

VERİTABANI TASARIMI

Öğretim Görevlisi A. Berika VAROL MALKOÇOĞLU

İçindekiler

- Saklı Yordamlar
- Saklı Yordam örnekleri

DELIMITER İfadesi Nedir?

- SQL ifadeleri yazarken, iki ifadeyi ayırmak için noktalı virgöl (;) kullanılır.
- Ancak bir saklı yordam, noktalı virgülle (;) ayrılmış birden çok ifadeden oluşur.
- Bunun önüne geçmek için DELIMITER komutu kullanılmalı.

DELIMITER delimiter_character

- Tek bir karakter veya birden fazla karakter için // veya \$\$ kullanılmalı.

Parametresiz Saklı Yordamlar

```
--  
54 DELIMITER //  
55  
56 • CREATE PROCEDURE yazarList()  
57 BEGIN  
58     SELECT * FROM yazar ORDER BY yazar_ad;  
59 END//  
60 DELIMITER ;  
61  
62 • CALL yazarList();  
63
```

<

Result Grid | Filter Rows: | Export: | Wrap Ce

	yazar_id	yazar_ad	yazar_soyad
▶	10	ahmet	ümit
	12	alan	moore
	4	ayşe	kulin
	3	can	yücel
	6	ernest	hemingway
	5	halide edip	adivar
	2	orhan	pamuk
	1	sabahattin	ali
	9	stefan	zweig
	8	william	faulker
	7	yahya kemal	beyatlı
	11	yaşar	kemal

Parametresiz Saklı Yordamlar

```
67 • CREATE PROCEDURE kitapGrupla()  
68 BEGIN  
69     SELECT kategori_id as kategori, COUNT(*) as kitap_sayisi FROM kitap GROUP BY kategori_id ;  
70 END//  
71 DELIMITER ;  
72  
73 • CALL kitapGrupla();  
74  
75
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	kategori	kitap_sayisi
▶	1	12
	2	3
	3	1
	4	2
	6	3
	7	13
	8	4
	9	1

Saklı Yordamlar Yapısı

```
DELIMITER $$  
    CREATE PROCEDURE sp_name()  
    BEGIN  
        -- statements  
    END $$  
DELIMITER ;
```

```
DELIMITER //  
    CREATE PROCEDURE sp_name()  
    BEGIN  
        -- statements  
    END //  
DELIMITER ;
```

Saklı Yordamlarda Değişkenler

- Saklı yordamlar içerisinde kullanabileceğimiz değişkenler tanımlayabiliriz.
- Değişken tanımlama DECLARE direktifi ile yapılır.

DECLARE degisken_adi veritipi(boyut) DEFAULT varsayılan_deger;

DECLARE x, y INT DEFAULT 0;

Saklı Yordamlarda Değişkenler

- Bir değişkene değer atamak için SET ifadesi kullanılır.

`SET variable_name = value;`

Örnek:

```
DECLARE total INT DEFAULT 0;  
SET total = 10;
```


Saklı Yordamlarda Değişkenler

```
83
84 DELIMITER $$
85
86 • CREATE PROCEDURE toplamYazarSayisi()
87 BEGIN
88     DECLARE toplamYazar INT DEFAULT 0;
89     SELECT COUNT(*) INTO toplamYazar FROM yazar;
90     SELECT toplamYazar;
91
92 END$$
93
94 DELIMITER ;
95
96 • call toplamYazarSayisi();
97
```

<

Result Grid | Filter Rows: | Export: | Wrap Cell Content

	toplamYazar
▶	12

Parametrelili Saklı Yordamlar

```
DELIMITER //  
CREATE PROCEDURE AdUzunluk(gelendeger VARCHAR(50))  
BEGIN  
    SELECT CHAR_LENGTH(gelendeger) AS Uzunluk;  
END//  
DELIMITER ;  
  
CALL AdUzunluk('berika');
```


```
2
3 DELIMITER //
4 • CREATE PROCEDURE AdUzunluk(gelendeger VARCHAR(50))
5 BEGIN
6     SELECT CHAR_LENGTH(gelendeger) AS Uzunluk;
7 END//
8 DELIMITER ;
9
10 • CALL AdUzunluk('berika');
11
12
```

<	
Result Grid	Filter Rows: <input type="text"/>
Export: 	Wrap Cell Content: 
	Uzunluk
▶	6

```
12 • CALL AdUzunluk('AYŞE BERİKA VAROL MALKOÇOĞLU');
```

<	
Result Grid	Filter Rows: <input type="text"/>
Export: 	Wrap Cell Content: 
	Uzunluk
▶	28

```
1 • USE kitaplik;
2
3 DELIMITER //
4 • USE kitaplik//
5 • CREATE PROCEDURE kitap_getir(IN id INT)
6 BEGIN
7     select * FROM kitap WHERE kitap_id=id ;
8 END//
9
10 DELIMITER ;
11
12 • CALL kitap_getir(5);
13
```

<				
Result Grid				
Filter Rows: <input type="text"/>				
Export:  Wrap Cell Cont				
	kitap_id	kitap_ad	yazar_id	kategori_id
▶	5	kar	2	7

```
15 DELIMITER //
```

```
16 • CREATE PROCEDURE kitapSayisi(IN id INT, OUT toplam_kitap INT)
```

```
17 BEGIN
```

```
18     select Count(*) FROM kitap WHERE yazar_id=id ;
```

```
19 END//
```

```
20
```

```
21 DELIMITER ;
```

```
22
```

```
23 • CALL kitapSayisi(3,@toplam_kitap);
```

<	
Result Grid	
Filter Rows: <input type="text"/>	
Export: 	
Wrap Cell Content: 	
	Count(*)
▶	2

```

26 DELIMITER //
27 • CREATE PROCEDURE kitapEkle(IN kitap_ad VARCHAR(50), IN yazar_id INT, IN kategori_id INT)
28 BEGIN
29     INSERT INTO kitap(kitap_ad, yazar_id, kategori_id) VALUES(kitap_ad, yazar_id, kategori_id);
30 END//
31 DELIMITER ;
32
33 • CALL kitapEkle('XXXX KİTABI', 2, 3);
34
35 • select * from kitap;
36

```

Result Grid				
		Filter Rows:		
		Edit:		
		Export/Import:		
		Wrap Cell Content:		
kitap_id	kitap_ad	yazar_id	kategori_id	
34	kalp ağrısı	5	1	
35	sarı sıcak	11	1	
36	wathcmen	12	4	
37	olağanüstü beyfendil...	12	4	
38	XXXX KİTABI	2	3	
NULL	NULL	NULL	NULL	

```

36
37 DELIMITER //
38 • CREATE PROCEDURE kitapEkle2(IN inp_kitap_ad VARCHAR(50), IN inp_yazar_ad VARCHAR(50), IN inp_yazar_soyad VARCHAR(50), IN inp_kategori_ad VARCHAR(50))
39 BEGIN
40     INSERT INTO kitap(kitap_ad, yazar_id, kategori_id)
41     VALUES(inp_kitap_ad, (
42     SELECT yazar_id FROM yazar WHERE concat(yazar_ad, ' ', yazar_soyad)=concat(inp_yazar_ad, ' ', inp_yazar_soyad)), (
43     SELECT kategori_id FROM kategori WHERE kategori_ad=inp_kategori_ad));
44 END//
45 DELIMITER ;
46
47 • CALL kitapEkle2('veda', 'ayşe', 'kulin', 'roman');
48
49 • select * from kitap;
50

```

Result Grid				
Filter Rows:				
Edit:				
Export/Import:				
Wrap Cell Content:				
kitap_id	kitap_ad	yazar_id	kategori_id	
29	istanbul hatırası	10	6	
30	ölmez otu	11	1	
31	çanlar kimin için çalıyor	6	2	
32	can yücel'in şiirleri	3	8	
33	mor salkımlı ev	5	1	
34	kalp ağrısı	5	1	
35	sarı sıcak	11	1	
36	wathomen	12	4	
37	olağanüstü beyfendil...	12	4	
38	XXXX KİTABI	2	3	
40	veda	4	7	
*	NULL	NULL	NULL	

Listelemek için Saklı Yordamlar

- Saklı yordamları listelemek için:

SHOW PROCEDURE STATUS

sorgusunu kullanırız

Listelemek için Saklı Yordamlar

79 • SHOW PROCEDURE STATUS;

80

Db	Name	Type	Definer	Modified	Created	Security_type	Comment	character_set_client	collation_connection
kitaplik	kitapEkle	PROCEDURE	root@localhost	2020-10-15 18:12:42	2020-10-15 18:12:42	DEFINER		utf8mb4	utf8mb4_0900_ai
kitaplik	kitapEkle2	PROCEDURE	root@localhost	2020-10-15 18:12:31	2020-10-15 18:12:31	DEFINER		utf8mb4	utf8mb4_0900_ai
kitaplik	kitapGrupla	PROCEDURE	root@localhost	2020-10-19 10:07:39	2020-10-19 10:07:39	DEFINER		utf8mb4	utf8mb4_0900_ai
kitaplik	kitapSayisi	PROCEDURE	root@localhost	2020-10-15 17:39:56	2020-10-15 17:39:56	DEFINER		utf8mb4	utf8mb4_0900_ai
kitaplik	kitap_getir	PROCEDURE	root@localhost	2020-10-15 17:37:10	2020-10-15 17:37:10	DEFINER		utf8mb4	utf8mb4_0900_ai
kitaplik	kutuphaneIDarttir	PROCEDURE	root@localhost	2020-10-19 10:35:21	2020-10-19 10:35:21	DEFINER		utf8mb4	utf8mb4_0900_ai
kitaplik	yazarList	PROCEDURE	root@localhost	2020-10-19 10:00:50	2020-10-19 10:00:50	DEFINER		utf8mb4	utf8mb4_0900_ai
sakila	film_in_stock	PROCEDURE	root@localhost	2020-10-09 17:34:54	2020-10-09 17:34:54	DEFINER		utf8mb4	utf8mb4_0900_ai
sakila	film_not_in_stock	PROCEDURE	root@localhost	2020-10-09 17:34:54	2020-10-09 17:34:54	DEFINER		utf8mb4	utf8mb4_0900_ai
sakila	rewards_report	PROCEDURE	root@localhost	2020-10-09 17:34:54	2020-10-09 17:34:54	DEFINER	Provides a customizable report on best customers	utf8mb4	utf8mb4_0900_ai
sakli_...	AdUzunluk	PROCEDURE	root@localhost	2020-10-15 17:08:01	2020-10-15 17:08:01	DEFINER		utf8mb4	utf8mb4_0900_ai
sakli_...	AdUzunluk1	PROCEDURE	root@localhost	2020-10-15 17:08:42	2020-10-15 17:08:42	DEFINER		utf8mb4	utf8mb4_0900_ai
sakli_...	AdUzunluk_2	PROCEDURE	root@localhost	2020-10-15 17:11:54	2020-10-15 17:11:54	DEFINER		utf8mb4	utf8mb4_0900_ai
sys	create_synonym...	PROCEDURE	mysql.sys@lo...	2020-10-09 17:32:36	2020-10-09 17:32:36	INVOKER	Description ----- Takes a source database...	utf8mb4	utf8mb4_0900_ai
sys	diagnostics	PROCEDURE	mysql.sys@lo...	2020-10-09 17:32:36	2020-10-09 17:32:36	INVOKER	Description ----- Create a report of the cu...	utf8mb4	utf8mb4_0900_ai
sys	execute_prepare...	PROCEDURE	mysql.sys@lo...	2020-10-09 17:32:36	2020-10-09 17:32:36	INVOKER	Description ----- Takes the query in the ar...	utf8mb4	utf8mb4_0900_ai
sys	ps_setup_disable...	PROCEDURE	mysql.sys@lo...	2020-10-09 17:32:36	2020-10-09 17:32:36	INVOKER	Description ----- Disable all background th...	utf8mb4	utf8mb4_0900_ai
sys	ps_setup_disable...	PROCEDURE	mysql.sys@lo...	2020-10-09 17:32:36	2020-10-09 17:32:36	INVOKER	Description ----- Disables consumers withi...	utf8mb4	utf8mb4_0900_ai
sys	ps_setup_disable...	PROCEDURE	mysql.sys@lo...	2020-10-09 17:32:36	2020-10-09 17:32:36	INVOKER	Description ----- Disables instruments with...	utf8mb4	utf8mb4_0900_ai

Listelemek için Saklı Yordamlar

81 • `SHOW PROCEDURE STATUS WHERE Db LIKE 'kitap%';`

82

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

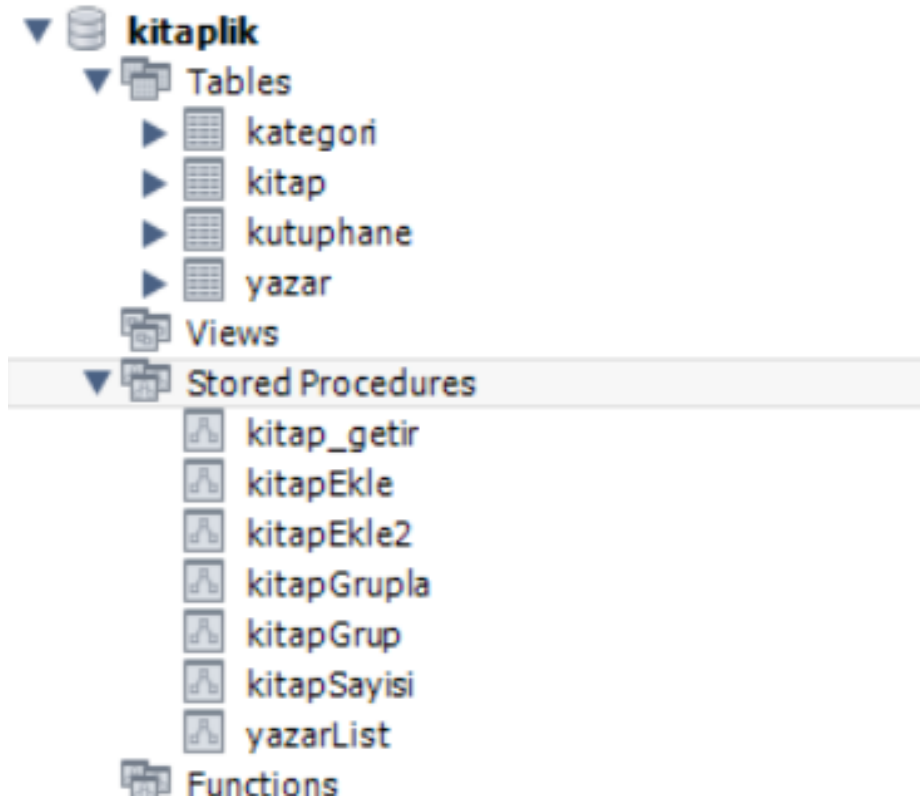
	Db	Name	Type	Definer	Modified	Created	Security_type	Comment	character_set_client	collation_connection	Database Collation
▶	kitaplik	kitapEkle	PROCEDURE	root@localhost	2020-10-15 18:12:42	2020-10-15 18:12:42	DEFINER		utf8mb4	utf8mb4_0900_ai_ci	utf8mb4_0900_ai_ci
	kitaplik	kitapEkle2	PROCEDURE	root@localhost	2020-10-15 18:12:31	2020-10-15 18:12:31	DEFINER		utf8mb4	utf8mb4_0900_ai_ci	utf8mb4_0900_ai_ci
	kitaplik	kitapGrupla	PROCEDURE	root@localhost	2020-10-19 10:07:39	2020-10-19 10:07:39	DEFINER		utf8mb4	utf8mb4_0900_ai_ci	utf8mb4_0900_ai_ci
	kitaplik	kitapSayisi	PROCEDURE	root@localhost	2020-10-15 17:39:56	2020-10-15 17:39:56	DEFINER		utf8mb4	utf8mb4_0900_ai_ci	utf8mb4_0900_ai_ci
	kitaplik	kitap_getir	PROCEDURE	root@localhost	2020-10-15 17:37:10	2020-10-15 17:37:10	DEFINER		utf8mb4	utf8mb4_0900_ai_ci	utf8mb4_0900_ai_ci
	kitaplik	kutuphaneIDarttir	PROCEDURE	root@localhost	2020-10-19 10:35:21	2020-10-19 10:35:21	DEFINER		utf8mb4	utf8mb4_0900_ai_ci	utf8mb4_0900_ai_ci
	kitaplik	yazarList	PROCEDURE	root@localhost	2020-10-19 10:00:50	2020-10-19 10:00:50	DEFINER		utf8mb4	utf8mb4_0900_ai_ci	utf8mb4_0900_ai_ci

Saklı Yordamı Silmek

- Bir saklı yordamı veritabanından silmek için

```
DROP PROCEDURE [IF EXISTS] stored_procedure_name;
```

Saklı Yordamı Silmek



```
5 DROP PROCEDURE kitapGrup;
```

```
9 10:23:11 DROP PROCEDURE kitapGrup
```

Saklı Yordamlarda Kontrol Deyimleri

- MySQL'de aşağıdaki kontrol deyimleri bulunur:
 - IF koşul deyimi
 - CASE koşul deyimi
 - WHILE döngü deyimi
 - UNTIL döngü deyimi
 - LOOP döngü deyimi
 - LEAVE deyimi (diğer dillerdeki break benzeri.)
 - ITERATE deyimi (diğer dillerdeki continue benzeri.)
- VTYS'ye göre kullanım farklılıkları olabilir.

IF Koşulu

```
16 DELIMITER //
```

```
17 • CREATE PROCEDURE AdiSoyadi(IN sayi INT, OUT durum VARCHAR(50))
```

```
18 BEGIN
```

```
19
```

```
20 IF sayi = 1 THEN
```

```
21     SET durum = 'Yahya';
```

```
22 ELSEIF sayi = 2 THEN
```

```
23     SET durum = 'Kemal';
```

```
24 ELSEIF sayi = 3 THEN
```

```
25     SET durum = 'Yahya Kemal';
```

```
26 ELSE
```

```
27     SET durum = 'Yahya Kemal Beyatlı';
```

```
28 END IF;
```

```
29
```

```
30 END//
```

```
31 DELIMITER ;
```

```
32
```

```
33 • SET @sonuc = 0;
```

```
34 • CALL AdiSoyadi(99, @sonuc);
```

```
35 • SELECT @sonuc;
```

```
33 • SET @sonuc = 0;
```

```
34 • CALL AdiSoyadi(1, @sonuc);
```

```
35 • SELECT @sonuc;
```

```
36
```

```
37
```

Result Grid		Filter Rows:
	@sonuc	
▶	Yahya	

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	@sonuc			
▶	Yahya Kemal Beyatlı			

IF Koşulu

```
1 • USE world;
2
3 DELIMITER //
4 • CREATE PROCEDURE nufusYounlugu1(IN sehirID INT, OUT durum VARCHAR(50))
5 BEGIN
6
7     select Name,population from city where ID=sehirID;
8
9     IF (select population from city where ID=sehirID) > 100000 THEN
10         SET durum='Nüfus yoğunluğu yüksek';
11     ELSE
12         SET durum= 'Nüfus yoğunluğu düşük';
13     END IF;
14 END//
15 DELIMITER ;
```

```
16
17 • CALL nufusYounlugu1(56, @durum);
18 • SELECT @durum;
19 • Select nufusYounlugu1();
20
```

Result Grid		Filter Rows:	Expo
	@durum		
▶	Nüfus yoğunluğu yüksek		

```
16
17 • CALL nufusYounlugu1(56, @durum);
18 • SELECT @durum;
19 • Select nufusYounlugu1();
20
```

Result Grid		Filter Rows:	Export:	Wrap C
	Name	population		
▶	Luanda	2022000		

IF Koşulu

```
1 • use world;
2 DELIMITER //
3 • CREATE PROCEDURE countryPopulationCount(IN startPopulation INT, IN endPopulation INT, OUT result Varchar(250))
4 BEGIN
5 declare countryCount INT;
6 select count(*) into countryCount from country where population between startPopulation and endPopulation;
7
8 IF countryCount>0 and countryCount<10 THEN
9     SET result = 'Belirttiğiniz nüfus aralığında 10 dan az sayıda ülke var';
10 ELSEIF countryCount>=10 and countryCount<100 THEN
11     SET result = 'Belirttiğiniz nüfus aralığında 10-100 arasında ülke var';
12 ELSEIF countryCount>=100 and countryCount<500 THEN
13     SET result = 'Belirttiğiniz nüfus aralığında 100-500 arasında ülke var';
14 ELSE
15     SET result = 'Belirttiğiniz kriterlere göre bir sonuç bulunamamıştır.';
16 END IF;
17 END //
18
19 DELIMITER ;
20
21 • call countryPopulationCount(0,1500,@result);
22 • select @result;
23
```

Result Grid |   Filter Rows: | Export:  | Wrap Cell Content: 

haahaha
Belirttiğiniz nüfus aralığında 10-100 arasında ülk...

CASE Koşulu

```
24 DELIMITER //
```

```
25 • CREATE PROCEDURE AdiSoyadiBul(IN sayi INT, OUT durum VARCHAR(50))
```

```
26 BEGIN
```

```
27
```

```
28 CASE sayi
```

```
29 WHEN 1 THEN
```

```
30     SET durum = 'yahya';
```

```
31 WHEN 2 THEN
```

```
32     SET durum = 'kemal';
```

```
33 WHEN 3 THEN
```

```
34     SET durum = 'yahya kemal';
```

```
35 ELSE
```

```
36     SET durum = 'yahya kemal beyatlı';
```

```
37 END CASE;
```

```
38
```

```
39 END//
```

```
40 DELIMITER ;
```

```
41
```

```
42 • call AdiSoyadiBul(2,@durum);
```

```
43 • select @durum;
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
@durum				
kemal				

```
42 • call AdiSoyadiBul(9,@durum);
```

```
43 • select @durum;
```

Result Grid		Filter Rows:	Export:
@durum			
yahya kemal beyatlı			

WHILE Döngüsü

```
46
47 DELIMITER //
48 • CREATE PROCEDURE while_dongusu(IN sayi INT, INOUT sonuc TEXT)
49 BEGIN
50
51     DECLARE x INT;
52     SET x = 1;
53
54     WHILE x <= sayi DO
55         SET sonuc = CONCAT(sonuc, x, ',');
56         SET x = x + 1;
57     END WHILE;
58
59 END //
60 DELIMITER ;
61
62 • SET @sonuc = '';
63 • CALL while_dongusu(10, @sonuc);
64 • SELECT @sonuc;
```

<	
Result Grid	Filter Rows: <input type="text"/>
Export: Wrap Cell Content:	
@sonuc	
▶ 1,2,3,4,5,6,7,8,9,10,	

LOOP-LEAVE-ITERATIVE ifadesi

```
1  DELIMITER //
```

```
2  • CREATE PROCEDURE calcTotal ( IN starting_value INT , OUT result INT )
```

```
3  BEGIN
```

```
4      SET result = 0;
```

```
5  label1: LOOP
```

```
6      IF starting_value > 0 THEN
```

```
7          SET result = result + starting_value;
```

```
8          SET starting_value = starting_value - 1;
```

```
9          ITERATE label1;
```

```
10     END IF;
```

```
11     LEAVE label1;
```

```
12     END LOOP label1;
```

```
13 END; //
```



```
14 DELIMITER ;
```

```
15
```

```
16
```

```
17 • call calcTotal(5,@result);
```

```
18 • select @result
```

<
Result Grid
Filter Rows: <input type="text"/>
Export:  Wrap Cell Content: 
@result
15