

Adatbázis Rendszerek I.

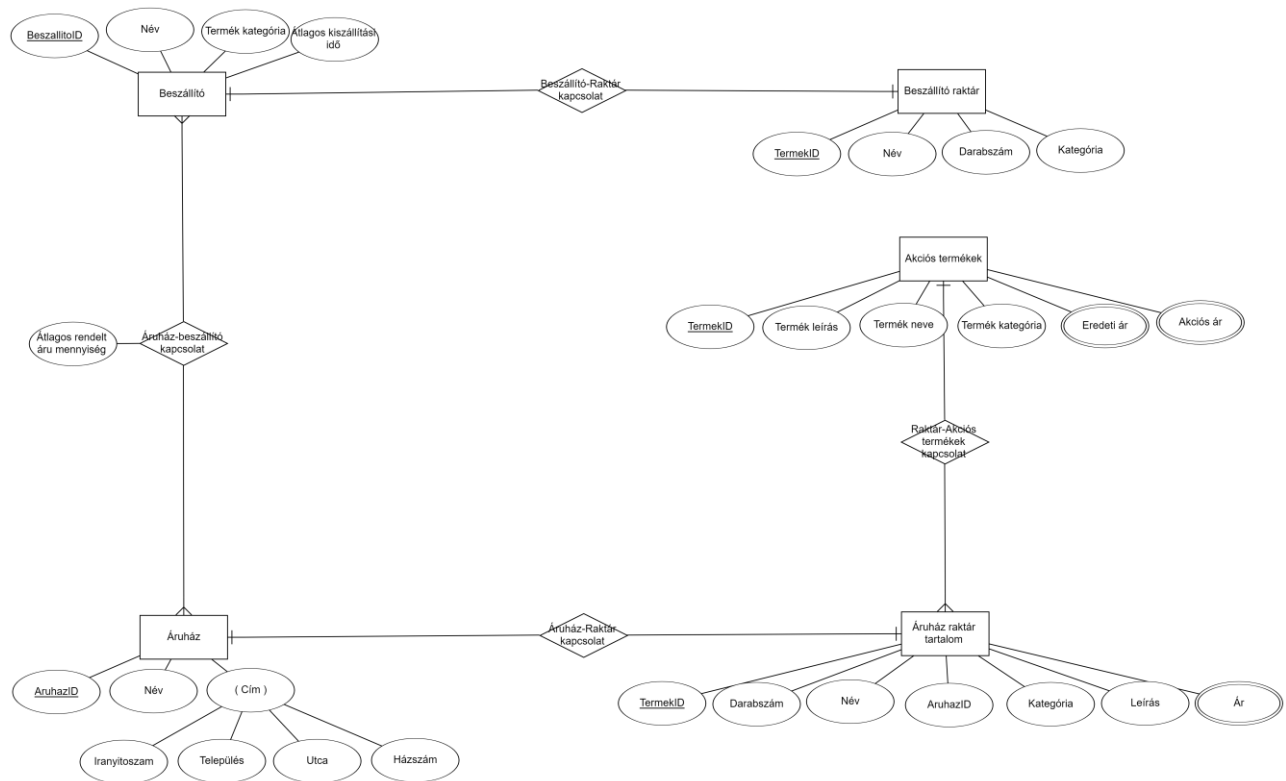
BSc

Féléves feladat jegyzőkönyv 2022-10-30

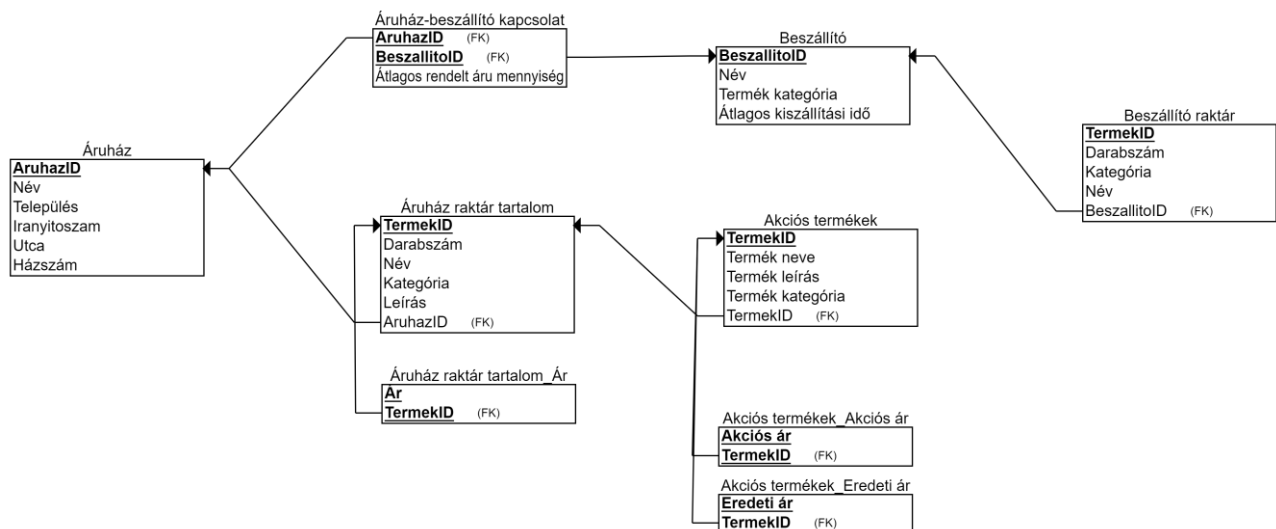
Készítette:
Stremler László
Programtervező informatikus
AQY08L

Miskolc,
2022

1.feladat:



2.feladat:



3.feladat:

Áruház[AruhazID, Név, Település, Iranyitoszam, Utca, Házsám, TermekID]

Áruház-beszállító kapcsolat[AruhazID, BeszallitoID, Átlagos rendelt áru mennyiség]

Beszállító[BeszallitoID, Név, Termék kategória, Átlagos kiszállítási idő]

Beszállító raktár[TermekID, Darabszám, Kategória, Név, BeszallitoID]

Áruház raktár tartalom[TermekID, Darabszám, Név, Kategória, Leírás, AruhazID]

Áruház raktár tartalom ár[Ár, TermekID]

Akciós termékek[TermekID, Termék neve, Termék leírás, Termék kategória, TermekID]

Akciós termékek_akciós_ár[Akciós ár, TermekID]

Akciós termékek_eredeti_ár[Eredeti ár, TermekID]

4.feladat:

```
CREATE TABLE Áruház
(
  AruhazID INT NOT NULL,
  Név VARCHAR(45) NOT NULL,
  Település VARCHAR(45) NOT NULL,
  Iranyitoszam INT NOT NULL,
  Utca VARCHAR(45) NOT NULL,
  Házszám INT NOT NULL,
  PRIMARY KEY (AruhazID)
);

CREATE TABLE Áruház_raktár_tartalom
(
  Darabszám INT NOT NULL,
  Név VARCHAR(45) NOT NULL,
  Kategória VARCHAR(45) NOT NULL,
  Leírás VARCHAR(145) NOT NULL,
  TermekID INT NOT NULL,
  AruhazID INT NOT NULL,
  PRIMARY KEY (TermekID),
  FOREIGN KEY (AruhazID) REFERENCES Áruház(AruhazID)
);

CREATE TABLE Akciós_termékek
(
  TermekID INT NOT NULL,
  Termék_neve VARCHAR(45) NOT NULL,
  Termék_leírás VARCHAR(145) NOT NULL,
  Termék_kategória VARCHAR(45) NOT NULL,
  PRIMARY KEY (TermekID),
  FOREIGN KEY (TermekID) REFERENCES Áruház_raktár_tartalom(TermekID)
);

CREATE TABLE Áruház_raktár_tartalom_Ár
(
  Ár INT NOT NULL,
  TermekID INT NOT NULL,
  PRIMARY KEY (Ár, TermekID),
  FOREIGN KEY (TermekID) REFERENCES Áruház_raktár_tartalom(TermekID)
);

CREATE TABLE Akciós_termékek_Eredeti_ár
(
  Eredeti_ár INT NOT NULL,
  TermekID INT NOT NULL,
  PRIMARY KEY (Eredeti_ár, TermekID),
  FOREIGN KEY (TermekID) REFERENCES Akciós_termékek(TermekID)
);

CREATE TABLE Akciós_termékek_Akciós_ár
(
  Akciós_ár INT NOT NULL,
  TermekID INT NOT NULL,
  PRIMARY KEY (Akciós_ár, TermekID),
  FOREIGN KEY (TermekID) REFERENCES Akciós_termékek(TermekID)
);

CREATE TABLE Beszállító
(
  BeszallitoID INT NOT NULL,
  Név VARCHAR(45) NOT NULL,
  Termék_kategória VARCHAR(145) NOT NULL,
  Átlagos_kiszállítási_idő INT NOT NULL,
  PRIMARY KEY (BeszallitoID)
);

CREATE TABLE Beszállító_raktár
(
  TermekID INT NOT NULL,
  Darabszám INT NOT NULL,
  Kategória VARCHAR(145) NOT NULL,
  Név VARCHAR(45) NOT NULL,
  BeszallitoID INT NOT NULL,
  PRIMARY KEY (TermekID),
  FOREIGN KEY (BeszallitoID) REFERENCES Beszállító(BeszallitoID)
);

CREATE TABLE Áruház_beszállító_kapcsolat
(
  Átlagos_rendelt_áru_mennyiség INT NOT NULL,
  AruhazID INT NOT NULL,
  BeszallitoID INT NOT NULL,
  PRIMARY KEY (AruhazID, BeszallitoID),
  FOREIGN KEY (AruhazID) REFERENCES Áruház(AruhazID),
  FOREIGN KEY (BeszallitoID) REFERENCES Beszállító(BeszallitoID)
);
```

5.feladat:

```
INSERT INTO `áruház` (`AruhazID`, `Név`, `Település`, `Iranyitoszam`, `Utca`, `Házszám`) VALUES ('1', 'Coop', 'Mucsomröcsöge', '1234', 'Bánat utca', '21'), ('2', 'Spar', 'Bekecs', '6789', 'Károly utca', '98'), ('3', 'Penny Market', 'Andalujza', '9183', 'Petőfi utca', '56'), ('4', 'Marika kisboltja', 'Sajószöged', '6756', 'Korda György utca', '69');
```

```
INSERT INTO `beszállító` (`BeszallitoID`, `Név`, `Termék_kategória`, `Átlagos_kiszállítási_idő`) VALUES ('1', 'Beszál-Litok', 'Gumiabroncs', '120'), ('2', 'Hűtő feltöltők', 'Fagyasztott áru', '45'), ('3', 'Kertészkedés', 'Kertészet', '60'), ('4', 'Rip-Rop', 'Rágcsa', '70');
```

```
INSERT INTO `áruház_beszállító_kapcsolat` (`AruhazID`, `BeszallitoID`) VALUES ('1', '2'), ('1', '4'), ('4', '4'), ('4', '2'), ('3', '2'), ('3', '1'), ('3', '3'), ('3', '4'), ('2', '4'), ('2', '3'), ('2', '2'), ('2', '1');
```

```
INSERT INTO `áruház_raktár_tartalom` (`Darabszám`, `Név`, `Kategória`, `Leírás`, `TermekID`, `AruhazID`) VALUES ('20', 'Mogyoróvaj', 'Rágcsa', 'Finomság :)', '1', '2'), ('10', 'Kecsketej', 'Fagyasztott áru', 'Finom, frissen fejt kecsketejcsike', '2', '4'), ('200', 'Kinder MaxiKing', 'Rágcsa', 'A létező legjobb édesség', '3', '3'), ('11', 'Kerekerdő Gumiabroncs', 'Jármű', 'Jó kis abroncs', '4', '2'), ('5', 'Tuja', 'Kertészet', 'Szimpla tuja', '5', '2'), ('6', 'Gyorsfagyasztott pizza', 'Fagyasztott áru', 'Egy gyorsan megfagyasztott pizza', '6', '1');
```

```
INSERT INTO `áruház_raktár_tartalom_ár` (`Ár`, `TermekID`) VALUES ('200', '3'), ('500', '1'), ('300', '2'), ('15000', '4'), ('600', '5'), ('1200', '6');
```

```
INSERT INTO `akciós_termékek` (`TermekID`, `Termék_neve`, `Termék_leírás`, `Termék_kategória`) VALUES ('1', 'Aksiós mogyoróvaj', 'Finom, krémes és még akciós is!', 'Rágcsa'), ('4', 'Kerekerdő Gumiabroncs', 'Jól megy és még akciós is!', 'Jármű');
```

```
INSERT INTO `akciós_termékek_eredeti_ár` (`Eredeti_ár`, `TermekID`) VALUES ('600', '1'), ('15000', '4');
```

```
INSERT INTO `akciós_termékek_akciós_ár` (`Akciós_ár`, `TermekID`) VALUES ('450', '1'), ('10000', '4');
```

```
INSERT INTO `beszállító_raktár` (`TermekID`, `Darabszám`, `Kategória`, `Név`, `BeszallitoID`) VALUES ('1', '34', 'Rágcsa', 'Mogyoróvaj', '4'), ('2', '10', 'Fagyasztott áru', 'Birkatej', '2'), ('6', '200', 'Fagyasztott áru', 'Gyorsfagyasztott ananászos pizza', '2');
```

6.feladat:

1. Kérdezzük le az összes olyan beszállítót, aki fagyasztott árut szállít!

SELECT * FROM beszállító WHERE Termék_kategória='Fagyasztott áru'

$\sigma_{\text{termék_kategória} = \text{"Fagyasztott áru"}} \text{ beszállító}$

2. Kérdezzük le az összes olyan áru nevét a beszállító raktárból, amit ha bármelyik áruház berendelne, ugyanolyan névvel kapna meg!

SELECT b.Név FROM beszállító_raktár b JOIN áruház_raktár_tartalom a ON b.TermekID=a.TermekID
WHERE b.Név LIKE a.Név

$\pi_{b.Név}$

$\sigma_{b.Név \text{ LIKE } a.Név}$

$(\rho_{b \text{ beszállító_raktár}} \bowtie \rho_{a \text{ áruház_raktár_tartalom}})$

$\rho_{a \text{ áruház_raktár_tartalom}}$

3. Kérdezzük le az összes olyan akciós termék nevét, amit be tud rendelni az áruház a beszállítótól készlethiány esetén! (nem számít hogy az áruház kapcsolatban áll a beszállítóval)

SELECT a.Termék_neve FROM akciós_termékek a INNER JOIN beszállító_raktár b ON
a.TermekID=b.TermekID

$\pi_{a.Termék_neve}$

$(\rho_{a \text{ akciós_termékek}} \bowtie \rho_{b \text{ áruház_raktár_tartalom}})$

$\rho_{b \text{ beszállító_raktár}}$

4. Kérdezzük le az összes olyan akciós termék nevét, amit be tud rendelni az áruház a beszállítótól készlethiány esetén! (számít hogy az áruház kapcsolatban áll a beszállítóval)

SELECT a.Termék_neve FROM akciós_termékek a INNER JOIN beszállító_raktár b ON
a.TermekID=b.TermekID LEFT JOIN áruház_raktár_tartalom arak ON a.TermekID=arak.TermekID JOIN
áruház ar ON arak.ArühazID=ar.ArühazID JOIN áruház_beszállító_kapcsolat abk ON
ar.ArühazID=abk.BeszallitoID GROUP BY a.Termék_neve;

$\pi_{a.Termék_neve}$

$(\rho_{a \text{ akciós_termékek}} \bowtie \rho_{b \text{ áruház_raktár_tartalom}})$

$\rho_{b \text{ beszállító_raktár}} \bowtie \rho_{a \text{ áruház_raktár_tartalom}}$

$\rho_{arak \text{ áruház_raktár_tartalom}} \bowtie \rho_{ar \text{ áruház_raktár_tartalom}}$

$\rho_{ar \text{ áruház_raktár_tartalom}} \bowtie \rho_{ar \text{ áruház_raktár_tartalom}}$

$\rho_{abk \text{ áruház_beszállító_kapcsolat}}$

5. Kérdezzük le a Coop áruházi raktárában lévő termékek átlagárát!

SELECT AVG(arta.Ár) AS 'Átlagár' FROM áruház_raktár_tartalom a JOIN áruház_raktár_tartalom_ár
arta ON a.TermekID=arta.TermekID JOIN áruház ar ON a.ArühazID=ar.ArühazID WHERE
ar.Név='Coop'

$\pi_{AVG(ár)}$

$\gamma_{AVG(ár)}$

$\sigma_{ar.Név = \text{"Coop"}}$

$(\rho_{a \text{ áruház_raktár_tartalom}} \bowtie \rho_{arta \text{ áruház_raktár_tartalom_ár}})$

$\rho_{arta \text{ áruház_raktár_tartalom_ár}} \bowtie \rho_{ar \text{ áruház}}$

$\rho_{ar \text{ áruház}}$

6. Kérdezzük le az akciós termékek nevét és árának különbségét a normál áratól!

SELECT art.Név, (arta.Ár-ataa.Akciós_ár) AS 'Árkülönség' FROM akciós_termékek_akciós_ár ataa
JOIN áruház_raktár_tartalom_ár arta ON ataa.TermekID=arta.TermekID JOIN áruház_raktár_tartalom
art ON ataa.TermekID=art.TermekID

$\pi_{art.Név, arta.Ár - ataa.Akciós_ár}$

$(\rho_{ataa \text{ akciós_termékek_akciós_ár}} \bowtie \rho_{arta \text{ áruház_raktár_tartalom_ár}})$

$\rho_{arta \text{ áruház_raktár_tartalom_ár}} \bowtie \rho_{art \text{ áruház_raktár_tartalom}}$

$\rho_{art \text{ áruház_raktár_tartalom}}$

7. Kérdezzük le az akciós termékek nevét és árának különbségét az akcióban kiírt normál áratól!

SELECT art.Név, (area.Eredeti_ár-ataa.Akciós_ár) AS 'Árkülönség' FROM akciós_termékek_akciós_ár
ataa JOIN akciós_termékek_eredeti_ár area ON ataa.TermekID=area.TermekID JOIN
áruház_raktár_tartalom art ON ataa.TermekID=art.TermekID

$\pi_{art.Név, area.Eredeti_ár - ataa.Akciós_ár}$

$(\rho_{ataa \text{ akciós_termékek_akciós_ár}} \bowtie \rho_{area \text{ akciós_termékek_eredeti_ár}})$

$\rho_{area \text{ akciós_termékek_eredeti_ár}} \bowtie \rho_{ataa \text{ akciós_termékek_akciós_ár}}$

$\rho_{art \text{ áruház_raktár_tartalom}}$

8. Hány darab különböző 300 forint feletti termék kapható a Sparban?

SELECT COUNT(art.TermekID) AS 'Darabszám' FROM áruház_raktár_tartalom art JOIN
áruház_raktár_tartalom_ár arta ON art.TermekID=arta.TermekID JOIN áruház ar ON
art.ArühazID=a.ArühazID WHERE arta.Ár>300 AND a.Név LIKE 'Spar'

$\pi_{COUNT(termekid)}$

$\gamma_{COUNT(termekid)}$

$\sigma_{arta.Ár > 300 \text{ AND } a.Név \text{ LIKE "Spar"}}$

$(\rho_{art \text{ áruház_raktár_tartalom}} \bowtie \rho_{arta \text{ áruház_raktár_tartalom_ár}})$

$\rho_{arta \text{ áruház_raktár_tartalom_ár}} \bowtie \rho_{a \text{ áruház}}$

$\rho_{a \text{ áruház}}$

9. Mely termékeket szállítják le a legrövidebb idő alatt a beszállítók?

SELECT b.Átlagos_kiszállítási_idő AS 'Idő', br.Név AS 'Leggyorsabban leszállítható termékek' FROM
beszállító_raktár br JOIN beszállító b ON br.BeszallitoID=b.BeszallitoID ORDER BY
b.Átlagos_kiszállítási_idő ASC LIMIT 1

$\pi_{(Átlagos_kiszállítási_idő) \gamma_{(Átlagos_kiszállítási_idő)} \sigma_{(b \text{ beszállító}} \bowtie (b \text{ BeszallitoID}=br \text{ BeszallitoID} (b \text{ Átlagos_kiszállítási_idő} \times$

$br \text{ BeszallitoID})$

10. Mely termékeket szállítják le a leghosszabb idő alatt a beszállítók?

SELECT b.Átlagos_kiszállítási_idő AS 'Idő', br.Név AS 'Leglassabban leszállítható termékek' FROM
beszállító_raktár br JOIN beszállító b ON br.BeszallitoID=b.BeszallitoID ORDER BY
b.Átlagos_kiszállítási_idő DESC LIMIT 1

$\pi_{(Átlagos_kiszállítási_idő) \gamma_{(Átlagos_kiszállítási_idő)} \sigma_{(b \text{ beszállító}} \bowtie (b \text{ BeszallitoID}=br \text{ BeszallitoID} (b \text{ Átlagos_kiszállítási_idő} \times$

$br \text{ BeszallitoID})$

Miskolc,
2022