

3.4.1. Készítsen egy lakcím leíró **T_LAKCIM** táblát, amely a fontosabb postai cím komponenseket tartalmazza.

```
CREATE TABLE T_LAKCIM ( Sz_neve VARCHAR(30) PRIMARY KEY NOT NULL, Lakcim VARCHAR(30) NOT NULL, Kerulet_az INT(4) NOT NULL);
```

3.4.2. Szüntesse meg az előzőleg létrehozott **T_LAKCIM** táblát.

```
DROP TABLE T_LAKCIM;
```

3.4.3. Hozzon létre egy **DOLGOZÓK** táblát, melyben szerepelnek a név, születési év és lakcím mezők. Ez utóbbi idegen kulcsa legyen a **T_LAKCIM** [**ID**, Irányítószám, Város, Utca, Házszám] táblának.

```
CREATE TABLE DOLGOZOK (nev VARCHAR(30), szul_ev DATE, lakcim INT FOREIGN KEY REFERENCES T_LAKCIM(lakcim));
```

3.4.4. Vigyen fel egy új rekordot a **DOLGOZÓK** és a **T_LAKCIM** [**ID**, Irányítószám, Város, Utca, Házszám] táblába.

```
INSERT INTO T_LAKCIM VALUES (1, 6722, „Szeged”, „Gutenberg u.”, 99);  
INSERT INTO DOLGOZOK VALUES („Kiss István”, 1999, 1);
```

3.4.5. Listázza ki a **DOLGOZÓK** [**ID**, Név, Szülév, Város, Lakcím] táblából a dolgozók nevét és a városukat, város szerinti sorrendben.

```
SELECT DOLGOZOK.nev, T_LAKCIM.Lakcim  
FROM DOLGOZOK  
JOIN T_LAKCIM ON DOLGOZOK.lakcim = T_LAKCIM.Sz_neve  
ORDER BY T_LAKCIM.Lakcim;
```

3.4.6. Kérdezze le a **DOLGOZÓK** [**ID**, Név, Szülév, Város, Lakcím] táblából dolgozók létszámát városonkénti bontásban.

```
SELECT T_LAKCIM.Lakcim, COUNT(DOLGOZOK.nev) AS DolgozokSzama  
FROM DOLGOZOK  
JOIN T_LAKCIM ON DOLGOZOK.lakcim = T_LAKCIM.Sz_neve  
GROUP BY T_LAKCIM.Lakcim;
```

3.4.7. Hozzon létre egy **EMBEREK** táblát, melyben adattagként szerepel a név, születési év, igazolványszám és lakcím.

```
CREATE TABLE EMBEREK (  
    Nev VARCHAR(30),  
    Szulev DATE,  
    Igazolvanysszam INT,  
    Lakcim INT,  
    FOREIGN KEY (Lakcim) REFERENCES T_LAKCIM(Sz_neve)  
);
```

3.4.8. Vigyen fel egy új embert az **EMBEREK** [**Név**, **Szülév**, **Igazolványszám**, **Lakcím**] táblába.

```
INSERT INTO EMBEREK VALUES ('John Doe', '1980-05-15', 123456, 1);
```

3.4.9. Kérdezze le az **EMBEREK** [**Név**, **Szülév**, **Igazolványszám**, **Lakcím**] táblában szereplő emberek neveit.

```
SELECT Nev FROM EMBEREK;
```

3.4.10. Módosítsa az **IG1 igazolványszámú** ember **lakcímét** egy új értékre.

```
UPDATE EMBEREK  
SET Lakcim = 2  
WHERE Igazolvanysszam = 123456;
```

3.4.11. Hozzon létre egy **MUNKAK** táblát, és módosítsa az **EMBEREK** táblát.

MUNKAK táblának van egy azonosítója az **ID oszlop**, és két másik oszlopa van, a munkakör és a fizetés.

Az **EMBEREK** táblához adjon hozzá egy **Munkak_ID** oszlopot.

```
CREATE TABLE MUNKAK (  
    ID INT PRIMARY KEY,
```

```
Munkakor VARCHAR(50),
Fizetes INT
);
ALTER TABLE EMBEREK
ADD COLUMN Munkak_ID INT,
ADD FOREIGN KEY (Munkak_ID) REFERENCES MUNKAK(ID);
```

3.4.12. Adjunk hozzá egy embert az **EMBEREK** táblához.

Név	Szülév	Igazolványszám	Lakcím	Munkák_ID
Kovács János	1980	ABC987	Budapest Nagy utca 45	45

```
INSERT INTO EMBEREK (Nev, Szulev, Igazolvanyyszam, Lakcim, Munkak_ID)
VALUES ('Kovács János', '1980-01-01', 'ABC987', 3, 45);
```

3.4.13. Hozzon létre két táblát, az **ÜGYFÉL** és a **SZÁMLA** táblát.
Az **ÜGYFÉL** tábla, az ügyfelek adatait tartalmazza: az azonosító -> ID, a név, a születési év és a telefonszám. Az ID oszlop a tábla elsődleges kulcsa, és a **NUMBER(10)** adattípust használjuk a típus meghatározásához.
A **SZÁMLA** tábla a számlák adatait tárolja: az azonosító -> ID, az ügyfél azonosítója→Ügyfél ID, a számla dátuma és az összeg.
Az ID oszlop itt is a tábla elsődleges kulcsa, míg az Ügyfél ID az elsődleges kulcsra hivatkozik az **ÜGYFÉL** táblában. Az összeg típusának **NUMBER(10,2)** adattípust használjuk a meghatározáshoz, ami azt jelenti, hogy 10 számjegyig és 2 tizedesjegyig lehet értéke.

```
CREATE TABLE UGYFEL (
    ID NUMBER(10) PRIMARY KEY,
    Nev VARCHAR(30) NOT NULL,
    Szulev DATE NOT NULL,
    Telefon VARCHAR(15) NOT NULL
);
CREATE TABLE SZAMLA (
    ID NUMBER(10) PRIMARY KEY,
    Ugyfel_ID NUMBER(10) REFERENCES UGYFEL(ID),
    SzamlaDatum DATE NOT NULL,
    Osszeg NUMBER(10,2) NOT NULL
);
```

3.4.14. A **MUNKAVÁLLALÓ** nevű táblában módosítsd az **EMPLOYEE NAME** oszlop nevét **Munkavállaló név** -re és adj hozzá egy új **EMAIL** oszlopot az alkalmazottak e-mail címének tárolására.

```
ALTER TABLE MUNKAVALLALO
RENAME COLUMN EMPLOYEE_NAME TO Munkavallalo_nev;
```

```
ALTER TABLE MUNKAVALLALO
ADD COLUMN EMAIL VARCHAR(50);
```

3.4.15. Készítsünk egy **DIÁKOK** nevű táblát, mely tartalmazza a nevét, születési évét, osztályát és átlagát. Az adattípusok legyenek: **CHAR(50)** a név, **DATE** a születési év, **CHAR(10)** az osztály és **NUMBER(4,2)** az átlag.

Ezután adjunk hozzá néhány sort a táblához.

ID	Név	szülév	Osztály	Átlag
1	Kovács János	1980	10	4,56
2	Nagy Márta	1981	11	3,78
3	Sárga Norbert	1982	12	4,12

Módosítsuk az átlagot **NUMBER(4,2)** -ről **NUMBER(5,2)** -re.

```
CREATE TABLE DIAKOK (
    ID NUMBER PRIMARY KEY,
    Nev CHAR(50),
    Szulev DATE,
```

```
Osztaly CHAR(10),  
Atlag NUMBER(5,2)  
);
```

```
INSERT INTO DIAKOK VALUES (1, 'Kovács János', '1980-01-01', '10', 4.56);  
INSERT INTO DIAKOK VALUES (2, 'Nagy Márta', '1981-01-01', '11', 3.78);  
INSERT INTO DIAKOK VALUES (3, 'Sárga Norbert', '1982-01-01', '12', 4.12);  
ALTER TABLE DIAKOK  
MODIFY Atlag NUMBER(5,2);
```

3.4.16. Hozz létre egy **TANÁR** nevű táblát az alábbi oszlopokkal és típusokkal:
"id" NUM típus, elsődleges kulcs
"név" CHAR (50) típus
"szulev" DATE típus
"tantárgy" CHAR (10) típus
Majd módosítsd a táblát úgy, hogy az " tantárgy " oszlop típusa " CHAR (2)" legyen.

```
CREATE TABLE TANAR (  
id NUMBER PRIMARY KEY,  
nev CHAR(50),  
szulev DATE,  
tantargy CHAR(2)  
);
```

3.4.17. Hozza létre az alábbi szerkezetű **ALKALMAZOTT** nevű táblát:
KOD CHAR (4)
NEV CHAR (30) NOT NULL
FIZETES NUM
SZULDAT DATE
Bővítsd az **ALKALMAZOTT** táblát a CIM oszloppal, melynek típusa CHAR(30).
A NEV hosszát módosítsa 40-re!

```
CREATE TABLE ALKALMAZOTT (  
KOD CHAR(4),  
NEV CHAR(30) NOT NULL,  
FIZETES NUMBER,  
SZULDAT DATE,  
CIM CHAR(30)  
);  
ALTER TABLE ALKALMAZOTT  
MODIFY NEV CHAR(40),  
ADD COLUMN BEOSZTAS CHAR(10),  
ADD COLUMN RESZLEG CHAR(30);
```

3.4.18. Hozza létre azt a **NEZET** nevű nézettáblát, amely az **ALKALMAZOTT** [ID, KOD, NEV, FIZETES, SZULDAT, BEOSZTAS, RESZLEG] és a **RESZLEG** [KOD, NEV, CIM] táblából csak az **ELADÓ** beosztású dolgozók kódját, nevét, beosztását, részlegének kódját, részlegének nevét és részlegének címét tartalmazza!

-----?-----

3.4.19. Hozza létre az **ATLAG** nevű nézettáblát, amely az **ALKALMAZOTT** [ID, KOD, NEV, FIZETES, SZULDAT, BEOSZTAS, RESZLEG] és a **RESZLEG** [KOD, NEV, CIM] táblából tartalmazza a részleg neve, a részleg kódja és az ott dolgozók átlagfizetését!
Készítsen **KIMUTATÁS** nevű nézettáblát az **ATLAG** nevű nézettábla felhasználásával, amelyben a dolgozók neve, fizetése, részlegének kódja és a részleg átlagfizetése szerepel!

-----?-----