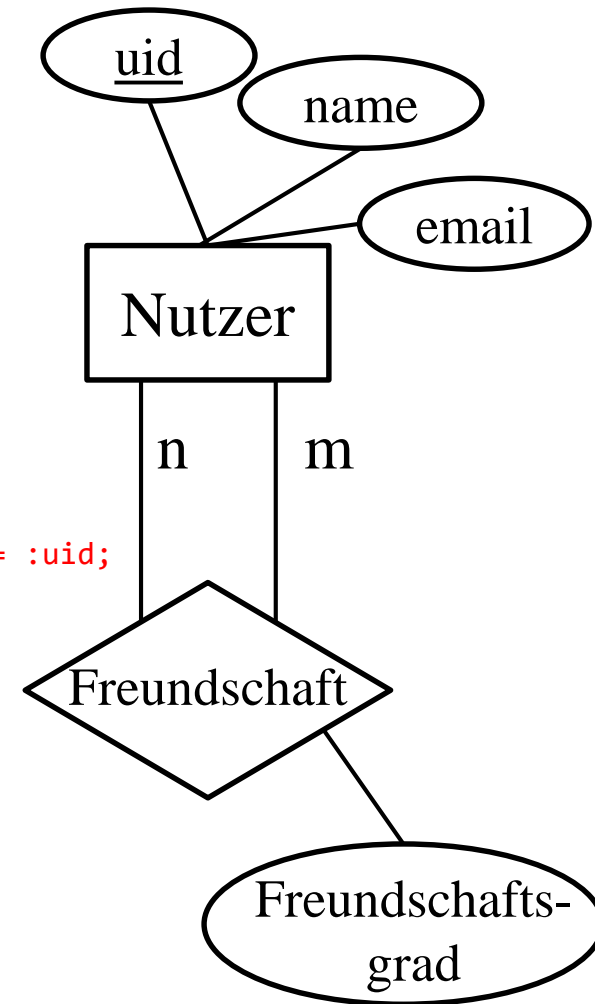


# Embedded SQL

**C-Programm, welches zu jedem Nutzer dessen Namen und die Anzahl seiner Freunde ausgibt.**

```
exec sql include sqlca;
main() {
    exec sql begin declare section;
        int    uid;
        int    count;
        char   name[100];
    exec sql end declare section;
    exec sql connect to dbname;
    exec sql declare C1 cursor for
        SELECT uid, name FROM Nutzer;
    exec sql declare C2 cursor for
        SELECT COUNT(*) FROM Freundschaft WHERE uid1 = :uid OR uid2 = :uid;
    exec sql open C1;
    while (sqlcode == ok) {
        exec sql fetch C1 into :uid, :name;
        printf ("%s: ", name);
        exec sql open C2;
        while (sqlcode == ok) {
            exec sql fetch C2 into :count;
            printf ("%d\n", count);
        }
        exec sql close C2;
    }
    exec sql close C1;
    exec sql disconnect;
}
```



# Verbindungsaufbau JDBC

---

```
import java.sql.*;

...
Class.forName("com.ibm.db2.jcc.DB2Driver");
String url = "jdbc:db2://host:50000/dbname";
Connection con = DriverManager.getConnection(url, user, pw);

Statement stmt = con.createStatement(
    "INSERT INTO Nutzer VALUES (1, aaa@bbb.de', 'Lisa')");
stmt.executeUpdate();

...
```

*Kompilieren:*

```
javac -cp .:jdbcdriver.jar Test2.java
```

*Ausführen:*

```
java -cp .:jdbcdriver.jar Test2
```

# Aufgabe 2d)

---

```
PreparedStatement ps = con.prepareStatement
```

```
("UPDATE Freundschaft SET Fgrad = ? WHERE (uid1 = ? AND uid2 = ? );");
```

```
...
```

```
public void changeFgrad(int uid1, int uid2, String fgradNew) throws SQLException {
```

```
    ps.setString (1, fgradNew);
```

```
    ps.setInt (2, uid1);
```

```
    ps.setInt (3, uid2);
```

```
    ps.executeUpdate();
```

```
    ps.setString (1, fgradNew);
```

```
    ps.setInt (2, uid2);
```

```
    ps.setInt (3, uid1);
```

```
    ps.executeUpdate();
```

```
}
```

```
// oder zusätzlich OR (uid1 = ? AND uid2 = ? ) in WHERE Klausel
```

```
// ps.setInt(4, uid2);
```

```
// ps.setInt(5, uid1);
```

# Aufgabe 2e)

---

```
PreparedStatement ps = con.prepareStatement
 („ SELECT a.name, b.name FROM Nutzer a, Freundschaft f, Nutzer b
  WHERE (f.uid1 = ? OR f.uid2 = ?)
  AND f.uid1 = a.uid
  AND f.uid2 = b.uid");
```

...

```
public void getFreunde(int uid) throws SQLException {

    String user1;
    String user2;

    ps.setInt (1, uid);
    ps.setInt (2, uid);
    ResultSet rs = ps.executeQuery();
    while (rs.next()) {
        user1 = rs.getString(1);
        user2 = rs.getString(2);

        System.out.println (
            String.format („%s ist befreundet mit %s.“, user1, user2));
    }
    rs.close();
}
```

# Aufgabe 2f)

---

```
PreparedStatement ps = con.prepareStatement
    ("UPDATE Freundschaft SET Fgrad = ?
     WHERE (uid1 = ? AND uid2 = ? )");
...
PreparedStatement psId = con.prepareStatement
    ("SELECT uid FROM Nutzer WHERE name = ?");
...

public void changeFgradByName(String unname1, String unname2, String fgradNew) throws
SQLException {
    int uid1;
    int uid2;

    psId.setString(1, unname1);
    ResultSet rsId = psId.executeQuery();
    if (rsId.next()) {
        uid1 = rsId.getInt(1);
    }
    rsId.close();

    psId.setString(1, unname2);
    rsId = psId.executeQuery();
    if (rsId.next()) {
        uid2 = rsId.getInt(1);
    }
    rsId.close();
    changeFgrad(uid1, uid2, fgradNew);
};
```

## 2g) Vergleich Cursor-Operatoren mit JDBC ResultSet

---

```
Class.forName("com.ibm.db2.jdbc.net.DB2Driver");
```

```
Connection con = DriverManager.getConnection ("jdbc:db2://host:6789/myDB", "login","pw");
```

```
PreparedStatement pstmt=
```

```
    con.prepareStatement("SELECT vorname, name, matrikel FROM Student WHERE studiengang= ?");
```

DECLARE

```
pstmt.setString(1, "Informatik");
```

```
ResultSet rs= pstmt.executeQuery();
```

OPEN

```
while(rs.next()) {
```

FETCH

```
    String vorname= rs.getString(1);
```

```
    String name= rs.getString(2);
```

```
    int matrikel= rs.getInt(3);
```

```
        System.out.println("vorname: " + vorname + ", nachname: " + name + ", matrikelnr: " + matrikel);
```

```
}
```

```
rs.close();
```

CLOSE