

ADATBÁZIS RENDSZEREK 2

Autócentrum

Szerző: Kiss Gergő - BKUPJ9

Gyakorlatvezető: Dr. Bednarik László

A következő feladatban egy autócentrum modellezését fogom bemutatni. Az adatbázisban 3 tábla szerepel, egy „Beszállító”, egy „Autó”, valamint egy „Vásárló”. A „Vásárló” tábla összeköttetésben áll az „Autó” táblával, a rendszámon keresztül.

A táblákat létrehozó SQL parancsok:

```
104
105 public static void StatikusTablaletrehozas() {
106     String sqlp_beszallito="CREATE TABLE beszallito"
107         + "("
108         + " ID int not null,"
109         + " cegnev char(30) NOT NULL,"
110         + " szekhely char(40) NOT NULL,"
111         + " raktkasszlet integer,"
112         + " szallitas date NOT NULL,"
113         + " PRIMARY KEY (ID)"
114         + ")";
115
116     String sqlp_auto="CREATE TABLE auto"
117         + "("
118         + " rsz char(6) NOT NULL,"
119         + " beszallito_ID int NOT NULL,"
120         + " tipus char(20) NOT NULL,"
121         + " szin char(10) NOT NULL,"
122         + " motor char(10) NOT NULL,"
123         + " an integer NOT NULL,"
124         + " gyartasi_ido date NOT NULL,"
125         + " PRIMARY KEY (rsz)"
126         + ")";
127
128     String sqlp_vasarlo="CREATE TABLE vasarlo"
129         + "("
130         + " adoszam numeric(10) NOT NULL,"
131         + " nev char(25) NOT NULL,"
132         + " lakhely char(30) NOT NULL,"
133         + " szulido date NOT NULL,"
134         + " PRIMARY KEY (adoszam)"
135         + ")";
136
137     if(conn!=null) {
138         try {
139             s=conn.createStatement();
140             s.executeUpdate(sqlp_beszallito);
141             System.out.println("Beszállító tábla létrejött\n");
142             s.executeUpdate(sqlp_auto);
143             System.out.println("Autó tábla létrejött\n");
144             s.executeUpdate(sqlp_vasarlo);
145             System.out.println("Vásárló tábla létrejött\n");
146             s.close();
147         } catch (Exception ex) {
148             System.err.println(ex.getMessage());
149         }
150     }
151 }
152
```

A táblát feltöltő SQL parancsok:

```
156 public static void StatikusAdatfelvetel() {
157     if(conn != null) {
158         String sqlp_besz[]={
159             "insert into beszallito values (1, 'Ford Hungaria St.', 'Budapest, Fő út 101.', 110, to_date('2022.05.12', 'yyyy.mm.dd'))",
160             "insert into beszallito values (2, 'Skoda Hungaria St.', 'Budapest, Mállek út 101.', 200, to_date('2022.06.11', 'yyyy.mm.dd'))",
161             "insert into beszallito values (3, 'Opel Hungaria St.', 'Budapest, Sárány út 3.', 300, to_date('2022.04.12', 'yyyy.mm.dd'))",
162             "insert into beszallito values (4, 'Mazda Hungaria St.', 'Budapest, Vágány út 101.', 400, to_date('2022.07.09', 'yyyy.mm.dd'))"
163         };
164
165         String sqlp_auto[]={
166             "insert into auto values('jdc761',1, 'Ford Mondeo', 'piros', '1.8dt', 14000000, to_date('2022.01.12', 'yyyy.mm.dd'))",
167             "insert into auto values('jbc767',2, 'Skoda Octavia', 'kék', '2.0 crtdi', 16000000, to_date('2021.03.12', 'yyyy.mm.dd'))",
168             "insert into auto values('axk455',3, 'Opel Insignia', 'fehér', '1.7dti', 19000000, to_date('2022.09.21', 'yyyy.mm.dd'))",
169             "insert into auto values('okl767',4, 'Mazda 6', 'piros', '1.6b', 11000000, to_date('2022.03.12', 'yyyy.mm.dd'))"
170         };
171
172         String sqlp_vas[]={
173             "insert into vasarlo values(8485825968, 'Próba László', 'Szerencs, Fő út 101.', to_date('1998.01.12', 'yyyy.mm.dd'),'jdc761')",
174             "insert into vasarlo values(8485825847, 'Próba Béla', 'Kazincbarcika, Fő út 101.', to_date('1994.01.30', 'yyyy.mm.dd'),'jbc767')",
175             "insert into vasarlo values(8485824724, 'Próba Péter', 'Eger, Fő út 101.', to_date('1991.01.18', 'yyyy.mm.dd'),'axk455')",
176             "insert into vasarlo values(8485827896, 'Próba Imre', 'Debrecen, Fő út 101.', to_date('1997.01.21', 'yyyy.mm.dd'),'okl767')"
177         };
178
179         for(int i = 0; i < sqlp_besz.length; i++) {
180             try {
181                 s=conn.createStatement();
182                 s.executeUpdate(sqlp_besz[i]);
183                 System.out.println("Beszállító felvéve\n");
184                 s.close();
185             } catch (Exception ex) {
186                 System.err.println(ex.getMessage());
187             }
188         }
189
190         for(int i = 0; i < sqlp_auto.length; i++) {
191             try {
192                 s=conn.createStatement();
193                 s.executeUpdate(sqlp_auto[i]);
194                 System.out.println("Auto felvéve\n");
195                 s.close();
196             } catch (Exception ex) {
197                 System.err.println(ex.getMessage());
198             }
199         }
200     }
201 }
202
```

```

208    }
209    }
210
211    for(int i = 0; i < sqlp_vas.length; i++) {
212        try {
213            s.conn.createStatement();
214            s.executeUpdate(sqlp_vas[i]);
215            System.out.println("vásárló felvéve\n");
216            s.close();
217        } catch (Exception ex) {
218            System.err.println(ex.getMessage());
219        }
220    }
221 }
222 }
223

```

A program funkciói:

-Driver Regisztrálás, Kapcsolódás a szerverhez:

```

67
68 public static void DriverReg() {
69     try {
70         Class.forName("oracle.jdbc.driver.OracleDriver");
71         System.out.println("Sikeres driver Regisztrálás!\n");
72     } catch (Exception e) {
73         System.err.println(e.getMessage());
74     }
75 }
76
77 public static void Connect() {
78     try {
79         conn = DriverManager.getConnection(url, user, pwd);
80         System.out.println("Sikeres kapcsolódás!\n");
81     } catch (Exception e) {
82         System.err.println(e.getMessage());
83     }
84 }
85

```

```

Sikeres driver Regisztrálás!
Sikeres kapcsolódás
Beszállító tábla törölve!
Vásárló tábla törölve!
Auto tábla törölve!
Beszállító tábla létrejött
Auto tábla létrejött
Vásárló tábla létrejött
Auto tábla módosítva!
Beszállító felvéve
Beszállító felvéve
Beszállító felvéve
Beszállító felvéve
Auto felvéve
Auto felvéve
Auto felvéve
Auto felvéve

```

-Adatbázis „letisztítása”:

```
100 public static void StatikusTablaTorles() {
101     if(conn != null) {}
102     try {
103         String sqlp_beszallito="DROP TABLE BESZALLITO";
104         String sqlp_auto="DROP TABLE auto";
105         String sqlp_vasarlo="DROP TABLE vasarlo";
106         s=conn.createStatement();
107         s.executeUpdate(sqlp_beszallito);
108         System.out.println("Beszallito tábla törölve!\n");
109         s.executeUpdate(sqlp_vasarlo);
110         System.out.println("Vasarlo tábla törölve!\n");
111         s.executeUpdate(sqlp_auto);
112         System.out.println("Auto tábla törölve!\n");
113         s.close();
114     }catch(Exception ex) {
115         System.err.println(ex.getMessage());
116     }
117 }
```

```
Sikeres driver regisztrálás!
Sikeres kapcsolat
Beszallito tábla törölve!
Vasarlo tábla törölve!
Auto tábla törölve!
Beszallito tábla létrejött
Auto tábla létrejött
Vasarlo tábla létrejött
Auto tábla módosítva!
Beszallito felvéve
Beszallito felvéve
Beszallito felvéve
Beszallito felvéve
Auto felvéve
Auto felvéve
Auto felvéve
Auto felvéve
```

-Statikus Tábla Módosítás:

```
153 public static void StatikusTablaModositas() {
154     if(conn != null) {}
155     try {
156         String sqlp="alter table vasarlo add(auto_id references auto)";
157         s=conn.createStatement();
158         s.executeUpdate(sqlp);
159         System.out.println("Autó tábla módosítva!\n");
160         s.close();
161     }catch(Exception ex) {
162         System.err.println(ex.getMessage());
163     }
164 }
165
```

```
Sikeres driver regisztrálás!
Sikeres kapcsolat
Beszallito tábla törölve!
Vasarlo tábla törölve!
Auto tábla törölve!
Beszallito tábla létrejött
Auto tábla létrejött
Vasarlo tábla létrejött
Auto tábla módosítva!
Beszallito felvéve
Beszallito felvéve
Beszallito felvéve
Beszallito felvéve
Auto felvéve
Auto felvéve
Auto felvéve
Auto felvéve
```

-Műveletválasztó eljárás:

```
33 public static void MilyenMuvelet() {
34     System.out.println("Milyen műveletet szeretnél végrehajtani? Az alábbi lehetőségek közül válassz (a megadott formátumokat képes kiértékelni a program):");
35     System.out.println("StatikusAdattörölés - SAT\n" + "StatikusLekérdezés - SL\n" + "Módosíthatókurzor - MK\n"
36     + "DinamikusLekérdezés - DL\n" + "DinamikusAdattörölés - DA");
37     String muvelet = sc.next();
38     if (conn != null) {
39         if (muvelet.equals("SAT")) {
40             StatikusAdattorles();
41         } else if (muvelet.equals("SL")) {
42             StatikusLekerdzes();
43         } else if (muvelet.equals("MK")) {
44             Modosithatokurzor();
45         } else if (muvelet.equals("DL")) {
46             DinamikusLekerdzes();
47         } else if (muvelet.equals("DA")) {
48             DinamikusAdattorles();
49         } else {
50             System.out.println("Rossz választ adtál, kérlek a fentebb felsoroltakat add meg!");
51             MilyenMuvelet();
52         }
53     }
54 }
55 }
```

```
Milyen műveletet szeretnél végrehajtani? Az alábbi lehetőségek közül válassz (a megadott formátumokat képes kiértékelni a program):
StatikusAdattörölés - SAT
StatikusLekérdezés - SL
Módosíthatókurzor - MK
DinamikusLekérdezés - DL
DinamikusAdattörölés - DA
|
```

-Statikus Adattörölés:

```
223
224 public static void StatikusAdattorles() {
225     System.out.println("Törlendő beszállító: ");
226     String id = sc.next();
227     String sqlp = "delete from beszallito where raktarkeszlet > 0 AND ID like '"+id+"'";
228     if (conn != null) {
229         try {
230             s = conn.createStatement();
231             s.executeUpdate(sqlp);
232             s.close();
233             System.out.println(id + " azonosítójú beszállító törölve!\n");
234         } catch (Exception ex) {
235             System.err.println(ex.getMessage());
236         }
237     }
238
239     System.out.println("Szeretnél még egy műveletet?(I/N)");
240     String igen_nem = sc.next().trim();
241     SzeretnelMegEgyet(igen_nem);
242 }
243 }
```

```
DinamikusAdattörölés - DA
SAT
Törlendő beszállító:
2
2 azonosítójú beszállító törölve!
```

-Statikus Lekérdezés:

```
243
244 public static void StatikusLekerdezes() {
245     System.out.println("Add meg a táblát, amelynek szeretné az adatait:");
246     String tabla = sc.next().trim();
247     if(conn != null) {
248         if(tabla.equals("auto")) {
249             if(conn != null) {
250                 String sqlp="select * from auto";
251                 System.out.println("Rendszám BeszállítóID Típus Szín Motor Ár Évjárat");
252                 System.out.println("-----");
253                 try {
254                     s=conn.createStatement();
255                     s.executeQuery(sqlp);
256                     rs=s.getResultSet();
257                     while(rs.next()) {
258                         String rsz = rs.getString("rsz");
259                         int beszallito_id=rs.getInt("beszallito_id");
260                         String tipus = rs.getString("tipus");
261                         String szin = rs.getString("szin");
262                         String motor = rs.getString("motor");
263                         int ar = rs.getInt("ar");
264                         String evjarat = rs.getString("gyartasi_ido");
265
266                         System.out.println(rsz+" "+beszallito_id+" "+tipus+" "+szin+" "+motor+" "+ar+" "+evjarat);
267                     }
268                     ELdontendo();
269                     rs.close();
270                 }catch(Exception ex) {
271                     System.out.println(ex.getMessage());
272                 }
273             }
274         }
275     }
276 }
```

```
274
275 }else if(tabla.equals("beszallito")) {
276     if(conn != null) {
277         String sqlp="select * from beszallito";
278         System.out.println("ID Cegnev Székhely Raktarkeszlet Szállítás");
279         System.out.println("-----");
280         try {
281             s=conn.createStatement();
282             s.executeQuery(sqlp);
283             rs=s.getResultSet();
284             while(rs.next()) {
285                 int id=rs.getInt("id");
286                 String cegnev = rs.getString("cegnev");
287                 String szekhely = rs.getString("szekhely");
288                 int raktarkeszlet = rs.getInt("raktarkeszlet");
289                 String szallitas = rs.getString("szallitas");
290
291                 System.out.println(id+" "+cegnev+" "+szekhely+" "+raktarkeszlet+" "+szallitas);
292             }
293             ELdontendo();
294             rs.close();
295         }catch(Exception ex) {
296             System.out.println(ex.getMessage());
297         }
298     }
299 }
```

```
299
300 }else if(tabla.equals("vasarlo")){
301     if(conn != null) {
302         String sqlp="select * from vasarlo";
303         System.out.println("Adoszam Nev Lakhely Születési idő");
304         System.out.println("-----");
305         try {
306             s=conn.createStatement();
307             s.executeQuery(sqlp);
308             rs=s.getResultSet();
309             while(rs.next()) {
310                 long adoszam=rs.getLong("adoszam");
311                 String nev = rs.getString("nev");
312                 String lakhely = rs.getString("lakhely");
313                 String szulido = rs.getString("szulido");
314
315                 System.out.println(adoszam+" "+nev+" "+lakhely+" "+szulido);
316             }
317             ELdontendo();
318             rs.close();
319         }catch(Exception ex) {
320             System.out.println(ex.getMessage());
321         }
322     }
323 }else {
324     System.out.println("Ilyen tábla nincs!");
325     StatikusLekerdezes();
326 }
327 System.out.println("Szeretnél még egy műveletet?(I/N)");
328 String igen_nem = sc.next().trim();
329 SzeretnelMegEgyet(igen_nem);
330 }
331 }
```

```
SL
Add meg a táblát, amelynek szeretné az adatait:
auto
Rendszám BeszállítóID Típus Szín Motor Ár Évjárat
-----
jdc761 1 Ford Mondeo piros 1.8dt 14000000 2022-01-12 00:00:00.0
jbc767 2 Skoda Octavia kék 2.0 crtdi 16000000 2021-03-12 00:00:00.0
axk455 3 Opel Insignia fehér 1.7dti 19000000 2022-09-21 00:00:00.0
okl767 4 Mazda 6 piros 1.6b 11000000 2022-03-12 00:00:00.0
Szeretné egy másik tábla adatait?(I/N)
```

-Dinamikus lekérdezés:

```
332 public static void DinamikusLekerdezes() {
333     System.out.println("Beszállító ID-je: ");
334     String id = sc.next().trim();
335     String sqlp = "select cegnev from beszallito where raktarkezeslet>0 AND ID= '"+id+"'";
336     if(conn != null) {
337         try {
338             s=conn.createStatement();
339             s.executeQuery(sqlp);
340             rs=s.getResultSet();
341             while(rs.next()) {
342                 String cegnev = rs.getString("cegnev");
343                 System.out.println("Cégnév: "+cegnev);
344             }
345             rs.close();
346         }catch(Exception ex) {
347             System.out.println(ex.getMessage());
348         }
349     }
350     System.out.println("Szeretnél még egy műveletet?(I/N)");
351     String igen_nem = sc.next().trim();
352     SzeretneLMegEgyet(igen_nem);
353 }
354 }
```

```
DL
Beszállító ID-je:
3
Cégnév: Opel Hungaria Bt.
Szeretnél még egy műveletet?(I/N)
```

-Dinamikus Adattörölés

```
369 public static void DinamikusAdattorles() {
370     System.out.println("Törlendő beszállító: ");
371     String id = sc.next();
372     String sqlp="delete from "+ user +".BESZALLITO where ID=?";
373     if(conn != null) {
374         try {
375             ps=conn.prepareStatement(sqlp);
376             ps.setString(1, id);
377             ps.executeUpdate();
378             ps.close();
379             System.out.println(id + " azonosítójú beszállító törölve!\n");
380         }catch(Exception ex) {
381             System.err.println(ex.getMessage());
382         }
383     }
384     System.out.println("Szeretnél még egy műveletet?(I/N)");
385     String igen_nem = sc.next().trim();
386     SzeretneLMegEgyet(igen_nem);
387 }
388 }
389 }
```

```
DinamikusAdattorles - OK
DA
Törlendő beszállító:
2
2 azonosítójú beszállító törölve!
```

-Módosítható Kurzor:

```
391 public static void ModosithatoKurzor() {  
392     System.out.println("Rendszám: ");  
393     String rsz = sc.next().trim();  
394     String sqlp = "select szin from auto where rsz= '"+rsz+"'";  
395     if(conn != null) {  
396         try {  
397             s=conn.createStatement(ResultSet.TYPE_FORWARD_ONLY, ResultSet.CONCUR_UPDATABLE);  
398             rs=s.executeQuery(sqlp);  
399             while(rs.next()) {  
400                 rs.updateString("szin", "lila");  
401                 rs.updateRow();  
402             }  
403             System.out.println(rsz+" rendszámú autó színe lila lett!");  
404         } catch (Exception ex) {  
405             System.err.println(ex.getMessage());  
406         }  
407         System.out.println("Szeretnél még egy műveletet?(I/N)");  
408         String igen_nem = sc.next().trim();  
409         SzeretneMegEgyet(igen_nem);  
410     }  
411 }  
412 }  
413 }  
414 }  
415 }
```

```
DinamikusAdattörles - DA  
MK  
Rendszám:  
jdk672  
jdk672 rendszámú autó színe lila lett!  
szeretnél még egy műveletet?(I/N)
```