

Prijava na bazu podataka

Web stranica: **http://195.130.59.11:5560**

Username : ???

Password: ???

Host string: ???

© Emir Buza

Trigeri baze podataka

Trigeri baze podataka su jedni od onih objekata baze podataka koji se izvršavaju kada se implicitno izvršava jedna od DML operacija: INSERT, UPDATE ili DELETE nad nekom od tabela nad kojom je kreiran dati objekat triger. Svrha triger je da se aktiviraju, odnosno izvrše svoj korisni dio koda u bazi podataka, u svakom momentu kada se konektuje neki od korisnika na bazu podataka, i nekom od aplikacija razvijenoj u ma kojem programskom jeziku pokušava da mijenja sadržaj slogova ili cijele slogove u bazi podatka navede tabeli.

Trigeri kojima ćemo se mi baviti u ovom materijalu su trigeri koji se isključivo odnose na trigere vezane za neku od tabela baze podataka, no međutim, postoje trigeri koji nisu vezani samo za tabele, tako na primjer, postoje trigeri na nivou baze podataka koji registruju korisnika prilikom logiranja i od-logiranja na/sa baze podataka. Zatim postoje trigeri koji registruju startanje i gašenje instance baze podataka, dakle opet na nivou same baze podataka. Ovi trigeri su jako korisni kada se žele spasiti informacije o tome kad, ko, kada i u koje vrijeme je pokušao da se poveže i koristiti objekte baze podataka.

Prilikom dizajniranja triger je treba imati na umu sljedeće stvari:

- Svi trigeri garantuju da kada se izvršava specifična operacija, da će se izvršiti i akcija definisana od strane triger.
Samo upotrebom triger baze podataka centralizirane, globalne operacije će se automatski registrovati i obraditi od strane triger, ako je tako definisano, bez obzira koji korisnik ili aplikacija je trenutno povezana na bazu podataka.
- Pametno je ne koristiti trigere za definisanje ili zamjenu funkcionalnosti koje su već kreirane u okviru baze podataka. Na primjer, nije potrebno u okviru triger definisati pravila integriteta baze podataka ako su ona već deklarirana kroz constraint-e.
Prekomijerna upotreba triger može rezultirati u složene među-zavisnosi, koje mogu biti složene za održavanje u velikom broju aplikacijama nad definisanom bazom podataka. Dakle, treba koristiti trigere samo kada je to neophodno, i čuvati se rekursivnih i kaskadnih efekata.

Sintaksa za kreiranje triger u bazi podataka nad tabelom data je sljedećom strukturom:

```
SQL> CREATE OR REPLACE TRIGGER naziv_trigera
Vrijeme okidanja triger
događaj1 /OR događaj2/ /OR događaj3/
ON naziv_tabele
/REFERENCING OLD AS old | NEW as new/
/FOR EACH ROW/
/WHEN uslov/
BEGIN
tijelo_trigera
```

```
END
;
```

U sintaksi je:

CREATE OR REPLACE	- ključne riječi za kreiranje i re-kreiranje trigerera ako već postoji kreiran u bazi podataka.
naziv_trigera	- je naziv trigerera koji se kreira u bazi podataka.
vrijeme okidanja trigerera	- je indikator koji specificira vrijeme kada će triger okinuti na događaj: <ul style="list-style-type: none"> - AFTER i - BEFORE
događaj 1..3	- označava operaciju manipulacije podataka u slogu za koju triger okida: <ul style="list-style-type: none"> - INSERT - DELETE i - UPDATE /OF kolona1, kolona2, .../
naziv_tabele	- je naziv tabele na koju je triger vezan.
REFERENCING	- je ključna riječ koja označava koleracijski naziv za staru i novu vrijednost trenutnog sloga. Default-e vrijednosti su OLD i NEW.
FOR EACH ROW	- označava da će triger okidati za svaki slog, nad kojim se vrši specificirana DML operacija, odnosno događaj 1..3. Ako se ovaj dio izostavi prilikom kreiranja trigerera, onda triger okida na nivou izraza.
WHEN uslov	- označava restrikciju za triger, kada treba da izvrši komande definisane u tijelu trigerera, a kada ne. Ovaj uslov se provjerava za svaki slog kada triger okine.
tijelo trigerera	- je tijelo trigerera koje sadrži akcije koje je potrebno izvršiti kada triger okine. Tijelo trigerera se piše između BEGIN END ključnih riječi.

Sve kreirane trigere za korisnika koji je prijavljen na bazu mogu se vidjeti kroz poglede data dictionary-a:

- ALL_TRIGGERS
- USER_TRIGGERS

Brisanje trigger moguće je izvršiti putem komande DROP SEQUENCE

```
SQL> DROP TRIGGER naziv_trigera;
```

Zadaci

1. Kreirajte vašu tabelu zaposlenih koja će sadržavati sljedeće kolone: šifru zaposlenog, naziv zaposlenog, naziv odjela, šifra posla, naziv posla, platu i dodatak na platu. Potom modificirati vašu tabelu zaposlenih sa novom kolonom šifra koja će biti primarni ključ. Potom kreirati trigger koji će onemogućiti promjene nad podacima onih zaposlenih koji su počeli da rade od 1998 godine.
2. Kreirati trigger nad vašom tabelom zaposlenih koji će onemogućiti promjene nad podacima za radne dane, tj. od ponedjeljka do petka, u periodu od 16:23 i 23:16, i subotom i nedjeljom od 06:34 i 23:56.
3. Kreirati trigger nad vašom tabelom zaposlenih koji se za sve promjene nad podacima zaposlenih insertovati stare podatke o zaposlenom u vašu tabelu arhive, koja će pored svih kolona kao i vaša tabela zaposlenih sadržavati još i informacije o korisniku koji se ažurirao podatke, kao i vrijeme ažuriranja podataka.
4. Kreirajte vašu tabelu odjela i poslova identičnih tabelama departments i jobs, sa dodatnim kolonama id, korisnik i datum, gdje će te proglasiti kolone id primarnim ključevima tabela, respektivno. Potom uvesti dodatne kolone nad vašim tabelama zaposlenih, kako bi ste kreirali odgovarajuće strane ključeve između odgovarajućih tabela. Potom kreirajte odgovarajuće ključeve.
Nakon što kreirate ove tabele kreirajte trigere nad bazom koji će omogućiti automatsko dodijeljivanje id, korisnika i datum nad vašim novo kreiranim tabelama. Id je potrebno osigurati iz sekvenci brojeva.
5. Modificirajte predhodni trigger za tabelu odjela na takava način da se za menagera odjela, koji se trenutno insertuje u bazu, automatski dodijeli onaj manager koji ima najmanje uposlenih.
6. Modificirajte trigger za insertovanje slogova za tabelu zaposlenih na takav način da se za sifru odjela uzme onaj odjel kojim rukovodi onaj šef, koji rukovodi sa najmanje 2 odjela.
7. Kreirajte trigger nad tabelom zaposlenih koji će za brisanje slogova iz vaše tabele zaposlenih kreirati tabelu arhive. Tabela arhive mora imati primarni ključ po koloni id tabele arhive.
8. Kreirajte trigger nad vašom tabelom zaposlenih koja će osigurati, integritet visine plate za odgovarajuće poslove po iznosima koji su definisani sa maksimalnom i minimalnom vrijednošću. U slučaju da je prebačen limit, tj. da je plata i dodatak na platu veći od naznačenog limita umanjiti platu za 20 odsto. U slučaju da i pored ovog umanjivanja plata sa dodatkom na platu je veća od dozvoljenih limita, umanjiti platu za 30 odsto i tako redom, dok se ne zadovolji kriterij visine plate po naznačenom poslu. A ako je u pitanju plata i dodatak na platu koji je manja od naznačenih limita, tada je potrebno uraditi obratan postupak, tj. prvo platu uvećati za 20 odsto, pa ako ne zadovoljava kriterij za 30 odsto i tako redom.
9. Kreirati trigger nad vašom tabelom odjela koji će onemogućiti promjenu odjela iz Amerike u Englesku, a potom kreirati najpodesniju tabelu logova, u kojoj će te čuvati informaciju o pokušajima promjene odjela iz jedne u drugu lokaciju. Voditi računa da je neophodno u log tabeli imati informaciju o staroj i novoj lokaciji.