

LAPORAN
PRAKTEK BASIS DATA
FUNGSI STRING & DATE TIME



SEMESTER 3

SULIS TIYAH

2001081002

JURUSAN TEKNOLOGI INFORMASI
PROGRAM STUDI TEKNIK KOMPUTER
POLITEKNIK NEGERI PADANG

2021

A. DASAR TEORI

Fungsi String ialah fungsi yang digunakan untuk melakukan manipulasi data teks (string). Database MySQL juga memiliki fungsi string yang dapat digunakan untuk memanipulasi setiap data yang akan Anda olah.

Fungsi	Keterangan	Struktur
LCASE atau LOWER	Konversi String menjadi huruf kecil.	SELECT LCASE(String) atau SELECT LOWER(String)
UCASE atau UPPER	Konversi String menjadi huruf besar.	SELECT UCASE(String) atau SELECT UPPER(String)
ASCII atau ORD	Konversi bilangan kedalam bentuk ASCII	SELECT ASCII('karakter') atau SELECT ORD ('karakter')
HEX	Untuk mengkonversi dari suatu karakter (angka dan huruf) ke bilangan HEXADESIMAL	SELECT HEX('karakter')
CHAR	Untuk mengkonversi bilangan desimal ke dalam bilangan biner	SELECT CHAR ('bilangan desimal')
BIN	Untuk mengkonversi bilangan desimal ke dalam bilangan biner	SELECT BIN(bilangan desimal)
LENGTH Atau CHAR_LENGTH	Untuk mengetahui panjang suatu string atau teks	SELECT LENGTH(String) atau SELECT CHAR_LENGTH(String)
LOCATE	Untuk mengetahui letak suatu substring (bagian string)	SELECT LOCATE(substring, string)
SPACE	Untuk membuat spasi kosong digunakan	SELECT SPACE(jumlah)

REPLACE	Untuk mengganti suatu karakter dengan karakter lainnya	SELECT REPLACE(String, String asal, String pengganti)
REPEAT	Untuk membuat string berulang sebanyak jumlah yang ditentukan	SELECT REPEAT(String, jumlah)
REVERSE	Untuk membalik susunan karakter (seakan dibaca dari belakang)	SELECT REVERSE(String)
QUOTE	Untuk menampilkan data yang didalamnya terdapat tanda petik tunggal maupun petik ganda	SELECT QUOTE(string)
CONCAT	Untuk menggabungkan string atau teks di MySQL dapat menggunakan fungsi	SELECT CONCAT(teks1, teks2)
FORMAT	digunakan untuk memformat angka.	SELECT FORMAT(nilai)
SUBSTRING atau SUBSTR	digunakan untuk mengambil sebagian string/ memotong string membutuhkan 2 atau 3 argumen	SELECT SUBSTRING (argumen1, argumen2)
LEFT dan RIGHT	digunakan untuk mengambil sebagian string dari posisi kiri (LEFT) dan kanan (RIGHT)	SELECT LEFT(argumen1, argumen2) dan SELECT RIGHT(argumen1, argumen2)

Fungsi date time ialah fungsi yang digunakan untuk melakukan manipulasi date (tanggal) dan time (waktu) pada database MySQL.

Fungsi	Keterangan	Struktur
CURRENT_TIMESTAMP, LOCALTIME, LOCALTIMESTAMP, NOW	untuk menampilkan waktu saat ini (real time) / sesuai dengan waktu dikomputer server.	SELECT CURRENT_TIMESTAMP(); SELECT LOCALTIME(); SELECT LOCALTIMESTAMP(); SELECT NOW();
DATE dan TIME	untuk menampilkan bagian tanggal/waktu dari input date/time. dengan argumen berupa nilai tanggal/field dengan tipe date/time.	SELECT DATE(); SELECT TIME();
CURDATE atau CURRENT_DATE	menampilkan tanggal real/saat ini = date(NOW())	SELECT CURDATE()
MONTH dan YEAR	menampilkan bulan dan tahun. bisa diikuti oleh fungsi lain dan bisa diikuti oleh nilai tanggal/field yang bertipe date.	SELECT YEAR(NOW()) SELECT MONTH(NOW())
MONTHNAME dan DAYNAME	menampilkan nama dan bulan nama hari dari sebuah tanggal.	SELECT MONTHNAME(NOW())
ADDDATE atau DATE_ADD	menampilkan hasil penambahan tanggal dan waktu	ADDDATE(DATE, INTERVAL jumlah_penambahan); DATE_ADD(DATE, INTERVAL jumlah_penambahan);
SUBDATE atau DATE_SUB	digunakan untuk proses pengurangan tanggal dan waktu	SUBDATE(DATE, INTERVAL jumlah_penambahan);

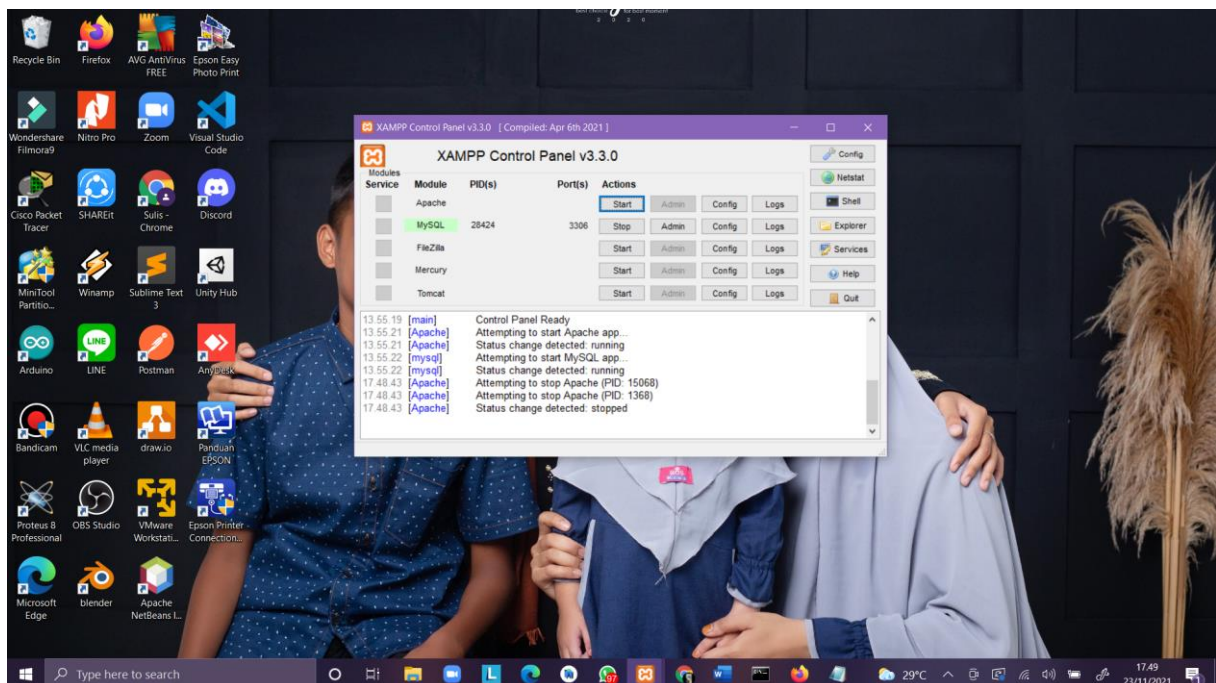
		DATE_SUB(DATE, INTERVAL jumlah_penambahan);
DATEDIFF	digunakan untuk menghitung selisih 2 buah tanggal.	SELECT DATEDIFF()

B. ALAT DAN BAHAN

1. PC / Laptop
2. Windows
3. Software XAMPP Control Panel

C. LANGKAH KERJA

1. Jalankan XAMPP pada MySQL.



2. Fungsi String

🚦 UCASE() atau UPPER()

```
MariaDB [mysql]> SELECT UCASE('Selamat Pagi PNP');
+-----+
| UCASE('Selamat Pagi PNP') |
+-----+
| SELAMAT PAGI PNP          |
+-----+
1 row in set (0.001 sec)

MariaDB [mysql]>
MariaDB [mysql]> SELECT UPPER('Teknik Komputer');
+-----+
| UPPER('Teknik Komputer') |
+-----+
| TEKNIK KOMPUTER          |
+-----+
1 row in set (0.000 sec)
```

🚦 LCASE atau LOWER()

```
MariaDB [mysql]> SELECT LCASE('Selamat Pagi PNP');
+-----+
| LCASE('Selamat Pagi PNP') |
+-----+
| selamat pagi pnp          |
+-----+
1 row in set (0.004 sec)

MariaDB [mysql]> SELECT LOWER('Teknik Komputer');
+-----+
| LOWER('Teknik Komputer') |
+-----+
| teknik komputer          |
+-----+
1 row in set (0.001 sec)
```

```
MariaDB [dbpenjualan]> SELECT UCASE(nmlgn) AS nama, LCASE(almlgn) AS alamat from tpelanggan;
+-----+-----+
| nama      | alamat      |
+-----+-----+
| GALAXY MEDIA | birugo      |
| GAMATECHNO  | anduring    |
| CITRA MOZAIC | jl. veteran |
| JJ ENT      | khatib sulaiman |
| GREEN MEDIA  | jambu air   |
| TK. INTAN    | jl.s.hatta  |
| ONEXINDO     | pauh        |
| RUDI         | NULL        |
| CITRA        | NULL        |
+-----+-----+
9 rows in set (0.012 sec)
```

CONCAT()

```
MariaDB [mysql]> SELECT CONCAT('Teknik','Komputer');
+-----+
| CONCAT('Teknik','Komputer') |
+-----+
| TeknikKomputer              |
+-----+
1 row in set (0.000 sec)

MariaDB [mysql]> SELECT CONCAT('Teknik',' ','Komputer');
+-----+
| CONCAT('Teknik',' ','Komputer') |
+-----+
| Teknik Komputer                |
+-----+
1 row in set (0.000 sec)

MariaDB [mysql]> SELECT LCASE(CONCAT('Teknik',' ','Komputer'));
+-----+
| LCASE(CONCAT('Teknik',' ','Komputer')) |
+-----+
| teknik komputer                  |
+-----+
1 row in set (0.000 sec)
```

```
MariaDB [classicmodels]> SELECT CONCAT(contactFirstName,' ',contactLastName) AS ContactFullName, country
-> FROM customers
-> WHERE country like '%USA%'
-> ORDER BY ContactFullName limit 10;
+-----+-----+
| ContactFullName | country |
+-----+-----+
| Allen Nelson   | USA    |
| Brian Chandler | USA    |
| Dan Lewis      | USA    |
| Dorothy Young  | USA    |
| Francisca Cervantes | USA    |
| Jean King      | USA    |
| Jeff Young     | USA    |
| Jerry Tseng    | USA    |
| Julie Brown    | USA    |
| Julie King     | USA    |
+-----+-----+
10 rows in set (0.024 sec)
```

FORMAT()

```
MariaDB [mysql]> SELECT FORMAT(7526000.34567, 2);
+-----+
| FORMAT(7526000.34567, 2) |
+-----+
| 7,526,000.35             |
+-----+
1 row in set (0.002 sec)

MariaDB [mysql]> SELECT FORMAT(7526000.341212, 4);
+-----+
| FORMAT(7526000.341212, 4) |
+-----+
| 7,526,000.3412            |
+-----+
1 row in set (0.000 sec)
```

```

MariaDB [dbpenjualan]> SELECT tgl, kdbrg, jml, CONCAT('Rp.', ' ',FORMAT(hrg,2)) Harga
-> FROM ttrans;
+-----+-----+-----+-----+
| tgl      | kdbrg | jml  | Harga      |
+-----+-----+-----+-----+
| 2008-09-01 | B001  | 5    | Rp. 750,000.00 |
| 2008-09-01 | B002  | 5    | Rp. 175,000.00 |
| 2008-09-01 | B002  | 5    | Rp. 175,000.00 |
| 2008-09-01 | B003  | 20   | Rp. 25,000.00  |
| 2008-09-01 | B002  | 15   | Rp. 175,000.00 |
| 2008-09-01 | B005  | 10   | Rp. 50,000.00  |
| 2008-09-02 | B004  | 10   | Rp. 220,000.00 |
| 2008-09-02 | B001  | 20   | Rp. 750,000.00 |
| 2008-09-02 | B002  | 20   | Rp. 175,000.00 |
| 2008-09-03 | B002  | 30   | Rp. 225,000.00 |
| 2008-09-03 | B005  | 30   | Rp. 50,000.00  |
| 2008-10-01 | B001  | 10   | Rp. 750,000.00 |
| 2008-10-10 | B001  | 5    | Rp. 175,000.00 |
| 2007-01-10 | B001  | 5    | Rp. 750,000.00 |
| 2007-01-10 | B002  | 1    | Rp. 175,000.00 |
| 2007-01-10 | B003  | 3    | Rp. 25,000.00  |
| 2006-02-06 | B001  | 12   | Rp. 750,000.00 |
| 2005-03-01 | B001  | 10   | Rp. 750,000.00 |
| 2005-03-01 | B004  | 15   | Rp. 220,000.00 |
| 2006-05-31 | B001  | 40   | Rp. 750,000.00 |
+-----+-----+-----+-----+
20 rows in set (0.005 sec)

```

SUBSTRING() atau SUBSTR()

```

MariaDB [mysql]> SELECT SUBSTRING('Teknik Komputer',4);
+-----+
| SUBSTRING('Teknik Komputer',4) |
+-----+
| nik Komputer                    |
+-----+
1 row in set (0.000 sec)

MariaDB [mysql]> SELECT SUBSTRING('Teknik Komputer',4,3);
+-----+
| SUBSTRING('Teknik Komputer',4,3) |
+-----+
| nik                               |
+-----+
1 row in set (0.003 sec)

MariaDB [mysql]> SELECT SUBSTRING('Teknik Komputer',4,6);
+-----+
| SUBSTRING('Teknik Komputer',4,6) |
+-----+
| nik Ko                           |
+-----+
1 row in set (0.000 sec)

MariaDB [mysql]> use dbpenjualan;
Database changed
MariaDB [dbpenjualan]> SELECT SUBSTRING(nmbrg,5,10) AS NamaBarang FROM tbrg;
+-----+
| NamaBarang |
+-----+
| uter Penti |
| book ASUS  |
| et         |
| isk Ext WD |
| disk Kings |
| tor Samsun |
| e Wireless |
+-----+
7 rows in set (0.009 sec)

```


LEFT() dan RIGHT()

```
MariaDB [mysql]> SELECT LEFT('Teknik Komputer',5);
```

```
+-----+
| LEFT('Teknik Komputer',5) |
+-----+
| Tekni                     |
+-----+
1 row in set (0.001 sec)
```

```
MariaDB [mysql]> SELECT RIGHT('Teknik Komputer',5);
```

```
+-----+
| RIGHT('Teknik Komputer',5) |
+-----+
| puter                      |
+-----+
1 row in set (0.000 sec)
```

```
MariaDB [dbpenjualan]> SELECT nmbrg AS NamaBarang
```

```
-> FROM tbrg
```

```
-> WHERE LEFT(nmbrg,2) = 'Mo';
```

```
+-----+
| NamaBarang                |
+-----+
| Monitor Samsung 15 Inch  |
| Mouse Wireless Logitech  |
+-----+
2 rows in set (0.004 sec)
```

```
MariaDB [classicmodels]> SELECT LEFT(customerNumber,2), RIGHT(checkNumber,4), SUBSTRING(paymentDate,6,2), FORMAT(amount,0)
-> FROM payments
-> WHERE amount <= 50000
-> ORDER BY amount DESC limit 10;
```

```
+-----+-----+-----+-----+
| LEFT(customerNumber,2) | RIGHT(checkNumber,4) | SUBSTRING(paymentDate,6,2) | FORMAT(amount,0) |
+-----+-----+-----+-----+
| 35                     | 1193                | 01                         | 49,706           |
| 14                     | 3410                | 01                         | 49,615           |
| 14                     | 5280                | 10                         | 49,539           |
| 11                     | 4694                | 02                         | 49,524           |
| 40                     | 7687                | 01                         | 49,165           |
| 39                     | 4275                | 11                         | 48,928           |
| 44                     | 2727                | 09                         | 48,810           |
| 18                     | 4726                | 03                         | 48,426           |
| 46                     | 6846                | 11                         | 48,356           |
| 24                     | 4084                | 09                         | 48,299           |
+-----+-----+-----+-----+
10 rows in set (0.004 sec)
```

3. FUNGSI DATE / TIME

📌 CURRENT_TIMESTAMP(), LOCALTIME(),
LOCALTIMESTAMP(), NOW()

```
MariaDB [mysql]> SELECT CURRENT_TIMESTAMP();
+-----+
| CURRENT_TIMESTAMP() |
+-----+
| 2021-11-23 18:32:10 |
+-----+
1 row in set (0.003 sec)

MariaDB [mysql]> SELECT LOCALTIME();
+-----+
| LOCALTIME() |
+-----+
| 2021-11-23 18:32:16 |
+-----+
1 row in set (0.000 sec)

MariaDB [mysql]> SELECT LOCALTIMESTAMP();
+-----+
| LOCALTIMESTAMP() |
+-----+
| 2021-11-23 18:32:21 |
+-----+
1 row in set (0.000 sec)

MariaDB [mysql]> SELECT NOW();
+-----+
| NOW() |
+-----+
| 2021-11-23 18:32:27 |
+-----+
1 row in set (0.000 sec)
```

📌 DATE() dan TIME()

```
MariaDB [mysql]> SELECT DATE(NOW());
+-----+
| DATE(NOW()) |
+-----+
| 2021-11-23 |
+-----+
1 row in set (0.000 sec)

MariaDB [mysql]> SELECT TIME(NOW());
+-----+
| TIME(NOW()) |
+-----+
| 18:35:43 |
+-----+
1 row in set (0.001 sec)
```

🚦 CURDATE() atau CURRENT_DATE()

```
MariaDB [mysql]> SELECT CURDATE();
+-----+
| CURDATE() |
+-----+
| 2021-11-23 |
+-----+
1 row in set (0.001 sec)
```

🚦 CURTIME() atau CURRENT_TIME()

```
MariaDB [mysql]> SELECT CURTIME();
+-----+
| CURTIME() |
+-----+
| 18:40:52 |
+-----+
1 row in set (0.001 sec)
```

🚦 MONTH() dan YEAR()

```
MariaDB [mysql]> SELECT YEAR(NOW());
+-----+
| YEAR(NOW()) |
+-----+
| 2021 |
+-----+
1 row in set (0.001 sec)
```

```
MariaDB [mysql]> SELECT MONTH(NOW());
+-----+
| MONTH(NOW()) |
+-----+
| 11 |
+-----+
1 row in set (0.001 sec)
```

```
MariaDB [classicmodels]> SELECT orderNumber, orderDate, status
-> FROM orders
-> WHERE MONTH(orderDate) = 9 AND YEAR(orderDate) = 2003;
+-----+-----+-----+
| orderNumber | orderDate | status |
+-----+-----+-----+
| 10146 | 2003-09-03 | Shipped |
| 10147 | 2003-09-05 | Shipped |
| 10148 | 2003-09-11 | Shipped |
| 10149 | 2003-09-12 | Shipped |
| 10150 | 2003-09-19 | Shipped |
| 10151 | 2003-09-21 | Shipped |
| 10152 | 2003-09-25 | Shipped |
| 10153 | 2003-09-28 | Shipped |
+-----+-----+-----+
8 rows in set (0.008 sec)
```

🚦 MONTHNAME() dan DAYNAME()

```
MariaDB [mysql]> SELECT MONTHNAME(NOW());
+-----+
| MONTHNAME(NOW()) |
+-----+
| November          |
+-----+
1 row in set (0.004 sec)
```

```
MariaDB [mysql]> use classicmodels;
Database changed
MariaDB [classicmodels]> SELECT orderNumber, orderDate, status
-> FROM orders
-> WHERE MONTHNAME(orderDate) = 'September' AND YEAR(orderDate) = 2003;
+-----+-----+-----+
| orderNumber | orderDate | status |
+-----+-----+-----+
| 10146       | 2003-09-03 | Shipped |
| 10147       | 2003-09-05 | Shipped |
| 10148       | 2003-09-11 | Shipped |
| 10149       | 2003-09-12 | Shipped |
| 10150       | 2003-09-19 | Shipped |
| 10151       | 2003-09-21 | Shipped |
| 10152       | 2003-09-25 | Shipped |
| 10153       | 2003-09-28 | Shipped |
+-----+-----+-----+
8 rows in set (0.001 sec)
```

🚦 ADDDATE() atau DATE_ADD()

```
MariaDB [mysql]> SELECT ADDDATE(NOW(), INTERVAL 15 DAY);
+-----+
| ADDDATE(NOW(), INTERVAL 15 DAY) |
+-----+
| 2021-12-08 18:58:14              |
+-----+
1 row in set (0.000 sec)
```

```
MariaDB [mysql]> SELECT ADDDATE(NOW(), INTERVAL 5 MONTH);
+-----+
| ADDDATE(NOW(), INTERVAL 5 MONTH) |
+-----+
| 2022-04-23 18:58:27              |
+-----+
1 row in set (0.000 sec)
```

```
MariaDB [mysql]> SELECT ADDDATE(NOW(), INTERVAL 2 YEAR);
+-----+
| ADDDATE(NOW(), INTERVAL 2 YEAR) |
+-----+
| 2023-11-23 18:58:42              |
+-----+
1 row in set (0.000 sec)
```

```
MariaDB [mysql]> SELECT ADDDATE(NOW(), INTERVAL 45 DAY);
+-----+
| ADDDATE(NOW(), INTERVAL 45 DAY) |
+-----+
| 2022-01-07 18:59:01              |
+-----+
1 row in set (0.000 sec)
```

```
MariaDB [mysql]> SELECT ADDDATE(NOW(), INTERVAL 2 HOUR);
+-----+
| ADDDATE(NOW(), INTERVAL 2 HOUR) |
+-----+
| 2021-11-23 20:59:16              |
+-----+
1 row in set (0.000 sec)
```

```

MariaDB [classicmodels]> SELECT orderDate, ADDDATE(orderDate, interval 1 week) as 'payDate'
-> FROM orders
-> WHERE MONTH(orderDate) = 11 AND YEAR(orderDate) = 2004;
+-----+-----+
| orderDate | payDate |
+-----+-----+
| 2004-11-01 | 2004-11-08 |
| 2004-11-02 | 2004-11-09 |
| 2004-11-02 | 2004-11-09 |
| 2004-11-03 | 2004-11-10 |
| 2004-11-03 | 2004-11-10 |
| 2004-11-04 | 2004-11-11 |
| 2004-11-04 | 2004-11-11 |
| 2004-11-05 | 2004-11-12 |
| 2004-11-05 | 2004-11-12 |
| 2004-11-05 | 2004-11-12 |
| 2004-11-09 | 2004-11-16 |
| 2004-11-10 | 2004-11-17 |
| 2004-11-12 | 2004-11-19 |
| 2004-11-15 | 2004-11-22 |
| 2004-11-16 | 2004-11-23 |
| 2004-11-17 | 2004-11-24 |
| 2004-11-17 | 2004-11-24 |
| 2004-11-18 | 2004-11-25 |
| 2004-11-19 | 2004-11-26 |
| 2004-11-19 | 2004-11-26 |
| 2004-11-20 | 2004-11-27 |
| 2004-11-21 | 2004-11-28 |
| 2004-11-22 | 2004-11-29 |
| 2004-11-23 | 2004-11-30 |
| 2004-11-24 | 2004-12-01 |
| 2004-11-24 | 2004-12-01 |
| 2004-11-24 | 2004-12-01 |
| 2004-11-24 | 2004-12-01 |
| 2004-11-25 | 2004-12-02 |
| 2004-11-25 | 2004-12-02 |
| 2004-11-29 | 2004-12-06 |
| 2004-11-29 | 2004-12-06 |
| 2004-11-01 | 2004-11-08 |
+-----+-----+
33 rows in set (0.002 sec)

```

SUBDATE() atau DATE_SUB()

```

MariaDB [mysql]> SELECT SUBDATE(NOW(), INTERVAL 15 DAY);
+-----+
| SUBDATE(NOW(), INTERVAL 15 DAY) |
+-----+
| 2021-11-08 19:37:22 |
+-----+
1 row in set (0.000 sec)

MariaDB [mysql]> SELECT SUBDATE(NOW(), INTERVAL 5 MONTH);
+-----+
| SUBDATE(NOW(), INTERVAL 5 MONTH) |
+-----+
| 2021-06-23 19:37:28 |
+-----+
1 row in set (0.000 sec)

MariaDB [mysql]> SELECT SUBDATE(NOW(), INTERVAL 2 YEAR);
+-----+
| SUBDATE(NOW(), INTERVAL 2 YEAR) |
+-----+
| 2019-11-23 19:37:33 |
+-----+
1 row in set (0.000 sec)

MariaDB [mysql]> SELECT SUBDATE(NOW(), INTERVAL 54 MINUTE);
+-----+
| SUBDATE(NOW(), INTERVAL 54 MINUTE) |
+-----+
| 2021-11-23 18:43:39 |
+-----+
1 row in set (0.000 sec)

```


D. TUGAS DAN PEMBAHASAN

1. Kerjakan langkah pada jobsheet bab 16.
 - a. Ketikkan perintah berikut untuk mengetahui tanggal dan waktu saat ini

```
MariaDB [mysql]> SELECT NOW();
+-----+
| NOW() |
+-----+
| 2021-11-23 19:42:53 |
+-----+
1 row in set (0.001 sec)
```

- b. Ketikkan perintah berikut untuk mengeluarkan tahun, bulan dan hari dari sebuah tanggal.

```
MariaDB [mysql]> SELECT YEAR('2021-11-11');
+-----+
| YEAR('2021-11-11') |
+-----+
| 2021 |
+-----+
1 row in set (0.001 sec)

MariaDB [mysql]> SELECT MONTH('2021-11-11');
+-----+
| MONTH('2021-11-11') |
+-----+
| 11 |
+-----+
1 row in set (0.000 sec)

MariaDB [mysql]> SELECT DAYOFMONTH('2021-11-11');
+-----+
| DAYOFMONTH('2021-11-11') |
+-----+
| 11 |
+-----+
1 row in set (0.000 sec)
```

- c. Ketikkan perintah berikut untuk mengetahui nama hari dan bulan dari sebuah tanggal.

```
MariaDB [mysql]> SELECT DAYNAME('2012-11-11') AS Hari, MONTHNAME('2012-11-11') AS Bulan;
+-----+-----+
| Hari | Bulan |
+-----+-----+
| Sunday | November |
+-----+-----+
1 row in set (0.001 sec)
```

- d. Ketikkan perintah berikut untuk mengetahui tanggal 15 hari kedepan setelah 11 November 2012 dan 10 hari sebelumnya.

```
MariaDB [mysql]> SELECT ADDDATE('2012-11-11',15), ADDDATE('2012-11-11',-10);
+-----+-----+
| ADDDATE('2012-11-11',15) | ADDDATE('2012-11-11',-10) |
+-----+-----+
| 2012-11-26                | 2012-11-01                |
+-----+-----+
1 row in set (0.000 sec)
```

- e. Ketikkan perintah berikut ini untuk mengubah format penanggalan

```
MariaDB [mysql]> SELECT DATE_FORMAT('2012-11-11 18:04:05', '%W %M %Y');
+-----+
| DATE_FORMAT('2012-11-11 18:04:05', '%W %M %Y') |
+-----+
| Sunday November 2012                            |
+-----+
1 row in set (0.001 sec)

MariaDB [mysql]> SELECT STR_TO_DATE('11-11-2011', '%d-%m-%Y');
+-----+
| STR_TO_DATE('11-11-2011', '%d-%m-%Y') |
+-----+
| 2011-11-11                            |
+-----+
1 row in set (0.003 sec)
```

2. SOAL 2:

DBPERKULIAHAN

1. Hitunglah usia dosen pada tabel tdosen, tampilkan juga nama dan tanggal lahir.

```
MariaDB [dbperkuliahahan]> SELECT nama, tgl_lahir, TRUNCATE(DATEDIFF(NOW(), tgl_lahir)/365,0) AS 'Usia Dosen'
-> FROM t_dosen;
+-----+-----+-----+
| nama          | tgl_lahir | Usia Dosen |
+-----+-----+-----+
| rinaldi munir | 1966-04-12 | 55         |
| sri purwanti  | 1960-07-16 | 61         |
| inggriani liem | 1952-10-28 | 69         |
| reza satria   | 1970-11-21 | 51         |
| rila mandala  | 1964-03-21 | 57         |
+-----+-----+-----+
5 rows in set (0.016 sec)
```


2. Tampilkan nama, tgl_lahir dan hari kelahiran mahasiswa.

```
MariaDB [dbperkuliah]> SELECT nama, tgl_lahir, DAYNAME(tgl_lahir) AS 'Hari Kelahiran'
-> FROM t_mahasiswa;
```

nama	tgl_lahir	Hari Kelahiran
darmawan p	1983-06-14	Tuesday
albinanto	1984-02-20	Monday
rizky yulianto	1983-06-14	Tuesday
nova kurniawan	1983-09-23	Friday
gama melisa	1984-07-03	Tuesday
andik taufik	1983-02-12	Saturday
anggun hapsari	1984-10-10	Wednesday

7 rows in set (0.009 sec)

DBPENJUALAN

3. Tanggal jatuh tempo pembayarang adalah 30 hari setelah transaksi. Tampilkan semua data transaksi dan data jatuh tempo pembayaran untuk yang jatuh temponya pada bulan februari saja.

```
MariaDB [dbpenjualan]> SELECT *, ADDDATE(tgl, interval 30 day) AS 'Tempo Pembayaran'
-> FROM ttrans
-> WHERE MONTH(tgl) = 2;
```

nonota	tgl	kdlgn	kdbrg	jml	hrg	Tempo Pembayaran
K003	2006-02-06	P004	B001	12	750000	2006-03-08

1 row in set (0.002 sec)

CLASSICMODELS

4. Tampilkan waktu tunggu yang dibutuhkan dari waktu pengiriman hingga pemesanan dari tabel order. untuk barang yang diorder pada bulan februari 2003. Tampilkan juga orderdate dan shippedadate nya dalam bentuk format nama hari, nama bulan dan tahun. Contoh: Monday, February 2003.

```
MariaDB [classicmodels]> SELECT(shippedDate - orderDate) AS 'Waktu Tunggu',
-> CONCAT(DAYNAME(orderDate),',',MONTHNAME(orderDate),' ',YEAR(orderDate)) AS 'orderDate',
-> CONCAT(DAYNAME(shippedDate),',',MONTHNAME(shippedDate),' ',YEAR(shippedDate)) AS 'shippedDate'
-> FROM orders
-> WHERE MONTH(orderdate) = 2 AND YEAR(orderdate) = 2003;
```

Waktu Tunggu	orderDate	shippedDate
1	Tuesday, February 2003	Wednesday, February 2003
4	Monday, February 2003	Friday, February 2003
2	Monday, February 2003	Wednesday, February 2003

3 rows in set (0.001 sec)

FUNGSI STRING

- ✚ Tampilkan nama mahasiswa yang dua huruf didepan namanya adalan An

```
MariaDB [dbperkuliahahan]> SELECT nama
-> FROM t_mahasiswa
-> WHERE LEFT(nama,2) = 'an';
+-----+
| nama      |
+-----+
| andik taufik |
| anggun hapsari |
+-----+
2 rows in set (0.001 sec)
```

- ✚ Tampilkan nama mahasiswa yang dua huruf dibelakang namanya adalan To

```
MariaDB [dbperkuliahahan]> SELECT nama
-> FROM t_mahasiswa
-> WHERE RIGHT(nama,2) = 'to';
+-----+
| nama      |
+-----+
| albinanto  |
| rizky yulianto |
+-----+
2 rows in set (0.000 sec)
```

- ✚ Tampilkan 3 karakter awal dari nama dosen

```
MariaDB [dbperkuliahahan]> SELECT SUBSTRING(nama,1,3)
-> FROM t_dosen;
+-----+
| SUBSTRING(nama,1,3) |
+-----+
| rin
| sri
| ing
| rez
| ril
+-----+
5 rows in set (0.000 sec)
```

- ✚ Tampilkan 5 karakter terakhir dari nama mahasiswa dalam bentuk huruf kapital dan tampilkan juga alamat mahasiswa sebanyak 4 karakter saja dimulai dari karakter ke 5 dalam bentuk huruf kecil

```
MariaDB [dbperkuliahahan]> SELECT UCASE(RIGHT(nama,5)) AS 'Nama', LCASE(SUBSTRING(alamat,5,4)) AS 'Alamat'
-> FROM t_mahasiswa;
```

Nama	Alamat
WAN P	isit
NANTO	ubag
IANTO	ma b
IAWAN	ma k
ELISA	sira
AUFIK	ubag
PSARI	. ta

```
7 rows in set (0.001 sec)
```

E. REFERENSI

-  <https://www.eplusgo.com/fungsi-string-mysql/>
-  <https://modulmakalah.blogspot.com/2015/08/fungsi-string-pada-mysql-database.html>
-  <https://www.eplusgo.com/fungsi-date-time-mysql/>
-  [Job Sheet Basis Data.pdf](#)