

JEGYZŐKÖNYV

GYAKORLAT CÍME

GYAKORLAT ADATOK	
NÉV	Csatlós Tamás Péter
NEPTUN KÓD	TALXCE
HALLGATÓ TANSZÉKE/SZAKIRÁNYA	IIT, RENDSZERFEJLESZTÉS
GYAKORLATVEZETŐ	Dr. Lengyel László
GYAKORLAT HELYE	IL206
GYAKORLAT IDŐPONTJA	2016.03.21.
GYAKORLAT SORSZÁMA	1.

FELADATLISTA	
1. FELADAT	<input type="checkbox"/>
2. FELADAT	<input type="checkbox"/>
3. FELADAT	<input type="checkbox"/>
4. FELADAT	<input type="checkbox"/>
5. FELADAT	<input type="checkbox"/>
6. FELADAT	<input type="checkbox"/>
7. FELADAT	<input type="checkbox"/>
8. FELADAT	<input type="checkbox"/>
9. FELADAT	<input type="checkbox"/>
10. FELADAT	<input type="checkbox"/>
11. FELADAT	<input type="checkbox"/>

FELADATOK

1.FELADAT

Feladat neve és leírása:

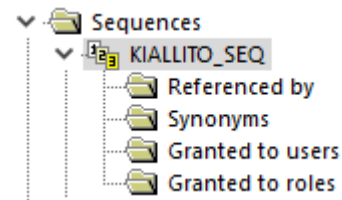
Hozz létre egy szekvenciát a Számla kiállító tábla azonosítójának generálásához!

Megoldás:

```
create sequence kiallito_seq  
start with 1000 increment by 1;
```

Indoklás:

Ezzel a kóddal 1000-től kezdődően lehet egyedi azonosítókat, kulcsokat generálni a Számla kiállító tábla rekordjaihoz.



2.FELADAT

Feladat neve és leírása:

Írj tárolt eljárást, amivel fel lehet venni egy új számla kiállítót. Az eljárás bemenő paraméterei legyenek a szükséges információk (név, irányítószám, város, utca, adószám, számlaszám), az azonosítót pedig az eljárásan belül az előző feladatban létrehozott szekvenciával generáld!

Megoldás:

```
create or replace procedure uj_szamlakiallito_sp  
(pnev nvarchar2,  
pirszam char,  
pvaros nvarchar2,  
putca nvarchar2,  
padoszam nvarchar2,  
pszamlaszam nvarchar2  
)
```

```
is
```

```
begin  
    insert into szamlakiallito  
    values (kiallito_seq.nextval,  
pnev,pirszam,pvaros,putca,padoszam,pszamlaszam);  
end;
```

Indoklás:

Ez a tárolt eljárás képes új rekordokat beszúrni a Számla Kiállító táblába, bár hibás adatokat nem képes kiszűrni.

Csatlós Tamás Péter - Adatbázis szerver oldali programozása - Informatikai technológiák laboratórium 1.

Példának egy sort beillesztettem a GUI segítségével.

	ID	NEV	IR	VAROS	UTCA	ADOSZAM	SZAMLASZAM
▶ 1	1000	Senki	0012	Budapest	A utca	21121212	231132132
2	1	Regio J?t?k ?ruh?z Kft	1119	Budapest	N?ndorfej?v?ri u. 23	15684995-2-32	259476332-15689799-10020065
3	2	Regio J?t?k ?ruh?z Zrt	1119	Budapest	N?ndorfej?v?ri u. 23	68797867-1-32	259476332-15689799-10020065

3.FELADAT

Feladat neve és leírása:

Egészítsd ki az előző feladatban megírt eljárást egy ellenőrző funkcióval, ami csak akkor vesz fel egy új számla kiállítót, ha annak neve még nem szerepelt az adatbázisban, valamint, ha az adószáma 13 karakter hosszú, és a 9. és 11. karaktere kötőjel. A megoldáshoz használhatod a substr(str,pos,len), az instr(string, substring, position, occurrence) és a length(char) beépített SQL függvényeket. Ha nem lehet felvenni az új adatot, dobj alkalmazáshibát!

Megoldás:

```
create or replace procedure uj_szamlakiallito_sp
(pnev nvarchar2,
pirszam char,
pvaros nvarchar2,
putca nvarchar2,
padoszam nvarchar2,
pszamlaszam nvarchar2
)
is
dbnev nvarchar2(50);
hossz int;
karakter1 char;
karakter2 char;

begin
select nev into dbnev
from szamlakiallito
where upper(nev) = upper(pnev);
raise_application_error(-20101, 'Ilyen nevu mar van!');

exception
when no_data_found then
hossz:= length(padoszam);
if hossz!= 13 then
raise_application_error(-20102, 'Nem 13 hosszú az adoszam!');
end if;

karakter1 := substr(padoszam,9,1);
karakter2 := substr(padoszam,11,1);
if(karakter1 = '-' and karakter2 = '-') then
insert into szamlakiallito
```

Csatlós Tamás Péter - Adatbázis szerver oldali programozása - Informatikai technológiák laboratórium 1.

```
values (kiallito_seq.nextval,  
pnev,pirszam,pvaros,putca,padoszam,pszamlaszam);  
else  
raise_application_error(-20103, 'Rossz formatum!');  
end if;  
end;
```


Indoklás:

Az elején a név meglétét ellenőrzi a szkript. Ha minden rendben, akkor a SELECT utasítás no_data_found hibát dob, mert nem talált az adatbázisban paraméternek megfelelő személyt. Ellenkező esetben program hibával terminál. A hiba kezelésnél rögtön ellenőrizzük az adószám helyességét. A kiegészített tárolt eljárás képes ellenőrizni a feladatban leírt adatok helyességét.

Első tesztet a nem 13 karakter hosszú adószám megadására kiváltott hiba üzenet. A képen a GUI látható, amivel a beszúrandó rekordot állítottam elő. A Paraméterek rendre: Név, Irányítószám, Város, Utca, Adószám, Számlaszám.

<input type="checkbox"/>	Variable	Type	Value
<input checked="" type="checkbox"/>	pnev	String	Senki
<input checked="" type="checkbox"/>	pirszam	String	0000
<input checked="" type="checkbox"/>	pvaros	String	322
<input checked="" type="checkbox"/>	putca	String	132
<input checked="" type="checkbox"/>	padoszam	String	542
<input checked="" type="checkbox"/>	pszamlaszam	String	1000
<input checked="" type="checkbox"/>			

Error

 ORA-20102: Nem 13 hosszú az adoszam!
ORA-06512: at "TALXCE.UJ_SZAMLAKIALLITO_SP", line 28
ORA-06512: at line 3

View program sources of error stack?

☐ Don't show this message again

Második teszt esetnek még egyszer beszúrtam ugyan azt a személyt, Senkit, és a megfelelő hibát dobta a rendszer.

4.FELADAT

Feladat neve és leírása:

Írj tárolt eljárást, aminek bemeneti paramétere egy státusz szövegesen, az eljárás pedig kilistázza a standard kimenetre azon megrendelések adatait, amik a bemenetként megadott státusszal

Csatlós Tamás Péter - Adatbázis szerver oldali programozása - Informatikai technológiák laboratórium 1.

rendelkeznek. Az eljárás listázza ki a megrendelések azonosítóját, a feladás dátumát, a határidőt, a telephely adatait (IR, város, utca, vevőnév) valamint a fizetési módot szövegesen.

Megoldás:

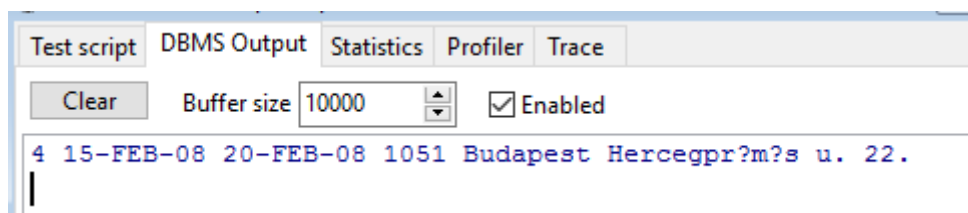
```
create or replace procedure megrendeles_lista_sp
(pstatus nvarchar2)
is
cursor c1 is select m.id, m.datum, m.hatarido,
                    (t.ir || ' ' || t.varos || ' ' || t.utca) as t_cim,
                    f.mod
from megrendeles m, telephely t, fizetesmod f, statusz s
where m.telephelyid = t.id
and m.fizetesmodid = f.id
and m.statuszid = s.id
and upper(s.nev) = upper(pstatus);

begin
    for clrec in c1 loop
        dbms_output.put_line(clrec.id || ' ' || clrec.datum || ' ' ||
clrec.hatarido || ' ' || clrec.t_cim);
    end loop;
end;
```

Indoklás:

Egy kurzort hozok létre, ami kigyűjti a megadott státuszú megrendelések kívánt adatait. Majd ezt egy ciklussal a kimenetre írom. A Konkatenációs műveletetre két példa is látható a szkriptben. Az egyik a kurzor definiálásakor a t_cim létrehozásánál látható.

Teszteléshez egy olyan státuszt adtam meg, aminél csak az egy rekord az eredmény.



5.FELADAT

Feladat neve és leírása:

Módosítsd a megrendelés táblát. Adj hozzá egy új oszlopot, ami a megrendelés összértékét fogja tartalmazni. (A megoldásként a módosító utasítást kell beadni, nem elég, ha grafikus felületen sikerült az új oszlopot felvenni.)

Megoldás:

A következő sorral vettem fel új oszlopot:

```
alter table megrendeles add (OsszMegrendeles float)
```

Indoklás:

Így lehet SQL utasítással már meglévő tábla oszlopainak a számát bővíteni 1 új oszloppal.

6.FELADAT

Feladat neve és leírása:

Írj blokkot, ami kitölti az előző feladatban létrehozott oszlop értékeit!

Megoldás:

```
declare cursor c is
  select m.id, sum(mt.nettoar * mt.mennyiseg) as ossz
  from megrendeles m
       join megrendelestetel mt
       on mt.megrendelesid = m.id
  group by m.id;

begin
  for rec in c loop
    update megrendeles
    set osszmegrendeles = rec.ossz
    where id = rec.id;
  end loop;
  commit;
end;
```

Indoklás:

Ez a kurzor kiszámolja, hogy mennyit kell fizetni a megrendelésekért. Az összeghez a megrendelés tétel táblát is kell használni, hiszen a nettó árat és a megrendelt mennyiséget onnan tudhatom meg.

7.FELADAT

Feladat neve és leírása:

Írj trigger, ami új megrendelés tétel felvételekor vagy törlésekor karban tartja az előző feladatban létrehozott összértékek tárolására szolgáló oszlopot. (Most ne foglalkozunk a termékek árának változásával).

Megoldás:

```
create or replace trigger AIUD_megrendeles
  after insert or update or delete
  on megrendelestetel
  for each row
declare
```

Csatlós Tamás Péter - Adatbázis szerver oldali programozása - Informatikai technológiák
laboratórium 1.

```
begin
  update megrendeles m
  set osszmegrendeles = nvl( osszmegrendeles, 0)
                        - nvl( :old.mennyiseg * :old.nettoar, 0)
                        + nvl( :new.mennyiseg * :new.nettoar, 0)
  where id = nvl(:old.megrendelesid, :new.megrendelesid);

end AIUD_megrendeles;
```

Indoklás:

Ez a trigger a tipikus DML utasítások után, amik a Megrendelés tétel táblán következnek be, frissíti a Megrendelés tábla összegrendelés attribútumát minden érintett rekordnak. Az összérték beállításánál ha nem volt megrendelés az előtt, akkor beszúrás és update esetén kerül új érték a rekordba, törlés esetén marad a nulla érték. Ha volt már értéke azelőtt, akkor beszúrás és update esetén hozzá adódik egy új érték, törlés esetén pedig levonódik a törölt összeg.

A következő feladat ennek a triggernek a teszteléséről szól, ott végeztem el a tesztelést.

8.FELADAT

Feladat neve és leírása:

Próbáld ki az előző feladatban megírt trigger mindkét kívánt esetre! (A megoldásban az utasításokat kell beadni, amiket erre a célra használtál!)

Megoldás:

```
select mt.id as MT_ID, m.osszmegrendeles, mt.nettoar, mt.mennyiseg
from megrendeles m join megrendelestetel mt on mt.megrendelesid = m.id
where m.id = 2;

insert into megrendelestetel
values(1200, 2, 100, 2, 3, 1);

update megrendelestetel
set mennyiseg = 3, nettoar = 700
where id = 1200;

delete from megrendelestetel
where id = 1200;

commit;
```

Indoklás:

Teszteléshez a fenti szkripteket használtam oly módon, hogy először lefuttattam a lekérdezést, hogy megtudjam milyen az adatok vannak az adatbázisban. Majd futtattam a következő szkriptet és utána megint a lekérdezést, amit egy újabb módosító szkript követett. Ezt így folytattam egészen a végéig.

Csatlós Tamás Péter - Adatbázis szerver oldali programozása - Informatikai technológiák laboratórium 1.

A lekérdezésben a Megrendelés tétel táblában szereplő tételek ID-át, az összeget, a nettó árat és a mennyiséget kérdeztem le. Mind ez a 2-es azonosítóval ellátott megrendeléshez tartozik.

Az első lekérdezés eredménye, mindenféle módosítás előtt.

	MT_ID	OSSZMEGRENDELES	NETTOAR	MENNYISEG
▶ 1	4	26151	7488	2
2	5	26151	3725	3

Beszúrás után az összeg értéke is változott.

	MT_ID	OSSZMEGRENDELES	NETTOAR	MENNYISEG
1	1200	26351	100	2
2	4	26351	7488	2
3	5	26351	3725	3

A beszúrt rekord megváltozása is hatással volt az összegre.

	MT_ID	OSSZMEGRENDELES	NETTOAR	MENNYISEG
1	1200	28251	700	3
2	4	28251	7488	2
3	5	28251	3725	3

Törlés után az első lekérdezéshez hasonló eredményt kaptam.

	MT_ID	OSSZMEGRENDELES	NETTOAR	MENNYISEG
1	4	26151	7488	2
2	5	26151	3725	3

Működik akkor is ha nem csak az én általam beszúrt rekordot módosítottam volna.

9.FELADAT

Feladat neve és leírása:

Hozz létre egy nézetet amiben két oszlop van, a kategória megnevezése, és a szülőkategória megnevezése, amennyiben létezik, vagy null.

Megoldás:

```
create view kategoria_nezet
as
select a.Nev as Kategoria, b.Nev as Szulo_kategoria
from Kategoria a left outer join
      Kategoria b on a.SzuloKategoria = b.ID
```

Indoklás:

Azokat a kategóriákat is megjeleníti, amelyeknek nincsen szülő kategóriájuk. Mindezt olvashatóbb kivitelben.

	Kategoria	Szulo_kategoria
1	Játék	NULL
2	Játszóház	NULL
3	Bébijáték	Játék
4	Építőjáték	Játék
5	Fajáték	Játék
6	Plüss figurák	Játék
7	Közeledési eszközök	Játék
8	0-6 hónapos kor	Bébijáték
9	6-18 hónapos kor	Bébijáték
10	18-24 hónapos kor	Bébijáték
11	DUPLO	Építőjáték
12	LEGO	Építőjáték
13	Építő elemek	Építőjáték
14	Építő kockák	Fajáték
15	Készségfejlesztő játékok	Fajáték
16	Logikai játékok	Fajáték
17	Ügyességi játékok	Fajáték
18	Bébi taxik	Közeledési eszközök
19	Motorok	Közeledési eszközök

10.FELADAT

Feladat neve és leírása:

Írj trigger, ami lehetővé teszi új kategória beszúrását a nézetten keresztül. Amennyiben a szülő kategória nem létezik, dobj hibát, és ne vedd fel az adatot a táblába.

Megoldás:

```
create trigger uj_kategoria
on kategoria_nezet
instead of insert

as

declare c1 cursor for
select kategoria, szulo_kategoria
from inserted

declare @kat nvarchar(50), @szul_kat nvarchar(50)

open c1
fetch c1 into @kat, @szul_kat
while @@FETCH_STATUS = 0
begin
```

Csatlós Tamás Péter - Adatbázis szerver oldali programozása - Informatikai technológiák laboratórium 1.

```
declare @szulkatid int
select @szulkatid = id
from kategoria
where nev = @szul_kat

if @szulkatid is not null
begin
    insert into kategoria
    values(@kat, @szulkatid)
end
else
    raiserror('Hibas szulo kategoria!', 16, 1)

fetch c1 into @kat, @szul_kat
end

close c1
deallocate c1
```

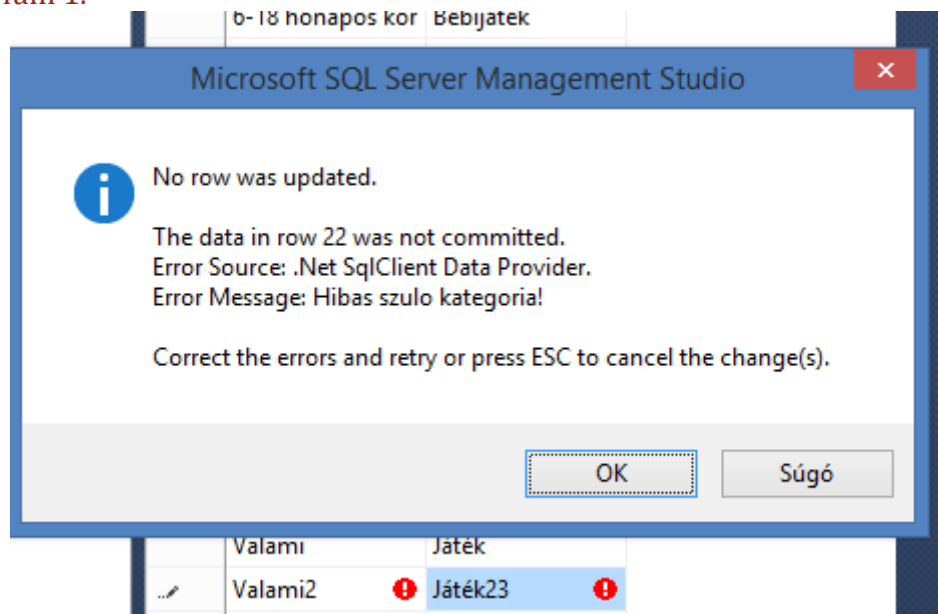
Indoklás:

Majd egy helyes szülőkategóriával beszúrtam egy kategóriát, amit engedett. Sikeres beszúráshoz „Valami” kategóriát adtam meg „Játék” szülő kategóriának.

Kategoria	Szulo_kategoria
Játék	NULL
Játszóház	NULL
Bébijáték	Játék
Építőjáték	Játék
Fajáték	Játék
Plüss figurák	Játék
Közlekedési esz...	Játék
0-6 hónapos kor	Bébijáték
6-18 hónapos kor	Bébijáték
18-24 hónapos ...	Bébijáték
DUPLO	Építőjáték
LEGO	Építőjáték
Építő elemek	Építőjáték
Építő kockák	Fajáték
Készségfejleszt...	Fajáték
Logikai játékok	Fajáték
Ügyességi játék...	Fajáték
Bébi taxik	Közlekedési esz...
Motorok	Közlekedési esz...
Triciklik	Közlekedési esz...
Valami	Játék

Második teszt során hibás szülő kategóriát próbáltam beilleszteni a táblába kevés sikerrel.

A „Valami2” Kategóriát szerettem volna beszúrni „Játék23” szülő kategóriával.



11.FELADAT

Feladat neve és leírása:

Írj eljárást, aminek bemeneti paramétere egy számlaazonosító. Az eljárás ellenőrizze le, hogy a számlatételek adatai egyeznek-e a megrendelés tételek megfelelő adataival (mennyiség, nettó ár, ÁFA kulcs). Amennyiben eltérés található a kettőben, úgy mindkettő értékét valamint a termék adatait írja ki a standard outputra.

Megoldás:

```
create procedure szamla_check_ps
    @SzamlaId int
as
begin tran

declare c cursor fast_forward read_only for
    select szt.mennyiseg as Szamla_mennyiseg, szt.nettoar as Szamla_nettoar,
    szt.afakulcs as Szamla_afa,
    mt.Mennyiseg as Tetel_mennyiseg, mt.NettoAr as Tetel_nettoar,
    a.Kulcs as Afa_kulcs
    from SzamlaTetel szt
    join MegrendelesTetel mt on mt.ID = szt.MegrendelesTetelID
    join AFA a on szt.AFAKulcs = a.Kulcs
    where @SzamlaId = szt.id

DECLARE @Szamla_Mennyiseg int
DECLARE @Tetel_Mennyiseg int

DECLARE @Szamla_NettoAr int
DECLARE @Tetel_NettoAr int
```

Csatlós Tamás Péter - Adatbázis szerver oldali programozása - Informatikai technológiák laboratórium 1.

```
DECLARE @Szamla_AFA int
DECLARE @Tetel_AFA int

open c
    FETCH NEXT FROM c
    INTO @Szamla_Mennyiseg, @Szamla_NettoAr, @Szamla_AFA, @Tetel_Mennyiseg,
@Tetel_NettoAr, @Tetel_AFA

    WHILE @@FETCH_STATUS = 0
    BEGIN
        --print kell még --egyeztetés kell még
        if (@Szamla_Mennyiseg <> @Tetel_Mennyiseg) or
            (@Szamla_NettoAr <> @Tetel_NettoAr) or
            (@Szamla_AFA <> @Tetel_AFA)
        begin
            print 'Eltérés az adatok között: '+
                'Mennyiség: '+convert(varchar(2),@Szamla_Mennyiseg)+' <-->
'+convert(varchar(2), @Tetel_Mennyiseg)+'
                'Nettó ár: '+convert(varchar(7), @Szamla_NettoAr)+' <-->
'+convert(varchar(7), @Tetel_NettoAr)+'
                'Áfa: '+convert(varchar(7), @Szamla_AFA)+' <-->
'+convert(varchar(7), @Tetel_AFA)
        end

        FETCH NEXT FROM c
        INTO @Szamla_Mennyiseg, @Szamla_NettoAr, @Szamla_AFA,
@Tetel_Mennyiseg, @Tetel_NettoAr, @Tetel_AFA
    END

close c

deallocate c

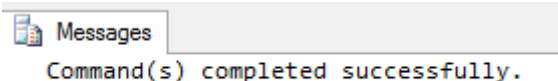
commit
```

Indoklás:

Kikeresem a fontos tulajdonságokat, majd ha a feltételnek eleget tesznek, akkor kiíratom a alapértelmezett kimenetre.

Sikeres teszt esetén nem kellene kiírnia semmit. A következő kóddal néztem meg a sikeres működést.

```
exec szamla_check_ps 2
```



Arról, hogy ez tényleg helyes a következő lekérdezéssel győződtem meg:

```
select szt.mennyiseg as Szamla_mennyiseg, szt.nettoar as Szamla_nettoar, szt.afakulcs
as Szamla_afa,
        mt.Mennyiseg as Tetel_mennyiseg, mt.NettoAr as Tetel_nettoar,
```

Csatlós Tamás Péter - Adatbázis szerver oldali programozása - Informatikai technológiák laboratórium 1.

```
        a.Kulcs as Afa_kulcs
from SzamlaTetel szt
join MegrendelesTetel mt on mt.ID = szt.MegrendelesTetelID
join AFA a on szt.AFAKulcs = a.Kulcs
where szt.id = 2
```

Results		Messages				
	Szamla_mennyiseg	Szamla_nettoar	Szamla_afa	Tetel_mennyiseg	Tetel_nettoar	Afa_kulcs
1	1	1854	20	1	1854	20