

# Szálloda adatbázis projekt dokumentáció

---

- Név: Barna Gergely
- Neptun kód: FJKXGG
- H-s azonosító: H144988

A webalkalmazás éles verziója jelenleg is fut a szerveremen. Ezen az URL-en elérhető:

<https://hotel.barnagergely.hu/>

## Rendszerspecifikáció

Ebben a projektben egy hotel nyilvántartási webes alkalmazást fogok PHP és MySQL alapokra felépíteni.

A rendszernek képesnek kell lennie arra, hogy a cég takarítói, vendégei, ellenőrei és foglalásai nyilvántartását végezzék benne.

### Funkciók

- Személyek nyilvántartása
  - Lehesen nyilvántartást vezetni a hotelben megfordult emberekről
  - Tárolandó adatok: név, sz.ig. száma, Állapot, Rang
  - Lehesen személyeket létrehozni, módosítani és törölni
  - A személyeknek lehesen rangot adni (egyszerre akár több rangja is lehesen egy személynek)
  - A személyket meg lehesen tekinteni egy listában és rendezni adataik alapján
- Dolgozók nyilvántartása
  - A dolgozók legyenek speciális személyek, akikről több adatot tárolunk
  - Tárolandó adatok: a személy adatain felül beosztás és fizetés
  - Lehesen személyeket létrehozni, módosítani és törölni
  - Ha a dolgozóhoz tartozó személyt töröljük, törlődjön a dolgozó is
- Ellenőrök nyilvántartása
  - Az ellenőrök legyenek speciális személyek, akikről több adatot tárolunk
  - Tárolandó adatok: a személy adatain felül ellenőr állapota, leírás
  - Lehesen ellenőröket létrehozni, módosítani és törölni
  - Az ellenőrök leírásába be lehesen írni egy hosszabb szöveget az ellenőről.
  - Ha az ellenőrhöz tartozó személyt töröljük, törlődjön az ellenőr is
- Szobák nyilvántartása
  - Az adatbázisban tároljuk el a hotelben lévő szobákat és néhány adatot róluk
  - Tárolandó adatok: szobaszám, Szoba neve, Ki takarította utoljára, Mikor takarították utoljára
  - Lehesen szobákat létrehozni, módosítani és törölni
  - Minden szobánál meg lehesen adni melyik dolgozó és mikor takarította legutóbb a szobát
- Foglalások nyilvántartása
  - Követni lehesen ki, mikorra és melyik szobát foglalta le.
  - Tárolandó adatok: Foglalás száma, Foglalt szoba, Foglalo személy, Foglalás kezdeti dátuma, Foglalás vég dátuma, Foglalás állapota
  - Lehesen foglalásokat létrehozni, módosítani és törölni
  - Lehesen egy szobát egy időpontban több személynek is kiadni

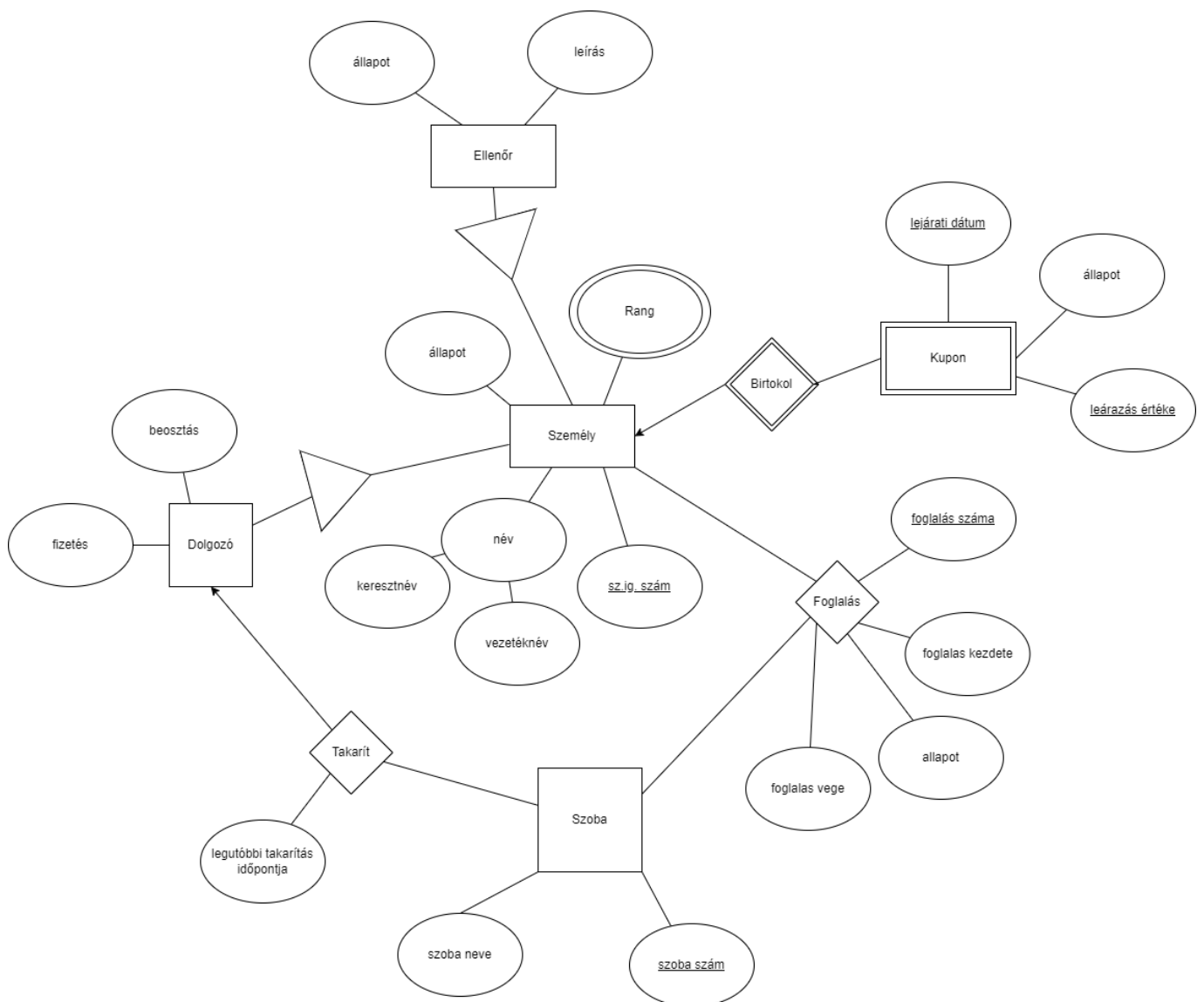
- Egy személy egy időpontra akár több szobát is lefoglalhasson
- Be lehessen állítani a foglalások állapotát

## Technológia

A projekt a következő technológiákat/szoftvereket fogja használni a munka során:

- Egyszerű CloudLinux alapú Cpanel webtárhely MySQL adatbázissal
- PHP (backend)
- Github szoftverprojekt menedzselő szolgáltatás
- VSCode IDE
- Git verziókövető (Github)

## Egyed-kapcsolat diagram



### Egyed-kapcsolat diagram leírása

Az hogy az egyes tábláknak milyen attribútumaik vannak vagy vagy hogy épp mi a nevük, szerintem teljesen egyértelmű, ezért nem fogom most újra leírni. A következő párszáz sor ezt taglalja, most foglalkozzunk inkább az izgalmasabb jelölésekkel.

A Személy táblának van egy több értékű attribútuma, amely leképezés után egy külön táblába fog kerülni. Ennek hála egy személynek egyszerre akár több rangja is lehet. E mellett van egy összetett attribútuma is (név), amelynek részattribútumai kerülnek csak bele leképezésnél a táblába.

A Kupon gyenge egyed egy meghatározó kapcsolaton keresztül megkapja leképezésnél a személy kulcsát, ami nélkül nem lehetne egyértelműen hivatkozni rekordjaira (nem lenne pontos kulcsa).

Az Ellenőr és a Dolgozó a Személy specializációja. Úgy döntöttem a redundancia elkerülése végett leképezésnél a szülő kulcsát veszem hozzá a gyerek attribútumaihoz. Ez lassítja és összetettebbé teszi a gyerekek összes adatának lekérdezését, cserébe kisebb helyet foglal (Tegyük úgy mintha számítana egy ekkora adatbázisnál a méret).

A Foglалás egy N:M kapcsolat, szóval külön táblát kap leképezésnél (nem lehet össze vonni) attribútumaival és a két összekapcsolt tábla kulcsával.

A Takarít egy 1:M kapcsolat az az az M oldali táblával összevonható, így abba fognak bele olvadni attribútumai és a másik tábla kulcsa leképezéskor.

## Egyed-kapcsolat diagram leképezése konsolidált relációs adatbázissémákká

Személy(Keresztnév, Vezetéknév, Személyi igazolvány szám, Személy állapota)

Ellenőr(Személyi igazolvány szám, Ellenőr állapota, Leírás)

Dolgozó(Személyi igazolvány szám, beosztás, fizetés)

Rang(Személyi igazolvány szám, Rang név)

Kupon(Lejárat dátum, Leárazás értéke, Tulajdonos, Állapot)

Szoba(Szobaszám, Szoba név, Legutóbbi takarító, Legutóbbi takarítás időpontja)

Foglалás(Foglалás száma, Szobaszám, Foglалás kezdete, Foglалás vége, Foglалás állapota, *Foglалó személy*)

## Relációs adatbázissémák normalizálása

### Funkcionális függőségek:

Személy:

{Személyi igazolvány szám} -> {Keresztnév, Vezetéknév, Személy állapota}

{Személyi igazolvány szám, Keresztnév} -> {Vezetéknév, Személy állapota}

{Személyi igazolvány szám, Személy állapota} -> {Vezetéknév, Keresztnév}

{Személyi igazolvány szám, Vezetéknév} -> {Személy állapota, Keresztnév}

{Személyi igazolvány szám, Keresztnév, Vezetéknév} -> {Személy állapota}

{Személyi igazolvány szám, Keresztnév, Személy állapota} -> {Vezetéknév}

{Személyi igazolvány szám, Vezetéknév, Személy állapota} -> {Keresztnév}

Ellenőr:

{Személyi igazolvány szám} -> {Ellenőr állapota, Leírás}

{Személyi igazolvány szám, Ellenőr állapota} -> {Leírás}

{Személyi igazolvány szám, Leírás} -> {Ellenőr állapota}

{Személyi igazolvány szám, Beosztás} -> {Fizetés}

{Személyi igazolvány szám, Fizetés} -> {Beosztás}

Rang:

A táblának minden attribútuma kulcs is, ezért kicsit furán néz ki a függősége. Lényegében saját magát határozza meg.

{Személyi igazolvány szám, Rang név} -> {Rang név}

Kupon:

{Lejárat dátum, Leárazás értéke, Tulajdonos} -> {Állapot}

Szoba:

{Szobaszám} -> {Szoba név, Legutóbbi takarító, Legutóbbi takarítás időpontja}

{Szobaszám, Szoba név} -> {Legutóbbi takarító, Legutóbbi takarítás időpontja}

{Szobaszám, Legutóbbi takarító} -> {Szoba név, Legutóbbi takarítás időpontja}

{Szobaszám, Legutóbbi takarítás időpontja} -> {Szoba név, Legutóbbi takarító}

{Szobaszám, Szoba név, Legutóbbi takarító} -> {Legutóbbi takarítás időpontja}

{Szobaszám, Szoba név, Legutóbbi takarítás időpontja} -> {Legutóbbi takarító}

{Szobaszám, Legutóbbi takarító, Legutóbbi takarítás időpontja} -> {Szoba név}

Foglalás:

{Foglalás száma} -> {Szobaszám, Foglalás kezdete, Foglalás vége, Foglalás állapota, Foglalo személy}

{Foglalás száma, Szobaszám} -> {Foglalás kezdete, Foglalás vége, Foglalás állapota, Foglalo személy}

{Foglalás száma, Foglalás kezdete} -> {Szobaszám, Foglalás vége, Foglalás állapota, Foglalo személy}

{Foglalás száma, Foglalás vége} -> {Szobaszám, Foglalás kezdete, Foglalás állapota, Foglaló személy}

{Foglalás száma, Foglalás állapota} -> {Szobaszám, Foglalás kezdete, Foglalás vége, Foglaló személy}

{Foglalás száma, Foglaló személy} -> {Szobaszám, Foglalás kezdete, Foglalás vége, Foglalás állapota}

{Foglalás száma, Szobaszám, Foglalás kezdete} -> {Foglalás vége, Foglalás állapota, Foglaló személy}

{Foglalás száma, Szobaszám, Foglalás vége} -> {Foglalás kezdete, Foglalás állapota, Foglaló személy}

{Foglalás száma, Szobaszám, Foglalás állapota} -> {Foglalás kezdete, Foglalás vége, Foglaló személy}

{Foglalás száma, Szobaszám, Foglaló személy} -> {Foglalás kezdete, Foglalás vége, Foglalás állapota}

{Foglalás száma, Foglalás kezdete, Foglalás vége} -> {Szobaszám, Foglalás állapota, Foglaló személy}

{Foglalás száma, Foglalás kezdete, Foglalás állapota} -> {Szobaszám, Foglalás vége, Foglaló személy}

{Foglalás száma, Foglalás kezdete, Foglaló személy} -> {Szobaszám, Foglalás vége, Foglalás állapota}

{Foglalás száma, Foglalás vége, Foglalás állapota} -> {Szobaszám, Foglalás kezdete, Foglaló személy}

{Foglalás száma, Foglalás vége, Foglaló személy} -> {Szobaszám, Foglalás kezdete, Foglalás állapota}

{Foglalás száma, Foglalás állapota, Foglaló személy} -> {Szobaszám, Foglalás kezdete, Foglalás vége}

{Foglalás száma, Szobaszám, Foglalás kezdete, Foglalás vége} -> {Foglalás állapota, Foglaló személy}

{Foglalás száma, Szobaszám, Foglalás kezdete, Foglalás állapota} -> {Foglalás vége, Foglaló személy}

{Foglalás száma, Szobaszám, Foglalás kezdete, Foglaló személy} -> {Foglalás vége, Foglalás állapota}

{Foglalás száma, Szobaszám, Foglalás vége, Foglalás állapota} -> {Foglalás kezdete, Foglaló személy}

{Foglalás száma, Szobaszám, Foglalás vége, Foglaló személy} -> {Foglalás kezdete, Foglalás állapota}

{Foglalás száma, Szobaszám, Foglalás állapota, Foglaló személy} -> {Foglalás kezdete, Foglalás vége}

... - Had ne írjam ki az összeset ehhez a táblához. Lényegében a kulcs (Foglalás száma) után bármilyen attribútumot fűzünk, meg fogja határozni a tábla többi attribútumát.

Ha elvonatkoztatok a példa megoldástól és elkezdek gondolkodni, simán kijön, hogy két esélyes kulcsa van a foglalás táblának (na jó 3, de kettő lényegében ugyan az):

{Foglalás száma} -> {Szobaszám, Foglalás kezdete, Foglalás vége, Foglalás állapota, Foglaló személy} --- *a foglalás számát lényegében kulcsnak terveztem, nem csoda hát hogy egyértelműen meghatározza a tábla összes többi attribútumát. Ezt használtam később is kulcsnak, mert ő a legegyszerűbb.*

{Szobaszám, Foglalás kezdete, Foglaló személy} -> {Foglalás vége, Foglalás állapota, Foglalás száma} --- *Egy szobát egy személy egy időben nem bérelhet ki kétszer (legalábbis nem lenne értelme, ezért felételezem hogy nem gond, ha nem teheti meg), ezért akár ez is lehetne kulcs*

{Szobaszám, Foglалás vége, Foglалó személy} -> {Foglалás kezdete, Foglалás állapota, Foglалás száma} --- *Ugyan ez érvényesül, ha a foglалás végét vesszük kulcsnak akkor is.*

*Bármilyen ezeknél több nem lehet kulcs, mert ezek szűkebbek nála, ezért nincs is értelme foglалkozni vele.*

## Kulcsok

Külső kulcs más néven: idegen kulcs, Foreign key

Személy(Keresztnév, Vezetéknév, Személyi igazolvány szám, Személy állapota)

- Kulcs: Személyi igazolvány szám
  - Ez a legszűkebb halmaz, amelynek lezárta visszaadja a teljes attribútumhalmazt.

Ellenőr(Személyi igazolvány szám, Ellenőr állapota, Leírás)

- Kulcs: Személyi igazolvány szám
  - Ez a legszűkebb halmaz, amelynek lezárta visszaadja a teljes attribútumhalmazt.
- Külső kulcs: Személyi igazolvány szám
  - Személy tábla elsődleges kulcsára hivatkozik

Dolgozó(Személyi igazolvány szám, beosztás, fizetés)

- Kulcs: Személyi igazolvány szám
  - Ez a legszűkebb halmaz, amelynek lezárta visszaadja a teljes attribútumhalmazt.
- Külső kulcs: Személyi igazolvány szám
  - Személy tábla elsődleges kulcsára hivatkozik

Rang(Személyi igazolvány szám, Rang név)

- Kulcs: Személyi igazolvány szám
  - Ez a legszűkebb halmaz, amelynek lezárta visszaadja a teljes attribútumhalmazt.
- Külső kulcs: Személyi igazolvány szám
  - Személy tábla elsődleges kulcsára hivatkozik

Kupon(Lejárat dátum, Leárazás értéke, Tulajdonos, Állapot)

- Kulcs: Lejárat dátum, Leárazás értéke, Tulajdonos
  - Ez a legszűkebb halmaz, amelynek lezárta visszaadja a teljes attribútumhalmazt.
- Külső kulcs: Tulajdonos
  - Személy tábla elsődleges kulcsára hivatkozik

Szoba(Szobaszám, Szoba név, Legutóbbi takarító, Legutóbbi takarítás időpontja)

- Kulcs: Szobaszám
  - Ez a legszűkebb halmaz, amelynek lezárta visszaadja a teljes attribútumhalmazt.
- Külső kulcs: Legutóbbi takarító
  - Dolgozó tábla elsődleges kulcsára hivatkozik

Foglалás(Foglалás száma, Szobaszám, Foglалás kezdete, Foglалás vége, Foglалás állapota, Foglалó személy)

- Kulcs: Foglалás száma

- Ez a legszűkebb halmaz, amelynek lezártja visszaadja a teljes attribútumhalmazt.
- Külső kulcs: Szobaszám
  - Szoba tábla elsődleges kulcsára hivatkozik
- Külső kulcs: Foglalo személy
  - Személy tábla elsődleges kulcsára hivatkozik

## Normalizálás

Minden tábla 3NF-ben lett a leképezés után, így nem igazán tudok normalizálni.

A minden táblán belül minden másodlagos attribútum atomi, teljesen függ bármely kulcstól és nincs tranzitív függés az attribútumok között, az az 3NF-ben van minden tábla.

Normalizáltság bizonyítása után a séma ugyan olyan maradt:

Személy(Keresztnév, Vezetéknév, Személyi igazolvány szám, Személy állapota)

Ellenőr(Személyi igazolvány szám, Ellenőr állapota, Leírás)

Dolgozó(Személyi igazolvány szám, beosztás, fizetés)

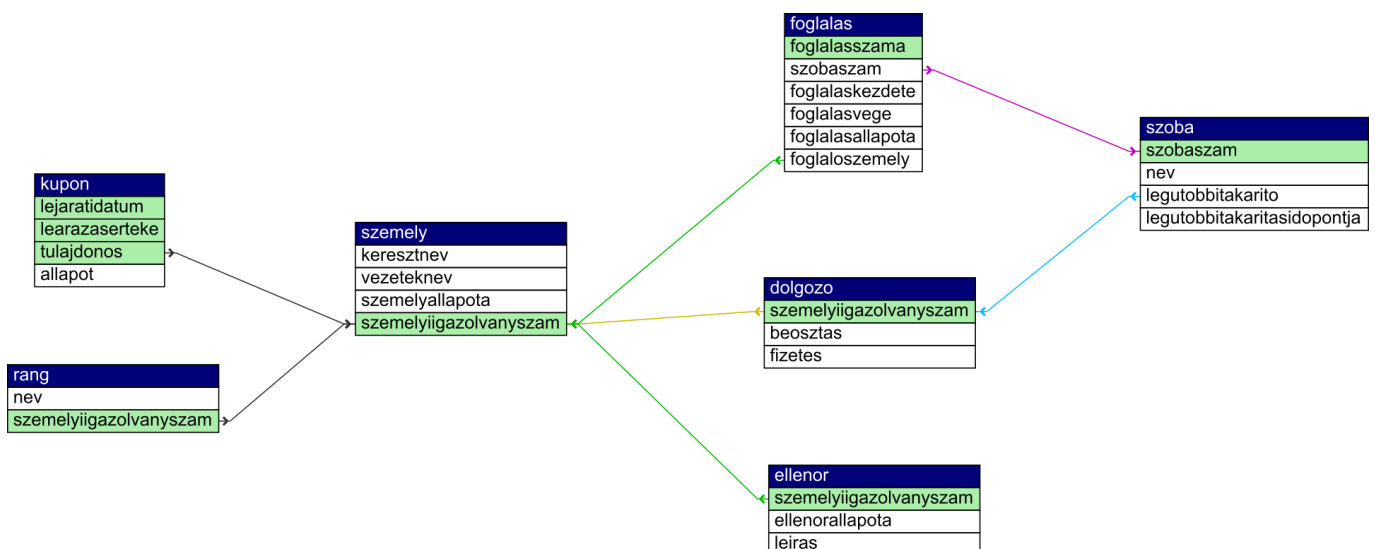
Rang(Személyi igazolvány szám, Rang név)

Kupon(Lejárat dátum, Leárazás értéke, Tulajdonos, Állapot)

Szoba(Szobaszám, Szoba név, Legutóbbi takarító, Legutóbbi takarítás időpontja)

Foglalás(Foglalás száma, Szobaszám, Foglалás kezdete, Foglалás vége, Foglалás állapota, Foglalo személy)

## Tábla tervek



Nem tudom ilyen tábla terv kellett e, ezért a dokumentum legaljára bemásoltam a tábla, kulcs és külső kulcs létrehozó SQL-t is.

## Összetett lekérdezések

1) Legutóbb takarított szoba(k):

```

SELECT szoba.szobaszam, szoba.nev, MIN(szoba.legutobbitakaritasidopontja)
FROM szoba
GROUP BY szoba.szobaszam, szoba.nev
HAVING MIN(szoba.legutobbitakaritasidopontja) = (
    SELECT MIN(legregebbi)
    FROM (
        SELECT szoba.szobaszam, MIN(szoba.legutobbitakaritasidopontja) AS legregebbi
        FROM szoba
        GROUP BY szoba.szobaszam
    ) as x
)

```

2) Legtöbbet foglalt személy(ek):

```

SELECT vezeteknev, keresztnév, személyiigazolványszám FROM személy
WHERE személyiigazolványszám
IN (
    SELECT foglaloszemely
    FROM (
        SELECT foglaloszemely, COUNT(*) AS darab
        FROM foglalas
        GROUP BY foglaloszemely
        having count(*) = (
            SELECT max(x.darab)
            FROM (
                SELECT foglaloszemely, COUNT(*) AS darab
                FROM foglalas
                GROUP BY foglaloszemely
            ) as x
        )
    ) as xy
)

```

3) Jelenleg szabad szobák száma:

```

SELECT COUNT(*) as uresszobakszama
FROM szoba
WHERE szoba.szobaszam IN (
    SELECT foglalas.szobaszam
    FROM foglalas
    WHERE CURRENT_DATE BETWEEN foglalaskezdetek AND foglalasvege
)

```

4) Legkevesebbet kereső dolgozó(k):



```

SELECT vezeteknev, keresztnév, dolgozo.szemelyiigazolvanszam
FROM dolgozo
LEFT OUTER JOIN személy ON személy.szemelyiigazolvanszam =
dolgozo.szemelyiigazolvanszam
WHERE fizetes = (
    SELECT MIN(fizetes) FROM dolgozo
)

```

#### 5) Legtöbbet kereső dolgozó(k):

```

SELECT vezeteknev, keresztnév, dolgozo.szemelyiigazolvanszam
FROM dolgozo
LEFT OUTER JOIN személy ON személy.szemelyiigazolvanszam =
dolgozo.szemelyiigazolvanszam
WHERE fizetes = (
    SELECT MAX(fizetes) FROM dolgozo
)

```

#### 6) Foglalások listázása (Foglalás kezdetének ideje szerint rendezve):

```

SELECT foglalasszama, foglalas.szobaszam, nev, foglalaskezdetek, foglalasvege,
foglalasallapota, foglaloszemely, személy.keresztnév, személy.vezeteknev
FROM foglalas
LEFT OUTER JOIN szoba ON foglalas.szobaszam = szoba.szobaszam
LEFT OUTER JOIN személy ON foglalas.foglaloszemely =
személy.szemelyiigazolvanszam
ORDER BY foglalaskezdetek

```

#### 7) Dolgozók listázása (Név szerint rendezve):

```

SELECT keresztnév, vezeteknev, személyallapota, személy.szemelyiigazolvanszam,
beosztas, fizetes
FROM személy
RIGHT OUTER JOIN dolgozo ON személy.szemelyiigazolvanszam =
dolgozo.szemelyiigazolvanszam
ORDER BY keresztnév, vezeteknev

```

### Tábla, kulcs, külső kulcs létrehozó SQL utasítássorozat

```

CREATE TABLE `dolgozo` (
    `szemelyiigazolvanszam` int(32) NOT NULL,
    `beosztas` varchar(32) COLLATE utf8_hungarian_ci DEFAULT NULL,
    `fizetes` int(16) DEFAULT NULL

```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_hungarian_ci;
```

```
CREATE TABLE `ellenor` (  
  `szemelyiigazolvanszam` int(32) NOT NULL,  
  `ellenorallapota` varchar(32) COLLATE utf8_hungarian_ci DEFAULT NULL,  
  `leiras` text COLLATE utf8_hungarian_ci DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_hungarian_ci;
```

```
CREATE TABLE `foglalas` (  
  `foglalasszama` int(11) NOT NULL,  
  `szobaszam` int(4) DEFAULT NULL,  
  `foglalaskezdetek` date DEFAULT NULL,  
  `foglalasvege` date DEFAULT NULL,  
  `foglalasallapota` varchar(32) COLLATE utf8_hungarian_ci DEFAULT NULL,  
  `foglaloszemely` int(32) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_hungarian_ci;
```

```
CREATE TABLE `kupon` (  
  `lejaratidatum` date NOT NULL,  
  `learazaserteke` int(2) NOT NULL,  
  `tulajdonos` int(32) NOT NULL,  
  `allapot` varchar(16) COLLATE utf8_hungarian_ci DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_hungarian_ci;
```

```
CREATE TABLE `rang` (  
  `nev` varchar(32) COLLATE utf8_hungarian_ci NOT NULL,  
  `szemelyiigazolvanszam` int(32) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_hungarian_ci;
```

```
CREATE TABLE `szemely` (  
  `keresztnev` varchar(128) COLLATE utf8_hungarian_ci DEFAULT NULL,  
  `vezeteknev` varchar(128) COLLATE utf8_hungarian_ci DEFAULT NULL,  
  `szemelyallapota` varchar(32) COLLATE utf8_hungarian_ci DEFAULT NULL,  
  `szemelyiigazolvanszam` int(32) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_hungarian_ci;
```

```
CREATE TABLE `szoba` (  
  `szobaszam` int(4) NOT NULL,  
  `nev` varchar(32) COLLATE utf8_hungarian_ci DEFAULT NULL,  
  `legutobbitakarito` int(32) DEFAULT NULL,  
  `legutobbitakaritasidopontja` datetime DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_hungarian_ci;
```

```
ALTER TABLE `dolgozo`  
  ADD PRIMARY KEY (`szemelyiigazolvanszam`);
```

```
ALTER TABLE `ellenor`  
  ADD PRIMARY KEY (`szemelyiigazolvanszam`);
```

```
ALTER TABLE `foglalas`  
  ADD PRIMARY KEY (`foglalasszama`),  
  ADD KEY `szobaFoglal` (`szobaszam`),  
  ADD KEY `szemelyFoglal` (`foglaloszemely`);
```

```

ALTER TABLE `kupon`
  ADD PRIMARY KEY (`lejaratidatum`,`learazaserteke`,`tulajdonos`),
  ADD KEY `kuponBirtokol` (`tulajdonos`);

ALTER TABLE `rang`
  ADD PRIMARY KEY (`nev`,`szemelyiigazolvanyaszam`) USING BTREE,
  ADD KEY `rangjaVan` (`szemelyiigazolvanyaszam`);

ALTER TABLE `szemely`
  ADD PRIMARY KEY (`szemelyiigazolvanyaszam`);

ALTER TABLE `szoba`
  ADD PRIMARY KEY (`szobaszam`),
  ADD KEY `takarit` (`legutobbitakarito`);

ALTER TABLE `foglalas`
  MODIFY `foglalasszama` int(11) NOT NULL AUTO_INCREMENT;

ALTER TABLE `dolgozo`
  ADD CONSTRAINT `Dolgozo_bovit` FOREIGN KEY (`szemelyiigazolvanyaszam`) REFERENCES `szemely` (`szemelyiigazolvanyaszam`) ON DELETE CASCADE ON UPDATE CASCADE;

ALTER TABLE `ellenor`
  ADD CONSTRAINT `ellenorBovit` FOREIGN KEY (`szemelyiigazolvanyaszam`) REFERENCES `szemely` (`szemelyiigazolvanyaszam`) ON DELETE CASCADE ON UPDATE CASCADE;

ALTER TABLE `foglalas`
  ADD CONSTRAINT `szemelyFoglal` FOREIGN KEY (`foglaloszemely`) REFERENCES `szemely` (`szemelyiigazolvanyaszam`) ON DELETE CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `szobaFoglal` FOREIGN KEY (`szobaszam`) REFERENCES `szoba` (`szobaszam`) ON DELETE CASCADE ON UPDATE CASCADE;

ALTER TABLE `kupon`
  ADD CONSTRAINT `kuponBirtokol` FOREIGN KEY (`tulajdonos`) REFERENCES `szemely` (`szemelyiigazolvanyaszam`) ON DELETE CASCADE ON UPDATE CASCADE;

ALTER TABLE `rang`
  ADD CONSTRAINT `rangjaVan` FOREIGN KEY (`szemelyiigazolvanyaszam`) REFERENCES `szemely` (`szemelyiigazolvanyaszam`) ON DELETE CASCADE ON UPDATE CASCADE;

ALTER TABLE `szoba`
  ADD CONSTRAINT `takarit` FOREIGN KEY (`legutobbitakarito`) REFERENCES `dolgozo` (`szemelyiigazolvanyaszam`) ON DELETE SET NULL ON UPDATE CASCADE;
COMMIT;

```