

VERİTABANI TASARIMI

Öğretim Görevlisi A. Berika VAROL MALKOÇOĞLU

İçindekiler

- Veri Tipleri
- Aritmetik Operatörler
- Karakter fonksiyonları
- Matematiksel fonksiyonlar
- Tarih ve zaman fonksiyonları

Veri Tipleri

- MySQL çeşitli kategorilerde birçok veri tipini destekler:
 - Nümerik tipler, tarih ve zaman tipleri ve dizge/karakter (string/character) tipleri.
 - Veri tipleri, spesifik bir işlevi yerine getirmek için oluşturulmuştur.
 - Bazı tipler küçük miktarlardaki yazıyı saklamak için uygun iken bazı tipler büyük miktardaki yazıları, bazı tipler tarihleri ve diğer bazı tipler sayıları tutar.
- MySQL kullanma kılavuzu şöyle der: “En uygun depolama için, her zaman en hassas (uygun) veri tipini kullanmaya çalışmalısınız. Örneğin, tam sayı (integer) tipindeki bir sütun 1 ile 99999 arasındaki değerler için kullanılacaksa MEDIUMINT UNSIGNED kullanmak en iyisidir. Verilen değerler için diğer tüm tipler arasından bu tip, en az miktarda depolama alanı kullanır.”
- **Unsigned Nedir?**
 - Tüm integer tipler için kullanılabilir.
 - Bir alan unsigned olarak işaretlendiğinde sadece pozitif değer alabilir.
 - Örneğin; tinyint alan unsigned olarak işaretlendiğinde 0 ile 255 arasında bir değer alabilir.

Metinsel Veri Tipleri

- **Text (boyut) :**

- Belirtilen değerden az veri girilse bile boyutu kadar yer kaplar.
- Unicode'u desteklemez.
- En fazla 255 karakter alır.

- **Char (boyut) :**

- Belirtilen değerden az veri girilse bile boyut kadar yer kaplar.
- Unicode'u desteklemez.
- En fazla 8.000 karaktere kadar depolama yapar.

- **Nchar (boyut) :**

- Belirtilen değerden az veri girilse bile boyut kadar yer kaplar.
- En fazla 4.000 karaktere kadar depolama yapar.

Metinsel Veri Tipleri

- **Varchar (boyut) :**

- Chardan farkı, verinin boyutu kadar yer kaplamasıdır.
- Unicode'u desteklemez.
- En fazla 8.000 karaktere kadar depolama yapar.
- Boyut değeri max olursa, depolama boyutu $2^{31} - 2$ (2 GB) olur.

- **Nvarchar (boyut) :**

- Verinin boyutu kadar yer kaplar.
- Unicode'u destekler.
- En fazla 4.000 karaktere kadar depolama yapar.
- Boyut değeri max olursa, depolama boyutu $2^{31} - 2$ (2 GB) olur.

Metinsel Veri Tipleri

- **Tinytext :**

- En fazla 255 karaktere kadar olan boşluklar dahil metinsel ifadeleri saklar.

- **Blob :**

- Binary yani ikili verilerin saklanacağı durumlarda kullanılır.
- En fazla 65.535 byte (yaklaşık 64 KB) veri saklanabilir.

- **Mediumtext :**

- 6 milyon karaktere kadar metinsel ifadeleri depolayabilir.

Metinsel Veri Tipleri

Değer	CHAR (4)	Depolama Boyutu	VARCHAR (4)	Depolama Boyutu
' '	' '	4 bytes	' '	1 byte
'ab'	'ab '	4 bytes	'ab'	3 bytes
'abcd'	'abcd'	4 bytes	'abcd'	5 bytes
'abcdefgh'	'abcd'	4 bytes	'abcd'	5 bytes

Sayısal Veri Tipleri

- **Binary (boyut) :**
 - İkili bilgi saklar.
 - Belirtilen değerden az veri girilse bile boyut kadar yer kaplar.
 - En fazla 8.000 karakter depolama yapar.
- **Bit (boyut) :**
 - Bir byte uzunluğunda tamsayı veri tipidir.
 - Sadece 0, 1 veya Null değerlerini içerir.
- **int :**
 - Yaklaşık -2 milyar ile +2 milyar arasında değer alabilen tamsayı veri tipidir.
 - 4 byte büyüklüğünde yer kaplar.

Sayısal Veri Tipleri

- **Tinyint (boyut) :**
 - -128 ile 127 arasında değer alabilen tamsayı veri tipidir.
 - 1 byte büyüklüğünde yer kaplar.
- **Smallint (boyut) :**
 - -32768 ile 32767 arasında değer alabilen tamsayı veri tipidir.
 - 2 byte büyüklüğünde yer kaplar.
- **Mediumint (boyut) :**
 - Yaklaşık -8 milyar ile 8 milyar arasında değer alabilen tamsayı veri tipidir.
 - 3 byte büyüklüğünde yer kaplar.

Sayısal Veri Tipleri

- **Bigint (boyut) :**
 - -2^{63} ile $+2^{63}$ arasında değer alabilen tamsayı veri tipidir.
 - 8 byte büyüklüğünde yer kaplar.
- **Decimal :**
 - Boyutu değişken olmakla birlikte -10^{38} ile $+10^{38}$ arasında ondalık ve tamsayı türünde verileri saklayabilir.
- **Float :**
 - Boyutu ve doğruluğu (ondalık kısım duyarlılığı) aldığı parametreye göre değişen kayan noktalı sayılar için kullanılır.
 - Max boyut değeri 23'dür.
- **Double :**
 - Float tipinde daha büyük kayan noktalı sayıları almak için kullanılır.
 - Max boyut değeri 53'dür.

Tarih-Zaman Veri Tipleri

- **Date :**

- Tarihleri YYYY-MM-DD şeklinde tutar.
- 3 byte uzunluğunda veri tipidir.
- 0001-01-01 ile 9999-12-31 tarihleri arasındaki tüm değerleri tutabilir.

- **Datetime :**

- YYYY-MM-DD hh:mm:ss:mmm şeklinde tarih ve zaman verilerini tutar.
- 8 byte uzunluğunda veri tipidir.
- 1753-01-01 00:00:00.000 ile 9999-12-31 23:59:59.999 arası değerlerini saklar.

- **Year :**

- 2 veya 4 basamaklı yıl bilgisini tutar.
- 4 basamaklı verilerde 1901 – 2155 arasında değer saklar.
- 2 basamaklı verilerde ise 70 – 69 (1970 ile 2069) değerlerini saklar.

Tarih-Zaman Veri Tipleri

- **Timestamp :**

- Depolama biçimi DATETIME() ile aynıdır.
- Ancak yazdığınız program üzerinden bu alan veri gönderimi yapılmaz.
- O anki tarih ve saati otomatik olarak ekler
- . Eğer elle bir tarih-saat eklenecekse o zaman veritipi DATETIME() olarak ayarlanmalıdır.

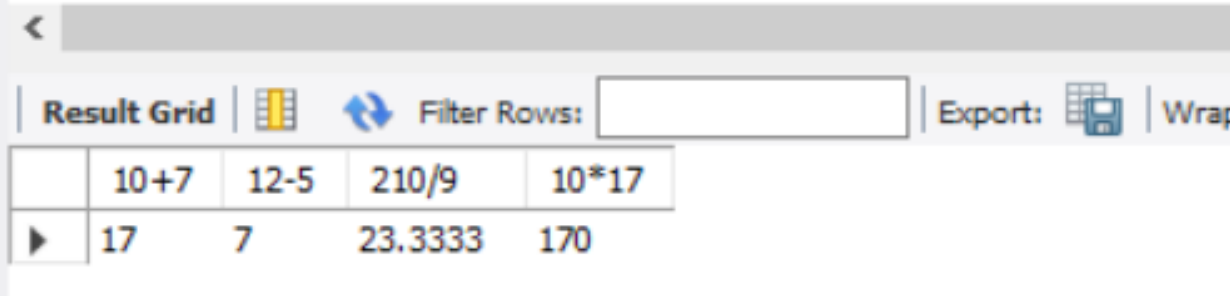
- **Time :**

- Sadece saat verilerini hh:mm:ss:nnnnnnnn şeklinde saklayan veri tipidir.
- 3 - 5 byte uzunluğunda veri tipidir.

Aritmetik Operatörler

- Sütunlar Üzerinde Matematiksel İşlevler;
 - () İşlemleri gruplandırma ve işlem önceliği belirleme
 - + Toplama
 - - Çıkarma
 - * Çarpma
 - / Bölme

```
10 • select 10+7, 12-5, 210/9, 10*17;  
11
```



The screenshot shows a SQL query execution interface. The query is `select 10+7, 12-5, 210/9, 10*17;`. Below the query, there is a "Result Grid" section. The grid has a header row with the expressions `10+7`, `12-5`, `210/9`, and `10*17`. The first row of data contains the results: 17, 7, 23.3333, and 170. The interface also includes a "Filter Rows" input field, an "Export" button, and a "Wrap" button.

	10+7	12-5	210/9	10*17
▶	17	7	23.3333	170

ASCII Nedir?

- Latin alfabesi üzerine kuruludur.
- İlk olarak ANSI tarafından 1963 yılında standart olarak sunulmuştur.
- 128 karakterin sayılarla eşleştirilmiş halidir.
- 128 adet sayı 7 bite karşılık gelir ($2^7=128$).
- Yani 7 bit ile gösterilebilecek son sayı 127'dir. Dolayısıyla ASCII 7 bitlik bir sistemdir.
- ASCII'de 32 tane basılmayan kontrol karakteri ve 95 tane basılan karakter yer alır.
- Kontrol karakterleri metnin akışını kontrol eden ve ekranda çıkmayan karakterlerdir.
- Basılan karakterlerse ekranda görünen ve okuduğumuz metni oluşturan karakterlerdir.

ASCII Tablosu

İkilik sistem	Sekizlik sistem	Onluk sistem	On altılık sistem	Karakter	İkilik sistem	Sekizlik sistem	Onluk sistem	On altılık sistem	Karakter	İkilik sistem	Sekizlik sistem	Onluk sistem	On altılık sistem	Karakter
010 0000	040	32	20	space	100 0000	100	64	40	@	110 0000	140	96	60	`
010 0001	041	33	21	!	100 0001	101	65	41	A	110 0001	141	97	61	a
010 0010	042	34	22	"	100 0010	102	66	42	B	110 0010	142	98	62	b
010 0011	043	35	23	#	100 0011	103	67	43	C	110 0011	143	99	63	c
010 0100	044	36	24	\$	100 0100	104	68	44	D	110 0100	144	100	64	d
010 0101	045	37	25	%	100 0101	105	69	45	E	110 0101	145	101	65	e
010 0110	046	38	26	&	100 0110	106	70	46	F	110 0110	146	102	66	f
010 0111	047	39	27	'	100 0111	107	71	47	G	110 0111	147	103	67	g
010 1000	050	40	28	(100 1000	110	72	48	H	110 1000	150	104	68	h
010 1001	051	41	29)	100 1001	111	73	49	I	110 1001	151	105	69	i
010 1010	052	42	2A	*	100 1010	112	74	4A	J	110 1010	152	106	6A	j
010 1011	053	43	2B	+	100 1011	113	75	4B	K	110 1011	153	107	6B	k
010 1100	054	44	2C	,	100 1100	114	76	4C	L	110 1100	154	108	6C	l
010 1101	055	45	2D	-	100 1101	115	77	4D	M	110 1101	155	109	6D	m
010 1110	056	46	2E	.	100 1110	116	78	4E	N	110 1110	156	110	6E	n
010 1111	057	47	2F	/	100 1111	117	79	4F	O	110 1111	157	111	6F	o
011 0000	060	48	30	0	101 0000	120	80	50	P	111 0000	160	112	70	p
011 0001	061	49	31	1	101 0001	121	81	51	Q	111 0001	161	113	71	q
011 0010	062	50	32	2	101 0010	122	82	52	R	111 0010	162	114	72	r
011 0011	063	51	33	3	101 0011	123	83	53	S	111 0011	163	115	73	s
011 0100	064	52	34	4	101 0100	124	84	54	T	111 0100	164	116	74	t
011 0101	065	53	35	5	101 0101	125	85	55	U	111 0101	165	117	75	u
011 0110	066	54	36	6	101 0110	126	86	56	V	111 0110	166	118	76	v
011 0111	067	55	37	7	101 0111	127	87	57	W	111 0111	167	119	77	w
011 1000	070	56	38	8	101 1000	130	88	58	X	111 1000	170	120	78	x
011 1001	071	57	39	9	101 1001	131	89	59	Y	111 1001	171	121	79	y
011 1010	072	58	3A	[:]	101 1010	132	90	5A	Z	111 1010	172	122	7A	z
011 1011	073	59	3B	;	101 1011	133	91	5B	[111 1011	173	123	7B	{
011 1100	074	60	3C	[<]	101 1100	134	92	5C	\	111 1100	174	124	7C	
011 1101	075	61	3D	=	101 1101	135	93	5D]	111 1101	175	125	7D	}
011 1110	076	62	3E	[>]	101 1110	136	94	5E	^	111 1110	176	126	7E	~
011 1111	077	63	3F	?	101 1111	137	95	5F	_					

Karakter Fonksiyonları

- Char() fonksiyonu ile ASCII karakterleri öğrenilebilir.

```
8 • SELECT CHAR(67 USING utf8);
```

Result Grid	Filter Rows:	Export:
CHAR(67 USING utf8)		
C		


```
8 • SELECT CHAR(67, 97, 116 USING utf8);
```

Result Grid	Filter Rows:	Export:	Wrap Cell
CHAR(67, 97, 116 USING utf8)			
Cat			

- ASCII komutu ile harflerin sayısal değerleri öğrenilebilir

```
4 • SELECT ASCII ('C') ;
```

Result Grid	Filter Rows:
ASCII ('C')	
67	

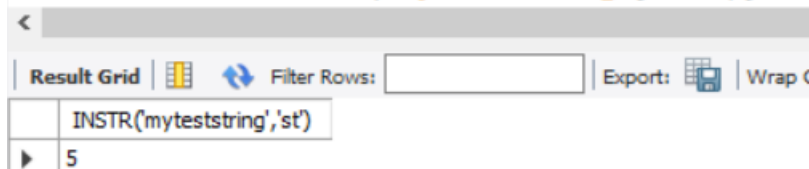

```
4 • SELECT ASCII ('C'), ASCII ('a'), ASCII ('t') ;
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
ASCII ('C')	ASCII ('a')	ASCII ('t')	
67	97	116	

Karakter Fonksiyonları

- INSTR () fonksiyonu bağımsız değişken olarak bir kaydı alır ve kayıttaki alt dizenin ilk oluşumunun konumunu gösteren bir tamsayı döndürür.

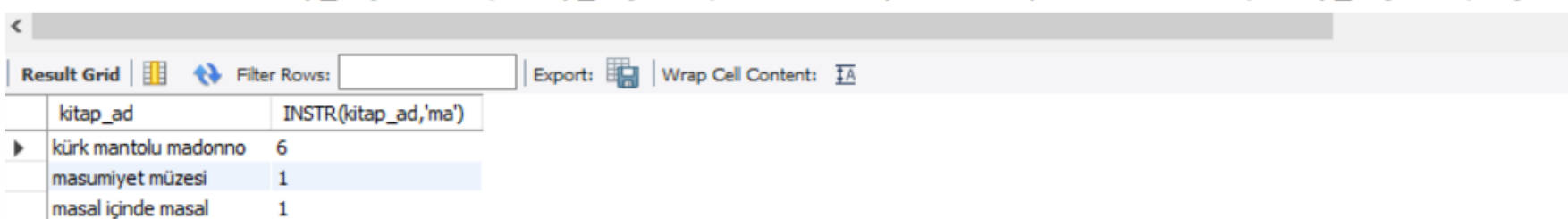
```
13 • SELECT INSTR('myteststring','st');
```



The screenshot shows a database query result interface. At the top, the query is: `13 • SELECT INSTR('myteststring','st');`. Below the query, there is a toolbar with options: "Result Grid", "Filter Rows:" (with a search box), "Export:" (with a download icon), and "Wrap C". The main area displays a table with one row:

INSTR('myteststring','st')
5

```
15 • SELECT kitap_ad, INSTR(kitap_ad,'ma') FROM kitaplik.kitap WHERE INSTR(kitap_ad,'ma')>0;
```



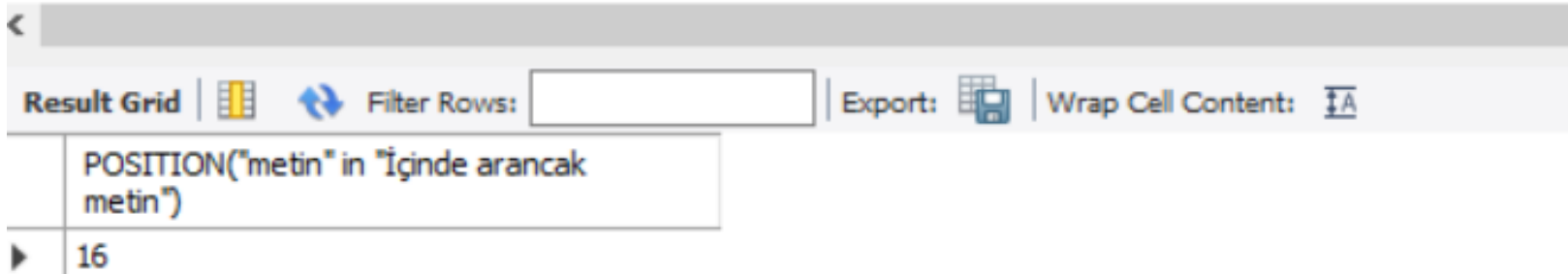
The screenshot shows a database query result interface. At the top, the query is: `15 • SELECT kitap_ad, INSTR(kitap_ad,'ma') FROM kitaplik.kitap WHERE INSTR(kitap_ad,'ma')>0;`. Below the query, there is a toolbar with options: "Result Grid", "Filter Rows:" (with a search box), "Export:" (with a download icon), and "Wrap Cell Content: I A". The main area displays a table with three rows:

kitap_ad	INSTR(kitap_ad,'ma')
kürk mantolu madonno	6
masumiyet müzesi	1
masal içinde masal	1

Karakter Fonksiyonları

- POSITION() komutu metin içerisinde aranmak istenilen metni arar.

```
17
18 • SELECT POSITION("metin" in "İçinde arancak metin");
19
```



The screenshot shows a SQL query execution interface. The query is: `SELECT POSITION("metin" in "İçinde arancak metin");`. The result is displayed in a table with one row and one column. The value in the cell is 16. The interface includes a toolbar with options like 'Result Grid', 'Filter Rows', 'Export', and 'Wrap Cell Content'.




POSITION("metin" in "İçinde arancak metin")
16

Karakter Fonksiyonları

- LEFT() Fonksiyonu metin içinde sol taraftan başlayarak istenilen uzunluktaki metni alır.

```
22 • SELECT LEFT("ayşeberikavarolmalkoçoğlu",4);  
23
```

<

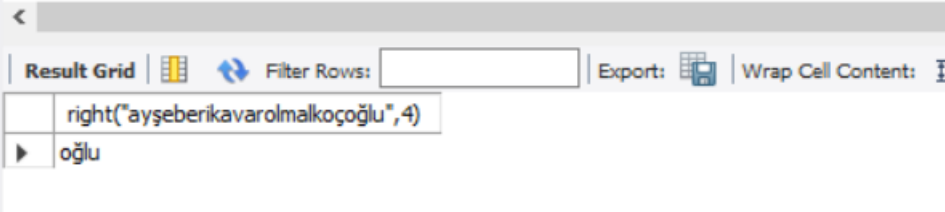
Result Grid |  Filter Rows: | Export:  | Wrap Cell Content: 

	LEFT("ayşeberikavarolmalkoçoğlu",4)
▶	ayşe

Karakter Fonksiyonları

- RIGHT() Fonksiyonu metin içinde sağ taraftan başlayarak istenilen uzunluktaki metni alır.

```
23
24 • SELECT right("ayşeberikavarolmalkoçoğlu",4);
25
```






right("ayşeberikavarolmalkoçoğlu",4)	
▶	oğlu

Karakter Fonksiyonları

- SUBSTRING fonksiyonu bir metinde istenen bölümden istenen bir parçayı almak için kullanılır.

```
77 • SELECT SUBSTRING('foobarbar' FROM 4);
```

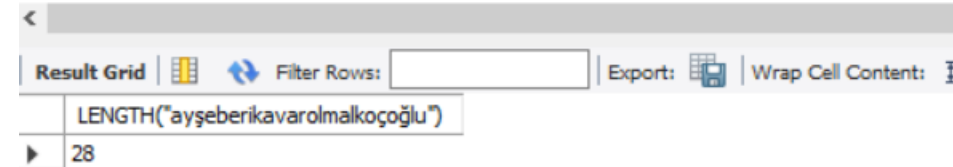
<   Filter Rows: | Export:  | Wrap On

	SUBSTRING('foobarbar' FROM 4)
▶	barbar

Karakter Fonksiyonları

- LENGTH() fonksiyonu uzunluk verir.

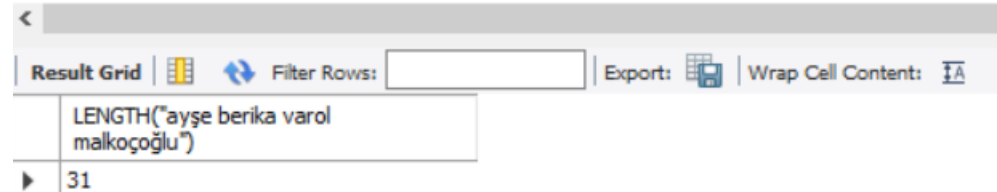
```
24 • SELECT LENGTH("ayşeberikavarolmalkoçoğlu");
25
```



The screenshot shows a SQL query result in a table. The table has one column with the header `LENGTH("ayşeberikavarolmalkoçoğlu")` and one row with the value `28`. The interface includes a "Result Grid" tab, a "Filter Rows" input field, and buttons for "Export" and "Wrap Cell Content".

LENGTH("ayşeberikavarolmalkoçoğlu")
28

```
24 • SELECT LENGTH("ayşe berika varol malkoçoğlu");
25
```



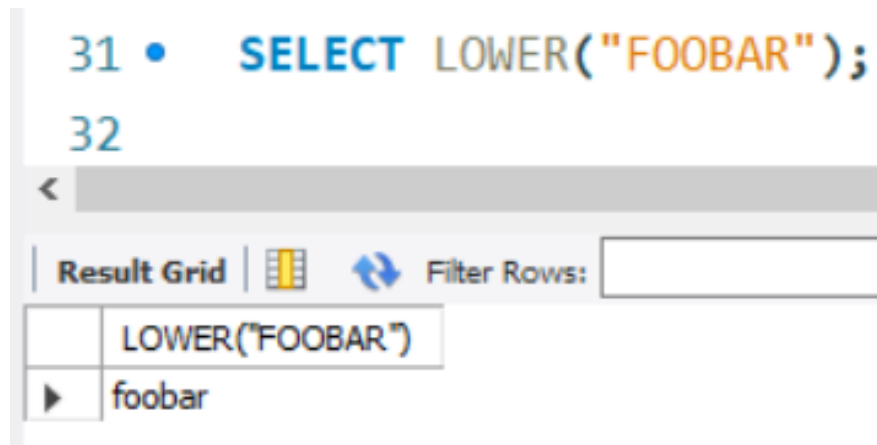
The screenshot shows a SQL query result in a table. The table has one column with the header `LENGTH("ayşe berika varol malkoçoğlu")` and one row with the value `31`. The interface includes a "Result Grid" tab, a "Filter Rows" input field, and buttons for "Export" and "Wrap Cell Content".

LENGTH("ayşe berika varol malkoçoğlu")
31

Karakter Fonksiyonları

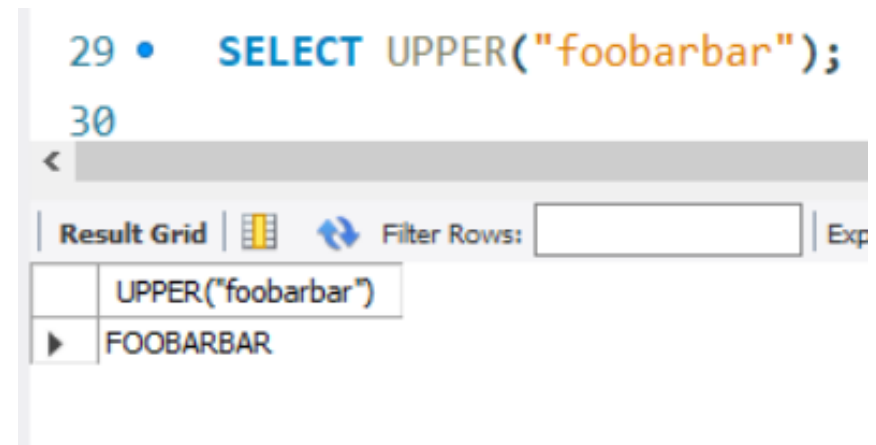
- Lower () , Upper() Fonksiyonu Küçük ve büyük harf yapmak için kullanılır.

```
31 • SELECT LOWER("FOOBAR");  
32
```



Result Grid	Filter Rows:
LOWER("FOOBAR")	
▶ foobar	

```
29 • SELECT UPPER("foobarbar");  
30
```



Result Grid	Filter Rows:
UPPER("foobarbar")	
▶ FOOBARBAR	

Karakter Fonksiyonları

- LTRIM() fonksiyonu metnin solundaki boşlukları silmek için kullanılır.

```
33 • SELECT (" foobarbar");
```

Result Grid | Filter Rows: | Export:

	foobarbar
▶	foobarbar

```
35 • SELECT LTRIM(" foobarbar");
```

36

Result Grid | Filter Rows: | Export:


	LTRIM(" foobarbar")
▶	foobarbar

Karakter Fonksiyonları

- RTRIM() fonksiyonu metnin sağındaki boşlukları silmek için kullanılır.

```
75 • SELECT RTRIM('foobarbar ');
```

<

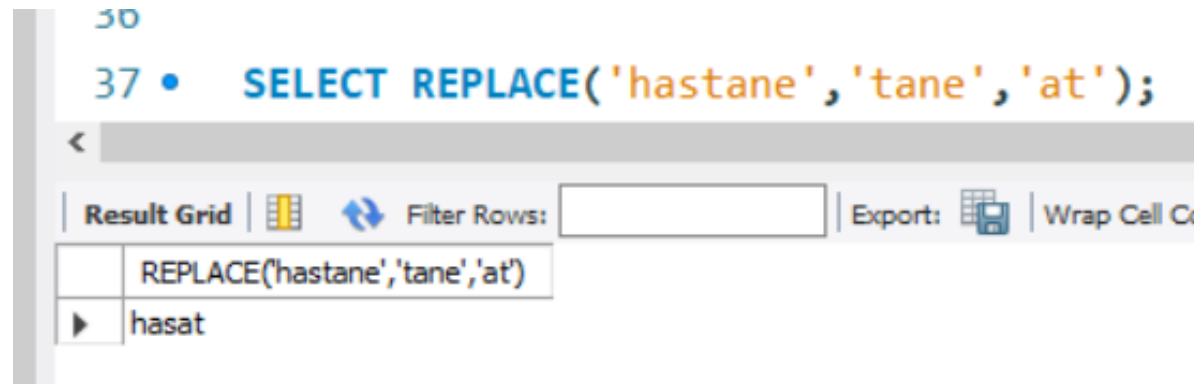
Result Grid |  Filter Rows: Export:

	RTRIM('foobarbar ')
▶	foobarbar

Karakter Fonksiyonları

- REPLACE fonksiyonu parametre olarak verilen birinci metin içerisinde ikinci metni gördüğü yerde ikinci metni silerek verilen üçüncü parametredeki metni yerleştirir.

```
30
37 • SELECT REPLACE('hastane', 'tane', 'at');
```



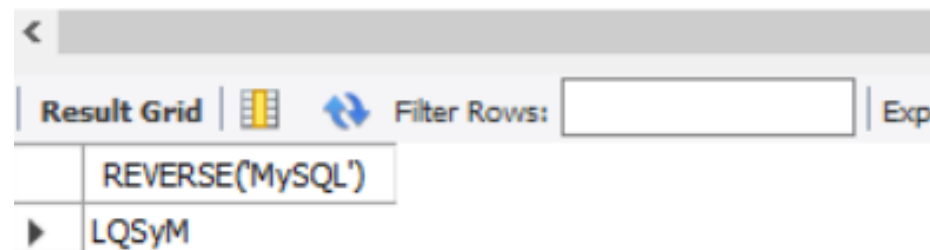
The screenshot shows a SQL query execution interface. The query is `SELECT REPLACE('hastane', 'tane', 'at');`. Below the query, there is a toolbar with options like "Result Grid", "Filter Rows", "Export", and "Wrap Cell Co". The result is displayed in a table with one row containing the value "hasat".

	REPLACE('hastane','tane','at')
▶	hasat

Karakter Fonksiyonları

- REVERSE() fonksiyonu verilen metni ters çevirmek için kullanılır.

```
36  
37 • SELECT REVERSE('MySQL');  
38
```



The screenshot shows a SQL query execution interface. At the top, the query `SELECT REVERSE('MySQL');` is entered. Below the query, the results are displayed in a table. The table has one row with the value `REVERSE('MySQL')`. To the right of the table, there is a 'Filter Rows' input field and an 'Exp' button. The table is titled 'Result Grid'.

Result Grid
REVERSE('MySQL')



Matematiksel Fonksiyonlar

- POWER() / POW() bir sayının verilen diğer sayıya göre üstünü alır.
- SELECT POWER(P1,P2)
 - P1 parametresi ile girilen sayının P2 parametresi ile girilen sayı kadar kuvvetini alır.

```
43 • SELECT POWER(4, 2);
```

```
44
```

<


Result Grid   Filter Rows:


	POWER(4, 2)
▶	16

```
43 • SELECT POW(4, 2);
```

```
44
```

<

Result Grid 

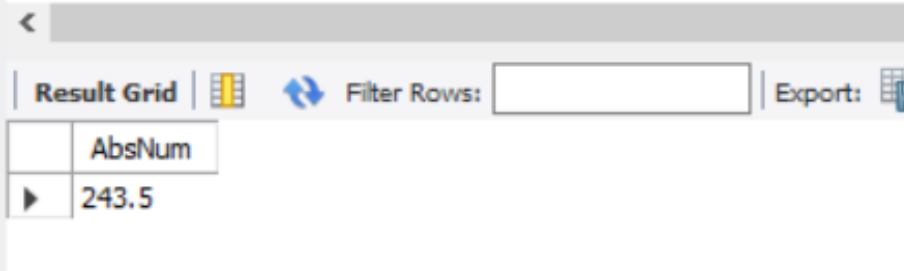
 Filter Rows:

	POW(4, 2)
▶	16

Matematiksel Fonksiyonlar

- ABS () fonksiyonu Sayının mutlak değerini döndürür.

```
45 • SELECT Abs(-243.5) AS AbsNum;  
46
```

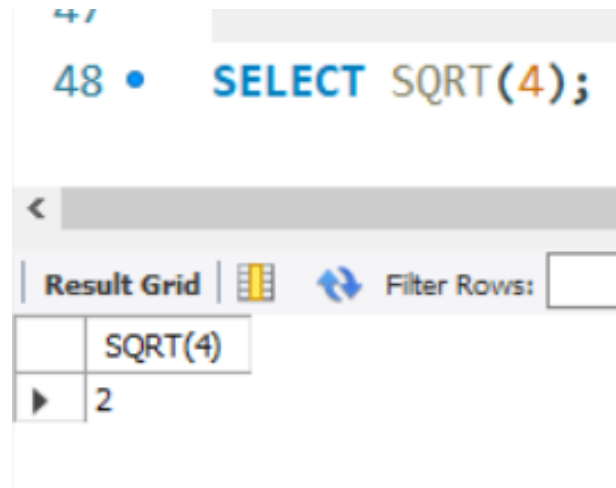


The screenshot shows a SQL query execution interface. The query is `SELECT Abs(-243.5) AS AbsNum;`. Below the query, there is a "Result Grid" tab with a table containing one row and one column. The column is labeled "AbsNum" and the value is "243.5".

AbsNum
243.5

Matematiksel Fonksiyonlar

- SQRT () fonksiyonu sayının kare kökünü alır.



Matematiksel Fonksiyonlar

- RAND () fonksiyonu 0 ve 1 arasında rastgele ondalıklı değer üretir.

```
50 • SELECT RAND( );
```

51

Result Grid		Filter Rows:
	RAND()	
▶	0.3097211702817317	

```
52 • SELECT RAND()*(10-5)+5;
```

53

Result Grid		Filter Rows:	Export
	RAND()*(10-5)+5		
▶	7.845261416253877		

Matematiksel Fonksiyonlar

- ROUND () fonksiyonu yuvarlama fonksiyonudur.

ROUND(P1,P2)

- P1 = Her hangi bir tam veya ondalıklı sayı yazılır.
- P2 = Sayının ondalıklı kısmının kaç basamaklı olmasını istiyorsak o sayı girilir.
Sayı negatif girilirse tam sayı kısmını da yuvarlar

```
52 • SELECT ROUND(235.415456, 2) AS RoundValue;  
53
```

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

RoundValue
235.42

Tarih ve Zaman Fonksiyonları

- DATE_ADD () fonksiyonu yazılan parametre cinsinden zamanı belirtilen ölçüde arttırır.

```
58 • SELECT DATE_ADD('2018-05-01',INTERVAL 1 YEAR);  
59
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
DATE_ADD('2018-05-01',INTERVAL 1 YEAR)			
2019-05-01			





```
56 • SELECT DATE_ADD('2018-05-01',INTERVAL 1 DAY);  
57
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
DATE_ADD('2018-05-01',INTERVAL 1 DAY)			
2018-05-02			

Tarih ve Zaman Fonksiyonları



- DATE_SUB () fonksiyonu yazılan parametre cinsinden zamanı belirtilen ölçüde azaltır.

```
61 • SELECT DATE_SUB('2018-05-01',INTERVAL 1 YEAR);  
62
```

<   Filter Rows: | Export:  | Wrap Cell Content: 

DATE_SUB('2018-05-01',INTERVAL 1 YEAR)
▶ 2017-05-01

```
63 • SELECT DATE_SUB('1998-01-02', INTERVAL 31 DAY);  
64
```

<   Filter Rows: | Export:  | Wrap Cell Content: 




DATE_SUB('1998-01-02', INTERVAL 31 DAY)
▶ 1997-12-02

Tarih ve Zaman Fonksiyonları

- DATEIFF() fonksiyonu iki tarih arasındaki zamanı döner.

```
65 • SELECT DATEDIFF('2017-10-31 23:59:59','2007-12-30');  
66
```

<

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

	DATEDIFF('2017-10-31 23:59:59','2007-12-30')
▶	3593

Tarih ve Zaman Fonksiyonları

- DATENAME() fonksiyonu yazılan tarihin gününü döner.
- DAYOFMONTH() fonksiyonu yazılan tarihin ayın kaçınıcı günü olduğunu döner.

```
67 • SELECT DAYNAME('2020-08-12');
```

```
68
```

DAYNAME('2020-08-12')
Wednesday

```
69 • SELECT DAYOFMONTH('2020-08-12');
```

```
70
```

DAYOFMONTH('2020-08-12')
12

Tarih ve Zaman Fonksiyonları

- DAYOFWEEK() fonksiyonu yazılan tarihin haftanın kaçıncı günü olduğunu döner.
- DAYOFYEAR() fonksiyonu yazılan tarihin yılın kaçıncı günü olduğunu döner.

71 • `SELECT DAYOFWEEK('2020-08-12');`

Result Grid		Filter Rows:	Export:
	DAYOFWEEK('2020-08-12')		
▶	4		

73 • `SELECT DAYOFYEAR('2020-08-12');`

Result Grid		Filter Rows:	Export:
	DAYOFYEAR('2020-08-12')		
▶	225		