

Adatbázis rendszerek II.

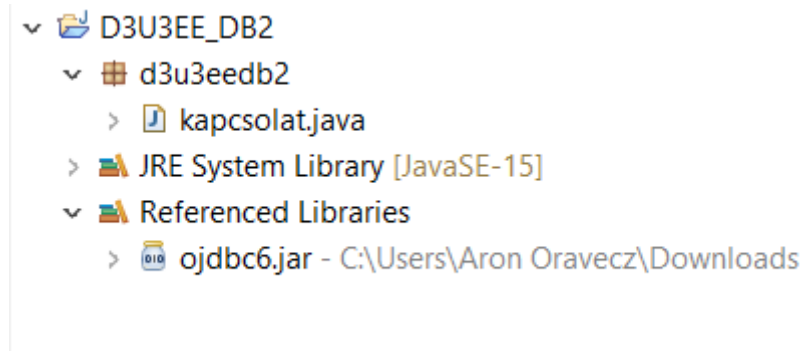
Gyakorlati feladatsor

Készítette:
Oravecz Áron
D3U3EE

Dátum:
2022 második félév

A feladat:

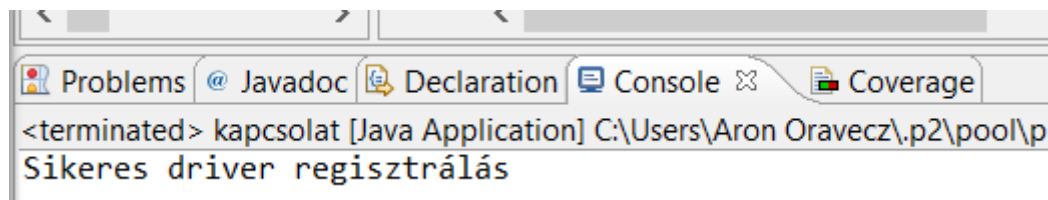
szükséges drive-ok feltöltése az eclipse-be.



B feladat:

DriverReg metódus megírása

```
public static void DriverReg() {  
    try {  
        Class.forName("oracle.jdbc.driver.OracleDriver");  
        System.out.println("Sikeres driver regisztrálás\n");  
    } catch (Exception e) {  
        System.err.println(e.getMessage());  
    }  
}
```



Sikeres driver használat ellenőrzése.

C feladat:

Adatbázishoz való kapcsolódási metódus megírása és alkalmazása.

URL – elérése az adatbázishoz.

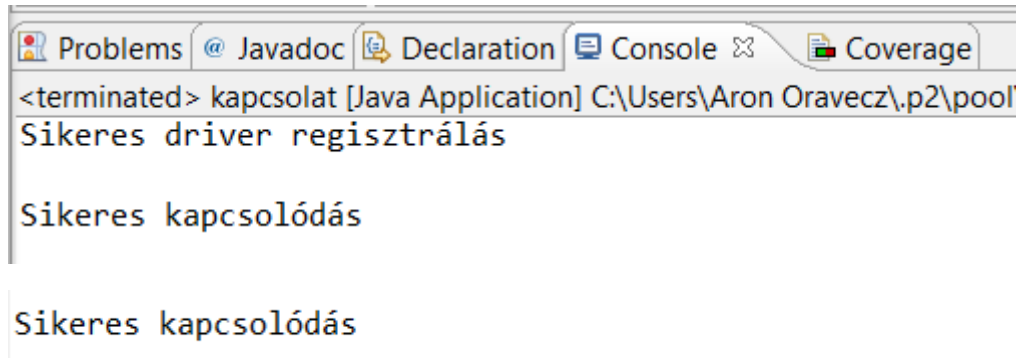
User - adatbázishoz kapcsolt felhasználó bejelentkezéshez szükséges.

pwd – jelszó, ami szükséges a kapcsolódáshoz az adatbázishoz

```

public static void Connect() {
    String url = "jdbc:oracle:thin:@193.6.5.58:1521:XE";
    String user = "H22_d3u3ee";
    String pwd = "D3U3EE";
    try {
        conn = (Connection) DriverManager.getConnection(url, user, pwd);
        System.out.println("Sikeres kapcsolódás\n");
    } catch (Exception e) {
        System.err.println(e.getMessage());
    }
}

```



D feladat:

```

public static void LeKapcs() {
    if (conn != null) {
        try {
            conn.close();
            System.out.println("Sikeres lekapcsolódás\n");
        } catch (Exception ex) {
            System.err.println(ex.getMessage());
        }
    }
}

```

el kellett készíteni a lekapcsolódási metódust. Lekapcsolódik az adatbázisról.

1. feladat.

```
// 1 feladat auto és tulaj tábla létrehozása
public static void StatikusTablaLetrehozas() {
    String sqlp_auto = "CREATE TABLE auto(rsz char(6) primary key, "
        + "tipus char(10) not null, szin char(10) default 'feher',"
        + "evjarat number(4), ar number(8) CHECK(ar>0))";

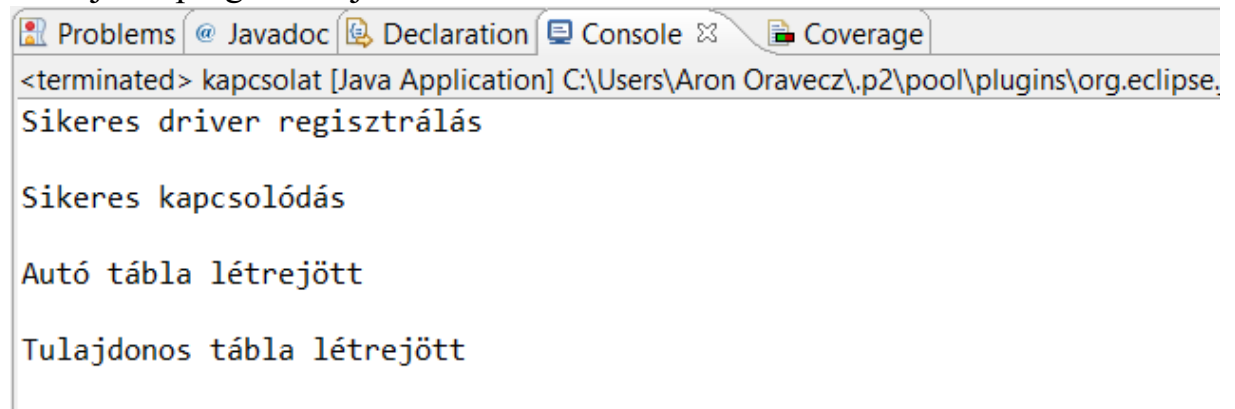
    String sqlp_tulaj = "CREATE TABLE tulaj(id number(3) primary key, "
        + "nev char(20) not null, cim char(20), szuldatum date)";

    if (conn != null) {
        try {
            s = conn.createStatement();
            s.execute(sqlp_auto);
            System.out.println("Autó tábla létrejött\n");
            s.execute(sqlp_tulaj);
            System.out.println("Tulajdonos tábla létrejött\n");
            s.close(); // erőforrás felszabadítása
        } catch (Exception ex) {
            System.err.println(ex.getMessage());
        }
    }
}
```

sqlp_auto nevezetű string változóban leírjuk milyen sql- parancsot, vagyis létrehozást szeretnénk most használni. Itt megadjuk milyen típusúak a benne lévő belső elemek

Majd hasonlóképpen létrehozunk egy sqlp_tulaj-t is.

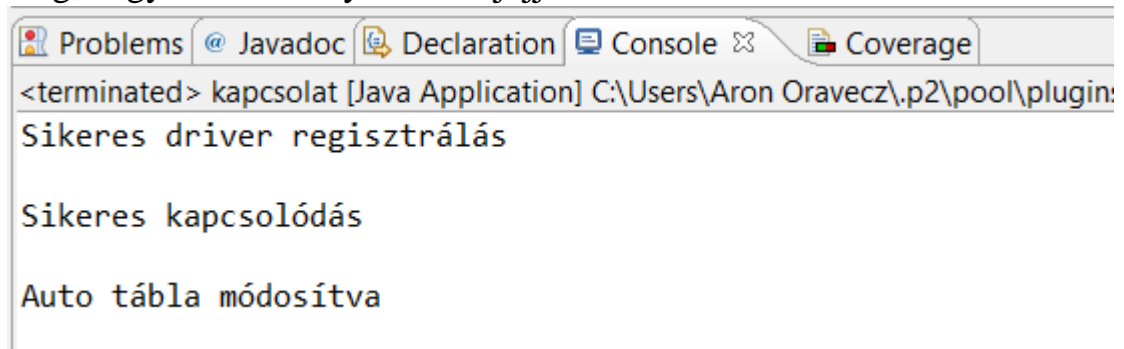
Majd egy try ctach segítségével hiba ellenőrzést végzünk és kiírjuk a felhasználónak, a sikeres táblázat létrejöttét vagy éppenséggel ha véletlen formán nem jönne létre a tábla akkor a catch részben, a hibánk típusát közöljük a program írójával.



2. feladat

```
// 2. feladat
public static void StatikusTablaModositas() {
    if (conn != null) {
        try {
            String sqlp = "ALTER TABLE auto ADD(tulaj_id REFERENCES tulaj)";
            s = conn.createStatement();
            s.executeUpdate(sqlp);
            System.out.println("Auto tábla módosítva\n");
            s.close();
        } catch (Exception ex) {
            System.err.println(ex.getMessage());
        }
    }
}
```

Úgy módosítjuk az autó táblát, hogy az autó tulajdonosát is eltároljuk ezt egy idegen kulcs befűzésével hozzá Alter-ezésével, oldhatjuk meg. s változó egy Statment ami connect-el az adatbázishoz és az executeUpdate-val módosítani tudjuk a meglévő táblánkat. Természetesen mind ezt egy hiba ellenőrzési folyamaton belül tesszük meg, hogy minél könnyedébben jöjjünk rá a hiba forrására.



3. feladat

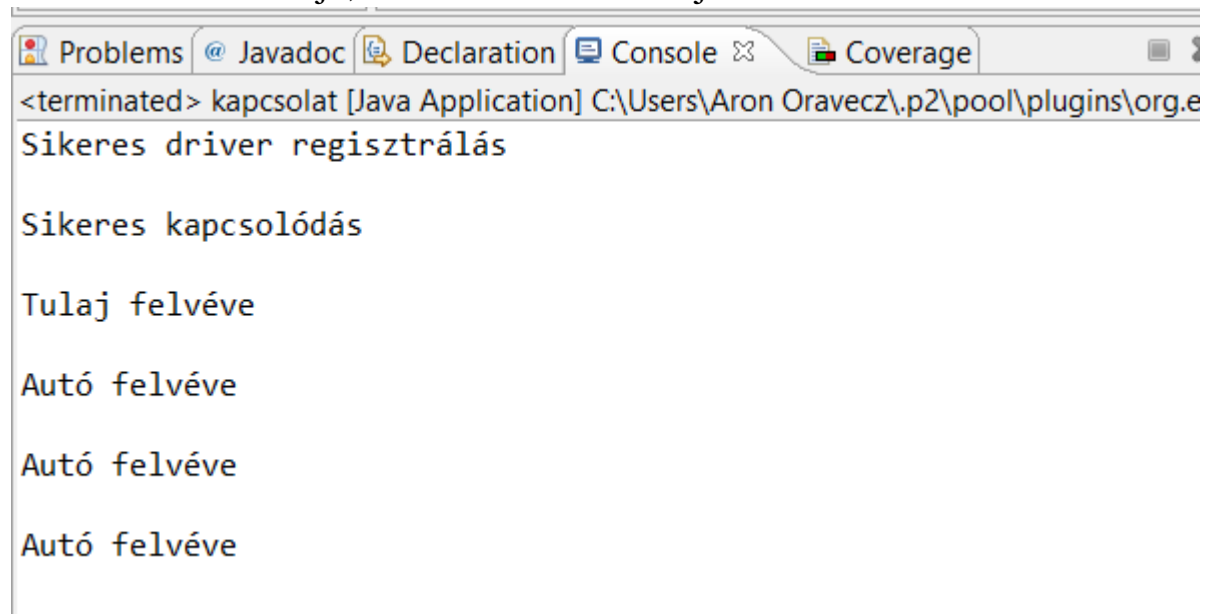
```
// 3. feladat
public static void StatikusAdatfelvitel() {
    if (conn != null) {
        String sqlp_tul = "INSERT INTO tulaj VALUES(1, 'Tóth Máté', "
            + " ' Miskolc', to_date('1980.05.12', 'yyyy.mm.dd'))";

        String[] sqlp = { "INSERT INTO auto VALUES('aaa111','opel','piros',2014,1650000,1)",
            "INSERT INTO auto VALUES('bbb222','mazda',null,2016,2800000,1)",
            "INSERT INTO auto VALUES(rsz, tipus, evjarat, ar) VALUES ('ccc333','ford',2009,1500000)" };

        try {
            s = conn.createStatement();
            s.execute(sqlp_tul);
            System.out.println("Tulaj felvéve\n");
            s.close();
        } catch (Exception ex) {
            System.err.println(ex.getMessage());
        }

        for (int i = 0; i < sqlp.length; i++) {
            try {
                s = conn.createStatement();
                s.executeUpdate(sqlp[i]);
                System.out.println("Autó felvéve\n");
                s.close();
            } catch (Exception ex) {
                System.err.println(ex.getMessage());
            }
        }
    }
}
```

Statikus adatfelvétel. Először megírjuk az adatokat miket és hogyan akarunk felvenni majd, hiba ellenőrzéssel elérjük az adatbázis és



4. feladat

A felhasználótól kérünk be adatot vagyis dinamikus adat felvételt alkalmazunk.

```
// 4. feladat
public static void DinamikusAdatFelvetel() {
    if (conn != null) {
        // Az SQL parancsban a ? helyére kerülnek a paraméterek
        String sqlp = "INSERT INTO auto(rsz, tipus, szín, evjarat, ar, tulaj_id)" + "VALUES (?, ?, ?, ?, ?, ?)";

        System.out.println("Kérem a rendszámot: ");
        String rsz = sc.next().trim();

        System.out.println("Kérem a típust: ");
        String tipus = sc.next().trim();

        System.out.println("Kérem adja meg a színt: ");
        String szín = sc.next().trim();

        System.out.println("Kérem adja meg az évjáratot: ");
        int evjarat = sc.nextInt();

        System.out.println("Kérem adja meg az árat: ");
        float ar = sc.nextFloat();

        System.out.println("Kérem a tulajdonos azonosítóját: ");
        int tulaj_id = sc.nextInt();
        try {
            ps = conn.prepareStatement(sqlp);
            ps.setString(1, rsz);
            ps.setString(2, tipus);
            ps.setString(3, szín);
            ps.setInt(4, evjarat);
            ps.setFloat(5, ar);
            ps.setInt(6, tulaj_id);
            ps.executeUpdate();
            System.out.println("Autó felvétele\n");
        } catch (Exception ex) {
            System.err.println(ex.getMessage());
        }
    }
}
```

DinamikusAdatFelvetel

5. feladat

6. feladat

7. feladat

8. feladat

9. feladat

10.feladat

11.feladat