

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

Bartul Brajković, 0036507098

23. ožujka 2021.

LABORATORIJ PROFILA 2

Odjeljak Sustavi baza podataka

6. Vježba

2.1. Zadatak

```
3.
4. GO
5. CREATE PROCEDURE montirajBezTrans @sifAlat INT, @oznStroj CHAR(10)
6. AS
7. BEGIN
8.
9.     BEGIN TRY
10.
11.         BEGIN TRY
12.
13.             INSERT INTO montaza VALUES (@oznStroj, @sifAlat)
14.
15.         END TRY
16.
17.         BEGIN CATCH
18.
19.             DECLARE @errorPKMontaza INT = ERROR_NUMBER()
20.
21.             --error 2627 is violation of unique constraint
22.             (including primary key). Cannot insert duplicate key in object
23.             IF (@errorPKMontaza = 2627)
24.                 THROW 50511, 'Alat je zauzet', 1;
25.
26.         END CATCH
27.
28.         UPDATE alat SET montiran = 'D' WHERE sifAlat = @sifAlat
29.
30.         BEGIN TRY
31.             UPDATE stroj SET brojMontiranih = brojMontiranih+1 WHERE
32.             oznStroj = @oznStroj
33.         END TRY
34.
35.         BEGIN CATCH
36.
37.             --error 547 is used dor any constraint violation
38.             DECLARE @errorKapacitet INT = ERROR_NUMBER()
39.
40.             IF (@errorKapacitet = 547)
41.                 THROW 50512, 'Kapacitet stroja je popunjen', 1
42.
43.         END CATCH
44.
45.     END TRY
46.
47.     BEGIN CATCH
48.
49.         THROW
50.     END CATCH
51. END
52. GO
53. SELECT * FROM montaza;
54. SELECT * FROM alat;
```

```
55. SELECT * FROM stroj;  
56.  
57. --test  
58. EXEC dbo.montirajBezTrans @sifAlat = 137, @oznStroj = 'S2'
```

2.2. Zadatak

a) montiranje slobodnog alata na stroj čiji kapacitet nije popunjen

```
EXEC dbo.montirajBezTrans @sifAlat = 137, @oznStroj = 'S2'
```

Upit se uspješno izvodi – ne narušuje nijedno integritetsko ograničenje

b) montiranje slobodnog alata na stroj čiji je kapacitet popunjen

```
EXEC dbo.montirajBezTrans @sifAlat = 101, @oznStroj = 'S4'
```

Poziv procedure narušava integritetsko ograničenje ckhKapacitet (prva dva upita prolaze, na trećem se javlja greška).

```
Msg 50512, Level 16, State 1, Procedure dbo.montirajBezTrans, Line 35 [Batch Start Line 61]  
Kapacitet stroja je popunjen
```

c) montiranje zauzetog alata na stroj čiji kapacitet nije popunjen

```
EXEC dbo.montirajBezTrans @sifAlat = 103, @oznStroj = 'S6'
```

Rezultat izvođenja je greška da je alat već zauzet, do greške dolazi jer je sifra alata (103) već primarni ključ u relaciji montaza

```
Msg 50511, Level 16, State 1, Procedure dbo.montirajBezTrans, Line 19 [Batch Start Line 64]  
Alat je zauzet
```

d) montiranje zauzetog alata na stroj čiji je kapacitet popunjen

```
EXEC dbo.montirajBezTrans @sifAlat = 103, @oznStroj = 'S4'
```

Dobivamo istu grešku kao u prethodnom primjeru. Alat je zauzet te se neće ni pokušati izvoditi ostali dijelovi koda.

```
Msg 50511, Level 16, State 1, Procedure dbo.montirajBezTrans, Line 19 [Batch Start Line 67]  
Alat je zauzet
```

e) montiranje nepostojećeg alata na stroj čiji kapacitet nije popunjen

```
EXEC dbo.montirajBezTrans @sifAlat = 103, @oznStroj = 'S4'
```

Upit se uspješno obavlja, ali je jedino rezultiralo u tablici stroj gdje se povećao brojMontiranih.

To je pogrešno.

3.1. Zadatak

```
GO
CREATE PROCEDURE montirajUzTrans @sifAlat INT, @oznStroj CHAR(10)
AS
BEGIN TRANSACTION

    BEGIN TRY

        BEGIN TRY

            INSERT INTO montaza VALUES (@oznStroj, @sifAlat)

        END TRY

        BEGIN CATCH

            ROLLBACK TRANSACTION
            DECLARE @errorPKMontaza INT = ERROR_NUMBER()

            --error 2627 is violation of unique constraint (including primary
            key). Cannot insert duplicate key in object
            IF (@errorPKMontaza = 2627)
                THROW 50511, 'Alat je zauzet', 1;

        END CATCH

        UPDATE alat SET montiran = 'D' WHERE sifAlat = @sifAlat

        BEGIN TRY
            UPDATE stroj SET brojMontiranih = brojMontiranih+1 WHERE oznStroj =
@oznStroj
        END TRY

        BEGIN CATCH

            ROLLBACK TRANSACTION
            DECLARE @errorKapacitet INT = ERROR_NUMBER()

            --error 547 is used dor any constraint violation
            IF (@errorKapacitet = 547)
                THROW 50512, 'Kapacitet stroja je popunjen', 1

        END CATCH
```

```

        END TRY

        BEGIN CATCH

            ROLLBACK TRANSACTION
            THROW

        END CATCH

    COMMIT TRANSACTION

GO

```

3.2. Zadatak

a) montiranje slobodnog alata na stroj čiji kapacitet nije popunjen

```
EXEC dbo.montirajUzTrans @sifAlat = 137, @oznStroj = 'S2'
```

Upit se uspješno izvod – ne narušuje nijedno integritetsko ograničenje

b) montiranje slobodnog alata na stroj čiji je kapacitet popunjen

```
EXEC dbo.montirajUzTrans @sifAlat = 101, @oznStroj = 'S4'
```

Poziv procedure narušava integritetsko ograničenje ckhKapacitet – nijedan upit ne prolazi

c) montiranje zauzetog alata na stroj čiji kapacitet nije popunjen

```
EXEC dbo.montirajUzTrans @sifAlat = 103, @oznStroj = 'S6'
```

Rezultat izvođenja je greška da je alat već zauzet, do greške dolazi jer je sifra alata (103) već primarni ključ u relaciji montaza

```
Msg 50511, Level 16, State 1, Procedure dbo.montirajBezTrans, Line 19 [Batch Start Line 64]
```

Alat je zauzet

d) montiranje zauzetog alata na stroj čiji je kapacitet popunjen

```
EXEC dbo.montirajUzTrans @sifAlat = 103, @oznStroj = 'S4'
```

Dobivamo istu grešku kao u prethodnom primjeru. Alat je zauzet te se neće ni pokušati izvoditi ostali dijelovi koda.

```
Msg 50511, Level 16, State 1, Procedure dbo.montirajBezTrans, Line 19 [Batch Start Line 67]
```

Alat je zauzet

e) montiranje nepostojećeg alata na stroj čiji kapacitet nije popunjen

```
EXEC dbo.montirajUzTrans @sifAlat = 103, @oznStroj = 'S4'
```

Upit se ne obavlja uspješno.

Poruka:

Msg 3902, Level 16, State 1, Procedure dbo.montirajBezTrans2, Line 50 [Batch Start Line 77]

The COMMIT TRANSACTION request has no corresponding BEGIN TRANSACTION.

`SELECT XACT_STATE()` vraća:

- 1 : Trenutni upit ima aktivnu transakciju
- 0 : Ne postoji aktivna transakcija za upit
- -1 : Trenutni upit ima aktivnu transakciju, ali se dogodila greška koja klasificirala transakciju kao uncommittable

4. Zadatak

```
import java.sql.CallableStatement;  
import java.sql.Connection;  
import java.sql.DriverManager;  
import java.sql.SQLException;
```

```
public class ZoviMontirajUzTrans {  
  
    public static void main(String[] args) {  
  
        Connection connection = otvoriKonekciju();  
  
        try {  
            int sifAlata = Integer.parseInt(args[0]);  
            String oznStroj = args[1];  
  
            CallableStatement cstmt = connection.prepareCall("{call  
dbo.montirajUzTrans(?, ?)}");  
  
            cstmt.setInt("sifAlata", sifAlata);  
            cstmt.setString("oznStroj", oznStroj);  

```

```

        cstmt.execute();

        System.out.println("Alat je montiran.");

        cstmt.close();

        connection.close();

    } catch (SQLException exception) {

        ispisiPogresku(exception);
        System.exit(-1);
        // I u slučaju pogreske trebalo bi osloboditi resurse,
        // ali program ovdje ionako završava.
    }
}

private static void ispisiPogresku(SQLException exception) {
    System.out.println(exception.getErrorCode() + "; "
        + exception.getMessage() + "; "
        + "State=" + exception.getSQLState());
}

private static Connection otvoriKonekciju() {
    // sastavljanje JDBC URL:
    String url =
        "jdbc:sqlserver://localhost:1433;"
        + "databaseName=labprof6;"
        + "user=sa;"
        + "password=xxxxxxx;";

    // ucitavanje i registriranje SQL Server JDBC driver-a
    try {
        // connection = DriverManager.getConnection(connectionString);
        Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
        System.out.println("SQL Server JDBC driver je ucitan i registriran.");
    } catch (ClassNotFoundException exception) {
        System.out.println("Pogreska: nije uspjelo ucitavanje JDBC driver-a.");
        System.out.println(exception.getMessage());
        System.exit(-1);
    }

    // uspostavljanje konekcije
    Connection conn = null;
    try {
        conn = DriverManager.getConnection(url);
        System.out.println("Konekcija je uspostavljena.");
    } catch (SQLException exception) {
        System.out.println("Pogreska: nije uspjelo uspostavljanje konekcije.");
        System.out.println(exception.getErrorCode() + " " +
exception.getMessage());
        System.exit(-1);
    }
    return conn;
}

```

```
}
```

5. Zadatak

```
import java.sql.CallableStatement;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

public class ZoviMontirajBezTrans {

    public static void main(String[] args) {

        Connection connection = otvoriKonekciju();

        try {
            int sifAlata = Integer.parseInt(args[0]);
            String oznStroj = args[1];

            CallableStatement cstmt = connection.prepareCall("{call
dbo.montirajBezTrans(?, ?)}");

            cstmt.setInt("sifAlata", sifAlata);
            cstmt.setString("oznStroj", oznStroj);

            cstmt.execute();

            System.out.println("Alat je montiran.");

            cstmt.close();

            connection.close();

        } catch (SQLException exception) {

            ispisiPogresku(exception);
            System.exit(-1);
            // I u slučaju pogreske trebalo bi osloboditi resurse,
            // ali program ovdje ionako završava.
        }

    }

    private static void ispisiPogresku(SQLException exception) {
        System.out.println(exception.getErrorCode() + "; "
            + exception.getMessage() + "; "
            + "State=" + exception.getSQLState());
    }
}
```



```

}

private static Connection otvoriKonekciju() {
    // sastavljanje JDBC URL:
    String url =
        "jdbc:sqlserver://localhost:1433;"
        + "databaseName=labprof6;"
        + "user=sa;"
        + "password=xxxxxxx;";

    // ucitavanje i registriranje SQL Server JDBC driver-a
    try {
        // connection = DriverManager.getConnection(connectionString);
        Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
        System.out.println("SQL Server JDBC driver je ucitan i registriran.");
    } catch (ClassNotFoundException exception) {
        System.out.println("Pogreska: nije uspjelo ucitavanje JDBC driver-a.");
        System.out.println(exception.getMessage());
        System.exit(-1);
    }

    // uspostavljanje konekcije
    Connection conn = null;
    try {
        conn = DriverManager.getConnection(url);
        System.out.println("Konekcija je uspostavljena.");
    } catch (SQLException exception) {
        System.out.println("Pogreska: nije uspjelo uspostavljanje konekcije.");
        System.out.println(exception.getErrorCode() + " " +
exception.getMessage());
        System.exit(-1);
    }
    return conn;
}
}

```