SAKLI YORDAM (Stored Procedure)

Sibel Somyürek

- * Saklı yordamlar veritabanı kataloğunda saklanan SQL kod bloklarının tanımlanmasıdır.
- * Mesela, iki sayı alıp bunların toplamlarını hesaplayan bir kod parçasını toplayıcı adında bir prosedür içerisine paketleyebiliriz.
- * Saklı yordamlar pek çok sorgulamayı daha kolaylaştırır.
- * MYSQL 5.0.1 versiyonundan itibaren MySQl'de yer alır.

- * Saklı yordamlar sadece ilk kez çalıştırıldıklarında derlenirler.
- * Daha sonraki çalıştırma işlemlerinde derlenmezler.
- * Bu sayede uygulamalar hız kazanılmış olur. Yani normal kullanılan bir SQL sorgusunun tekrar tekrar çalıştırılmasına oranla daha fazla performans elde edilmesini sağlarlar.
- * Bir kez yazılıp, tekrar tekrar kullanıldığı için modüler bir yapıda program geliştirilmiş olur

- * Oluşturulan veya var olan saklı yordamlar dışarıdan parametre alabilirler.
- * Bu parametrelere göre tanımlanan işlemleri yaparlar ve/veya dışarıya değer kümesi dönderirler.

Neden kullanılır?

1. Kodların yeniden kullanımı

* Aynı veritabanını kullanan farklı uygulamalar tekrar aynı kodları yazmak yerine saklı yordamları kullanabilir.

2. Daha kolay kodlama

- Bir geliştirici sorgu ya da tabloların tam olarak bilmeden bir saklı yordamı çağırabilir. Veritabanını daha kullanıcı dostu kılarlar.
- * Call profilikaydet(1, 'ayşe', 'ak')
- * Böylece veritabanı yöneticisi geliştiricileri bilgilendirmeden veritabanını tekrar yapılandırabilir.

Neden kullanılır?

- 3. API'lerden ötürü güvenlik
 - * Saklı yordamlar vertabanında API gibi davranabilirler. Eğer kullanıcılar sadece saklı yordamları kullanma iznine sahipse, sadece bu yordamlara verilen işlemleri gerçekleştirebilirler.
- * Kullanıcının bir tabloya erişim izni olmasa bile o tablo üzerinde işlem yapan bir saklı yordamı kullanma izni olabilir.

- * Tetikleyicilerden farklı olarak prosedürler uygulama tarafından ya da script tarafından bir şekilde çağrılmak zorundadırlar.
- * Otomatik devreye giremezler.

Saklı Prosedür Yaratmak

* CREATE PROCEDURE ya da kısaca CREATE PROC ifadesi ile yaratılır.

Saklı Prosedür Yaratmak

```
id SMALLINT(3) NOT NULL,
ad VARCHAR(25) NOT NULL,
soyad VARCHAR(25) NOT NULL
)
```

Saklı Prosedür Yaratmak

```
DELIMITER //
CREATE PROCEDURE veriekle()
    INSERT INTO tablo(id,ad,soyad) VALUES (1,"Ayşe","Ak")
//
DELIMITER;
```

Saklı Yordama değer gönderme/alma

- * Saklı yordamlar parametreli olabilir.
- * Parametreler saklı yordamları daha esnek ve kullanılabilir kılar.
- * Mysql de saklı yordam parametreleri üç yöntem belirtecinden birini alabilir.
 - * IN
 - * OUT ve
 - * INTOUT

* IN

* Değişkenin değerine erişilir ancak değişiklik yapılamaz.

* OUT

* Değişkenin değerine erişilemez ancak değişiklik yapılabilir.

* INTOUT

* Değişkenin değerine erişilir ve değişiklik yapılabilir.

* Parametre girilmezse IN parametresi girilmiş gibi algılar

* Saklı yordamı çağırmak için CALL deyimi kullanılır

* CALL saklıprosedürismi();

* CALL deneme();

* Hata: #1146 - Table 'elektronik.elektronik' doesn't exist

* < MYSQL 5.1.7 versiyon

* Güncellemek gerekli

Saklı yordama parametre gönderme

```
DELIMITER //
CREATE PROCEDURE verieklepm (eskiad VARCHAR(25))
UPDATE tablo SET ad='fatma' where ad=eskiad
//
DELIMITER;
```

CALL verieklepm('Ayşe');

IN parametresi-örnek

```
DELIMITER //
CREATE PROCEDURE veriekle3 (IN eskiad VARCHAR(25))
UPDATE tablo SET ad='fatma' where ad=eskiad
//
DELIMITER;

CALL verikle3('Ayşe');
```

IN parametresi-örnek 2

```
* DELIMITER //
* CREATE PROCEDURE desikenleveriekle (IN eskiad VARCHAR(25))
* UPDATE tablo SET ad='fatma' where ad=eskiad
* //
* DELIMITER;
* SET @eskiad='emine';
* CALL desikenleveriekle(@eskiad);
```

Değişken tanımlama

- * Saklı Yordam içerisinde kullanabileceğimiz değişkenler tanımlayabiliriz.
- * Değişken tanımlama DECLARE direktifi ile yapılır.
- * DECLARE degisken_ismi veri_tipi(boyutu) DEFAULT varsayılan_deger;

IN parametresi

```
DELIMITER //
CREATE PROCEDURE veriekle3 (IN eskiad VARCHAR(25))
UPDATE tablo SET ad='fatma' where ad=eskiad //
DELIMITER;
CALL veriekle3('ayşe')
```

OUT parametresi

```
DELIMITER //
CREATE PROCEDURE kayitsayisi (OUT toplam INT(11))
SELECT COUNT(*) AS toplam FROM tablo //
DELIMITER;
```

CALL kayitsayisi(@toplam)

OUT parametresi

```
DELIMITER //
CREATE PROCEDURE kayitsayisi2 (OUT toplam INT(11))
SELECT COUNT(*) INTO toplam FROM tablo //
DELIMITER;
```

CALL kayitsayisi2(@toplam)

Birden fazla parametre

```
DELIMITER //
CREATE PROCEDURE kayitsayisi2 (IN eskiad VARCHAR(25), OUT b INT(11))
SELECT count(*) INTO b FROM tablo where ad=eskiad
//
DELIMITER;
```

* CALL kayitsayisi2('fatma', @b);

INOUT parametresi – örnek 1

```
DELIMITER //
CREATE PROCEDURE deneme8 (INOUT n INT(11))
UPDATE tablo SET id=id+5 WHERE id=n;
//
DELIMITER;

SET @n=1;
Call deneme8(@n);
```

INOUT parametresi –örnek 2

```
create table kayit(
a INT( 11 ) NOT NULL
)
```

INOUT parametresi –örnek 2

```
* DELIMITER //
* CREATE PROCEDURE kayitsayisiaktar (IN adi VARCHAR(25), OUT k
INT(11))
* BEGIn
* SELECT count(*) INTO k FROM tablo where ad=adi;
* INSERT INTO kayit(a) VALUES (k);
* END
* //
* DELIMITER;
* CALL kayitsayisiaktar ('fatma', @k);
```

Prosedürleri göstermek

* Show Procedure Status

Saklı yordamı kaldırmak

- * DROP pocedure prodesür_adı
- * DROP PROCEDURE veriekle;

GÖREV

- okul isimli yeni bir veritabanı oluşturun
- Dersler (derskod,ad, saat) isimli bir tablo oluşturun.
 - 1. Çalıştırıldığında ders tablosuna 2 adet derse ait kayıt ekleyen saklı yordamı oluşturun.
 - 2. Ders kodu gönderilen dersin adını yeni ders adı ile güncelleyen saklı yordamı oluşturun ve çalıştırın.
 - Derssayisi (adet) isimli bir tablo oluşturun.
 - 4. Dersler tablosundaki ders sayısını derssayisi tablosuna kaydeden bir saklı yordam oluşturun ve çalıştırın.
 - 5. INOUT parametresini kullanabileceğiniz bir saklı yordam senaryosu düşünerek bunun için gereklki tabloları oluşturun, saklı yordamı yazın ve çalıştırın.
 - 6. Saklı yordamları görüntüleyin
 - 7. Oluşturduğunuz saklı yordamlardan birini silin.

Kaynakça

- * http://www.mysqltutorial.org/stored-proceduresparameters.aspx
- * http://www.sinanakyazici.com/mysqlde-storedprocedure-sakli-yordam-kullanimi
- * http://emircem.wordpress.com/2011/07/20/mysqldesakli-yordamlar/