Adatbázis rendszerek

9. Gyakorlat

2025. 04. 16

**Készítette:**

Rácz László

Szak: 1.PTI

CI880V

**Sárospatak, 2025**

1. Task

Téma: SQL nyelv, DDL. DML, DQL utasítások

A feladat megvalósítása: MySQL (*MarioDB vagy mysql Workbench 8.0 CE*) és APEX.

Oracle Apex felület. URL: https://apex.oracle.com/en/

Mindkét felületen készítse el a feladatokat!

Javaslat: az utasításokat egy TXT fájlba írja le, majd onnan másolja be Mysql, ill. Oracle Apex felületre.

a.) Lépjen be az adatbázisba: NEPTUNKOD

CREATE DATABASE NEPTUNKOD;

USE NEPTUNKOD;

b.) Kérdezze le az auto, tulajdonos tábla szerkezetét!

Tulajdonos



MariaDB

CREATE TABLE Tulajdonos (

Tkód INT(3) NOT NULL PRIMARY KEY,

Név CHAR(15) NOT NULL,

Cím CHAR(10),

Telefon INT(10)

);

Apex

CREATE TABLE Tulajdonos (

Tkód NUMBER (3) NOT NULL primary key,

Név CHAR (15) NOT NULL,

Cím CHAR (10),

Telefon NUMBER (10)

);

Auto



Apex

CREATE TABLE Auto (

Rendszám CHAR (7) NOT NULL primary key,

Típus CHAR (25) NOT NULL,

Szín CHAR (15),

Ár NUMBER (11),

Kor NUMBER (11),

Tulaj NUMBER (11)

);

MariaDB

CREATE TABLE Auto (

Rendszám CHAR (7) NOT NULL,

Típus CHAR (25) NOT NULL,

Szín CHAR (15),

Ár INT (11),

Kor INT (11),

Tulaj INT (11)

);

d.) Ékezetes karakterek létrehozása!

ALTER DATABASE NEPTUNKOD CHARACTER SET utf8mb4 COLLATE utf8mb4\_hungarian\_ci;

Vagy

CREATE TABLE Példa (

Szöveg VARCHAR(100)

) CHARACTER SET utf8mb4 COLLATE utf8mb4\_hungarian\_ci;

e.) Kérdezze le az auto, majd a tulajdonos tábla *szerkezetét*!

DESCRIBE Auto;

DESCRIBE Tulajdonos;

f.) Milyen utasítással hozta létre a táblákat, kérdezze le!

SHOW CREATE TABLE Auto;

SHOW CREATE TABLE Tulajdonos;

2. Feladat

Végezzen az auto, ill tulajdonos tábla szerkezetén módosításokat.

1. Nevezze át az auto táblát jarmu, majd nevezze vissza!

Apex

ALTER TABLE Auto RENAME TO Jarmu;

ALTER TABLE Jarmu RENAME TO Auto;

MariaDB

ALTER TABLE Auto RENAME TO Jarmu;

ALTER TABLE Jarmu RENAME TO Auto;

2. Nevezze át a tulajdonos táblában a név mezőt name mezőre, majd nevezze vissza!

Apex:

ALTER TABLE Tulajdonos RENAME COLUMN Név TO name;

ALTER TABLE Tulajdonos RENAME COLUMN name TO Név;

MariaDB:

ALTER TABLE Tulajdonos CHANGE Név name CHAR(15) NOT NULL;

ALTER TABLE Tulajdonos CHANGE name Név CHAR(15) NOT NULL;

3. Adja hozzá az auto táblához a *Koros int(2) check (Koros>0)* mezőt az elejére!

Apex:

ALTER TABLE Auto ADD Koros NUMBER(2) CHECK (Koros >0);

MariaDB:

ALTER TABLE Auto ADD Koros INT(2) CHECK (Koros >0);

4. Törölje a Koros mezőt.

Apex:

ALTER TABLE Auto DROP COLUMN Koros;

MariaDB:

ALTER TABLE Auto DROP COLUMN Koros;

5. Adja hozzá a *Koros* mezőt a végére!

Apex:

ALTER TABLE Auto ADD Koros NUMBER(2) CHECK (Koros > 0);

MariaDB:

ALTER TABLE Auto ADD Koros INT(2) CHECK (Koros > 0);

6. Módosítsa a Tulajdonos táblába Telefon mező *adattípus/adatszélességet,* majd vissza!

Apex:

ALTER TABLE Tulajdonos MODIFY Telefon VARCHAR2(30);

MariaDB:

ALTER TABLE Tulajdonos MODIFY Telefon VARCHAR2(20);

7. Adja hozzá a Tulajdonos tábla Tkód mezőhöz egy UNIQUE integritási feltételt, majd törölje le!

ALTER TABLE Tulajdonos ADD CONSTRAINT uq\_Tkód UNIQUE (Tkód);

ALTER TABLE Tulajdonos DROP CONSTRAINT uq\_Tkód;

3. Feladat

Töltse fel az auto, ill. a tulajdonos táblákat adatokkal a minta alapján!

Auto

Apex:

AUTO

INSERT ALL

INTO AUTO VALUES ('FER-831', 'Opel Corsa', 'Piros', 18, 390, 101)

INTO AUTO VALUES ('GDF-525', 'Renault Twingo', 'Fekete', 16, 580, 101)

INTO AUTO VALUES ('HUB-936', 'Suzuki Swift', 'Fekete', 13, 200, 103)

INTO AUTO VALUES ('IXL-239', 'Suzuki Swift', 'Zöld', 11, 920, 103)

INTO AUTO VALUES ('JAH-425', 'Skoda Fabia', 'Piros', 13, 670, 102)

INTO AUTO VALUES ('JCD-443', 'Opel Astra', 'Fehér', 12, 920, 102)

INTO AUTO VALUES ('KAP-290', 'BMW 316', 'Fekete', 6, 3250, 107)

INTO AUTO VALUES ('KFT-204', 'Opel Astra', 'Szürke', 7, 1250, 106)

INTO AUTO VALUES ('MLM-211', 'Toyota Yaris', 'Fehér', 3, 1850, 105)

SELECT \* FROM dual;

TULAJDONOS

INSERT ALL

INTO TULAJDONOS VALUES (100, 'Kis János', 'Eger', '209555666')

INTO TULAJDONOS VALUES (101, 'Kis János', 'Eger', '209555666')

INTO TULAJDONOS VALUES (102, 'Kis Éva', 'Szerencs', '308764432')

INTO TULAJDONOS VALUES (103, 'Retek Ödön', 'Miskolc', '308764432')

INTO TULAJDONOS VALUES (104, 'Virág Zoltán', 'Nyék', '703355440')

INTO TULAJDONOS VALUES (105, 'Nagy Eszter', 'Ózd', '703355443')

INTO TULAJDONOS VALUES (106, 'Kovács Magor', 'Szerencs', '703855446')

INTO TULAJDONOS VALUES (107, 'Kovács Magor', 'Szerencs', '703855446')

INTO TULAJDONOS VALUES (109, 'Asztal Antal', 'Eger', '209955666')

SELECT \* FROM dual;

MariaDB

Auto

INSERT INTO Auto VALUES ('FER-831','Opel Corsa','Piros','18','390','101');

INSERT INTO Auto VALUES ('GDF-525','Renault Twingo','Fekete','16','280','103');

INSERT INTO Auto VALUES ('HUB-936','Suzuki Swift','Fekete','16','500','104');

INSERT INTO Auto VALUES ('IXL-239','Suzuki Swift','Zöld','15','450','105');

INSERT INTO Auto VALUES ('JAH-425','Skoda Fabia','Piros','13','620','102');

INSERT INTO Auto VALUES ('JCD-443','Opel Astra','Fehér','12','990','107');

INSERT INTO Auto VALUES ('KAP-290','BMW 316','Fekete','6','3250','102');

INSERT INTO Auto VALUES ('KFT-204','Opel Astra','Szürke','7','1250','106');

INSERT INTO Auto VALUES ('MLM-211','Toyota Yaris','Fehér','3','1850','105');

Tulajdonos2

INSERT INTO Tulajdonos VALUES ('100', 'Kis János', 'Eger', '209555666');

INSERT INTO Tulajdonos VALUES ('101', 'Kis János', 'Eger', '209555666');

INSERT INTO Tulajdonos VALUES ('102', 'Kis Éva', 'Szerencs', '308764432');

INSERT INTO Tulajdonos VALUES ('103', 'Retek Ödön', 'Miskolc', '308764432');

INSERT INTO Tulajdonos VALUES ('104', 'Virág Zoltán', 'Nyék', '703355440');

INSERT INTO Tulajdonos VALUES ('105', 'Nagy Eszter', 'Ózd', '703355440');

INSERT INTO Tulajdonos VALUES ('106', 'Kovács Magor', 'Szerencs', '703855444');

INSERT INTO Tulajdonos VALUES ('107', 'Kovács Magor', 'Szerencs', '703855445');

INSERT INTO Tulajdonos VALUES ('109', 'Asztal Antal', 'Eger', '209555666');



Tulajdonos



Kérdezze le a táblákat.

4. Feladat

A NEPTUNKÓD adatbázisba hozza létre a Dolgozó táblát, úgy, hogy Ne legyen a Kód mezőnek PK, ill. NN!

Apex

CREATE TABLE Dolgozó (

Kód NUMBER(4),

Név CHAR(20),

Város CHAR(20),

Beosztás CHAR (20),

Belépés date,

Fizetés NUMBER(7),

Osztály CHAR(15)

);

MariaDB

CREATE TABLE Dolgozó (

Kód INT(4),

Név CHAR(20),

Város CHAR(20),

Beosztás CHAR (20),

Belépés date,

Fizetés INT(7),

Osztály CHAR(15)

);



DML és DQL utasítások használata!

1. Adja hozzá a dolgozó tábla Kód mezőhöz PK-t!

ALTER TABLE Dolgozó

ADD PRIMARY KEY (Kód);

2. Adja hozzá a Név mezőhöz egy NN integritási feltételt!

ALTER TABLE Dolgozó

MODIFY Név VARCHAR(256) NOT NULL;

2a. Töltse fel a dolgozó táblát adatokkal (1173 rekord kivételével)!



INSERT INTO Dolgozó VALUES ('1016','Kis Elek','Eger','Programozó','2010-01-01','290000','Fejlesztés');

INSERT INTO Dolgozó VALUES ('1026','Kis Jenő','Eger','Tesztelő','2005-01-01','270000','Fejlesztés');

INSERT INTO Dolgozó VALUES ('1028','Kis Géza','Miskolc','Programozó','2011-03-15','250000','Fejlesztés');

INSERT INTO Dolgozó VALUES ('1039','Kő Benő','Budapest','Elnök','2005-01-01','650000','Irányítás');

INSERT INTO Dolgozó VALUES ('1099','Mezei Virág','Eger','Tesztelő','2006-03-01','280000','Fejlesztés');

INSERT INTO Dolgozó VALUES ('1113','Merész Teréz','Budapest','Elemző','2009-08-05','250000','Fejlesztés');

INSERT INTO Dolgozó VALUES ('1180','SP','DrBl','Oktató','2024-10-01','100000','Tanszék');

INSERT INTO Dolgozó VALUES ('1214','Sztár Pál','Budapest','Eladó','2012-01-08','270000','Terjesztés');

2b. Kérdezze le a dolgozó táblát!

SELECT \* FROM Dolgozó;

3. Töltse fel a saját magát a dolgozó táblába!

INSERT INTO Dolgozó VALUES (1014, 'Rácz László', 'Királyhelmec', 'Hallgató', CURDATE(), 1000000, 'Fejlesztő');

4. Kérdezze le a saját rekordját!

SELECT \* FROM Dolgozó

WHERE Név = 'Rácz László';

5a. Vigye fel „Hibás Ilona miskolci elemző adatait: Kód, Név, Belépés, a dátum a mai legyen.

INSERT INTO Dolgozó VALUES (1073, 'Hibás Ilona', 'Miskolc', 'Elemző', CURDATE(), 3000000, 'Fejlesztés');

5b. Vigye fel Hibás Ilona a *fejlesztésre* került, *300000 fizetéssel* – adatokat!

UPDATE Dolgozó

SET Beosztás = 'Fejlesztő', Fizetés = 300000

WHERE Név = 'Hibás Ilona';

6. Kérdezze le az egri emberek nevét!

SELECT Név

FROM Dolgozó

WHERE Város = 'Eger';

7. Kérdezze le a nem egri emberek nevét, városát!

SELECT Név, Város

FROM Dolgozó

WHERE Város <> 'Eger';

8. Kérdezze le a nem budapesti emberek nevét, városát!

SELECT Név, Város

FROM Dolgozó

WHERE Város <> 'Budapest';

9. Kérdezze le azokat, akik a fejlesztés osztályon dolgozók nevét, ABC sorrendben!

SELECT Név

FROM Dolgozó

WHERE Osztály = 'Fejlesztés'

ORDER BY Név ASC;

10. Kérdezze le azoknak a neveket, fizetéseket, fizetés szerinti csökkenő sorrendben!

SELECT Név, Fizetés

FROM Dolgozó

ORDER BY Fizetés DESC;

10a. Kérdezze le az M betűvel kezdődő neveket!

SELECT Név

FROM Dolgozó

WHERE Név LIKE 'M%';

11. Kérdezze le az a és a z betűre végződő neveket!

SELECT Név

FROM Dolgozó

WHERE Név LIKE '%a' OR Név LIKE '%z';

12. Kérdezze le azoknak a kódját, nevét, akiknek a kódjában van 4 es számjegy!

SELECT Kód, Név

FROM Dolgozó

WHERE Kód LIKE '%4%';

13. Kérdezze le azokat, akinek a kódját, nevét, akiknek a kódjában van 1-es, 2-es, és 4-es számjegy is!

SELECT Kód, Név

FROM Dolgozó

WHERE Kód LIKE '%1%' AND Kód LIKE '%2%' AND Kód LIKE '%4%';

14. Kérdezze le azoknak a nevét, kódját, akik kódja 1030 és 1130 közötti!

SELECT Név, Kód

FROM Dolgozó

WHERE Kód BETWEEN 1030 AND 1130;

15. Kérdezze le azokat, akinek nincs megadva a városa!

SELECT \*

FROM Dolgozó

WHERE Város IS NULL;

16. Kérdezze le azoknak a nevét, belépési dátumát, akik 2024-ben léptek be!

SELECT Név, Belépés

FROM Dolgozó

WHERE YEAR(Belépés) = 2024;

17. Kérdezze le a 2012-01-08 és 2024-10-01 között belépett dolgozók nevét, belépési dátumát!

SELECT Név, Belépés

FROM Dolgozó

WHERE Belépés BETWEEN '2012-01-08' AND '2024-10-01';

18. Kérdezze le az osztályokat!

SELECT Osztály

FROM Dolgozó;

19. Kérdezze le *osztályokat* úgy, hogy ne ismétlődjenek!

SELECT DISTINCT Osztály

FROM Dolgozó;

20. Kérdezze le a *fejlesztésen* dolgozó miskolciak nevét!

SELECT Név

FROM Dolgozó

WHERE Osztály = 'Fejlesztés' AND Város = 'Miskolc';

21. Kérdezze le a fejlesztésen és a terjesztésen dolgozó nem budapestiek nevét!

SELECT Név

FROM Dolgozó

WHERE Osztály IN ('Fejlesztés', 'Terjesztés')

AND Város <> 'Budapest';

4. Feladat

a) Hozzon létre egy adatbázist, neve: PTINeptunkod!

CREATE DATABASE PTICI880V;

b.) Törölje a PTINeptunkod adatbázis!

DROP DATABASE PTICI880V;

4. Feladat

a.) Hozzon létre egy adatbázist, neve: OktatoHallgato, majd lépjen be az adatbázisba!

CREATE DATABASE OktatoHallgato;

USE OktatoHallgato;

b.) A 6. Practice elkészített relációs séma alapján hozza létre a táblákat.

Tervezze meg a mezők *adattípusait*, *értékeit*, ill. *integritási feltételeit*!

OKTATÓ [neptunkód PK, név, tanszék]

TÁRGY [kód PK, megnevezés, kredit, oktató FK]

HALLGAT [hallgató FK, tárgy FK, félév, vizsgajegy]

HALLGATÓ [neptunkód PK, név, szül.d, tankör, irsz, város, utca, hsz]

CREATE TABLE OKTATO (

neptunkod CHAR(6) PRIMARY KEY,

nev VARCHAR(100) NOT NULL,

tanszek VARCHAR(100) NOT NULL

);

CREATE TABLE TARGY (

kod CHAR(6) PRIMARY KEY,

megnevezes VARCHAR(100) NOT NULL,

kredit INT NOT NULL CHECK (kredit > 0),

oktato CHAR(6),

FOREIGN KEY (oktato) REFERENCES OKTATO(neptunkod)

);

CREATE TABLE HALLGATO (

neptunkod CHAR(6) PRIMARY KEY,

nev VARCHAR(100) NOT NULL,

szul\_d DATE NOT NULL,

tankor VARCHAR(10),

irsz CHAR(4),

varos VARCHAR(50),

utca VARCHAR(100),

hsz VARCHAR(10)

);

CREATE TABLE HALLGAT (

hallgato CHAR(6),

targy CHAR(6),

felev VARCHAR(10),

vizsgajegy INT CHECK (vizsgajegy BETWEEN 1 AND 5),

PRIMARY KEY (hallgato, targy),

FOREIGN KEY (hallgato) REFERENCES HALLGATO(neptunkod),

FOREIGN KEY (targy) REFERENCES TARGY(kod)

);