Táborhely Üzemeltető segédszoftver Implementációs dokumentáció.

# Program

A program futtatására a program.sql skript hivatott. Mely meghívja az inicializációs, és a menűt kirajzoló szkripteket.

## program.sql:

@@0\_Init

@@menu

ACCEPT m PROMPT 'Ha folytatni akarja a munkát írja be, hogy menu (vagy egyet az almenűk közül):'

@@ &m

# Inicializáció

Ez a rész állítja be a környezet fontosabb paramétereit, illetve a tábla létrehozó, illetve a tesztadatokkal feltöltő szkripteket.

## 0\_Init:

SET feedback OFF

ALTER SESSION set nls\_date\_format = 'YYYY-MM-DD';

@@ 13\_Jogosultságok

connect Admin1/passAdmin01

@@ 1\_create\_tables\_2\_constraints

@@ 3\_b\_Insert\_Csoportok

@@ 3\_a\_Insert\_Házak

@@ 3\_c\_Insert\_Lakók

@@15\_Alprogramok

@@16\_triggerek

SET feedback ON

SET verify OFF

SET SERVEROUTPUT ON

## 13\_Jogosultságok

# CONNECT sys/rendszergazda AS SYSDBA

# cl scr

# --szerepkörök CampAdmin; CampBoss

# --alkalmazottak: Admin1; Admin2 - Boss1; Boss2

# --13\_Jogosultságok

# DROP USER Admin1 CASCADE;

# DROP USER Admin2 CASCADE;

# DROP USER Boss1 CASCADE;

# DROP USER Boss2 CASCADE;

# DROP ROLE CampAdmin;

# DROP ROLE CampBoss;

# CREATE ROLE CampAdmin;

# GRANT

# CREATE SESSION,

# CREATE TABLE

# TO

# CampAdmin;

# CREATE ROLE CampBoss;

# GRANT

# CREATE SESSION,

# SELECT ANY TABLE

# TO

# CampBoss;

# 

# CREATE USER Admin1

# IDENTIFIED BY passAdmin01;

# PROMPT admin1 gran\_t

# GRANT CampAdmin

# TO Admin1;

# CREATE USER Admin2

# IDENTIFIED BY passAdmin02;

# GRANT CampAdmin

# TO Admin2;

# CREATE USER Boss1

# IDENTIFIED BY passBoss01;

# GRANT CampBoss

# TO Boss1;

# 

# CREATE USER Boss2

# IDENTIFIED BY passBoss02;

# GRANT CampBoss

# TO Boss2;

## 15\_Alprogramok

CREATE OR REPLACE FUNCTION KorSzámít(szül\_dátum IN DATE)

RETURN NUMBER

AS

BEGIN

RETURN TRUNC(MONTHS\_BETWEEN(SYSDATE,szül\_dátum)/12);

END;

/

# Táblák és függőségeik

A táblák tervben foglaltak szerint készültek.

Táblák létrehozása (1\_create\_tables\_2\_constraints.sql)

## A Lakók tábla:

DROP TABLE Lakók;

CREATE TABLE Lakók (

Név VARCHAR2(30),

Egészségbiztosítási\_szám VARCHAR2(9) UNIQUE,

SzületésiDátum DATE,

Nem CHAR(1) CHECK( Nem IN ('f','n') ),

Személyigazolvány\_szám VARCHAR2(11) UNIQUE,

Tartózkodas\_tól DATE,

Tartózkodas\_ig DATE,

Ország VARCHAR2(50),

ház\_szám NUMBER(4),

Csoport\_név VARCHAR2(50),

Csoportvezető VARCHAR2(30)

);

A kívánt szöveges és szám mezők meghatározásakor, ahol tudtam, a lehető legnagyobb, még ésszerű foglalási méretet választani. Ezt néhol kutatás is megelőzte, például az ország név mező hosszának meghatározásakor számításba vettem ilyeneket, mint: The United Kingdom of Great Britain and Northern Ireland, mely 48 karakter hosszú. Az ország mezőt 50 hosszú VARCHAR2-ként vettem fel. A VARCHAR2 típusról tudni kell, hogy nem foglalódik le alapból a teljes meghatározott karakternyi méret (előző példában 50), ezért úgy láttam, érdemesebb túltervezni, ahol csak lehet. Az Egészségbiztosítási\_szám, illetve a Személyigazolvány\_szám fix hosszúságú szövegek, ennek megfelelő CHAR típusnak határoztam meg mind a kettőt. A szám típusoknál a ház\_szám, illetve a férőhely\_szám három helyértéken tárol, a csoport pontszám egy tizedes jegy pontossággal, és öt egész helyértékkel tárol, ami a táborban bőven elég lesz.

## A Csoportok tábla

DROP TABLE Csoportok;

CREATE TABLE Csoportok(

csoport\_név VARCHAR2(50),

csoportvezető VARCHAR2(30),

csoport\_pontszám NUMBER

);

## A Házak tábla

DROP TABLE Házak;

CREATE TABLE Házak (

ház\_szám NUMBER(4),

ház\_megnevezés VARCHAR2(25) NOT NULL,

férőhely\_szám NUMBER(4)

);

## Megszorítások

A terv alapján megalkottam az elsődleges, illetve idegen kulcsokat

ALTER TABLE Lakók ADD CONSTRAINTS p\_Lakók PRIMARY KEY (Név,Egészségbiztosítási\_szám);

ALTER TABLE Csoportok ADD CONSTRAINTS p\_Csoportok PRIMARY KEY (csoport\_név,csoportvezető);

ALTER TABLE Házak ADD CONSTRAINTS p\_Házak PRIMARY KEY (ház\_szám);

ALTER TABLE Lakók ADD CONSTRAINTS f\_Lakók\_Csoportok FOREIGN KEY (Csoport\_név,Csoportvezető)

REFERENCES Csoportok (csoport\_név,csoportvezető);

ALTER TABLE Lakók ADD CONSTRAINTS f\_Lakók\_Házak FOREIGN KEY (ház\_szám)

REFERENCES Házak (ház\_szám);

A fentiekben látható, hogy a Lakók táblában az Egészségbiztosítási\_szám, amellett, hogy az elsődleges kulcs része, még egyedinek is jelöltem. Egyedi továbbá a Személyigazolvány\_szám mező is. Az egészségbiztosítási szám, mivelhogy az elsődleges kulcs része, kötelező a kitöltése is. A Személyigazolvány\_szám kitöltése ellenben nem kötelező.

# A menürendszer

A program egy menüt rajzol ki, melyből a felhasználó parancsok begépelésével almenükbe mehet át.

## Házak kezelése almenü (házak)

## Coportok kezelése almenü (csoportok)

## Lakók kezelése almenü (lakók)

## DDL-Táblamódosítás almenü (ddl)