

Jegyzőkönyv
Adatbázis rendszerek I.
Féléves feladat

Készítette: Szabó Dániel
Neptunkód: XUXEJO
Gyakorlat ideje: Csütörtök 8-10
Gyakorlatvezető: Bednarik László

Feladat leírása:

Az általam megtervezett és létrehozott adatbázis Rally Versenyekről fog adatokat tárolni. Tárolja az adott Szakasz adatait és annak versenyzőit. Számon tartja a Versenyzők nemzetiségét. Szakaszokon lement verseny Autókat, pilótáik nemzetiségét. Tárolja hogy az adott Szakasz melyik Országban van. Ezek alapján könnyen le lehet kérdezni az adott Rallynak az alapvető információit (Autók, Versenyzők, Nemzetiségük, stb.). Az adott Szakaszon milyen hőmérsékletek uralkodnak, milyen hosszú és mi a versenyszakasz felülete, legyen az murva, aszfalt, durva kavicsos murva, hó, stb. Versenyzők melyik autót vezetik, milyen időket mentek a szakaszokon és még sok más.

Az ER modell egyedei és tulajdonságai

Egyedek: Szakaszz, Versenyző, Nemzetiség, Autó, Ország

Szakasz:

Név – primary key

Hőmérséklet

Hossz

Felület

Versenyző:

VKód – primary key

Név

Nemzetiség:

Nemzetkód - primary key

Név

Autó:

AutóNév – primary key

Versenben

Ország:

OrszágID – primary key

Egyedek közötti kapcsolat:

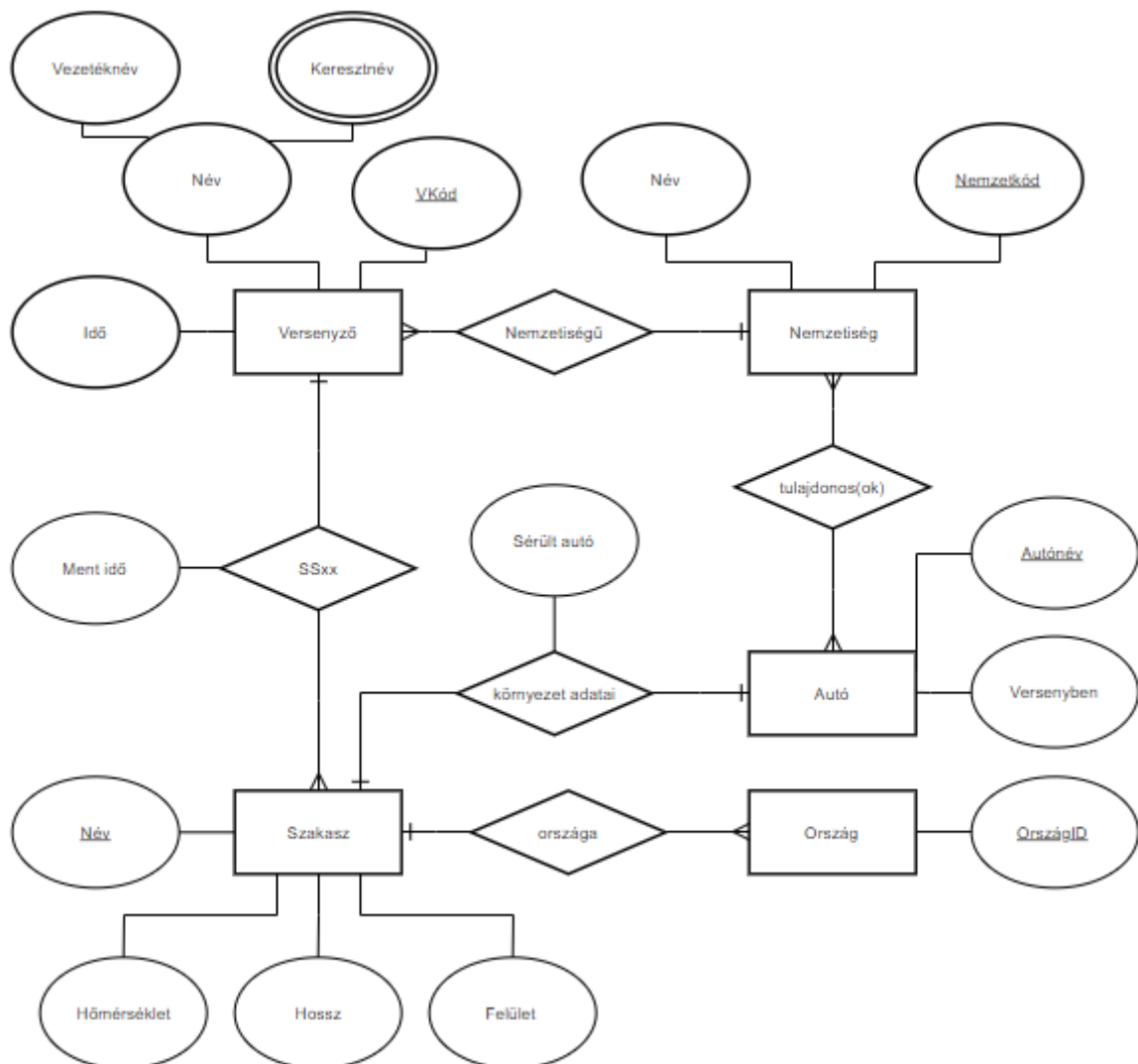
A **Szakasz** és a **Versenyző** között 1:N típusú kapcsolat van, mivel egy versenyző több szakaszon is részt vehet.

A **Versenyző** és a **Nemzetiség** között 1:N típusú kapcsolat van, mivel egy nemzethez tartozhat több versenyző, de egy versenyzőnek nem lehet több nemzetisége.

Az **Autó** és a **Nemzetiség** között N:M típusú kapcsolat van, mivel egy autója több nemzetnek is lehet és egy nemzetnek lehet több autója is.

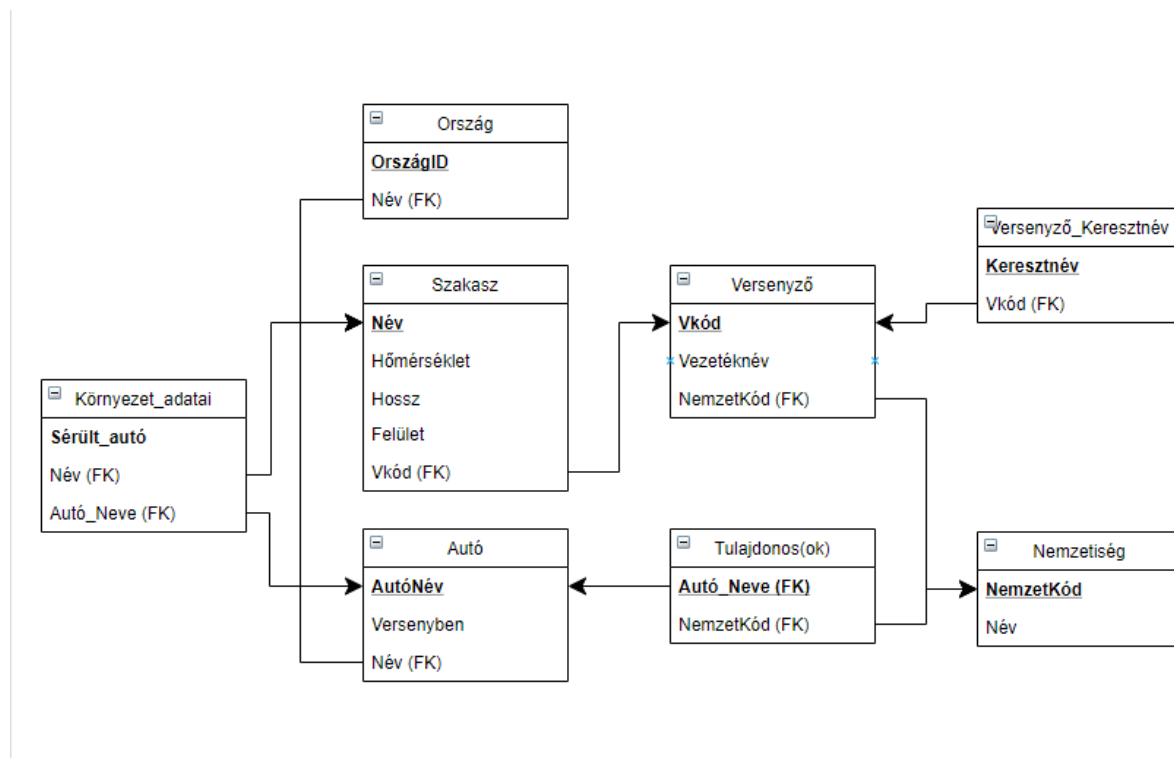
A **Szakasz** és az **Autó** között 1:1 típusú kapcsolat van.

Az **Ország** és a **Szakasz** között 1:N típusú kapcsolat van, mivel egy országnak lehet több szakasza, de egy adott szakasz csak egy országban lehet.



Relációs modell:

- Az **Ország** egyedből tábla lesz. Itt csak az **OrszágID** lesz, mint primary key tulajdonság, illetve a **Név** foreign key mutat az **Szakasz**ra.
- A **Szakasz** egyedből tábla lesz. Az elsődleges kulcsa a **Név** lesz. Tulajdonságai lesznek a **Hőmérséklet**, a **Hossz** és a **Felület**. Foreign key lesz a **VKód**, ami a **Versenyző** táblára mutat.
- A **Versenyző** is tábla lesz. Az elsődleges kulcs a **VKód**; tulajdonságai a **Vezetéknév** és a **Keresztnév**. Foreign key a **NemzetKód** a **Nemzetiség** táblára mutatóan.
- A Tulajdonos(ok) kapcsolatból kapcsolótábla lesz két foreign key-jel. Ezek az **Autó_Neve** és a **NemzetKód**, mivel a **Nemzetiség** táblát kapcsolja össze az **Autó** táblával.
- Az **Autó** egyedből tábla lesz, primary key az **AutóNév**; tulajdonság a **Versenyben**, hogy nem-e esett ki az autó az egyik szakaszon és foreign key a **Név** a **Szakasz** táblával összefüggésben.
- Mivel a **Keresztnév** egy többértékű tulajdonság, ezért új tábla jön belőle létre, ahol a **Vkód** lesz a foreign key.
- A **Környezet_adatai** kapcsolatból tábla lesz két foreign key-jel, **Név** a **Szakaszból** és **Autó_Neve** az **Autóból**, a kapcsolat két tulajdonsága pedig a tábla részét fogja képezni.



Relációs séma:

Ország [OrszágID, Név]
Szakas [Név, Hőmérséklet, Hossz, Felület, Vkód]
Autó [AutóNév, Versenyben, Név]
Környezet_adatok [Sérült autó_Név, Autó_Neve]
Versenyző [Vkód, Vezetéknév, NemzetKód]
Tulajdonos [Autó_Neve, NemzetKód]
Versenyző_Keresztnév [Keresztnév, Vkód]
Nemzetiség [NemzetKód, Név]

SQL Parancsok:

Táblák létrehozása:

```
CREATE TABLE Nemzetiség (
    NemzetKód INT NOT NULL,
    Név VARCHAR(50) NOT NULL,
    PRIMARY KEY (nemzetKód)
);

CREATE TABLE Versenyző(
    VKód INT NOT NULL,
    Vezetéknév VARCHAR(50) NOT NULL,
    NemzetKód INT NOT NULL,
    PRIMARY KEY (VKód),
    FOREIGN KEY (NemzetKód) REFERENCES Nemzetiség(NemzetKód)
);

CREATE TABLE Versenyző_Keresztnév(
    Keresztnév VARCHAR(50) NOT NULL,
    VKód INT NOT NULL,
    PRIMARY KEY (Keresztnév),
    FOREIGN KEY (VKód) REFERENCES Versenyző(VKód)
);

CREATE TABLE Szakasza (
    Név VARCHAR(50) NOT NULL,
    Hőmérséklet INT NOT NULL,
    Hossz INT NOT NULL,
    Felület VARCHAR(50) NOT NULL,
    VKód INT NOT NULL,
    PRIMARY KEY (Név),
    FOREIGN KEY (VKód) REFERENCES Versenyző(VKód)
);

CREATE TABLE Autó(
    AutóNév VARCHAR(50) NOT NULL,
    Versenyben VARCHAR(3) NOT NULL,
    Név VARCHAR(50) NOT NULL,
    PRIMARY KEY (AutóNév),
    FOREIGN KEY (Név) REFERENCES Szakasza(Név)
);

CREATE TABLE Ország(
    OrszágID INT NOT NULL,
    Név VARCHAR(50) NOT NULL,
    PRIMARY KEY (OrszágID),
    FOREIGN KEY (Név) REFERENCES Szakasza(Név)
);
```

```
CREATE TABLE Tulajdonos(  
    AutóNév VARCHAR(50) NOT NULL,  
    NemzetKód INT NOT NULL,  
    PRIMARY KEY (AutóNév, NemzetKód),  
    FOREIGN KEY (AutóNév) REFERENCES Autó(AutóNév),  
    FOREIGN KEY (NemzetKód) REFERENCES Nemzetiség(NemzetKód)  
);  
  
CREATE TABLE Környezet_adatai (  
    Sérült_autó VARCHAR(15) NOT NULL,  
    Név VARCHAR(50) NOT NULL,  
    Autó_Neve VARCHAR(50) NOT NULL,  
    FOREIGN KEY (Név) REFERENCES Szakasz(Név),  
    FOREIGN KEY (Autó_Neve) REFERENCES Autó(AutóNév)  
);
```


Táblák feltöltése:

```
INSERT INTO Nemzetiség (NemzetKód, Név) VALUES ('1', 'cseh');
INSERT INTO Nemzetiség (NemzetKód, Név) VALUES ('2', 'francia');
INSERT INTO Nemzetiség (NemzetKód, Név) VALUES ('3', 'német');
INSERT INTO Nemzetiség (NemzetKód, Név) VALUES ('4', 'olasz');
INSERT INTO Nemzetiség (NemzetKód, Név) VALUES ('5', 'magyar');
```

```
INSERT INTO Versenyző (VKód, Vezetéknév, NemzetKód) VALUES ('1', 'Tanak', '5');
INSERT INTO Versenyző (VKód, Vezetéknév, NemzetKód) VALUES ('2', 'Ogier', '3');
INSERT INTO Versenyző (VKód, Vezetéknév, NemzetKód) VALUES ('3', 'Rovenpera', '1');
INSERT INTO Versenyző (VKód, Vezetéknév, NemzetKód) VALUES ('4', 'Neuville', '2');
INSERT INTO Versenyző (VKód, Vezetéknév, NemzetKód) VALUES ('5', 'Perez', '4');
```

```
INSERT INTO Versenyző_Keresztnév (VKód, Keresztnév) VALUES ('1', 'Ott');
INSERT INTO Versenyző_Keresztnév (VKód, Keresztnév) VALUES ('2', 'Sebastian');
INSERT INTO Versenyző_Keresztnév (VKód, Keresztnév) VALUES ('3', 'Kella');
INSERT INTO Versenyző_Keresztnév (VKód, Keresztnév) VALUES ('4', 'Thierry');
INSERT INTO Versenyző_Keresztnév (VKód, Keresztnév) VALUES ('5', 'Sergio');
```

```
INSERT INTO Szakas (Név, Hőmérséklet, Hossz, Felület, VKód) VALUES ('magyar_szakas', '12', '25', 'murva', '1');
INSERT INTO Szakas (Név, Hőmérséklet, Hossz, Felület, VKód) VALUES ('olasz_szakas', '9', '30', 'murva', '3');
INSERT INTO Szakas (Név, Hőmérséklet, Hossz, Felület, VKód) VALUES ('francia_szakas', '15', '12', 'aszfalt', '5');
INSERT INTO Szakas (Név, Hőmérséklet, Hossz, Felület, VKód) VALUES ('nemet_szakas', '4', '25', 'aszfalt-murva', '4');
INSERT INTO Szakas (Név, Hőmérséklet, Hossz, Felület, VKód) VALUES ('orosz_szakas', '-1', '27', 'aszfalt', '2');
```

```
INSERT INTO Ország (OrszágID, Név) VALUES ('1', 'magyar_szakas');
INSERT INTO Ország (OrszágID, Név) VALUES ('2', 'olasz_szakas');
INSERT INTO Ország (OrszágID, Név) VALUES ('3', 'francia_szakas');
INSERT INTO Ország (OrszágID, Név) VALUES ('4', 'nemet_szakas');
INSERT INTO Ország (OrszágID, Név) VALUES ('5', 'orosz_szakas');
```

```
INSERT INTO Autó (AutóNév, Versenyben, Név) VALUES ('skoda_fabia_r5',  
'nem', 'orosz_szakaszi');  
INSERT INTO Autó (AutóNév, Versenyben, Név) VALUES ('bmw_m3_e30', 'igen',  
'magyar_szakaszi');  
INSERT INTO Autó (AutóNév, Versenyben, Név) VALUES ('ford_fiesta_r5',  
'nem', 'olasz_szakaszi');  
INSERT INTO Autó (AutóNév, Versenyben, Név) VALUES ('ford_focus_wrc',  
'nem', 'francia_szakaszi');  
INSERT INTO Autó (AutóNév, Versenyben, Név) VALUES ('wv_polo_r5', 'igen',  
'nemet_szakaszi');
```

```
INSERT INTO Környezet_adatai (Sérült_autó, Név, Autó_Neve) VALUES ('nem',  
'francia_szakaszi', 'ford_focus_wrc');  
INSERT INTO Környezet_adatai (Sérült_autó, Név, Autó_Neve) VALUES ('igen',  
'nemet_szakaszi', 'wv_polo_r5');  
INSERT INTO Környezet_adatai (Sérült_autó, Név, Autó_Neve) VALUES ('igen',  
'magyar_szakaszi', 'bmw_m3_e30');  
INSERT INTO Környezet_adatai (Sérült_autó, Név, Autó_Neve) VALUES ('nem',  
'olasz_szakaszi', 'ford_fiesta_r5');  
INSERT INTO Környezet_adatai (Sérült_autó, Név, Autó_Neve) VALUES ('nem',  
'orosz_szakaszi', 'skoda_fabia_r5');
```

```
INSERT INTO tulajdonos (AutóNév, NemzetKód) VALUES ('bmw_m3_e30', '5');  
INSERT INTO tulajdonos (AutóNév, NemzetKód) VALUES ('ford_fiesta_r5', '2');  
INSERT INTO tulajdonos (AutóNév, NemzetKód) VALUES ('skoda_fabia_r5', '4');  
INSERT INTO tulajdonos (AutóNév, NemzetKód) VALUES ('ford_focus_wrc', '3');  
INSERT INTO tulajdonos (AutóNév, NemzetKód) VALUES ('wv_polo_r5', '1');
```

SQL lekérdezések:

Kilistázza az összes Nemzetiség nevét:

```
SELECT Név FROM Nemzetiség;
```

π Név Nemzetiség

Kilistázza az összes Szakasz nevét:

```
SELECT Név FROM Szakasz;
```

π Név Szakasz

Megmondja, hogy van-e olyan Autó, amit Versenyzőről neveztek el:

```
SELECT Autó.AutóNév, Versenyző.Vezetéknév FROM Autó JOIN Versenyző ON  
Autó.AutóNév = Versenyző.Vezetéknév WHERE AutóNév = Vezetéknév;
```

π Autó.AutóNév, Versenyző.Vezetéknév [σ AutóNév = Vezetéknév (Versenyző \bowtie Autó.AutóNév = Versenyző.Vezetéknév Autó)]

Megmondja, hogy adott Szakasz hány darab Országban ment le:

```
SELECT COUNT(Ország.OrszágID) FROM Ország JOIN Szakasz WHERE Ország.Név =  
Szakasz.Név
```

Γ COUNT(Ország.OrszágID) [σ Ország.Név = Szakasz.név (Ország \bowtie Ország.Név=Szakasz.Név Szakasz)]

Kiírja a Versenyzők teljes nevét:

```
SELECT Versenyző.Vezetéknév, Versenyző_Keresztnév.Keresztnév FROM Versenyző  
JOIN Versenyző_Keresztnév ON Versenyző.VKód = Versenyző_Keresztnév.VKód
```

π Versenyző.Vezetéknév, Versenyző_Keresztnév.Keresztnév (Versenyző \bowtie Versenyző.VKód = Versenyző_Keresztnév.VKód Versenyző_Keresztnév)

Kiírja a versenyben lévő Autók adatait:

```
SELECT * FROM Autó WHERE Versenyben = 'igen';
```

π^* (σ Versenyben = 'igen' (Autó))

Megszámolja adott Szakasznak hány Autója van:

```
SELECT Szakasz.Név, COUNT(Autó.AutóNév) AS Autók_Száma FROM Autó LEFT JOIN  
Szakasz ON Autó.AutóNév = Szakasz.Név GROUP BY Szakasz.Név;
```

Γ Szakasz.Név Szakasz.Név, COUNT(Autó.AutóNév) Autó [Autó \bowtie Autó.AutóNév = Szakasz.Név Szakasz]

Megszámolja adott Nemzetiségeknek hány Versenyzője van számontartva:

```
SELECT Nemzetiség.Név as Nemzetiség, COUNT(Versenyző.Vezetéknév) AS  
Versenyzők_Száma FROM Nemzetiség LEFT JOIN Versenyző ON  
Nemzetiség.NemzetKód = Versenyző.NemzetKód GROUP BY Nemzetiség.Név;
```

Γ Nemzetiség.Név Nemzetiség.Név, COUNT(Versenyző.Vezetéknév) Nemzetiség [Nemzetiség \bowtie Nemzetiség.NemzetKód = Versenyző.NemzetKód Versenyző]

Kilistázza azon szakaszok nevét, ahol Ford Fiesta R5-ök versenyeztek

```
SELECT Szakaszh.Név FROM Autó JOIN Szakaszh WHERE AutóNév= 'ford_fiesta_r5 ';
```

$\pi_{\text{AutóNév}} [\text{AutóNév} = \text{'ford_fiesta_r5'} \text{ (Autó} \bowtie_{\text{Autó.Név=Szakaszh.Név}} \text{Szakaszh)}]$

Murva szakaszok száma:

```
SELECT COUNT(Név) from Szakaszh WHERE Felület = 'murva ';
```

$\sigma_{\text{Felület} = \text{'murva'}} [\tau_{\text{count(Név)}}]$

Módosítások:

```
UPDATE Versenyző SET Vezetéknév='daniel' WHERE Nemzetkód=5;
```

```
ALTER TABLE Autó RENAME COLUMN Versenyben TO DNF;
```

```
DROP TABLE tulajdonos(ok);
```