

Mérési jegyzőkönyv – Szoftver Laboratórium 5

3. mérés: JDBC

Név:	Kovács Levente Ákos
Neptun kód:	CM6UKU
Feladat kódja:	34-TOZSDE
Mérésvezető neve:	Ádám Anna Kinga
Mérés időpontja:	2016-03-31 10:15-12:00
Mérés helyszíne:	HSZK N
A működő alkalmazás elérhetősége:	http://rapid.eik.bme.hu/~cm6uku/jdbc
Megoldott feladatok:	1,2,3,4,5
Elérhető pontszám (plusz pontok nélkül):	50p

Felhasználói útmutató

Az alkalmazás elindítása után szükséges beírni a két felső dobozba a bejelentkezési adatokat

Felhasználónév: cm6uku

Jelszó: Aa36912151

A guin 4 fül látható:

*Az 1. fülön (**Log**) az alkalmazás használatakor keletkező loggolt információk olvashatóak.*

*A 2. fülön (**Search**) a tranzakciók listázása végezhető, ha a szöveg beviteli mezőt üresen hagyjuk akkor kilistáz minden tranzakciót, ha kitöltjük egy 4 jegyű számmal akkor adott évi tranzakciókat listáz ki.*

*A 3. fülön (**Edit**) az adat bevitel és frissítés végezhető. Minden mezőt jó formátumú adatokkal kell feltölteni, ha az adatbázisban van már a megadott részvénytípus akkor frissíti ellenkező esetben beszúr egy új rekordot, ha az autocommit N-re (kikapcsolt) van állítva akkor beilleszt egy tranzakciót is a megadott befektető id-vel. A változtatás véglegesítéséhez a commit gombot kell megnyomni ha az auto commit ki volt kapcsolva legutóbb.*

*A 4.fülön (**Statistics**) a gomb megnyomásával statisztikai adatokat lehet listázni a tranzakciókról.*

Mérési feladatok megoldása

1. feladat

A megoldáshoz használt SQL utasítás

Listázáshoz szükséges sql kód:

```
SELECT b.NEV,rt.CEGNEV,t.MENNYISEG,t.EGYSEGAR
FROM Tranzakcio t
inner join Reszvenytípus rt on rt.RTID=t.RTID
inner join Befekteto b on b.BEFEKTETOID=t.BEFEKTETOID
```

Évszerinti listázáshoz szükséges sql kód:

```
SELECT b.NEV,rt.CEGNEV,t.MENNYISEG,t.EGYSEGAR
FROM Tranzakcio t
inner join Reszvenytípus rt on rt.RTID=t.RTID
inner join Befekteto b on b.BEFEKTETOID=t.BEFEKTETOID
where EXTRACT(YEAR FROM t.Datum)=?
```

Magyarázat

A feladat megoldásához a mintának megadott search fület használtam, annyi változtatással, hogy egy label-t raktam a szövegdoboz elé, ami felhívja a figyelmet a 2 funkcióra. A megoldáshoz a view osztály searcheventhandler()-jét, továbbá a controll és a model osztály search metódusát kellett implementálni, ezek a metódusok ebben a sorrendben hívják egymást. A model.search(...) felel azért hogy adatbázisból kinyerje az adatokat ezeket vissza adja a controllnak, ha szükséges a ? helyére az keresési évet helyettesíti be. A controll osztályban végzek bemenet validációt hogy a megadott keyword csak 4 jegyű szám lehessen és itt végzem a result set átalakítását String listává, így a viewnak nem kell tudnia semmit a visszkapot értékekről a kilistázáshoz. A viewnak visszatört sorokat kirakom a search fülön lévő táblázatba.

2. feladat

A megoldáshoz használt SQL utasítás

Egyediség vizsgálathoz:

```
SELECT rtid
FROM RESZVENYTIPUS rt
WHERE
rtid=?
```

Beillesztéshez:

```
INSERT INTO RESZVENYTIPUS
VALUES (?, ?, ?, ?, ?, ?)
```

Updatehez:

```
UPDATE reszvenytípus set
cegnev=?,
kibocsatas=?,
nevertek=?,
arfolyam=?,
megjegyzes=?
WHERE rtid=?
```

Magyarázat

A feladat megoldásához felvettem új mezőket (6 mezőt) az fxml fájlba és ezeket a megfelelő annotációval változóként a view osztályba, ezek a részvenytípus adatainak beviteléhez szükségesek.

A viewban lévő editeventhandler()-t kellett implementálni, ez meghívta a controll osztályt modifyData függvényét átadva a textfieldekben lévő adatokból képzett mapet. A controll validálta az textfieldbe beírt adatokat, majd ha ez sikerült, meghívta a model osztály modifyData függvényét. A model osztály megnézte, hogy létezik e már ilyen részvenytípus egy lekérdezés segítségével, amennyiben a lekérdezés nem adott

vissza adatot, akkor beszúrást hajtott végre a program, ha talált ilyen sort, akkor pedig frissítést hajtott végre, ezekhez a mapként megkapott adatokat parseolta és a ?-hez rendelte. A visszatérési értékből a controll tudta hogy milyen esemény történt és ez megfelelően logolta.

3. feladat

A megoldáshoz használt SQL utasítás

Nem kellett

Magyarázat

A feladat elvégzéséhez a `controller.verifyData(...)` metódust kellett megvalósítani reguláris kifejezések segítségével.

3 típusu adatot kellett vizsgálnom,

Szám: `\d+ regex-el` a `nevertek,rtid,arfolyam`, (később a befektetőid) mezőkhöz

Dátum: `\d{4}-[0-1]\d-[0-3]\d regex-el` a kibocsátás mezőhöz (Formátumát a guin jeleztem label-en)

Szöveg: `...toString().equals("") || ...toString().trim().isEmpty()` itt csak azt kellett ellenőriznem, hogy van megadva szöveg és nem csupa whitespace, ez a megjegyzés és a cégnév mezőhöz kellett, ehhez nem kellett reguláris kifejezés.

Legördülő panelt nem indokolt egyik adattípusom se, de később az auto commithoz vettem fel.

4. feladat

A megoldáshoz használt SQL utasítás

ID generáláshoz:

```
select max(trid) from tranzakcio
```

Beszúráshoz:

```
insert into TRANZAKCIO (RTID,BEFEKTETOID,MENNYISEG,EGYSEGAR,DATUM,TRID)
values(?, ?, ?, ?,SYSDATE,? )
```

Magyarázat

A megoldáshoz a guira fel kellett venni még egy textfieldet a befektető id nak, továbbá egy commit gombot és hozzá egy comboboxot.

A feladat megoldásához, a 2. feladatban megírt függvényeket kellett kiegészíteni, az modelben ki kellett kapcsolni az autocommitot, ha az volt megadva a legördülő fülön.

Ha meg van adva befektetőid akkor validálom, ha üresen van hagyva akkor az eredeti insert/update funkciót nyújtja a program. Ha megvan adva jó formátumban, továbbá az autocommit ki van kapcsolva és beszúrás történt akkor beszúr egy új tranzakciót is az eddigi bekért adatokkal továbbá a sysdate felhasználásával , az újonnan generált trid-vel ami az eddigi max trid+1 értéket képvisel és mennyiségnek égetve 1 el. Ha valami hiba történik akkor rollbackel a program.

A commit gomb megnyomására commitál a program, amennyiben már előzőleg commitált, úgy nincs hatása a gombnak.

Tesztelés:

MyJwsApplication

cm6uku Aa36912151 Connect Connection created

Log Search Edit Statistics

RTID: 1000 Cegnev: Skynet Kibocsatas(YYYY-MM-DD): 2016-01-15

Neverték: 2000 Arfolyam: 3000 Megjegyzes: T-1000 plans

Befekteto id: 2

Set autocommit N

Commit update/insert

Log Search Edit Statistics

statistics

Cegnev	Atlagos havi tranzakcioszam
EGIS	0.0571
MOL	0.0667
MAV	0.0851
MAGYAR TELEKOM	0.1875
SYNERGON	0.1957
VOLANBUSZ	0.2
OTP	0.2115
Skynet	1.0

Látható hogy a tesztelés eredménye sikeres jól beszúrta a mai dátummal a tranzakciót és természetesen így a havi átlag tranzakciója 1.

Teszt rossz befektetőid vel:

MyJwsApplication

cm6uku Aa36912151 Con... Connection created

Log Search Edit Statistics

RTID: 1500 Cegnev: BefektetoNincs Kibocsatas(YYYY-MM-DD): 2015-10-10

Neverték: 2000 Arfolyam: 3000 Megjegyzes: rollback

Befekteto id: 32115

Set autocommit N

Commit update/insert

logba kiírta hogy rollback occurred, és sql developerben megnézve nem vette fel a részvénytípus sem.

4.2

Ha megszakítjuk commit nélkül a program és kilépünk akkor elvesznek az előző commit óta beszűrt adatok/módosítások.

Ennek orvoslására a view osztályban az init függvényben onclose requestek közé felvettem a commit végrehajtását. Így már nem tűnnek el az adatok ha bezárjuk az alkalmazást, természetesen ez hirtelen áramszünet esetén/hálózati hibától nem véd meg minket.

5. feladat

A megoldáshoz használt SQL utasítás

```
Select rt.rtid,rt.cegnev,  
  
nvl(round(count(*)/Decode(round(MONTHS_BETWEEN(sysdate,MIN(t.datum))), 0,  
1,MONTHS_BETWEEN(sysdate,MIN(t.datum))),4),0)  
  
as tranzakcioszam  
  
FROM Tranzakcio t right join Reszvenytípus rt on t.RTID=rt.RTID  
  
left Outer join Befekteto b on t.BEFEKTETOID=b.BEFEKTETOID  
  
where nvl(b.KESZPENZ,10000000)>1000000  
  
group by rt.rtid,rt.cegnev order by 3
```

Magyarázat

Az előre gyártott statictis fül használható volt változtatás nélkül, a view kódjában a map kulcsait és oszlopneveit kellett csak megváltoztatni.

Kód szinten ez a viewban egy újabb event handler megvalósítását jelentette továbbá, a controller és a model getStatistics() függvényének implementálását

A feladat egy kicsit komplexebb lekérdezés megírását igényelte ami a gazdag részvényesek általi tranzakciókat figyelembe véve adott részvények átlagos havi tranzakciószámát listázta ki azokat is kilistázva akiknél ez 0.

A decodera azért volt szükség mert az eltelt hónapok töredékre pontosan vannak megadva a months_between által, de tervezői döntésként meghoztam hogy az egy perce beszűrt tranzakciónak ne legyen átlagos havi forgalma felszorozva addig amíg 1 nél kisebb az eltelt hónapok száma így nem ad a frissen beszűrt tranzakció se több ezres fals havi forgalmat.

Where ben az nvl re pedig azért hogy megmaradjon a 0 db tranzakciós rt is.

ID megjelenítését is fontosnak találtam hiszen a név nem feltétlen egyedi.

Vélemény(ek) a mérésről

Nagyon időigényes volt