

# TÖL303G

## Gagnasafnsfræði

### Vikublað 8

Snorri Agnarsson

11. október 2022

## Efni vikunnar — The Weeks Material

Við ræðum um notkun SQL inni í forritunarmálum. Þrjár aðferðir eru notaðar:

- Innfelld SQL (embedded SQL, ESQL), svo sem JSQL og fleiri.
- Fallasöfn og klasasöfn svo sem SQL/CLI, ODBC, JDBC.
- Forritunarmál sem innbyggð eru í gagnagrunnana.

Aðaláherslan er á JDBC.

Java forritin `JDBC1.java` og `Sample.java` eru meðal skráa í ZIP skránni `forritun.zip` í Canvas og ættu að vera gagnleg.

Einnig er í möppunni skipanaskráin `companysqlite.sql` sem nota má til að byggja COMPANY gagnagrunninn í SQLite, auk þess sem skráin `company.db` inniheldur COMPANY gagnagrunninn. Þá er jar-skrá í möppunni sem inniheldur JDBC klasa fyrir SQLite. Slíka jar-skrá (e.t.v. nýrri) má einnig finna á vefsíðum SQLite.

Athugið að til þess að unnt sé að nota ofangreind forrit þarf fyrst að byggja COMPANY gagnagrunn í viðeigandi gagnagrunnskerfi.

Almenna skjölun fyrir JDBC má finna út um allt á vefnum, meðal annars hjá Oracle<sup>1</sup>.

We will discuss using SQL within programming languages. Three methods are used:

- Embedded SQL, ESQL, such as JSQL and more.

---

<sup>1</sup><http://docs.oracle.com/javase/6/docs/technotes/guides/jdbc/>

- Function libraries and class libraries such as SQL/CLI, ODBC, JDBC.
- Programming languages that are built into databases (stored procedures).

The main emphasis is on JDBC.

The Java programs `JDBC1.java` and `Sample.java` are among files in the ZIP file `forritun.zip` in Canvas and should be useful.

Also contained in the folder is the command file `companysqlite.sql` which can be used to build the COMPANY database in SQLite, and the file `company.db` contains the COMPANY database. There is also a jar-file in the folder that contains JDBC classes for SQLite. Such a jar-file (perhaps newer) can also be found on the web pages for SQLite.

Note that to be able to use the aforementioned programs you first need to build the COMPANY database in the database management system in question.

General documentation for JDBC can be found all over the web, among other places at Oracle<sup>2</sup>.

## Verkefni — Assignments

Verkefnin sem skila skal í Gradescope eru eftirfarandi. Munið einnig Gradiance verkefnin.

The assignments that are to be turned in to Gradescope are the following. Also remember the Gradiance exercises.

1. Skrifðu forrit `V8a.java` í Java sem skrifar meðallaun allra starfsmanna í COMPANY gagnagrunninum, þ.e. meðallaun fyrirtækisins, sem er ein tala. Sýnið þá tölu.

Write a program `V8a.java` in Java that writes the average salary of all employees in the COMPANY database, i.e. the average salary of the company, which is one number. Show that number.

2. Skrifðu forrit `V8b.java` í Java sem gefur öllum í COMPANY gagnagrunninum 200 dala launahækkun. Forritið skal finna allar kennitölur (ssn) og beita runu UPDATE skipana sem hver um sig verkar fyrir eina kennitölu. Notið Statement fyrir fyrirspurnina og PreparedStatement fyrir UPDATE skipunina.

Write a program `V8b.java` in Java that gives everyone in the COMPANY database a 200 dollar salary raise. The program should find everyones social security number (ssn) and perform a sequence of UPDATE commands that each works for one ssn. Use a Statement for the query and a PreparedStatement for the updates.

---

<sup>2</sup><http://docs.oracle.com/javase/6/docs/technotes/guides/jdbc/>

3. Skrifðu forrit `V8c.java` í Java sem gefur öllum í `EMPLOYEE` töflunni í `COMPANY` gagnagrunninum 3% launahækkun. Forritið skal nota `SQL UPDATE` skipun sem verkar í heildsölu og breytir launum allra samtímis í einni skipun.

Write a program `V8c.java` in Java that gives everyone in the `EMPLOYEE` table in the `COMPANY` database a 3% raise of salary. The program should use an `SQL UPDATE` command that works in wholesale and changes everyones salary concurrently in one command.

4. Keyrið forritin einu sinni (hvert) á `COMPANY` gagnagrunninn ykkar og notið síðan forritið `JDBC1.java` sem er í Canvas til að finna launin fyrir James E. Borg (kennitala 888665555). Fyrir breytingarnar eru laun hans 55000. Eftir að þið keyrið bæði ykkar forrit ættu þau því að vera  $(55000 + 200) \times 1.03 = 56856$ . Sýnið þessa keyrslu, þ.e. textann á skipanalínunni. Sýnið einnig niðurstöðu lokakeyrslu á `V8a` sem sýnir meðallaunin eftir allar breytingarnar.

Run the programs once (each) on your `COMPANY` database and then use the program `JDBC1.java` which is in Canvas to find the salary of James E. Borg (ssn 888665555). Before the changes his salary is 55000. After you run both your program is should therefore be  $(55000 + 200) \times 1.03 = 56856$ . Show this run, i.e. the text on the command line. Show also the result from a final run of `V8a` which shows the average salary after changes.

Athugið að ef eitthvað fer úrskeiðis þá ættuð þið að geta endurbyggt gagnagrunninn með því að nota `SQL` skipanirnar í `companysqlite.sql`. Einnig má fá skrána `company.db` í upphaflegu ástandi úr Canvas.

Note that if something goes wrong you should be able to rebuild the database by using the `SQL` commands in `companysqlite.sql`. Also the original file `company.db` can be fetched from Canvas.