

A1. UML model je v dokumente schema_aerolinky.pdf. Kedže ide o UML je potrebné definovať OCL obmedzenia (minimálne kľúče):

context l1, l2: lietadlo inv

I1. seriove_cislo = l2.seriove_cislo implies l1 = l2 /* ostatne klúce by som definoval podobne */

context l: lietadlo inv

l.je_typu.vodicak->exists(p | p = l.ma_pilota) /* pilot lietadla musi mať pre dany typ lietadla opravnenie */

A2. Převeďte Vámi zhotovený UML model do schématu relační databáze. Vypište schémata relací a klíče (primárni, cizí).

zamestnanec(id_zam, meno, priezvisko)

letuska(id_zam), id_zam ⊑ zamestnanec.id_zam

pilot(id_zam), id_zam ⊑ zamestnanec.id_zam

lietadlo(id_lietadlo, seriove_cislo, id_typ, id_pilot1, id_pilot2), id_typ ⊑ typ.id_typ, id_pilot1 ⊑ pilot.id_zam, id_pilot2 ⊑ pilot.id_zam

letuskaVLietadlo(id_lietadlo, id_letuska), id_lietadlo ⊑ lietadlo.id_lietadlo, id_letuska ⊑ letuska.id_zam

typ(id_typ, kapacita, spotreba, dolet)

vodicak(id_pilot, id_typ), id_pilot ⊑ pilot.id_zam, id_typ ⊑ typ.id_typ

trasa(id_trasa, start, ciel, vzdialenosť)

let(id_let, den, hodina, id_lietadlo, id_trasa), id_lietadlo ⊑ lietadlo.id_lietadlo, id_trasa ⊑ trasa.id_trasa

pridelenitraza(id_lietadlo, id_trasa), id_lietadlo ⊑ lietadlo.id_lietadlo, id_trasa ⊑ trasa.id_trasa /* tato tabuľka je tu kvôli zadaniu triggera */

B1. Vypište lietadla, ktoré létají „zpäťeční“ lety, tzn. létají z letiště A do letiště B a taky z letiště B do Letiště A.

SELECT l.id_lietadlo

FROM lietadlo AS l

JOIN let AS let1 ON let1.id_lietadlo = l.id_lietadlo

JOIN trasa AS t1 ON let1.id_trasa = t1.id_trasa

JOIN let AS let2 ON let2.id_lietadlo = l.id_lietadlo

JOIN trasa AS t2 ON (let2.id_trasa = t2.id_trasa AND t1.start = t2.ciel AND t1.ciel = t2.start)

GROUP BY l.id_lietadlo

B2. Vytvořte přehled pilotů a nalétaných kilometrů za týden, přičemž piloty, kteří létají na 2 a méně letech neuvažujte.

/* predpokladam, ze tabulka let obsahuje casovy plan jedneho tyz dna (aktualny cestovny poriadok) */

SELECT p.id_zam, sum(vzdialenosť)

FROM pilot AS p

JOIN lietadlo AS l ON (p.id_zam = l.id_pilot1 OR p.id_zam = p.id_pilot2) /* predpokladam, ze id_pilot1 <> id_pilot2 */

JOIN let AS t ON let.id_lietadlo = l.id_lietadlo

JOIN trasa AS t ON let.id_trasa = t.id_trasa

GROUP BY p.id_zam

HAVING count(let.id_let) > 2

B3. Kteří piloti nelétají na žádné letiště, ze kterého letecká společnost léta do Prahy ?

(SELECT id_zam FROM pilot)

EXCEPT

(

SELECT p.id_zam

FROM pilot AS p

JOIN lietadlo AS l ON (p.id_zam = l.id_pilot1 OR p.id_zam = p.id_pilot2) /* predpokladam, ze id_pilot1 <> id_pilot2 */

JOIN let AS t ON let.id_lietadlo = l.id_lietadlo

JOIN trasa AS t ON let.id_trasa = t.id_trasa

WHERE t.ciel IN (SELECT t1.start FROM trasa AS t1 WHERE t1.ciel = 'Praha')

GROUP BY p.id_zam

)

C. Napište trigger, který při zavedení každé nové trasy, přiradí na tuto trasu všechna letadla (naplánování nových letů není potřeba), které tuto trasu dokážou obsloužit (dolet_letadla >= vzdálenost_letiště)

```
CREATE TRIGGER trgNovaTrasa ON trasa
AFTER INSERT
AS
BEGIN
    INSERT INTO pridelenaTrasa(id_lietadlo, id_trasa) VALUES
    (
        SELECT inserted.id_trasa, l.id_lietadlo
        FROM lietadlo AS l
        JOIN typ ON typ.id_typ = l.id_typ
        JOIN inserted ON inserted.vzdialenosť <= typ.dolet
    )
END
```