

Napredni MySQL



Uvod u MySQL Okidače

- **Što su Okidači?**
 - Specijalizirane skripte unutar baze podataka
 - Automatski se izvršavaju kao odgovor na određene događaje (npr. INSERT, UPDATE, DELETE)
- **Zašto koristiti Okidače?**
 - Automatizacija operacija nad podacima
 - Održavanje integriteta podataka
 - Evidencija promjena u bazi podataka

Vrste Okidača

- **Prema vremenu izvršenja:**
 - BEFORE (prije izvršenja akcije)
 - AFTER (nakon izvršenja akcije)
- **Prema vrsti događaja:**
 - INSERT
 - UPDATE
 - DELETE

Primjer Okidača za INSERT

- **Automatsko dodavanje vremena kreiranja**
 - Kada se novi red unese u tablicu, okidač dodaje trenutni vremenski zapis.

```
CREATE TRIGGER AddCreationDate
BEFORE INSERT ON employees
FOR EACH ROW
SET NEW.creation_date = NOW();
```

- **BEFORE INSERT:** Okidač se aktivira prije dodavanja novog reda
- **NEW.creation_date:** Postavlja vrijednost stupca **creation_date** za novi red

Primjer Okidača za UPDATE

- **Ažuriranje evidencije promjena**
 - Prati promjene na stupcu *salary* u tablici *employees*.

```
CREATE TRIGGER LogSalaryChanges
AFTER UPDATE ON employees
FOR EACH ROW
  WHEN (OLD.salary <> NEW.salary)
    INSERT INTO salary_changes(employee_id, old_salary,
new_salary, change_date)
  VALUES (OLD.id, OLD.salary, NEW.salary, NOW());
```

- **AFTER UPDATE:** Okidač se pokreće nakon ažuriranja reda
- **WHEN:** Izvršava se samo ako se vrijednost *salary* promijenila

Upravljanje Okidačima

- **Pregled i Brisanje Okidača**
 - Prikaz svih okidača: **SHOW TRIGGERS;**
 - Brisanje okidača: **DROP TRIGGER IF EXISTS naziv_okidača;**
- **Najbolje prakse i napomene**
 - Testirati okidače u razvojnom okruženju
 - Koristiti okidače oprezno zbog potencijalnog utjecaja na performanse
 - Dobro dokumentirati sve okidače u bazi

Hvala na pažnji!

