabase.manufacturername LIKE '%Zlin%'
OR dbo.aircraftDatabase.manufacturername LIKE '%Antonov%'
OR dbo.aircraftDatabase.manufacturername LIKE '%DE havilland%'
)
AND
-- icao24 musi byt zadano
dbo.AOC.icao24 IS NULL

```

```

CREATE VIEW [dbo].[vw_models_summary]
/*
    pohled pro získání hodnot letu pro jednotlivé typy letadel
*/
AS
SELECT
    f.model AS Model, -- model
    Min(f.Distance) AS MinDistance, -- nejmenší překonaná vzdálenost
    Avg(f.Distance) AS AvgDistance, -- průměrná překonaná vzdálenost
    Max(f.Distance) AS MaxDistance, -- největší překonaná vzdálenost
    Avg(CONVERT(float, f.distance) / DATEDIFF(SECOND, f.firstSeen, f.lastSeen))
    AS AvgSpeedIn, -- průměrná rychlost v m/s
    Avg(CONVERT(float, f.distance) / DATEDIFF(SECOND, f.firstSeen, f.lastSeen))* 3.6
    AS AvgSpeedInKmPerHour, -- průměrná rychlost v km/h
    Avg(DATEDIFF(SECOND, f.firstSeen, f.lastSeen))
    AS AvgTimeInSecs, -- průměrná doba letu v sekundách
    Avg(CONVERT(float, DATEDIFF(SECOND, f.firstSeen, f.lastSeen))) / 3600
    AS AvgTimeInHours, -- průměrná doba letu v hodinách
    Count(*) AS Total -- celkový počet agregovaných letů konkrétního modelu
FROM
    [dbo].[vw_flight] f
WHERE
    f.onground = 0 -- jen lety
    AND DATEDIFF(SECOND, f.firstSeen, f.lastSeen) > 0 -- délka letu nemůže být 0
    AND f.distance > 0 -- překonaná vzdálenost nemůže být 0
GROUP BY
    f.model

```

```

CREATE VIEW [dbo].[vw_flight_advanced]
/*
    rozšíření pohledu vw_flight o nové sloupce z pohledu agregovaných dat provázaných přes sloupec typ stroje
*/
AS
SELECT TOP (100) PERCENT
    f.*, -- všechny sloupce z pohledu vw_flight
    s.[AvgDistance], -- průměrná překonaná vzdálenost pro aktuální typ
    s.[AvgTimeInSecs], -- průměrný čas pro aktuální typ
    CASE WHEN f.onground = 0 AND f.Distance > s.AvgDistance
        THEN 1 ELSE 0 END AS DistanceAboveAvg, -- příznak překonané vzdálenosti: 1 = překonaná vzdálenost je nadprůměrná
    CASE WHEN f.onground = 0 AND DATEDIFF(SECOND, f.firstSeen, f.lastSeen) > s.AvgTimeInSecs
        THEN 1 ELSE 0 END AS TimeAboveAvg -- příznak trvání: 1 = doba trvání je nadprůměrná
FROM
    dbo.vw_flight f
    LEFT JOIN dbo.vw_models_summary s ON f.model = s.Model

```

```

CREATE VIEW [dbo].[vw_flight_with_context]
/*
    pohled doplňující další kontextové sloupce předchozích nebo navazujících událostí
*/
AS

SELECT
    *,

    -- příznak podobné doby trvání letu: 1 = let je v rozmezí od 0.8 do 1.2 násobku nejbližšího předcházejícího nebo následujícího letu
    CASE WHEN f.onground = 0 AND (
        (SELECT TOP 1 DATEDIFF(SECOND, firstSeen, lastSeen) FROM [dbo].[vw_flight]
         WHERE icao24 = f.[icao24] AND f.onground = 0 AND firstSeen < f.[firstSeen] ORDER BY [firstSeen] DESC) < f.AvgTimeInSecs * 1.2
        AND
        (SELECT TOP 1 DATEDIFF(SECOND, firstSeen, lastSeen) FROM [dbo].[vw_flight]
         WHERE icao24 = f.[icao24] AND f.onground = 0 AND firstSeen < f.[firstSeen] ORDER BY [firstSeen] DESC) > f.AvgTimeInSecs * 0.8)
    OR
    (
        (SELECT TOP 1 DATEDIFF(SECOND, firstSeen, lastSeen) FROM [dbo].[vw_flight]
         WHERE icao24 = f.[icao24] AND f.onground = 0 AND lastSeen > f.lastSeen ORDER BY lastSeen) < f.AvgTimeInSecs * 1.2
        AND
        (SELECT TOP 1 DATEDIFF(SECOND, firstSeen, lastSeen) FROM [dbo].[vw_flight]
         WHERE icao24 = f.[icao24] AND f.onground = 0 AND lastSeen > f.lastSeen ORDER BY lastSeen) > f.AvgTimeInSecs * 0.8))
    THEN 1 ELSE 0 END AS SimilarFlightDuration,

    -- příznak navazujících přistání: 1 = v předchozích nebo následujících 2 hodinách došlo k dalším přistáním
    CASE WHEN f.onground = 0 AND
        (SELECT COUNT(*) FROM [dbo].[vw_flight]
         WHERE icao24 = f.[icao24] AND f.onground = 1 AND (DATEDIFF(HOUR, f.lastSeen, firstSeen) < 2 OR DATEDIFF(HOUR, lastSeen, f.firstSeen) < 2)) < 2
    THEN 1 ELSE 0 END AS HasEventsIn2Hours,

    -- příznak zpátečního letu: 1 = příletové letiště je stejné jako odletové letiště předchozího letu
    CASE WHEN f.onground = 0 AND
        f.DepartureAirport = (SELECT TOP 1 ArrivalAirport FROM [dbo].[vw_flight]
                             WHERE icao24 = f.[icao24] AND f.onground = 0 AND firstSeen < f.[firstSeen] ORDER BY [firstSeen] DESC)
        AND
        f.ArrivalAirport = (SELECT TOP 1 DepartureAirport FROM [dbo].[vw_flight]
                           WHERE icao24 = f.[icao24] AND f.onground = 0 AND firstSeen < f.[firstSeen] ORDER BY [firstSeen] DESC)
    THEN 1 ELSE 0 END AS IsReturnFlight,

    -- příznak existence zpátečního: 1 = po tomto letu následuje let, který má letiště přistání stejné jako odletové tohoto
    CASE WHEN f.onground = 0 AND
        f.DepartureAirport = (SELECT TOP 1 ArrivalAirport FROM [dbo].[vw_flight]
                             WHERE icao24 = f.[icao24] AND f.onground = 0 AND lastSeen > f.lastSeen ORDER BY lastSeen)
        AND
        f.ArrivalAirport = (SELECT TOP 1 DepartureAirport FROM [dbo].[vw_flight]
                           WHERE icao24 = f.[icao24] AND f.onground = 0 AND lastSeen > f.lastSeen ORDER BY lastSeen)
    THEN 1 ELSE 0 END AS HasReturnFlight
FROM [dbo].[vw_flight_advanced] f

```