

Jegyzőkönyv
Adatbázis rendszerek I.
Féléves feladat
Állami Vasútközlekedési Vállalat

Készített:	Czikó Tivadar
Neptunkód:	O2IXLB
Gyakorlat:	Kedd 08-10
Gyakorlatvezető:	Dr.Bednarik László

Tartalomjegyzék:

1. A feladata leírása	3
2. Az adatbázis ER modellje	4
3. Az adatbázis konvertálása relációs modellre	4
4. Az adatbázis relációs modellje	5
5. Az adatbázis relációs sémái	5
6. A táblák létrehozás	6
7. A táblák feltöltése	8
8. A táblák módosítása	11
9. Lekérdezés	10

1. A feladat leírása:

Egy állam vasúti közlekedésének az adatbázisának ER-modellezése, átalakítása relációs adatmodellre, majd megvalósítása, feltöltése, módosítása SQL-ben, majd néhány lekérdezés az elkészült adatbázisból. Az ER-modell az alábbi jellemzőkkel épül fel:

A **Vonat** egyedhez tartoznak Azonosítószámok, a Vagonok száma (hány vagon tud húzni), a Gyártó és Típus tulajdonságok. Az azonosítószám kulcsként funkcionál, mert egyértelműen azonosítja az adott **vonatot**.

A **Mozdonyvezető** egyednek a *Mikor vezeti* lesz kulcsa. A **Mozdonyvezető** a **Vonattal** összekötött kapcsolat a VEZETI. A Mozdonyvezetőnek az *életkor* attribútuma származtatott, ugyanis az kiszámolható a *születési dátum* tulajdonságból. A Jogosítványok egy többértékű tulajdonság, hiszen egy **Vonatvezető**nek több, különböző jogosítványa is lehet. A *Név* egy összetett attribútum, amely a *Vezetéknév* és az *keresztnev* altulajdonságokból áll elő.

A **Vonat** és a **Mozdonyvezető** egyedek közötti kapcsolat kötelező, mert valakinek vezetnie kell az adott Vonatot és a Mozdonyvezetőnek és vezetnie kell egy Vonatot (ekkor kerül az adatbázisba). 1:1 típusú, hiszen egy vonatot egy ember vezethet és egy ember is egy Vonatot képes vezetni.

A **Menetrend** egyed tulajdonságai a *Menetrend* azonosító, az *Indulás* időpontja és az *Érkezés* időpontja. A *Menetrend* azonosító szolgál kulcsként.

A **Menetrend** és a **Vonat** egyedek között KÖZLEKEDIK kapcsolat 1:N Típusú, több **Vonat** is közlekedhet ugyan azon menetrend szerint. Kötelező kapcsolat.

A **Kalauz** egyed tulajdonsága az *Kalauz azonosító*, amely a kulcs szerepét is betölti. A **Kalauz** egyed továbbá jelentkezik egy *Név* összetett tulajdonságból, amely a *Vezetéknév* és *Keresztnev* tagokból áll össze.

A **Kalauz** és a **Mozdonyvezető** egyedek között a TARTÓZKODIK kapcsolat áll fenn, amely a **Vonat** felé kötelező, a **Kalauz** fel opcionális (a **kalauz**nak vonaton kell lennie, de nem kell minden vonatra kalauz) és 1:1 Típusú (egy **Vonaton** egy **Kalauz** lehet).

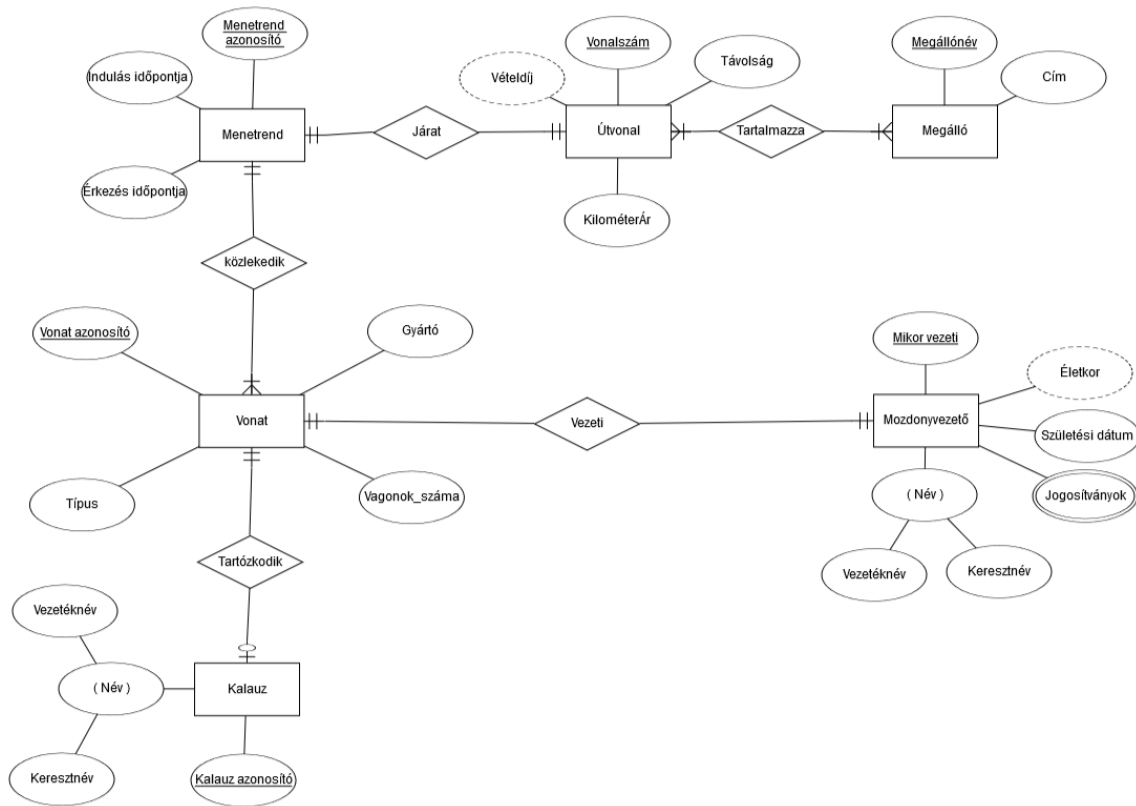
Az **Útvonal** egyedhez tartozó tulajdonságok a *Vonalszám*, a *Távolság*, a *Kilométerár* és a *Viteldíj*. A *Vonalszám* kulcs, amely egyértelműen azonosítja az adott **Útvonalat**. A *Viteldíj* egy származtatott tulajdonság, a *Távolság* és a *Kilométerár*ból számolható.

Az **Útvonal** és a **Menetrend** egyedek között lévő JÁRAT kapcsolat kötelező (minden útvonalon van menetrend) és 1:1 típusú (egy útvonalon egyféle menetrend).

A **Megálló** is egy egyed, amelyben a *Megállónév* tulajdonsága kulcsként funkcionál. A **Megálló** egyednek tulajdonsága továbbá a *Cím*.

Az **Útvonal** és a **Megálló** egyedek között a TARTALMAZZA kapcsolat van, amely N:N típusú, ugyanis egy útvonalhoz több megálló is tartozhat és egy megállóhoz is tartozhat többféle útvonal. Kötelező kapcsolat, hiszen megállók nélkül nincs útvonal és útvonal nélkül nincs értelme egy megállónak sem.

2. Az adatbázis ER modellje:



3. Az adatbázis konvertálása relációs modellre:

A **Mozdonyvezető** egyedből tábla lesz, a *Születési dátumból* mező lesz, az *Életkort* elhagyjuk, mivel származtatott mező, a *Név* tulajdonság kimarad, csak *Vezetéknévből* és *Keresztnévből* tagattribútumaiból lesznek mezők. A **Jogosítványok** többértékű tulajdonságból külön tábla lesz. A **Mozdonyvezető** Táblában altárolására kerülnek a **jogosítványok** elsődleges kulcsi idegen kulcsokként. A *Mikor vezeti* elsődleges kulcsa lesz a **Mozdonyvezető** táblának.

A **Vonat** egyedből tábla lesz, a *Vonat azonosító* tulajdonságból kulcsmező, a *Vagonok száma*, **Gyártó** és a **Típus** attribútumaiból lesznek mezők. A **MOZDONYVEZETŐ** kapcsolat helyett a *Mikor vezeti* idegen kulcsként a **Mozdonyvezető** azonosító lesz, a **Kalauz** táblához való kapcsolódást pedig a *Kalauz azonosító* idegen kulcsként való eltárolásával történik.

A **Kalauz** egyedből tábla lesz. A *Kalauz azonosító* kulcs tulajdonságából pedig kulcsmező. A *Név* származtatott mezőt elhagyjuk, helyette a *Vezetéknév* és *Keresztnév* kerül eltárolásra, külön mezőként.

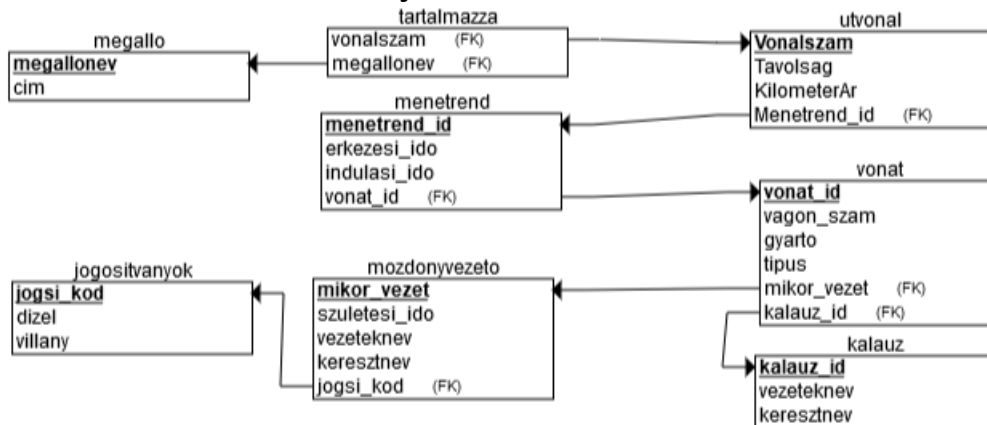
A **Menetrend** egyedből egy tábla lesz, a *Menetrend azonosító* kulcstulajdonságból kulcsmező, az *Indulás időpontja* és az *érkezés időpontjából* mezők lesznek. A **Vonat** tábla elsődleges kulcsa a *Vonat azonosító* néven idegen kulcsként eltárolódik a **Közlekedik** a kapcsolat helyett.

Az **Útvonal** egyedből tábla lesz, a *Vonalszám* tulajdonságból kulcs, a *Távolság* és *Kilométerár* tulajdonságokból mezők lesznek, a származtatott *Viteldíj* tulajdonságot elhagyjuk. A **Menetrend** táblával való **KÖZLEKEDIK** kapcsolat a **Menetrend** tábla *Menetrend azonosítójának* idegen kulcsként való eltárolásával valósul meg.

A **TARTALMAZZA** kapcsolatból tábla lesz, melynek a Vonalszám idegen kulcsa lesz, az **Útvonal** táblából és a **Megállónév** idegen kulcsa lesz nem üres mezője a **Megálló** táblából.

A **Megálló** egyedből egy tábla lesz, melynek **Megállónév** kulcstulajdonságából kulcsmező, a **Cím** attribútumából mező lesz.

4. Az adatbázis relációs modellje:



5. Az adatbázis relációs Schema-i:

Jogosítványok [Jogosítványkód, Dízelmozdony, Villamosmozdony]

Kalauz [Kalauz azonosító, Vezetéknév, Keresztnev]

Menetrend [Menetrend azonosító, Érkezés időpontja, Indulás időpontja, Vonat azonosító]

Megálló [Megállónév, Cím]

Mozdonyvezető [Mikor vezeti, Születési dátum, Vezetéknév, Keresztnev, Jogosítványkód]

Útvonal [Vonalszám, Távolság, Kilométerár, Menetrend azonosító]

Tartalmazza [Vonalszám, Megállónév]

Vonat [Vonat azonosító, Vagonok száma, gyártó, Típus, Kalauz azonosító, Mozdonyvezető azonosító]

6. A táblák létrehozása:

```
CREATE TABLE jogositvanyok (  
    jogsikod INT NOT NULL,  
    dize1 CHAR(4) NOT NULL,  
    villany CHAR(4) NOT NULL,  
    PRIMARY KEY (jogsikod)  
);  
  
CREATE TABLE kalauz (  
    kalauz_id INT NOT NULL,  
    vezeteknev VARCHAR(20) NOT NULL,  
    keresztnév VARCHAR(20) NOT NULL,  
    PRIMARY KEY (kalauz_id)  
);  
  
CREATE TABLE megallo (  
    cim VARCHAR(30) NOT NULL,  
    megallonev VARCHAR(20) NOT NULL,  
    PRIMARY KEY (megallonev)  
);  
  
CREATE TABLE mozdonyvezeto (  
    mikor_vezet DATE NOT NULL,  
    szuletesi_ido DATE NOT NULL,  
    vezeteknev CHAR(30) NOT NULL,  
    keresztnév CHAR(30) NOT NULL,  
    jogsikod INT NOT NULL,  
    PRIMARY KEY (mikor_vezet),  
    FOREIGN KEY (jogsikod) REFERENCES jogositvanyok(jogsikod)  
);  
  
CREATE TABLE vonat (  
    vonat_id CHAR(7) NOT NULL,  
    vagon_szam INT NOT NULL,  
    gyarto VARCHAR(20) NOT NULL,  
    tipus VARCHAR(20) NOT NULL,  
    mikor_vezet DATE NOT NULL,  
    kalauz_id INT NOT NULL,
```

```

PRIMARY KEY (vonat_id),
FOREIGN KEY (mikor_vezet) REFERENCES mozdonyvezeto(mikor_vezet),
FOREIGN KEY (kalauz_id) REFERENCES kalauz(kalauz_id)
);

CREATE TABLE menetrend (
    menetrend_id INT PRIMARY KEY NOT NULL,
    erkezesi_ido VARCHAR(5) NOT NULL,
    indulasi_ido VARCHAR(5) NOT NULL
);

CREATE TABLE utvonal (
    TavoIsag INT NOT NULL,
    KilometerAr INT NOT NULL,
    Vonalszam VARCHAR(3) NOT NULL,
    Menetrend_id INT NOT NULL,
    PRIMARY KEY (Vonalszam),
    FOREIGN KEY (Menetrend_id) REFERENCES menetrend(menetrend_id)
);

CREATE TABLE tartalmazza (
    vonalszam VARCHAR(3) NOT NULL,
    megallonev VARCHAR(20) NOT NULL,
    FOREIGN KEY (vonalszam) REFERENCES utvonal(Vonalszam),
    FOREIGN KEY (megallonev) REFERENCES megallo(megallonev)
);

```

7. A táblák feltöltése:

```

INSERT INTO jogositvanyok (jogsi_kod, dize1, villany) VALUES (1, 'IGEN', 'IGEN');
INSERT INTO jogositvanyok VALUES (2, 'IGEN', 'NEM');
INSERT INTO jogositvanyok VALUES (3, 'NEM', 'IGEN');
INSERT INTO jogositvanyok VALUES (4, 'IGEN', 'IGEN');

INSERT INTO `mav_info`.`mozdonyvezeto` (`mikor_vezet`, `szuletesi_ido`,
`vezeteknev`, `keresztnev`, `jogsi_kod`)
VALUES ('2020/11/18', '1990/05/03', 'Al Hag', 'Amin', '1');
INSERT INTO `mav_info`.`mozdonyvezeto` (`mikor_vezet`, `szuletesi_ido`,
`vezeteknev`, `keresztnev`, `jogsi_kod`)
VALUES ('2020/11/19', '1987/11/09', 'Schuldenfrei', 'Gustav', '2');

```

```

INSERT INTO `mav_info`.`mozdonyvezeto` (`mikor_vezet`, `szuletesi_ido`,
`vezeteknev`, `keresztnev`, `jogsi_kod`)
VALUES ('2020/11/20', '1972/03/02', 'Smirnov', 'Vladimir', '3');
INSERT INTO `mav_info`.`mozdonyvezeto` (`mikor_vezet`, `szuletesi_ido`,
`vezeteknev`, `keresztnev`, `jogsi_kod`)
VALUES ('2020/11/21', '1979/06/17', 'Kovalainen', 'Erik', '4');

INSERT INTO `mav_info`.`megallo` (`cim`, `megallonev`)
VALUES ('Kossuth ut 30', 'Hidasnemeti');
INSERT INTO `mav_info`.`megallo` (`cim`, `megallonev`)
VALUES ('Stanicnei Namestie 1', 'Kosice');
INSERT INTO `mav_info`.`megallo` (`cim`, `megallonev`)
VALUES ('Kando Kalman ut 4', 'Miskolc Tiszai');
INSERT INTO `mav_info`.`megallo` (`cim`, `megallonev`)
VALUES ('Vasarter utca 30', 'Szikszo');
INSERT INTO `mav_info`.`megallo` (`cim`, `megallonev`)
VALUES ('fo ut 3', 'Forro Encs');
INSERT INTO `mav_info`.`megallo` (`cim`, `megallonev`)
VALUES ('Moszkvai ut 15', 'Budapest Kelelti');
INSERT INTO `mav_info`.`megallo` (`cim`, `megallonev`)
VALUES ('Pesti ut 84', 'Hatvan');
INSERT INTO `mav_info`.`megallo` (`cim`, `megallonev`)
VALUES ('Egri ut 54', 'Fuzesabony');
INSERT INTO `mav_info`.`megallo` (`cim`, `megallonev`)
VALUES ('Allomas ut 4', 'Malyi');
INSERT INTO `mav_info`.`megallo` (`cim`, `megallonev`)
VALUES ('Fonoda utca 32', 'Miskolc Gomori');
INSERT INTO `mav_info`.`megallo` (`cim`, `megallonev`)
VALUES ('Peteri ut 3', 'Kazincbarcika');
INSERT INTO `mav_info`.`megallo` (`cim`, `megallonev`)
VALUES ('Petofi ut 12', 'Ozd');
INSERT INTO `mav_info`.`megallo` (`cim`, `megallonev`)
VALUES ('tavak utja 9', 'Nyek');

```



```
INSERT INTO kalauz (kalauz_id, vezeteknev, keresztnév) VALUES  
(101, 'Hatake', 'Kakashi');
```

```
INSERT INTO kalauz VALUES (102, 'Marchand', 'Jean Pierre');
```

```
INSERT INTO kalauz VALUES (103, 'Da Costa', 'Joao');
```

```
INSERT INTO kalauz VALUES (104, 'Nguen', 'Bayoden');
```

```
INSERT INTO `mav_info`.`menetrend` (`menetrend_id`, `erkezesi_ido`,  
`indulasi_ido`) VALUES ('1', '11:15', '07:30');
```

```
INSERT INTO `mav_info`.`menetrend` (`menetrend_id`, `erkezesi_ido`,  
`indulasi_ido`) VALUES ('2', '12:45', '10:25');
```

```
INSERT INTO `mav_info`.`menetrend` (`menetrend_id`, `erkezesi_ido`,  
`indulasi_ido`) VALUES ('3', '10:15', '09:30');
```

```
INSERT INTO `mav_info`.`menetrend` (`menetrend_id`, `erkezesi_ido`,  
`indulasi_ido`) VALUES ('4', '12:44', '11:00');
```

```
INSERT INTO `mav_info`.`vonat` (`vonat_id`, `vagon_szam`, `gyarto`, `tipus`,  
`mikor_vezet`, `kalauz_id`)
```

```
VALUES ('HU1010Z', '12', 'FLIRT', 'dizel', '2020/11/18', '101');
```

```
INSERT INTO `mav_info`.`vonat` (`vonat_id`, `vagon_szam`, `gyarto`, `tipus`,  
`mikor_vezet`, `kalauz_id`)
```

```
VALUES ('HU202KB', '6', 'Bzmot', 'dizel', '2020/11/19', '102');
```

```
INSERT INTO `mav_info`.`vonat` (`vonat_id`, `vagon_szam`, `gyarto`, `tipus`,  
`mikor_vezet`, `kalauz_id`)
```

```
VALUES ('HU303BP', '8', 'TAURUS', 'villany', '2020/11/20', '103');
```

```
INSERT INTO `mav_info`.`vonat` (`vonat_id`, `vagon_szam`, `gyarto`, `tipus`,  
`mikor_vezet`, `kalauz_id`)
```

```
VALUES ('SK404KS', '10', 'BDVmoz', 'villany', '2020/11/21', '104');
```

```
INSERT INTO `mav_info`.`utvonal` (`Tavolsag`, `KilometerAr`, `Vonalszam`,  
`Menetrend_id`)
```

```
VALUES ('80', '13', 'ECR', '1');
```

```
INSERT INTO `mav_info`.`utvonal` (`Tavolsag`, `KilometerAr`, `Vonalszam`,  
`Menetrend_id`)
```

```
VALUES ('215', '25', 'GYF', '2');
```

```
INSERT INTO `mav_info`.`utvonal` (`Tavolsag`, `KilometerAr`, `Vonalszam`,  
`Menetrend_id`)
```

```
VALUES ('70', '300', 'TEO', '3');
```

```
INSERT INTO `mav_info`.`utvonal` (`Tavolsag`, `KilometerAr`, `Vonalszam`,
`Menetrend_id`)
VALUES ('80', '8', 'SZE', '4');
```

```
INSERT INTO tartalmazza (vonalszam,megallonev)VALUES('ECR','Kosice');
INSERT INTO tartalmazza VALUES ('ECR','Forro Encs');
INSERT INTO tartalmazza VALUES ('ECR','Miskolc Tiszai');
INSERT INTO tartalmazza VALUES ('ECR','Budapest keleti');
INSERT INTO tartalmazza VALUES ('GYF','Budapest keleti');
INSERT INTO tartalmazza VALUES ('GYF','Hatvan');
INSERT INTO tartalmazza VALUES ('GYF','Fuzesabony');
INSERT INTO tartalmazza VALUES ('GYF','Malyi');
INSERT INTO tartalmazza VALUES ('GYF','Miskolc Tiszai');
INSERT INTO tartalmazza VALUES ('TE0','Miskolc Tiszai');
INSERT INTO tartalmazza VALUES ('TE0','Miskolc Gomori');
INSERT INTO tartalmazza VALUES ('TE0','Kazincbarcika');
INSERT INTO tartalmazza VALUES ('TE0','Ozd');
INSERT INTO tartalmazza VALUES ('SZE','Kosice');
INSERT INTO tartalmazza VALUES ('SZE','Hidasnemeti');
INSERT INTO tartalmazza VALUES ('SZE','Forro Encs');
INSERT INTO tartalmazza VALUES ('SZE','Szikszo');
INSERT INTO tartalmazza VALUES ('SZE','Miskolc tiszai');
```

8. Táblák módosítása:

1. Az egyik mozdonyvezető megszerezte dízel vonatra is a jogosítványt:

```
UPDATE `jogositvanyok` SET `dizel` = 'igen' WHERE jogs_i_kod=3;
```

```
UPDATE `jogositvanyok` SET `villany` = 'igen' WHERE jogs_i_kod=2;
```

2. Az egyik kalauz megváltoztatta a nevét (magyarosította a nevét):

```
UPDATE `kalauz` SET `keresztnev` = 'Janos', `vezeteknev` = 'Kereskedo' WHERE
kalauz_id=102;
```

3. Kihasználatlanság miatt megszüntetünk egy megállót.

```
DELETE FROM `megallo` WHERE `megallonev` = 'Nyek';
```

4. A MAV iroda kéri a vonatok kibocsátási évét:

```
ALTER TABLE `vonat` ADD `kibocsatas` INT(5);
```

5. A 10:15-kor beérkező járatnak 15 perccel nőtt a menetideje és a MAV szeretné ezt módosítani:

```
UPDATE `menetrend` SET `erkezesi_ido`='10:30' WHERE `menetrend_id`='3';
```

6. Egyik megálló címét A Trianon 100 éves évfordulója alkalmából átnevezték:

```
UPDATE `megallo` SET `cim`='Horty Miklos ut 4' WHERE `megallonev`='Malyi';
```

7. A MAV az ECR kódú EC RÁKOCZI nevű járatát a COVID-19 miatt növeli a Kilométerárat:

```
UPDATE `utvonal` SET `KilometerAr`=(13*1.25) WHERE `vonalszam`='ECR';
```

9. Lekérdezések:

1. A MAV szeretné tudni mely mozdonyvezetők vannak nyugdíj közelében:

```
SELECT vezeteknev, keresztnév FROM mozdonyvezeto INNER JOIN jogositvanyok
```

```
ON mozdonyvezeto.jogsi_kod = jogositvanyok.jogsi_kod
```

```
WHERE szuletesi_ido < '1980/04/27' ;
```

```
Πvezeteknev,keresztnév (σszuletesi_ido<'1980/04/27'(mozdonyvezeto ⋈ mozdonyvezeto.jogsi_kod=jogositvanyok.jogsi_kod  
jogositvanyok))
```

2. A MAV járatainak átlagos kilométerára:

```
SELECT avg(KilometerAr) FROM utvonal;
```

```
Γ avg(KilometerAr) (utvonal)
```

3. Van megálló a Fő út 3-on?

```
SELECT megallonev FROM megallo WHERE cim like 'fo ut 3';
```

```
Πmegallonev(σcim='fo út 3'(megallo))
```

4. A MÁV a régi útvonalak villanyosításán dolgozik az új EU-s szabvány környezet védelmi intézkedési miatt, ezért szükséges lekérni a villany jogosítvánnyal rendelkező emberek száma:

```
SELECT vezeteknev, keresztnév FROM mozdonyvezeto INNER JOIN jogositvanyok
```

```
ON mozdonyvezeto.jogsi_kod = jogositvanyok.jogsi_kod WHERE villany = 'igen';
```

```
Πvezeteknev,keresztnév (σvillany='Igaz' (mozdonyvezeto ⋈ mozdonyvezeto.jogsi_kod = jogositvanyok.jogsi_kod  
jogositvanyok))
```

5. A mozdonyvezetők nevének és a menetrendben szereplő, általa vezetett mozdony gyártójának lekérdezése:

SELECT DISTINCT vezeteknev, keresztnév, gyarto **FROM** mozdonyvezeto,
vonat **WHERE** mozdonyvezeto.mikor_vezet=vonat.mikor_vezet;

$\Pi_{\text{vezeteknev, keresztnév}}(\sigma_{\text{mozdonyvezeto.mikor_vezet=vonat.mikor_vezet}}(\text{mozdonyvezeto} \times \text{vonat}))$

6. A járatok vonalszámának és a hozzájuk kapcsolódó indulási és érkezési időpontok lekérdezése:

SELECT vonalszam, erkezesi_ido, indulasi_ido **FROM** menetrend

JOIN utvonal **USING** (menetrend_id);

$\Pi_{\text{vonalszam, erkezesi_ido, indulasi_ido}}(\text{menetrend} \bowtie \text{utvonal})$

7. Az útvonalaknak hány megállója van?

SELECT vonalszam, count(*) **FROM** tartalmazza **GROUP BY** vonalszam;

$\Gamma_{\text{vonalszam, count(*)}}^{\text{vonalszam}}(\text{tartalmazza})$

8. A MAV-nak nyilvánvalóan szüksége van annak a nyilvántartására, hogy az adott típusú mozdonyból hány van:

SELECT tipus, count(*) **FROM** vonat **GROUP BY** tipus;

$\Gamma_{\text{tipus, count(*)}}^{\text{tipus}}(\text{vonat})$

9. Azon mozdonyok azonosítószámainak és húzható vagonok lekérdezése, melynek több vagonja van mint az összes mozdony húzható vagonok számának összeségének az átlaga.

SELECT vonat_id, vagon_szam **FROM** vonat **WHERE** vagon_szam > (**SELECT** avg
(vagon_szam) **FROM** vonat);

$\Pi_{\text{vonat_id, vagon_szam}}(\sigma_{\text{vagon_szam} > (\Gamma_{\text{avg(vagon_szam)}}(\text{vonat}))}(\text{vonat}))$

10. Azon járatoknak a számának lekérdezése, amelyeknek a távolságra vetített kilométeráruk a vállalat szempontjából nézve az átlagtól kisebb:

SELECT vonalszam **FROM** utvonal **WHERE** KilometerAr * Tavolsag < (**SELECT** avg
(KilometerAr*Tavolsag) **FROM** utvonal);

$\Pi_{\text{vonalszam}}(\sigma_{\text{KilometerAr} * \text{Tavolsag} < (\Gamma_{\text{avg (KilometerAr * Tavolsag)}}(\text{utvonal}))}(\text{utvonal}))$

