**Adatbázisok gyakorlat**

**Projektmunka**

**Tömegközlekedés**

**Készítette:** László Noémi

Boros Dániel

Jauch Norbert

2020.05.06.

**Specifikáció**

Projektünk célja, hogy létrehozzunk egy olyan adatbázist, amelynek felhasználása lehetővé teszi egy olyan használható alkalmazás készítését, ami hasznos információkkal szolgálhat az egyes tömegközlekedési járatokról.

Ebben az adatbázisban fogjuk tárolni az egyes vonalakat, azok típusával, számával és betűjelével, üzemanyaguk típusával, a vonal hosszával.

Minden vonalon haladnak járatok, melyeknek különböző attribútumaik vannak: indulási idő, alacsony padlós, csuklós, ülőhelyek száma, rokkant helyek száma, kerékpár szállítási lehetőség.

Tároljuk továbbá az egyes járműveket is, amik az egyes járatokon járnak. Hasonló attribútumokkal rendelkeznek, mint a járatok, emellett tároljuk még a jármű típusát és hogy szükséges-e a javítása.

A naptípusok is tárolásra kerülnek, melyeknek lesz egy szöveges leírásuk, miszerint munkanap, munkaszüneti nap, szabadnap, stb-ről van szó.

Ezen kívül tároljuk az állomásokat (a kezdőállomást, köztes állomásokat, valamint a végállomást), nevükkel, földrajzi helyüknek megnevezésével, illetve GPS koordinátáikkal.

A feladat alapjául az egyes járatokat választottuk, amelyek a különböző vonalakon haladnak. Ennek megfelelően lesz majd az adatbázisban 8-as (troli) vonal, 2-es (villamos) vonal, 20-as (busz) vonal, stb, amelyen meghatározott időpontokban indulnak járatok.

Minden egyed elsődleges kulcsa egy sorszám lesz.

**Adatmodell**

**Az adatmodell leírása**

Az adatmodell célja egy olyan adatbázis létrehozása, amelynek segítségével alkalmazás készíthető, ami különböző tömegközlekedési járatokról szolgál információkkal.

Az adatmodellben helyet kapnak a vonalak, mint egyedek. A vonal egyed attribútumai az adott vonal száma (*vonalSzam int*), betűjele (*vonalBetujel varchar(2)*), a vonal hossza km-ben megadva (*vonalHossza float*). Ezen kívül még az egyes vonalak típusa (*vonalTipusa varchar (10)*) is külön attribútum, ami megadja, hogy busz, troli, vagy villamos az adott vonal. Minden vonalnak van üzemanyag típusa (*uzemanyagTipusa varchar (15)*), azaz, hogy gázolajjal, vagy villamos árammal működik. Mivel minden vonalnak van oda és vissza iránya, melyben a megállók különbözhetnek, így minden vonalhoz két sor fog tartozni a táblában, egy, ami az oda irányt jelöli, és egy másik, ami a vissza irányt. A vonal egyed elsődleges kulcsa egy sorszám (*vonalSorszam int*), melynek értéke automatikusan növekszik.

A következő egyed a járat, ami a vonal egyed része. Minden járatnak van saját indulási ideje (*indulasiIdo char (5)*), ami lehet 8:00, 8:10,.stb. További attribútumai még az ülőhelyek száma (*ulohelyekSzama int*), rokkant helyek száma (*rokkantHelyekSzama int*), hogy alacsony padlós-e (*alacsonyPadlos tiny(1)*), csuklós-e (*csuklos tiny(1)*) az adott járat, illetve, hogy lehet-e rajta biciklit szállítani, vagy sem (*kerekparSzallitasiLehetoseg tiny (1)*). Az elsődleges kulcsnak itt is automatikusan növekvő sorszámot adtunk meg (*jaratSorszam int*).

Az egyes járatokhoz tartoznak járművek. Ezek szintén egyedek, amik a járathoz hasonló attribútumokkal rendelkeznek, ám nincs indulási idejük. Emellett van típusuk (*tipus varchar (30)*), aminek meg kell egyeznie a vonal típusával, amin járnak. Továbbá van üzemeltetési költségük (*uzemeltetesiKoltseg float*), illetve egy attribútumuk, ami azt jelzi, hogy javításra szorul-e az adott jármű (*javitasSzukseges tiny (1)*). Az elsődleges kulcs szintén egy automatikusan növekvő sorszám (*jarmuSorszam int*).

A járat egyeddel kapcsolatban áll a nap típusa nevű egyed, melynek az automatikusan növekvő elsődleges kulcsán (*napTipusSorszam int*) kívül egyetlen attribútuma van: egy leírás (*leiras varchar(70)*), melyben rögzítjük, hogy munkanap, szabadnap, munkaszüneti nap, stb.

Az állomásokat is külön egyedben tároljuk, mind a kezdő- és végállomásokat, mind a köztes állomásokat. Minden állomásnak van neve (*nev varchar (40)*), X- és Y koordináta (*koordinataX varchar (30), koordinataY varchar (30)*) nevű attribútuma. Elsődleges kulcsként itt is egy sorszámot (*allomasSorszam int*) adtunk meg.

**Kapcsolatok**

Vonal-Állomás:

* *kezdete* 1-N
* *vége* 1-N
* *érint* N-M

Vonal-Járat: *része* 1-N

Járat-Jármű: *tartozik* 1-N

Járat-Naptípus: *jár* 1-N

Az egyes járatok az adott vonal részei. Tehát pl. a 8:00, 8:10, 8:20, stb. időpontokban induló járatok a 2-es számú, villamos típusú, x km hosszú, vonal része. Mivel egy vonalnak több járata van a nap folyamán, de egy járat csak egy bizonyos vonalhoz tartozhat, így ez egy 1-N kapcsolatként írható le.

A jármű és a járat egyedek közötti kapcsolat is 1-N kapcsolat, ugyanis egy járműhöz tartozhat több járat is, de egy járathoz csak egy jármű tartozik.

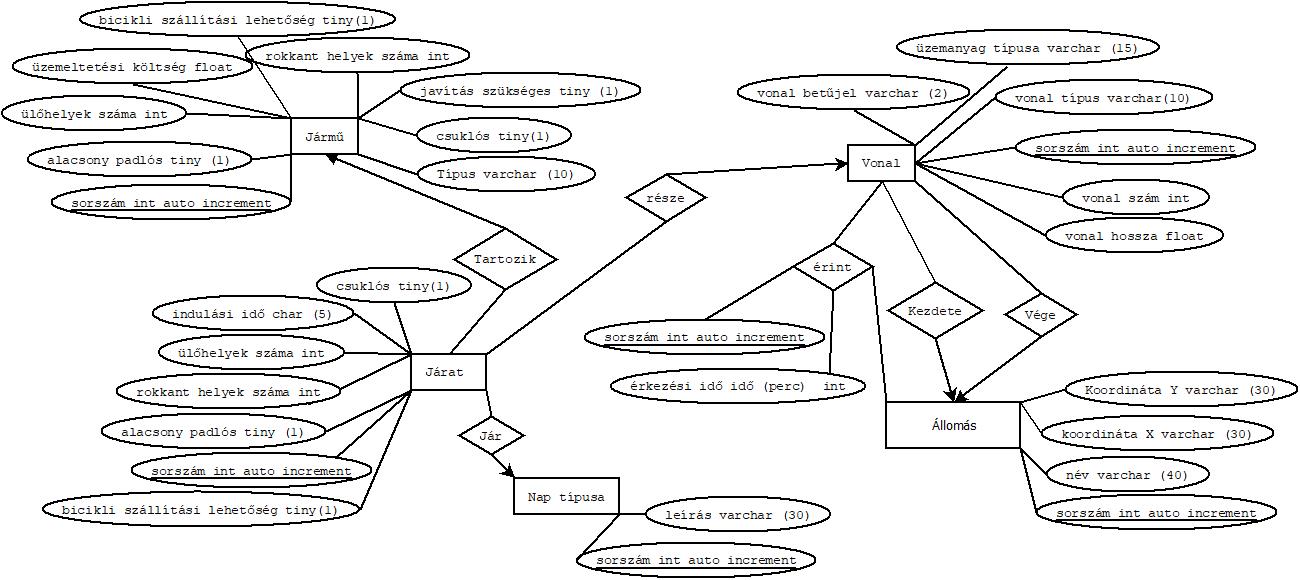
A járat és a nap típusa egyedek közötti kapcsolat szintén 1-N kapcsolat, mivel egy nap típushoz tartozhat több járat is, de egy járathoz csak egy nap típus tartozik.

Minden vonalnak van kezdőállomása, végállomása, valamint köztes állomásai. Ezeket külön kapcsolatként szükséges ábrázolni. Több vonal is indulhat ugyanarról a kezdőállomásról, emellett a végállomásuk is lehet azonos. Viszont egy vonalnak csak egy kezdőállomása és csak egy végállomása lehet. Emiatt a vonal és állomás egyedek közötti kezdete nevű kapcsolat 1-N kapcsolat és a szintén a vonal és állomás egyedek közötti vége nevű kapcsolat is 1-N kapcsolat.

Egy vonal több megállót is érint, valamint egy adott megállót több vonal is érint, így a vonal és állomás egyedek közötti érint nevű kapcsolat egy N-M kapcsolatként definiálható, melyhez kapcsolótábla felvitele szükséges, ami kapcsolódik a vonal egyedhez a vonal elsődleges kulcsán keresztül, és kapcsolódik az állomás egyedhez az állomás elsődleges kulcsán keresztül. Emellett az érint kapcsolathoz szükséges még felvenni az érkezési idő attribútumot, ami az egyes vonalaknak a kezdőállomástól való indulási idő és az adott állomásra történő érkezés időpontja között eltelt időt jelöli percben.

Az adatmodell alapján létrehozhatunk egy olyan adatbázist, amelyből lekérhetők az egyes járatok tulajdonságai, pl. a 12:45-kor a Személyi pályaudvar nevű állomásról a Tarján Víztorony tér nevű állomás irányába induló 90F-es busz mikor érkezik a Rókusi templom nevű megállóhoz, van-e rajta bicikli szállítási lehetőség, alacsony padlós-e, csuklós-e, mennyibe kerül a hozzá tartozó jármű üzemeltetése, a végállomásának mik a GPS koordinátái stb.

**Egyed-Kapcsolat diagram**



**Adatbázis létrehozása**

**Táblák létrehozása**

**1. Vonal tábla létrehozása**

CREATE TABLE vonal (

uzemanyagTipusa VARCHAR(15) NOT NULL,

vonalTipusa VARCHAR(10) NOT NULL,

vonalSorszam INT AUTO\_INCREMENT PRIMARY KEY,

vonalHossza FLOAT NOT NULL,

vonalBetujel VARCHAR(2) NOT NULL,

vonalSzam INT NOT NULL

)

**2.Járat tábla létrehozása**

CREATE TABLE jarat (

rokkantHelyekSzama INT NOT NULL,

ulohelyekSzama INT NOT NULL,

csuklos TINYINT(1) NOT NULL,

indulasiIdo CHAR(5) NOT NULL,

alacsonyPadlos TINYINT(1) NOT NULL,

jaratSorszam INT PRIMARY KEY AUTO\_INCREMENT NOT NULL,

kerekparSzallitasiLehetoseg TINYINT(1) NOT NULL

)

**3.Jármű tábla létrehozása**

CREATE TABLE jarmu (

rokkantHelyekSzama INT NOT NULL,

ulohelyekSzama INT NOT NULL,

csuklos TINYINT(1) NOT NULL,

alacsonyPadlos TINYINT(1) NOT NULL,

jarmuSorszam INT PRIMARY KEY AUTO\_INCREMENT NOT NULL,

kerekparSzallitasiLehetoseg TINYINT(1) NOT NULL,

uzemeltetesiKoltseg FLOAT NOT NULL,

tipus VARCHAR (10) NOT NULL,

javitasSzukseges TINYINT (1) NOT NULL

)

**4.Nap típus tábla létrehozása**

CREATE TABLE naptipus (

napTipusSorszam int PRIMARY KEY auto\_increment NOT NULL,

leiras VARCHAR (30) NOT null

)

**5.Állomás tábla létrehozása**

CREATE TABLE allomas (

koordianataX VARCHAR(30) NOT NULL,

koordianataY VARCHAR(30) NOT NULL,

nev VARCHAR (25) NOT NULL,

allomasSorszam INT PRIMARY KEY AUTO\_INCREMENT NOT NULL

)

**Kapcsolatok létrehozása**

**1.Vonal tábla és állomás táblák összekapcsolása**

1. **Kezdete kapcsolat létrehozása**

**Új oszlop felvitele a vonal táblába:**

ALTER TABLE vonal ADD kezdoAllomasSorszam INT

**Majd külső kulcs megadása a vonal táblához:**

ALTER TABLE vonal ADD FOREIGN KEY (kezdoAllomasSorszam)

REFERENCES allomas(allomasSorszam)

ON UPDATE CASCADE

ON DELETE SET NULL

1. **Vége kapcsolat létrehozása**

**Új oszlop megadása a vonal táblához:**

ALTER TABLE vonal ADD vegAllomasSorszam INT

**Majd külső kulcs megadása a vonal táblához:**

ALTER TABLE vonal ADD FOREIGN KEY (vegAllomasSorszam)

REFERENCES allomas(allomasSorszam)

ON UPDATE CASCADE

ON DELETE SET NULL

1. **Érint kapcsolótábla létrehozása**

CREATE TABLE erint (

erkezesiIdoPerc INT NOT NULL,

erintSorszam INT PRIMARY KEY AUTO\_INCREMENT NOT NULL,

allomasSorszam INT,

vonalSorszam INT,

FOREIGN KEY (allomasSorszam) REFERENCES allomas(allomasSorszam)

ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY (vonalSorszam) REFERENCES vonal(vonalSorszam)

ON DELETE CASCADE ON UPDATE CASCADE

)

**2.Vonal tábla és Járat tábla közötti kapcsolat létrehozása**

**Új oszlop felvitele a járat táblába:**

ALTER TABLE jarat ADD vonalSorszam int

**Majd külső kulcs megadása a járat táblához:**

ALTER TABLE jarat

ADD FOREIGN KEY (vonalSorszam)

REFERENCES jarat(jaratSorszam)

ON UPDATE CASCADE

ON DELETE SET NULL

**3.Járat és jármű táblák közötti kapcsolat létrehozása**

**Új oszlop felvitele a járat táblába:**

ALTER TABLE jarat ADD jarmuSorszam int

**Majd külső kulcs megadása a járat táblához:**

ALTER TABLE jarat

ADD FOREIGN KEY (jarmuSorszam)

REFERENCES jarmu(jarmuSorszam)

ON UPDATE CASCADE

ON DELETE SET NULL

**4.Járat és naptípus táblák közötti kapcsolat létrehozása**

**Új oszlop felvitele a járat táblába:**

ALTER TABLE jarat ADD napTipusSorszam int

**Majd külső kulcs megadása a járat táblához:**

ALTER TABLE jarat

ADD FOREIGN KEY (napTipusSorszam)

REFERENCES naptipus(napTipusSorszam)

ON UPDATE CASCADE

ON DELETE SET NULL

**Lekérdezések**

**Adatmanipuláció**

**1.Új akadálymentesített járatok (alacsony padlós es legalább egy rokkant hely**

**elérhető) felvétele, amelyek a Klinikák nevű állomásról indulnak.**

08:28-kor és 08:38-kor induló, alacsonypadlós, 5 rokkant hellyel rendelkező, Klinikákról induló 8-as troli, valamint 10:25-kor és 10:35-kor induló, alacsonypadlós, 5 rokkant hellyel rendelkező, Klinikákról induló 10-es troli felvétele a járat táblába:

**Kód:**

**INSERT** **INTO** jarat

(vonalSorszam,jarmuSorszam,indulasiIdo,napTipusSorszam,rokkantHelyekSzama,

uloHelyekSzama,csuklos,alacsonyPadlos,kerekparSzallitasiLehetoseg)

**VALUES**

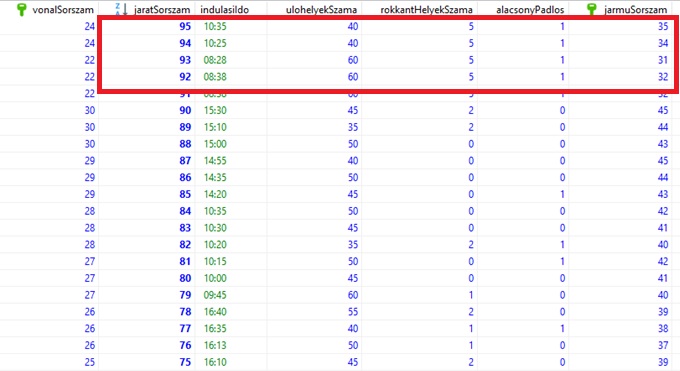
(22,32,'08:38',1,5,40,0,1,0), (22,31,'08:28',1,5,40,0,1,0),

(24,34,'10:25',1,5,40,0,1,0), (24,35,'10:35',1,5,40,0,1,0)

**Eredmény**:

*/\* Érintett sorok: 4 Talált sorok: 0 Figyelmeztetés: 0 Időtartam 1 lekérdezés: 0,094*

*mp. \*/*

**

**2.Reggeli csúcsforgalomban közlekedő járatok ülőhelyeinek kibővítése.**

Bővítsük ki a reggel 08:00 és 09:00 között induló, 60-nál kevesebb ülőhellyel rendelkező járatok ülőhelyeit 60-ra.

**Kód:**

**UPDATE** jarat **SET** ulohelyekSzama=60

**WHERE**

ulohelyekSzama<60 **AND**

indulasiIdo **LIKE** '%8:%'**OR** indulasiIdo **LIKE** '%9:%'

**AND** napTipusSorszam **IN** (

**SELECT** napTipusSorszam

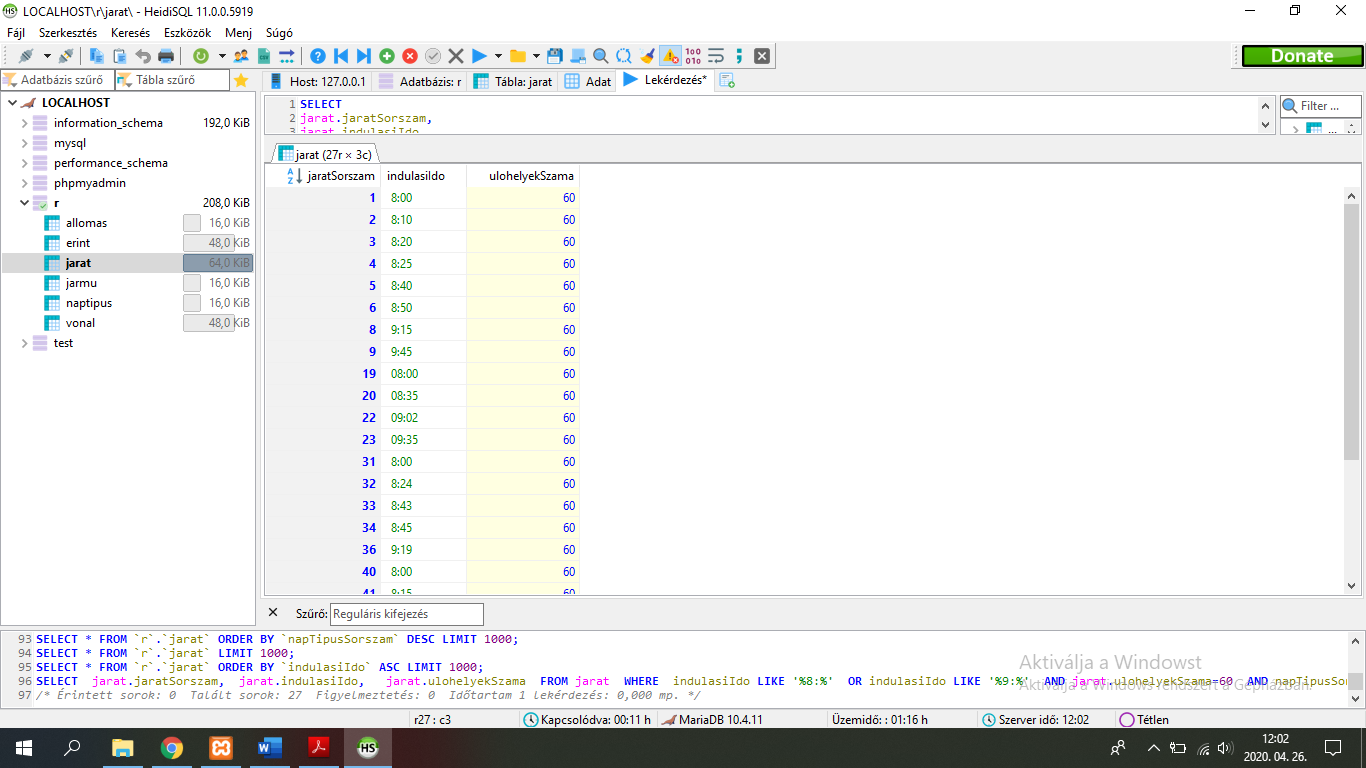
**FROM** naptipus

**WHERE** leiras = 'munkanap'

)

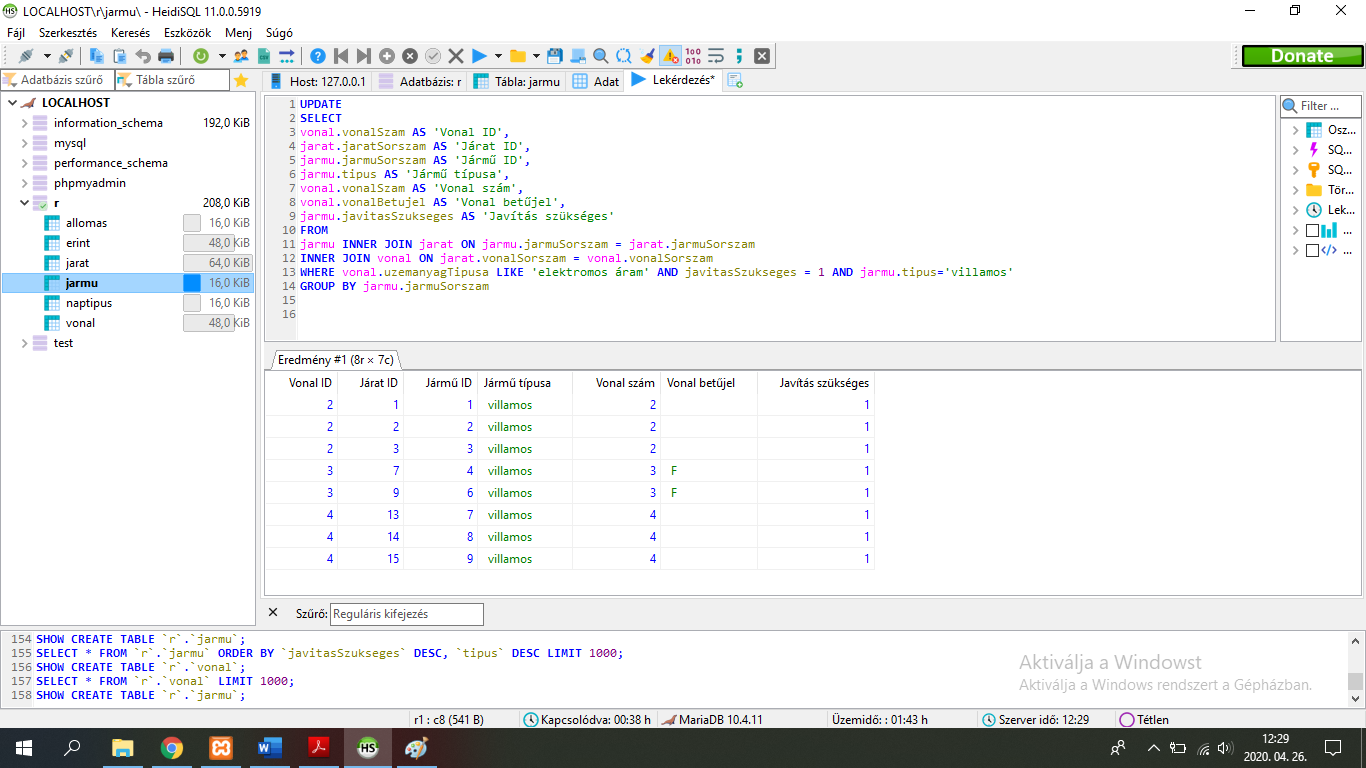
**Eredmény**:

*/\* Érintett sorok: 18 Talált sorok: 0 Figyelmeztetés: 0 Időtartam 1 lekérdezés: 0,079 mp. \*/*



**3.A javítandó villamosok típusát módosítjuk pótlóbusznak, majd átállítjuk a javítás szükségességét 0-ra, mivel a pótlóbusz nem igényel javítást.**

Először lekérdezzük a javítandó villamosok rekordjait:



Majd elvégezzük a módosítást:

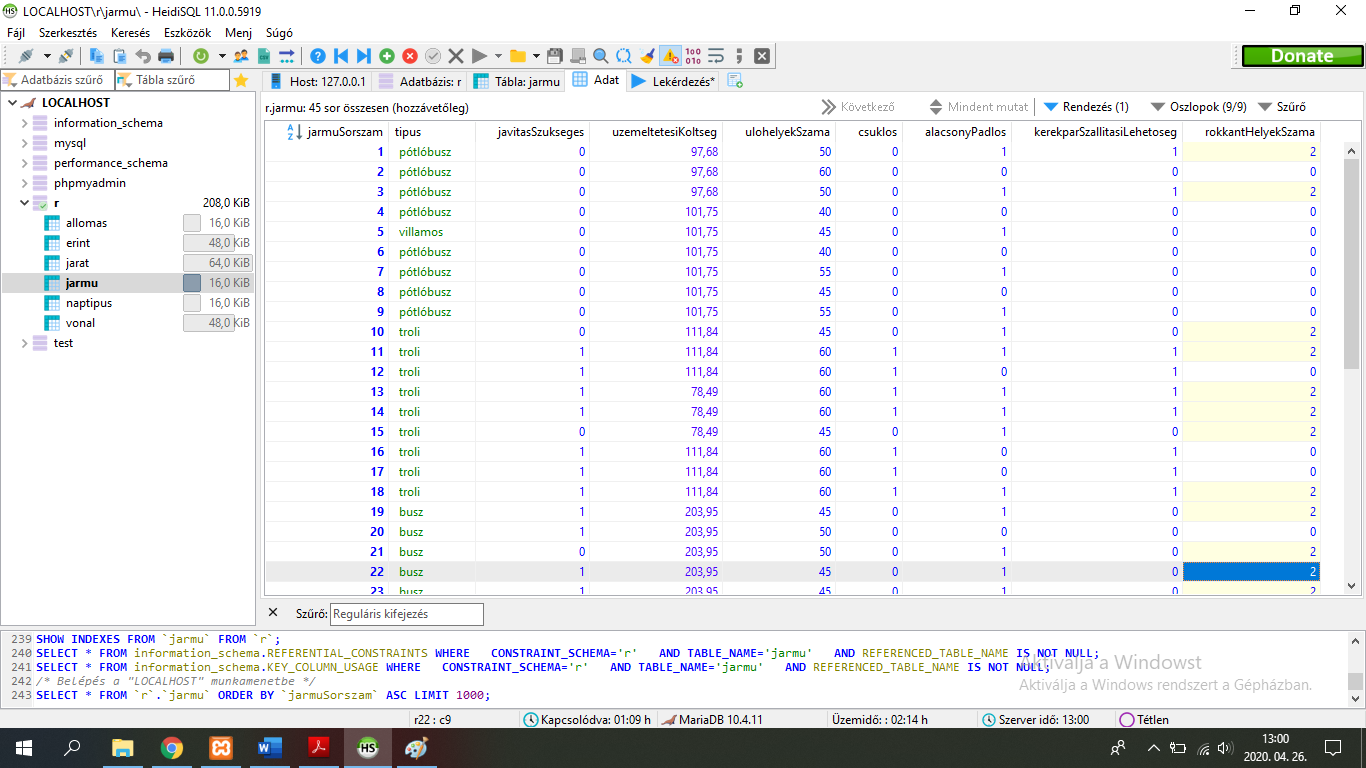
**Kód:**

**UPDATE** jarmu **SET** tipus='pótlóbusz', javitasSzukseges=0

**WHERE** tipus **LIKE** 'villamos' **AND** javitasSzukseges = 1

**Eredmény**:

*/\* Érintett sorok: 8 Talált sorok: 0 Figyelmeztetés: 0 Időtartam 1 lekérdezés: 0,079 mp. \*/*



**4.Töröljük a jármű táblából a javítandó buszokat**

**Kód:**

**DELETE** **FROM** jarmu

**WHERE** javitasSzukseges=1 **AND** tipus **LIKE** 'busz'

**Eredmény**:

*/\* Érintett sorok: 15 Talált sorok: 0 Figyelmeztetés: 0 Időtartam 1 lekérdezés: 0,093 mp. \*/*