**K2.1: Za entitetne tipe kreiraj relacije**

Popravljen konceptualni model ni imel povezav 1:1, tako da ni bilo potrebe po posebnem določanju, očeta in otrok.

**K2.2: Preveri relacije z normalizacijo**

Funkcionalne odvisnosti logičnega modela

Država = {ID\_Država -> (Ime\_Država)}

Pošta ={Postna\_Stevilka -> (Kraj)}

Program = {ID\_Program -> (Ime\_Program, Opis)}

Hotel = {ID\_Hotel -> (Ime\_Hotel, Ocena, Naslov\_Hotel)}

Poslovalnica = {ID\_Poslovalnica -> (Ime\_Poslovalnica, Naslov\_Poslovalnica}

Prevoz = {ID\_Prevoz -> (P\_Odhod, P\_Prihod)}

Letovanje = {ID\_Letovanje -> (L\_Odhod, L\_Prihod, Trajanje)}

Oseba = {EMSO -> (Ime\_Oseba, Priimek, Spol, Datum\_Rojstva, Email, Naslov)}

Prevoznik = {ID\_Prevoznik -> (Ime\_Prevoznik, Vrsta\_Prevoza)}

Potovanje = {ID\_Potovanje -> (Trajanje)}

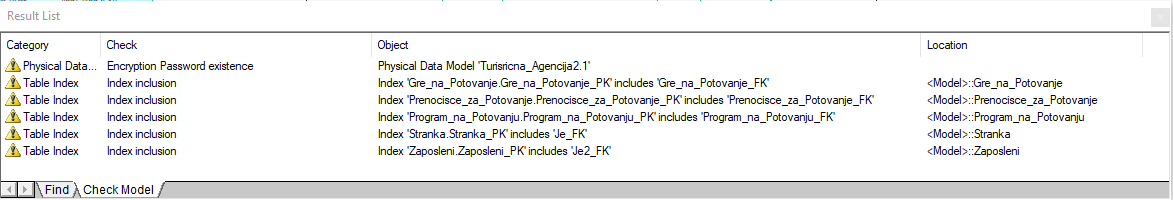
Stranka = {ID\_Stranka -> ()}

Zaposleni = {ID\_Zaposleni -> ()}

Ni bilo potrebe po normaliziranju, ker so vse relacije v 3. normalni obliki.

**K3.1: Izdelaj načrt osnovnih relacij**

Pred preslikavo v fizični model, PowerDesigner še enkrat preveri model, kateri je javil, da so se nekateri atributi indeksirali večkrat, kar je ne potrebno.



**K3.2: Izdelaj načrt predstavitve izpeljanih atributov**

Primer izpeljanega atributa v modelu je atribut Cas\_Trajanja v tabeli Letovanje\_1. Izračuna se ga tako, da se odšteje datum prihoda, z datumom odhoda. V atribut Cas\_Trajanja zapišemo, koliko dni traja letovanje. To naredimo tako, da najprej prepisemo tabelo Letovanje v tabelo Letovanje\_1, dodamo atribut TRAJANJE in nato za vsak vnos preračunamo trajanje letovanja.

CREATE TABLE LETOVANJE\_1 AS

SELECT \* FROM LETOVANJE;

ALTER TABLE LETOVANJE\_1

ADD (trajanje number);

UPDATE LETOVANJE\_1

SET trajanje = round(L\_PRIHOD - L\_ODHOD);

**K3.3: Izdelaj načrt splošnih omejitev**

Princip, ki smo ga uporabili za računanje časa trajanja letovanja, smo poenostavili tako, da smo v tabelo Letovanje dodali atribut Cas\_Trajanja in izdelali sprožilec, ki ob vsakem vnosu novega letovanja preračuna čas trajanja.

CREATE OR REPLACE TRIGGER trajanje\_letovanja

BEFORE INSERT OR UPDATE ON letovanje

FOR EACH ROW

BEGIN

:new.cas\_trajanja := round(:new.l\_prihod - :new.l\_odhod);

END;

/

**K4.1: Izdelaj načrt datotečne organizacije ter indeksov:**

V naslednjem koraku, smo realizirali transakcije s pomočjo SQL ukazov.

1. **V katerih državah so bila letovanja?**

select drzava.ime\_drzava from letovanje

inner join hotel on hotel.ID\_HOTEL = letovanje.ID\_HOTEL

inner join posta on posta.POSTNA\_STEVILKA = hotel.POSTNA\_STEVILKA

inner join drzava on drzava.ID\_DRZAVA = posta.ID\_DRZAVA;

2. **Kateri prevoznik je opravil največ prevozov?**

select prevoznik.id\_prevoznik, prevoznik.ime\_prevoznik from prevoz

inner join prevoznik on prevoznik.id\_prevoznik = prevoz.id\_prevoznik

group by prevoznik.id\_prevoznik, prevoznik.ime\_prevoznik

having count(\*) = (select max(count(\*)) from prevoz

inner join prevoznik on prevoznik.id\_prevoznik = prevoz.id\_prevoznik

group by prevoznik.ime\_prevoznik);

3. **Kateri je najbolj priljubljen hotel v Grčiji (imel največ letovanj)?**

select hotel.id\_hotel, hotel.ime\_hotel from letovanje

inner join hotel on letovanje.ID\_HOTEL = hotel.ID\_HOTEL

inner join posta on posta.POSTNA\_STEVILKA = hotel.POSTNA\_STEVILKA

inner join DRZAVA on DRZAVA.ID\_DRZAVA = posta.ID\_DRZAVA

where drzava.ime\_drzava = 'Grčija'

group by hotel.id\_hotel, hotel.ime\_hotel

having count(\*) = (select max(count(\*)) from letovanje

inner join hotel on letovanje.ID\_HOTEL = hotel.ID\_HOTEL

group by hotel.id\_hotel, hotel.ime\_hotel);

4. **Koliko ljudi je letovalo leta 2019 prek agencije?**

select count(\*) as Stevilo\_letovanj\_v\_2019 from letovanje

where to\_char(L\_ODHOD, 'YYYY') = 2019 and to\_char(L\_PRIHOD, 'YYYY') = 2019;

5. **Ali je več ljudi šlo na letovanje ali potovanje?**

select count(\*) from letovanje;

select count(\*) from vsa\_potovanja;

6. **V katerem hotelu je v juniju letovalo največ ljudi?**

select hotel.id\_hotel, hotel.ime\_hotel from letovanje

inner join hotel on letovanje.id\_hotel = hotel.id\_hotel

where to\_char(letovanje.l\_prihod, 'MM') = 6 and to\_char(letovanje.l\_odhod, 'MM') = 6

group by hotel.id\_hotel, hotel.ime\_hotel

having count(\*) = (select max(count(\*)) from letovanje

inner join hotel on letovanje.id\_hotel = hotel.id\_hotel

where to\_char(letovanje.l\_prihod, 'MM') = 6 and to\_char(letovanje.l\_odhod, 'MM') = 6

group by hotel.id\_hotel, hotel.ime\_hotel);

7. **Koliko zaposlenih ima vsaka poslovalnica?**

select id\_poslovalnica, ime\_poslovalnica, count(\*) as Stevilo\_zaposlenih from poslovalnica

group by id\_poslovalnica, ime\_poslovalnica;

8. **Koliko časa je trajalo najdaljše potovanje?**

select max(trajanje) as Najdaljsi\_cas\_potovanja from potovanje;

9. **Koliko ljudi je šlo na letovanje s prevozom?**

select count(\*) as Stevilo\_letovanj\_s\_prevozom from letovanje

inner join prevoz on prevoz.id\_prevoz = letovanje.id\_prevoz;

10. **V kateri državi je bilo najmanj letovanj?**

select drzava.id\_drzava, drzava.ime\_drzava from letovanje

inner join hotel on hotel.id\_hotel = letovanje.id\_hotel

inner join posta on posta.POSTNA\_STEVILKA = hotel.POSTNA\_STEVILKA

inner join drzava on drzava.ID\_DRZAVA = posta.ID\_DRZAVA

group by drzava.id\_drzava, drzava.ime\_drzava

having count(\*) = (select min(count(\*)) from letovanje

inner join hotel on hotel.id\_hotel = letovanje.id\_hotel

inner join posta on posta.POSTNA\_STEVILKA = hotel.POSTNA\_STEVILKA

inner join drzava on drzava.ID\_DRZAVA = posta.ID\_DRZAVA

group by drzava.id\_drzava, drzava.ime\_drzava);

**K4.3: Določi indekse**

Dodani sekundarni indeksi so:

* Indeks na Postna\_Stevilka v tabeli Program
* Indeks na Postna\_Stevilka v tabeli Prevoz
* Indeks na Pos\_Postna\_Stevilka v tabeli Prevoz
* Indeks na P\_Odhod v tabeli Prevoz
* Indeks na ID\_Prevoz v tabeli Potovanje
* Indeks na Postna\_Stevilka v tabeli Hotel
* Indeks na ID\_Stranka in EMSO v tabeli Letovanje
* Indeks na Postna\_Stevilka v tabeli Oseba

Primer sql ukaza za kreiranje indeksa:

create index postna\_st on Oseba (Postna\_Stevilka);

**K5: Izdelaj načrt uporabniški pogledov:**

Za primere uporabniških pogledov, sva se odločila za pogleda, koliko potovanj je od danes do naslednjega meseca (31 dni) ter od kje do kje vozi določen prevoznik.

create view potovanja\_naslednji\_mesec

AS

SELECT p1.kraj AS Odhod, p2.kraj AS Destinacija, prevoz.P\_Odhod AS "Datum odhoda" FROM potovanje

inner join prevoz on potovanje.ID\_Prevoz = prevoz.ID\_Prevoz

inner join posta p1 on p1.POSTNA\_STEVILKA = prevoz.POSTNA\_STEVILKA

inner join posta p2 on p2.POSTNA\_STEVILKA = prevoz.POS\_POSTNA\_STEVILKA

where ROUND(prevoz.P\_Odhod - (SELECT sysdate FROM dual)) <= 31;

create view Prevozniki as

select distinct prevoznik.ime\_prevoznik as Prevoznik, p1.kraj as Odhod, p2.kraj as Prihod from prevoz

inner join posta p1 on p1.POSTNA\_STEVILKA = prevoz.POSTNA\_STEVILKA

inner join posta p2 on p2.POSTNA\_STEVILKA = prevoz.POS\_POSTNA\_STEVILKA

inner join prevoznik on prevoznik.ID\_PREVOZNIK = prevoz.ID\_PREVOZNIK;