



Fußballverein

Dokumentation zur Semesteraufgabe



Inhaltsverzeichnis

- Einführung
- Angabe
- Kompetenzbereiche
- ER-Diagramm
- Einzelnen Schritte
- Abschließende Worte
- Quellen

Einführung

Diese Aufgabe muss von mir gemacht werden, da ich im ersten Semester ein paar Kompetenzbereiche nicht erfüllt habe.

Das ganze Projekt ist so aufgebaut, dass ich als erstes, ganz grob, ein Plan gemacht habe (handschriftlich) wie ich was angehe.

Zuerst wurde das ERD entwickelt, anschließend habe ich mir die CREATES rasuexportiert.

Bei der Erstellung der Datenbank habe ich ein paar Probleme gehabt, die ich später noch zeigen werde.

Die Dokumentation wurde sogar im PASSIV geschrieben, wodurch ein zusätzlicher Kompetenzbereich in Deutsch abgedeckt wird.

Angabe für die Datenbank

Ein österreichischer Fußballverein braucht eine neue Datenbank, um zumindest die Personenverwaltung auf eine professionelle Basis zu stellen. In dieser Datenbank werden folgende Daten verwaltet:

Jede Person ist identifiziert durch eine eindeutige Personalnummer (kurz "persnr"). Außerdem werden zu jeder Person folgende Informationen verwaltet: Vorname ("vname"), Nachname ("nname"), Geschlecht ("geschlecht") und Geburtsdatum ("gebdat"). Für "geschlecht" sind einzig die Eingaben "W", "M" und "N" gültig.

Jede Person, die am Vereinsleben beteiligt ist, gehört zu genau einer der folgenden 4 Kategorien: Angestellter, Vereinsmitglied (kurz "Mitglied"), Spieler oder Trainer. Für all diese Kategorien von Personen müssen zusätzliche Informationen verwaltet werden:

Von jedem Angestellten werden das Gehalt ("gehalt"), die Überstunden ("ueberstunden") und die E-Mail-Adresse ("e_mail") vermerkt.

Für jedes Vereinsmitglied werden ausständige Beiträge ("beitrag") abgespeichert.

Jeder Spieler hat eine Position ("position") (z.B. Tormann, Stürmer, etc.). Außerdem sind Gehalt ("gehalt") sowie Beginn ("von") und Ende ("bis") der Vertragsdauer abzuspeichern. Jeder Spieler ist zumindest einer Mannschaft zugeordnet. Die Spieler innerhalb einer Mannschaft müssen jeweils eine eindeutige Trikot-Nummer ("nummer") zwischen 1 und 99 haben.

Jeder Trainer hat ein Gehalt ("gehalt") sowie Beginn ("von") und Ende ("bis") der Vertragsdauer. Jeder Trainer ist genau einer Mannschaft zugeordnet.

Der Verein hat mehrere Mannschaften. Für jede Mannschaft sind folgende Informationen zu verwalten: eine eindeutige Bezeichnung ("bezeichnung") (wie 1. Mannschaft, Junioren, A-Jugend, ...), die Klasse ("klasse") sowie das Datum des nächsten Spiels ("naechstes_spiel"). Jede Mannschaft hat mindestens 11 Spieler, genau einen Chef-Trainer und genau einen Trainer-Assistenten. Beide sind als Trainer des Vereins registriert. Außerdem hat jede Mannschaft einen Kapitän (der ein Spieler des Vereins ist).

Der Verein hat mehrere Standorte. Jeder Standort ist identifiziert durch eine eindeutige ID ("sid"). Außerdem sind zu jedem Standort folgende Informationen verfügbar: Land ("land"), Postleitzahl ("plz") und Ort ("ort").

An jedem Standort gibt es 1 oder mehrere Fan-Clubs. Jeder Fan-Club hat einen für den jeweiligen Standort eindeutigen Namen ("name"), ein Gründungsdatum ("gegrundet") und einen Obmann ("obmann"), der Vereinsmitglied sein muss. Ein Vereinsmitglied kann von höchstens einem Fan-Club der Obmann sein. Die Fan-Clubs werden von Angestellten des Vereins betreut: Und zwar kann jeder Angestellte einen Fan-Club jeweils für einen gewissen Zeitraum betreuen. Dieser Zeitraum ist durch Anfangsdatum ("anfang") und Endedatum ("ende") bestimmt. Jeder Angestellte kann 0 oder mehrere Fan-Clubs betreuen, und jeder Fanclub kann von einem oder mehreren Angestellten betreut werden.

Der Verein führt Aufzeichnungen über die absolvierten Spiele. Jedes Spiel ist gekennzeichnet durch die Bezeichnung der Mannschaft ("mannschaft") und das Datum ("datum"). Für jedes Spiel werden der Gegner ("gegner") und das Ergebnis ("ergebnis") eingetragen, das einen der Werte "Sieg", "Unentschieden" oder "Niederlage" haben kann. Außerdem werden zu jedem Spiel die beteiligten Spieler abgespeichert, wobei für jeden Spieler die Dauer Einsatzes ("dauer") als Wert zwischen 1 und 90 vermerkt wird.

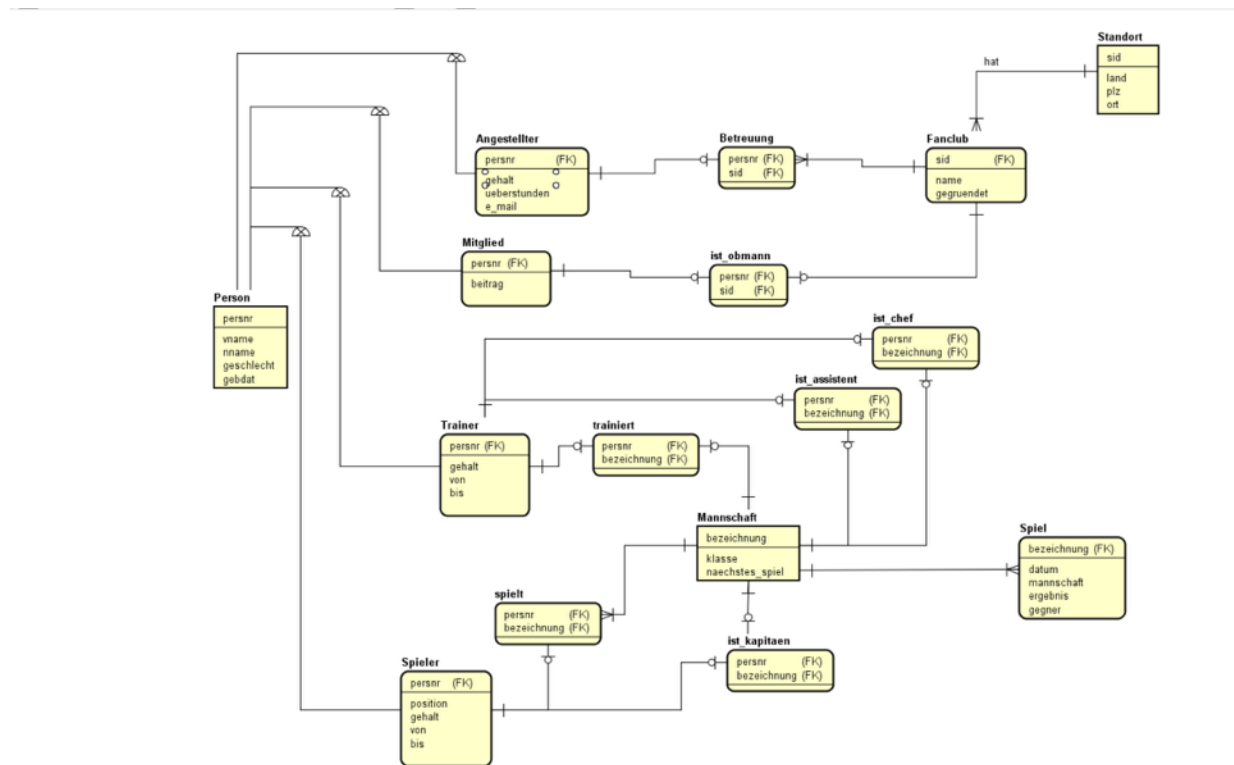
Kompetenzbereiche

-komplexe Abfragen für konkrete Problemstellungen entwickeln und optimieren.

-Anwendungssysteme unter Verwendung von Nebenläufigkeit entwickeln.

-Einfache Schnittstellen zur Kommunikation zwischen Anwendungen entwerfen und implementieren

ERD



SELECT-BEFEHLE

```
S1) SELECT sid, name FROM Betreuung WHERE CURRENT_DATE != BETWEEN
Betreuung.anfang AND Betreuung.ende;
```

Die Select Abfrage wurde nicht getestet, weil Betreuung in meiner Datenbank keine Anfangs- und Enddaten enthalten.

S2) `SELECT person.persnr, person.nname FROM Betreuung
ORDER BY nname ASC;`

S3)

S4)

S5)

```
S6) CREATE VIEW Trainer_view AS SELECT person.persnr, person.vname,
person.nname, person.geschlecht, person.gebdat, trainer.gehalt, trainer.von,
trainer.bis FROM trainer NATURAL JOIN person;
DROP VIEW Trainer_view;
```

```
You are now connected to database "fussballverein" as user "postgres".
fussballverein=# CREATE VIEW Trainer_view AS SELECT person.persnr, person.vname,
  person.nname, person.geschlecht, person.gebdat, trainer.gehalt, trainer.von, tr
  ainer.bis FROM trainer NATURAL JOIN person;
CREATE VIEW
fussballverein=# DROP VIEW Trainer_view;
ERROR:  syntax error at or near " "
LINE 1:
      ^
fussballverein=# DROP VIEW Trainer_view;
DROP VIEW
fussballverein=#
```

Vorgangsweise

1. Verstehen

Angabe wurde abgecheckt.

Anschließend wurde alles markiert, was ich leicht umsetzen konnte und was nicht.

So wie bei jeder Aufgabe habe ich bei manchen Sachen länger gebraucht, als bei anderen.

Ein österreichischer Fußballverein braucht eine neue Datenbank, um zumindest die Personenverwaltung auf eine professionelle Basis zu stellen. In dieser Datenbank werden folgende Daten verwaltet:

Jede Person ist identifiziert durch eine eindeutige Personalnummer (kurz "persnr"). Außerdem werden zu jeder Person folgende Informationen verwaltet: Vorname ("vname"), Nachname ("nname"), Geschlecht ("geschlecht") und Geburtsdatum ("gebdat"). Für "geschlecht" sind einzig die Eingaben "W", "M" und "N" gültig.

Jede Person, die am Vereinsleben beteiligt ist, gehört zu genau einer der folgenden 4 Kategorien: Angestellter, Vereinsmitglied (kurz "Mitglied"), Spieler oder Trainer. Für all diese Kategorien von Personen müssen zusätzliche Informationen verwaltet werden: Von jedem Angestellten werden das Gehalt ("gehalt"), die Überstunden ("ueberstunden") und die E-Mail-Adresse ("e_mail") vermerkt. Für jedes Vereinsmitglied werden ausstehende Beiträge ("beitrag") abgespeichert. Jeder Spieler hat eine Position ("position") (z.B. Tormann, Stürmer, etc.). Außerdem sind Gehalt ("gehalt") sowie Beginn ("von") und Ende ("bis") der Vertragsdauer abzuspeichern. Jeder Spieler ist zumindest einer Mannschaft zugeordnet. Die Spieler innerhalb einer Mannschaft müssen jeweils eine eindeutige Trikot-Nummer ("nummer") zwischen 1 und 99 haben. Jeder Trainer hat ein Gehalt ("gehalt") sowie Beginn ("von") und Ende ("bis") der Vertragsdauer. Jeder Trainer ist genau einer Mannschaft zugeordnet.

Der Verein hat mehrere Mannschaften. Für jede Mannschaft sind folgende Informationen zu verwalten: eine eindeutige Bezeichnung ("bezeichnung") (wie 1. Mannschaft, Junioren, A-Jugend, ...), die Klasse ("klasse") sowie das Datum des nächsten Spiels ("naechstes_spiel"). Jede Mannschaft hat mindestens 11 Spieler, genau einen Chef-Trainer und genau einen Trainer-Assistenten. Beide sind als Trainer des Vereins registriert. Außerdem hat jede Mannschaft einen Kapitän (der ein Spieler des Vereins ist).

Der Verein hat mehrere Standorte. Jeder Standort ist identifiziert durch eine eindeutige ID ("sid"). Außerdem sind zu jedem Standort folgende Informationen verfügbar: Land ("land"), Postleitzahl ("plz") und Ort ("ort").

An jedem Standort gibt es 1 oder mehrere Fan-Clubs. Jeder Fan-Club hat einen für den jeweiligen Standort eindeutigen Namen ("name"), ein Gründungsdatum ("gegruendet") und einen Obmann ("obmann"), der Vereinsmitglied sein muss. Ein Vereinsmitglied kann von höchstens einem Fan-Club der Obmann sein. Die Fan-Clubs werden von Angestellten des Vereins betreut: Und zwar kann jeder Angestellte einen Fan-Club jeweils für einen gewissen Zeitraum betreuen. Dieser Zeitraum ist durch Anfangsdatum ("anfang") und Enddatum ("ende") bestimmt. Jeder Angestellte kann 0 oder mehrere Fan-Clubs betreuen, und jeder Fanclub kann von einem oder mehreren Angestellten betreut werden.

Der Verein führt Aufzeichnungen über die absolvierten Spiele. Jedes Spiel ist gekennzeichnet durch die Bezeichnung der Mannschaft ("mannschaft") und das Datum ("datum"). Für jedes Spiel werden der Gegner ("gegner") und das Ergebnis ("ergebnis") eingetragen, das einen der Werte "Sieg", "Unentschieden" oder "Niederlage" haben kann. Außerdem werden zu jedem Spiel die beteiligten Spieler abgespeichert, wobei für jeden Spieler die Dauer Einsatzes ("dauer") als Wert zwischen 1 und 90 vermerkt wird.

2. Erstellung des ERD

Bei der Erstellung des Diagramms wurde bisschen mehr Zeit benötigt. Bin mir nicht sicher ob, so wie es von mir realisiert wurde auch richtig ist, da jeder aus unserer Klasse eine andere Lösung hat.

3. Hier werden Sie wahrscheinlich erwarten, dass über die Erstellung der Datenbank und über die Inserts berichtet wird, jedoch kann dazu von mir nichts mitgeteilt werden, da ich Probleme mit meinem Debian hatte.

Folgendes **Problem** ist aufgetreten:

```
root@debian:~# ifconfig
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 0 (Local Loopback)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Nur das Default-Interface ist „da“.

```
root@debian:~# mysqldump -u root -p --all-databases > dump_all.sql
Enter password:
root@debian:~# ls -la
total 1692
drwx----- 7 root root 4096 Apr 8 11:39 .
drwxr-xr-x 23 root root 4096 Oct 21 08:38 ..
-rw-r--r-- 1 root root 0 Apr 6 13:07 all_databases_dump.sql
-rw----- 1 root root 6665 Apr 8 11:02 .bash_history
-rw-r--r-- 1 root root 570 Jan 31 2010 .bashrc
-rw-r--r-- 1 root root 1672097 Apr 8 11:39 dump_all.sql
drwxr-xr-x 2 root root 4096 Oct 20 15:05 meinordner
-rw----- 1 root root 491 Apr 6 12:25 .mysql_history
drwxr-xr-x 2 root root 4096 Oct 21 11:35 .nano
-rw-r--r-- 1 root root 148 Aug 17 2015 .profile
drwxr-xr-x 2 root root 4096 Mar 7 14:24 slog
drwxr-xr-x 2 root root 4096 Mar 7 14:23 snort
drwxr-xr-x 2 root root 4096 Oct 21 13:04 .vim
-rw----- 1 root root 7362 Apr 6 12:21 .viminfo
root@debian:~# mv dump_all.sql /mnt/usb/
root@debian:~#
```

Andererseits ist das Interface eno16777736 wieder da.

```
root@debian:~# cat /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-hotplug eno16777736
iface eno16777736 inet dhcp
root@debian:~#
```

Lösung:

Alles in MySql dumpen. (In PostgreSQL hatte ich nichts zu dumpen)

Export

```
mysqldump -u root -p --all-databases > name.sql
```

Import

```
mysql -u root -p < alldb.sql
```

Was muss gemacht werden?

-USB-Stick muss gemountet werden

-Ordner muss erstellt werden

```
root@debian:~# mkdir /mnt/usb
root@debian:~# _
```

```
root@debian:~# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda          8:0    0   20G  0 disk
├─sda1       8:1    0  19.1G  0 part /
├─sda2       8:2    0    1K  0 part
└─sda5       8:5    0   880M  0 part [SWAP]
sdb          8:16    1  14.6G  0 disk
└─sdb1       8:17    1  14.6G  0 part /mnt/usb
sr0         11:0    1 1024M  0 rom
root@debian:~#
```

Anschließend wurde die Dump-Datei auf meinen USB Stick gegeben und mein Debian gelöscht.

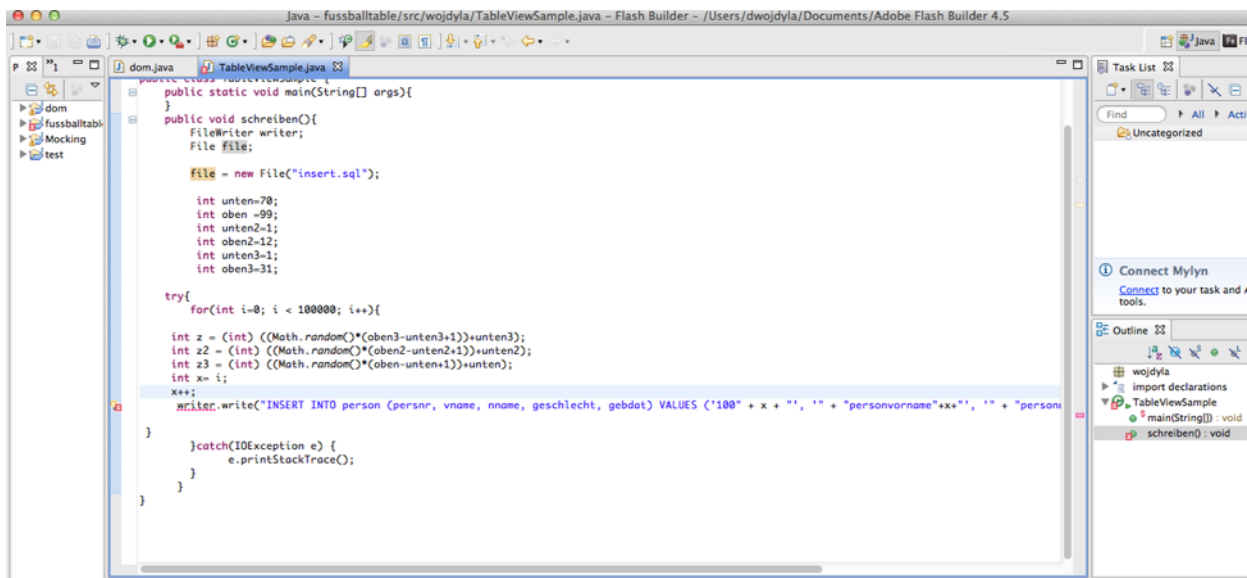
Hier meine neue ISO —>



4.Inserts

Die Inserts wurden in Java erstellt.

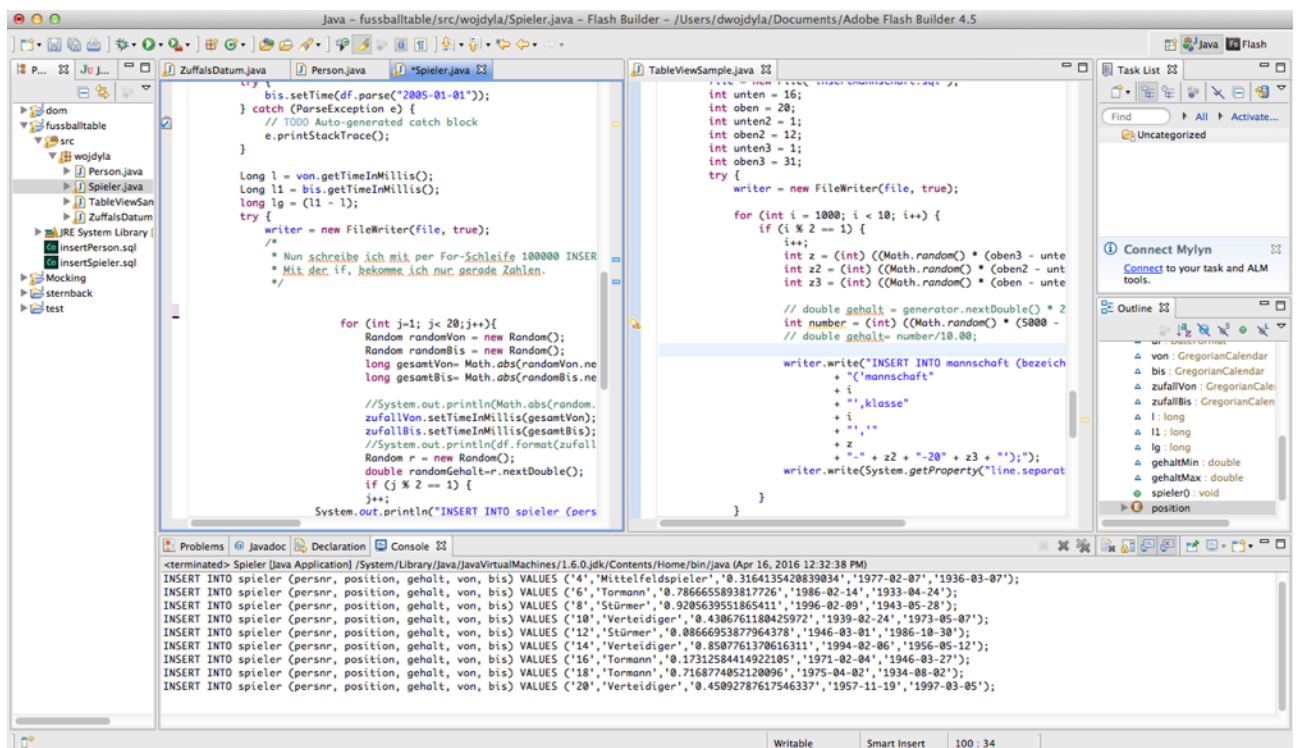
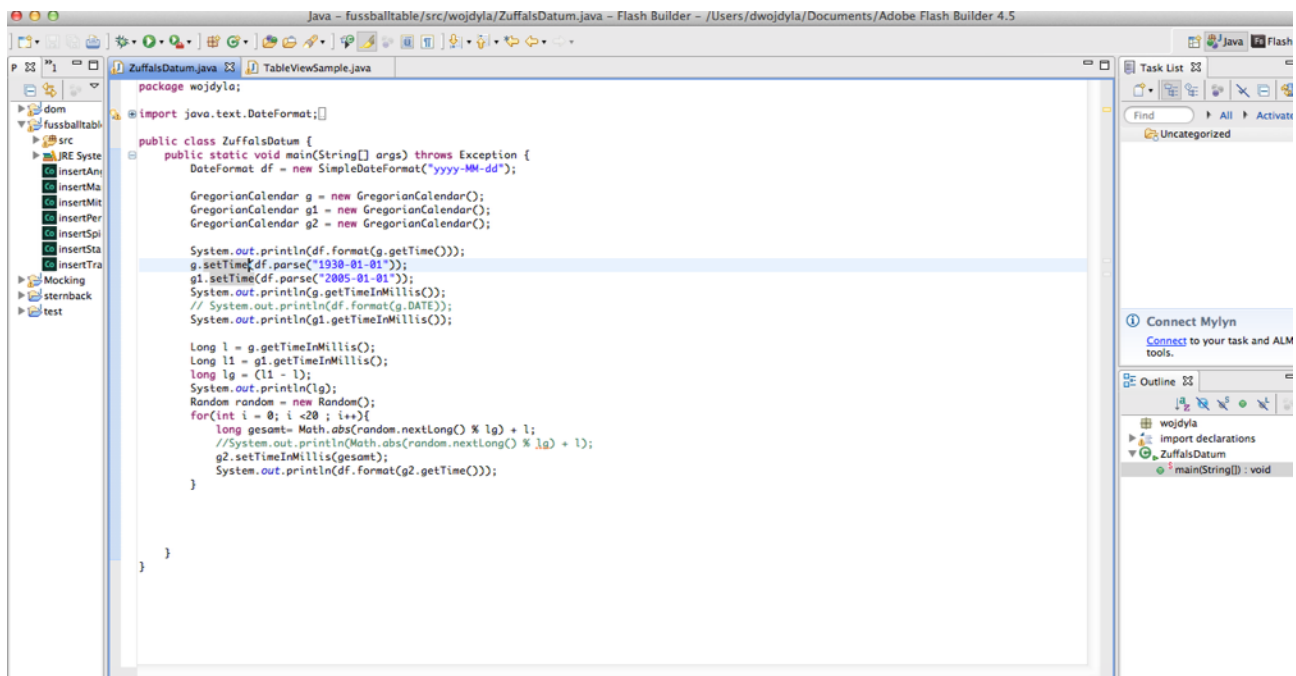
Der Datafiller wurde von mir nicht benutzt, da ich mir alle INSERTS selber schreiben wollte.



Am Anfang wurde alles in eine Klasse geschrieben, was natürlich ein Fehler war, da ich die Übersicht schnell verloren habe. Das nächste Dilemma war das Datum. bei der Version 1.0, wurde das Datum mit Random Zahlen gefüllt. Also 3 verschiedene Random Zahlen, die mit einem "-" dazwischen in das SQL File geladen wurde. Bei der Version 2.0, die gleichzeitig auch die Finale Version für das Datum war, wurde ein Gregoriancalendar erstellt. Das Anfangsdatum wurde auf den 01.01.1930. Dies wurde auch mit dem Enddatum gemacht, welches für mich 01.01.2005 war. Mit SDF (SimpleDateFormat) wurde das Format (yyy-MM-dd) gesetzt.

```
von.setTime(df.parse("1930-01-01"));
```


Von den beiden Daten holte ich mir die Millisekunden und rechnete mir dann davon eine Random Millisekunde aus, die zwischen dem „Von Datum“ und zwischen dem „Bis Datum“ lag. Anschließend wandelte ich diese in ein normales Datum wieder um.

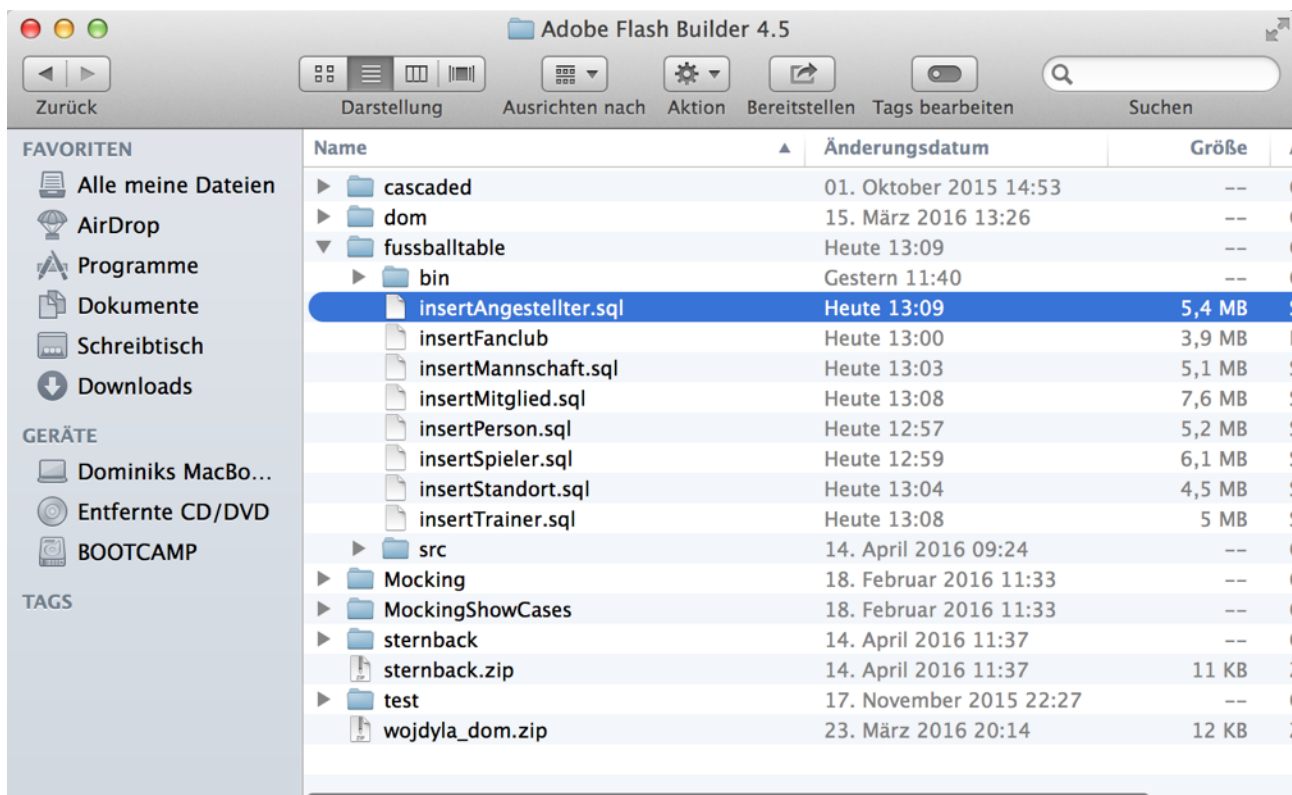


Das ganze wurde in einer Testklasse davor natürlich getestet. Damit bei der Ausgabe bzw. bei der Abspeicherung in ein SQL File, das „Von Datum“ nicht größer ist, als das „Bis Datum“, machte ich eine If-Anweisung über die Ausgabe.

```
if (gesamtVon < gesamtBis){
    zufallVon.setTimeInMillis(gesamtVon);
    zufallBis.setTimeInMillis(gesamtBis);
writer.write("INSERT INTO spieler (persnr, position, gehalt, von, bis) VALUES ('"+j+"','"+zufallsPosition()+"','"+randomGehalt+"','"+df.for
writer.write(System.getProperty("line.separator"));
}else{
    long temp = gesamtBis;
    long temp2 = gesamtVon;
    gesamtVon = temp;
    gesamtBis = temp2;
    zufallVon.setTimeInMillis(gesamtVon);
    zufallBis.setTimeInMillis(gesamtBis);
    writer.write("INSERT INTO spieler (persnr, position, gehalt, von, bis) VALUES ('"+j+"','"+zufallsPosition()+"','"+randomGehalt+"
    writer.write(System.getProperty("line.separator"));
}
}
```

Mit dem Filewriter erstellte ich mir eine SQL-Datei und speicherte die Inserts für die jeweilige Tabelle dort ab.

Das einzige Problem bzw. die einzige Schwierigkeit die ich bei den Inserts hatte, war das Gehalt zu speichern, deswegen bin ich davon ausgegangen, dass der Fußballverein chillig ist und keine Cent Münzen ausbezahlt.

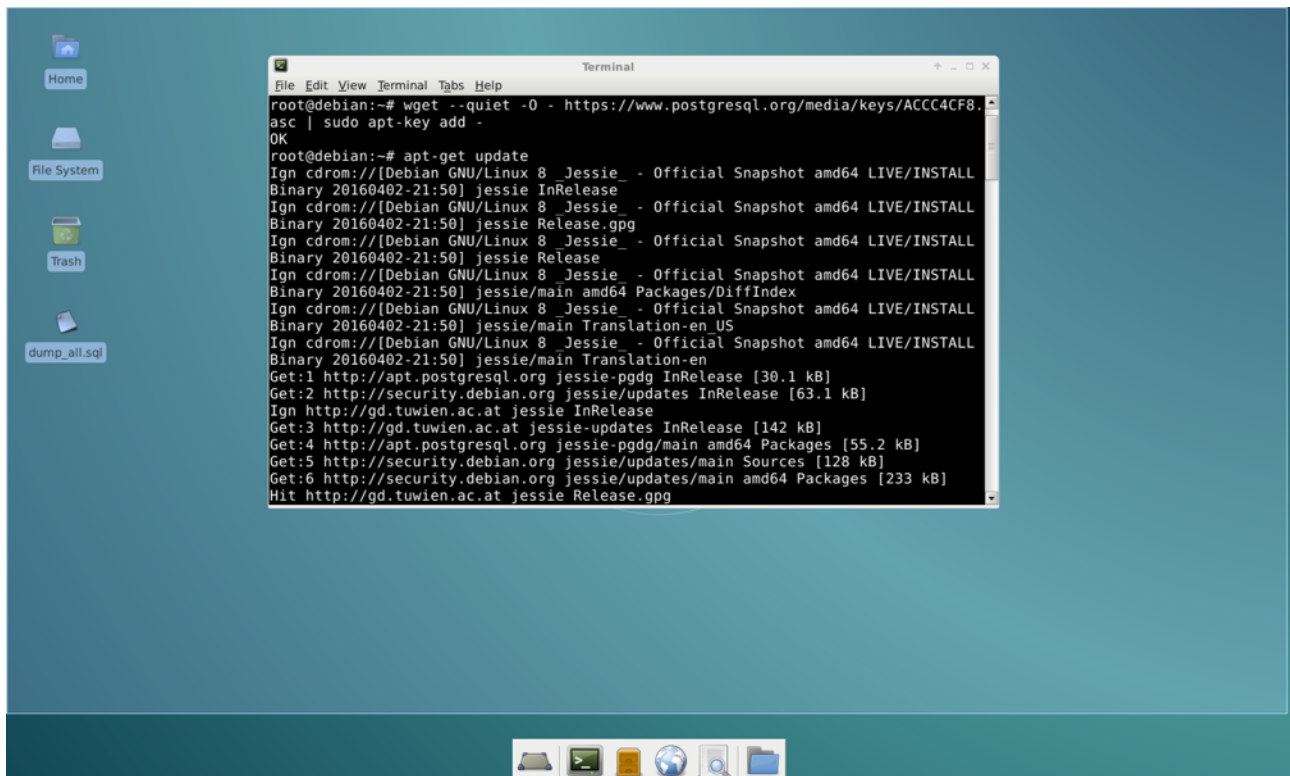


5. Astah Professional Creates von Astah hinaus exportieren



```
1
2
3 CREATE TABLE Mannschaft (
4   bezeichnung VARCHAR(10) NOT NULL,
5   klasse VARCHAR(10) NOT NULL,
6   naechstes_spiel DATE NOT NULL
7 );
8
9 ALTER TABLE Mannschaft ADD CONSTRAINT PK_Mannschaft PRIMARY KEY (bezeichnung);
10
11
12 CREATE TABLE Person (
13   persnr FLOAT(10) NOT NULL,
14   vname VARCHAR(10) NOT NULL,
15   nname VARCHAR(10) NOT NULL,
16   geschlecht CHAR(10) NOT NULL,
17   gebdat DATE NOT NULL
18 );
19
20 ALTER TABLE Person ADD CONSTRAINT PK_Person PRIMARY KEY (persnr);
21
22
23 CREATE TABLE Spiel (
24   bezeichnung VARCHAR(10) NOT NULL,
25   datum DATE,
26   mannschaft CHAR(10),
27   ergebnis VARCHAR(10),
28   gegner VARCHAR(10)
29 );
30
31 ALTER TABLE Spiel ADD CONSTRAINT PK_Spiel PRIMARY KEY (bezeichnung);
32
33
34 CREATE TABLE Spieler (
35   persnr FLOAT(10) NOT NULL,
36   position VARCHAR(10) NOT NULL,
37   gehalt DECIMAL(10) NOT NULL,
38   von DATE NOT NULL,
39   bis DATE NOT NULL
40 );
41
42 ALTER TABLE Spieler ADD CONSTRAINT PK_Spieler PRIMARY KEY (persnr);
43
44
45 CREATE TABLE spielt (
46   persnr FLOAT(10) NOT NULL,
47   bezeichnung VARCHAR(10) NOT NULL
48 );
49
50 ALTER TABLE spielt ADD CONSTRAINT PK_spielt PRIMARY KEY (persnr,bezeichnung);
51
```

6. Postgresql installieren



```
root@debian:~# wget --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.
asc | sudo apt-key add -
OK
root@debian:~# apt-get update
Ign cdrom://[Debian GNU/Linux 8 _Jessie_ - Official Snapshot amd64 LIVE/INSTALL
Binary 20160402-21:50] jessie InRelease
Ign cdrom://[Debian GNU/Linux 8 _Jessie_ - Official Snapshot amd64 LIVE/INSTALL
Binary 20160402-21:50] jessie Release.gpg
Ign cdrom://[Debian GNU/Linux 8 _Jessie_ - Official Snapshot amd64 LIVE/INSTALL
Binary 20160402-21:50] jessie Release
Ign cdrom://[Debian GNU/Linux 8 _Jessie_ - Official Snapshot amd64 LIVE/INSTALL
Binary 20160402-21:50] jessie/main amd64 Packages/DiffIndex
Ign cdrom://[Debian GNU/Linux 8 _Jessie_ - Official Snapshot amd64 LIVE/INSTALL
Binary 20160402-21:50] jessie/main Translation-en US
Ign cdrom://[Debian GNU/Linux 8 _Jessie_ - Official Snapshot amd64 LIVE/INSTALL
Binary 20160402-21:50] jessie/main Translation-en
Get:1 http://apt.postgresql.org jessie-pgdg InRelease [30.1 kB]
Get:2 http://security.debian.org jessie/updates InRelease [63.1 kB]
Ign http://gd.tuwien.ac.at jessie InRelease
Get:3 http://gd.tuwien.ac.at jessie-updates InRelease [142 kB]
Get:4 http://apt.postgresql.org jessie-pgdg/main amd64 Packages [55.2 kB]
Get:5 http://security.debian.org jessie/updates/main Sources [128 kB]
Get:6 http://security.debian.org jessie/updates/main amd64 Packages [233 kB]
Hit http://gd.tuwien.ac.at jessie Release.gpg
```

7. User erstellen

Hier wurde der User erstellt, mit dem ich auf die Datenbank zugreifen will.

```
postgres=# CREATE USER soccerdomi WITH PASSWORD 'user';
CREATE ROLE
postgres=#
```

```

List of roles
Role name | Attributes | Member of
-----+-----+-----
postgres | Superuser, Create role, Create DB, Replication, Bypass RLS | {}
soccerdomi | | {}
(END)

```

Nun kann ich mit GRANT ALL dem User die Rechte geben.

```
postgres=# CREATE USER soccerdomi PASSWORD 'soccerdomi';
CREATE ROLE
postgres=# CREATE DATABASE fussballverein;
ERROR:  database "fussballverein" already exists
postgres=# DROP DATABASE fussballverein;
DROP DATABASE
postgres=# CREATE DATABASE fussballverein;
CREATE DATABASE
postgres=# GRANT ALL ON DATABASE fussballverein TO soccerdomi;
GRANT
```

Folgendes Problem ist aufgetreten:

```
root@debian:~# sudo -u postgres psql -c "ALTER USER soccerdomi PASSWORD 'soccerdomi';"
could not change directory to "/root": Permission denied
ALTER ROLE
root@debian:~# su postgres
postgres@debian:/root$ su -soccerdomi
Password:
su: Authentication failure
postgres@debian:/root$
```

Beim Versuch sich anzumelden, habe ich immer ein Authentication failure bekommen. Das Passwort wurde von mir einige Male, auf verschiedene Art und Weise geändert, aber es hat nichts gebracht. Deswegen habe ich die Datenbank mit dem standard postgres User bearbeitet.

8. Creates in die Datenbank laden.

```
1 CREATE TABLE Entity11 (  
2 );  
3  
4  
5 CREATE TABLE Mannschaft (  
6 bezeichnung VARCHAR(30) NOT NULL,  
7 klasse VARCHAR(30) NOT NULL,  
8 naechstes_spiel DATE NOT NULL  
9 );  
10  
11 ALTER TABLE Mannschaft ADD CONSTRAINT PK_Mannschaft PRIMARY KEY (bezeichnung);  
12  
13  
14 CREATE TABLE Person (  
15 persnr FLOAT(30) NOT NULL,  
16 vname VARCHAR(30) NOT NULL,  
17 nname VARCHAR(30) NOT NULL,  
18 geschlecht CHAR(30) NOT NULL,  
19 gebdat DATE NOT NULL  
20 );  
21  
22 ALTER TABLE Person ADD CONSTRAINT PK_Person PRIMARY KEY (persnr);  
23  
24  
25 CREATE TABLE Spiel (  
26 bezeichnung VARCHAR(30) NOT NULL,  
27 datum DATE,  
28 mannschaft CHAR(30),  
29 ergebnis VARCHAR(30),  
30 gegner VARCHAR(30)  
31 );  
32  
33 ALTER TABLE Spiel ADD CONSTRAINT PK_Spiel PRIMARY KEY (bezeichnung);  
34  
35  
36 CREATE TABLE Spieler (  
37 persnr FLOAT(30) NOT NULL,  
38 position VARCHAR(30) NOT NULL,  
39 gehalt DECIMAL(30) NOT NULL,  
40 von DATE NOT NULL,  
41 bis DATE NOT NULL  
42 );  
43  
44 ALTER TABLE Spieler ADD CONSTRAINT PK_Spieler PRIMARY KEY (persnr);  
45  
46  
47 CREATE TABLE spielt (  
48 persnr FLOAT(30) NOT NULL,  
49 bezeichnung VARCHAR(30) NOT NULL  
50 );  
51  
52 ALTER TABLE spielt ADD CONSTRAINT PK_spielt PRIMARY KEY (persnr, bezeichnung);
```

```
fussballverein=# SELECT table_name FROM information_schema.tables WHERE table_sc  
hema = 'public';  
  table_name  
-----  
spielt  
ist_assistent  
spiel  
trainiert  
ist_obmann  
betreuung  
ist_kapitaen  
fanclub  
ist_chef  
standort  
person  
mannschaft  
trainer  
spieler  
angestellter  
mitglied  
(16 rows)  
  
fussballverein=#
```

```
fussballverein=# SELECT column_name FROM information_schema.columns WHERE table_
name = 'table';
column_name
-----
(0 rows)

fussballverein=# SELECT column_name FROM information_schema.columns WHERE table_
name = 'spieler';
column_name
-----
persnr
position
gehalt
von
bis
(5 rows)

fussballverein=# █
```

9. Threads & IPC

Hier wollte ich folgendes machen:

Ich wollte in QT eine grafische Oberfläche machen und mit Threads von 2 verschiedenen Benutzern auf die Datenbank zugreifen. Mit IPC gäbe es ein Kommunikation zwischen 2 Fenstern.

Abschließende Worte

Auch wenn ich diese Aufgabe nicht fertig geschafft habe, was sich bestimmt auf meinen Noten auswirken wird, denke ich, dass ich einiges dazugelernt habe.

Dadurch, dass ich mich viel mit dem ganzen auseinander gesetzt habe (selbstständig), fällt es mir leichter, mich in Debian/Java/PostgreSQL irgendwie zurecht zu finden und mich zu bewegen.

Ich möchte mit Ihnen ehrlich sein. Es ist mir nicht wichtig der Beste bzw. einer der Besten in SEW oder INSY zu sein. Ich will einfach nur, die Grundkompetenzen in jedem Bereich erfüllen und das machen was ich gerne mache. (Medientechnik) Schließlich bin ich auch deswegen auf das TGM gekommen. Ich werde versuchen das Beste von mir zu geben, aber ich denke die Noten zeigen einfach nur wie sehr man sich für ein Fach interessiert. Ich hatte andere Erwartungen, als ich auf das TGM kam und als ich letztes Jahr Medientechnik gewählt habe, habe ich das aus einem bestimmten Grund getan.

Ich sag einfach nur Industrial Design & Audiotechnik.

Mir ist natürlich bewusst, dass auch ein Medientechniker sich mit Datenbanken auskennen & programmieren können muss. Ich verstehe nur nicht wieso wir so wenig von der ganzen „Medientechnik“ machen. Wieso ist das ganze so aufgeteilt, dass wir im Labor nichts machen und wenn wir was machen dann nur oberflächlich, weil wir zu wenig Zeit haben. Das Tonstudio steht das ganze Jahr leer, in der Fotografie haben wir nur gelernt wie man die Blende, ISO und die Belichtungszeit einstellt, wenn ich letztes Jahr nicht das Freifach Film genommen hätte, dann wüsste ich jetzt noch immer nicht, wie ein Film aufgebaut ist.

Ich verstehe einfach nicht, wieso das TGM im Bereich Medientechnik so wenig zu bieten hat.

Sie können sich natürlich denken, mit welcher Motivation ich jeden Tag in die Schule komme, wissend, dass es in Prinzip völlig gleichgültig wäre ob ich SYT statt MEDT letztes Jahr gewählt hätte.

Das Wichtigste was ich wegen dem Semesterprojekt dazugelernt habe ist, dass man besser zu früh mit einer Aufgabe anfängt, als zu spät, so wie es in meinem Fall war. Ich tue mir sehr schwer in SEW und INSY und muss immer viel Zeit investieren um etwas hinzubekommen. In diesem Fall habe ich erst nach Ostern beginnen können, weil ich in den Osterferien keine Zugang zum Internet hatte, was natürlich kein Grund sein sollte das Projekt bis Montag 18.04.2016 fertig zu haben.

Ich hoffe Sie geben mir noch einmal die Möglichkeit beweisen zu können, dass ich den Stoff verstanden habe. Es wäre mir ein Anliegen, noch in diesem Semester, meine Kompetenzen unter Beweis zu stellen, da ich vorhabe in GGP vorgezogen zu maturieren.

Quellen

<http://stackoverflow.com/questions/9497869/export-and-import-all-mysql-databases-at-one-time>

<http://dev.mysql.com/doc/refman/5.5/en/mysqldump.html>

<https://wiki.ubuntuusers.de/mv/>

<http://www.computerhope.com/unix/fdisk.htm>

<http://stackoverflow.com/questions/15940862/java-random-number-between-0-and-0-06>

<http://stackoverflow.com/questions/2331513/convert-a-string-to-a-gregoriancalendar>