DATABASE SQL

Pesantren PeTIK II YBM PLN
Jl. KH. Bisri Syansuri RT/01 RW/05, Plosogeneng,
Kec. Jombang, Kabupaten Jombang, Jawa Timur

Pertemuan Ke-7

Materi

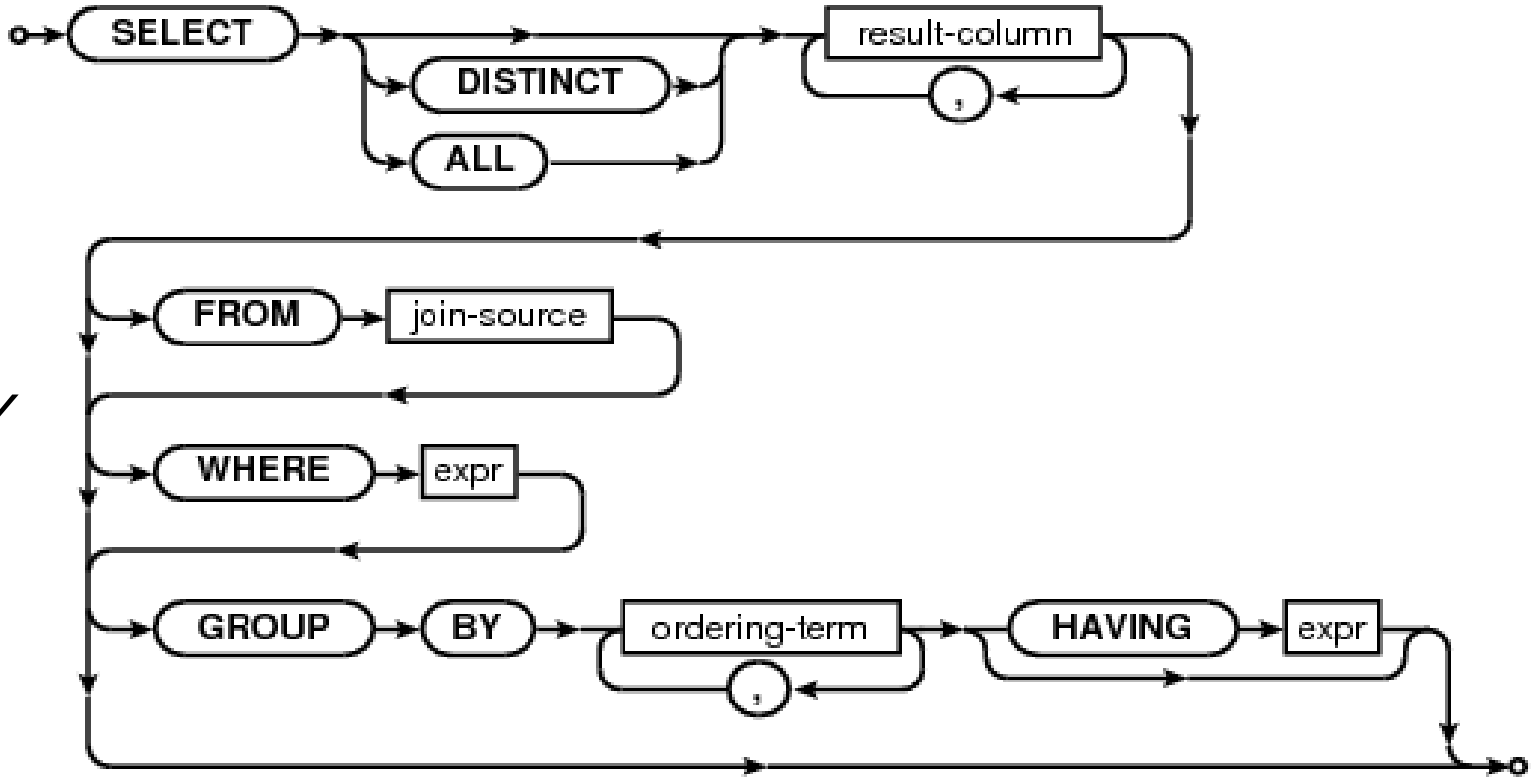
1. Pengantar Database
2. Pemodelan Data
3. Model Relasional Database
4. Normalisasi Database
5. Pengantar SQL
6. Perintah SQL SELECT 1
7. **Perintah SQL SELECT 2**
8. Fungsi Aggregate dan Grouping Data
9. Sub Query & SQL Join Table
10. View dan Analisa Query
11. Store Procedure dan Function
12. Trigger dan Transaction
13. Manajemen User
14. Backup dan Restore

7. Perintah SQL SELECT 2
Built-in Function

Perintah SELECT

URUTAN Perintah

- *SELECT*
- *FROM*
- *WHERE*
- *GROUP BY*
- *HAVING*



SQL Data Type : Date & Time

- DATE - format YYYY-MM-DD
- DATETIME - format: YYYY-MM-DD HH:MI:SS
- TIMESTAMP - format: YYYY-MM-DD HH:MI:SS
- YEAR - format YYYY or YY

Fungsi Waktu -1

Functions	Description
ADDDATE()	MySQL ADDDATE() adds a time value with a date.
ADDTIME()	In MySQL the ADDTIME() returns a time or datetime after adding a time value with a time or datetime.
CONVERT_TZ()	In MySQL the CONVERT_TZ() returns a resulting value after converting a datetime value from a time zone specified as the second argument to the time zone specified as the third argument.
CURDATE()	In MySQL the CURDATE() returns the current date in 'YYYY-MM-DD' format or 'YYYYMMDD' format depending on whether numeric or string is used in the function.
CURRENT_DATE()	In MySQL the CURRENT_DATE returns the current date in 'YYYY-MM-DD' format or YYYYMMDD format depending on whether numeric or string is used in the function.
CURRENT_TIME()	In MySQL the CURRENT_TIME() returns the current time in 'HH:MM:SS' format or HHMMSS.uuuuuu format depending on whether numeric or string is used in the function.

```
select curdate();
+------------+
| curdate()  |
+------------+
| 2021-06-17 |
+------------+
```

Fungsi Waktu -2

Functions	Description
CURRENT_TIMESTAMP()	In MySQL the CURRENT_TIEMSTAMP returns the current date and time in 'YYYY-MM-DD HH:MM:SS' format or YYYYMMDDHHMMSS.uuuuuu format depending on whether numeric or string is used in the function.
CURTIME()	In MySQL the CURTIME() returns the value of current time in 'HH:MM:SS' format or HHMMSS.uuuuuu format depending on whether numeric or string is used in the function.
DATE_ADD()	MySQL DATE_ADD() adds time values (as intervals) to a date value. The ADDDATE() is the synonym of DATE_ADD().
DATE_FORMAT()	MySQL DATE_FORMAT() formats a date as specified in the argument. A list of format specifiers given bellow may be used to format a date.
DATE_SUB()	MySql date_sub() function subtract a time value (as interval) from a date.
DATE()	MySQL DATE() takes the date part out from a datetime expression

```
select curtime();
+-----------+
| curtime() |
+-----------+
| 00:31:48  |
+-----------+
```

Fungsi Waktu -3

Functions	Description
DATEDIFF()	MySQL DATEDIFF() returns the number of days between two dates or datetimes.
DAY()	MySQL DAY() returns the day of the month for a specified date.
DAYNAME()	MySQL DAYNAME() returns the name of the week day of a date specified in the argument.
DAY OF MONTH()	MySQL DAYOFMONTH() returns the day of the month for a given date.
DAY OF WEEK()	MySQL DAYOFWEEK() returns the week day number (1 for Sunday,2 for Monday …… 7 for Saturday) for a date specified as an argument.
DAY OF YEAR()	MySQL DAYOFYEAR() returns day of the year for a date. The return value is within the range of 1 to 366.
EXTRACT()	MySQL EXTRACT() extracts a part of a given date.
FROM_DAYS()	MySQL FROM_DAYS() returns a date against a datevalue.
FROM_UNIXTIME()	MySQL FROM_UNIXTIME() returns a date /datetime from a version of unix_timestamp.

```
select curdate(),
-> datediff(curdate(),'2021-12-31')
-> AS day_to_newyear;
+------------+----------------+
| curdate()  | day_to_newyear |
+------------+----------------+
| 2021-06-17 |           -197 |
+------------+----------------+
```

Fungsi Waktu -4

Functions	Description
GET_FORMAT()	MySQL GET_FORMAT() converts a date or time or datetime in a formatted manner as specified in the argument.
HOUR()	MySQL HOUR() returns the hour of a time.
LAST_DAY()	MySQL LAST_DAY() returns the last day of the corresponding month for a date or datetime value.
LOCALTIME()	MySQL LOCALTIME returns the value of current date and time in 'YYYY-MM-DD HH:MM:SS' format or YYYYMMDDHHMMSS.uuuuuu format depending on the context (numeric or string) of the function.
LOCALTIMESTAMP()	MySQL LOCALTIMESTAMP returns the value of current date and time in 'YYYY-MM-DD HH:MM:SS' format or YYYMMDDHHMMSS.uuuuuu format depending on the context (numeric or string) of the function.
MAKEDATE()	MySQL MAKEDATE() returns a date by taking a value of a year and a number of days. The number of days must be greater than 0 otherwise a NULL will be returned.
MAKETIME()	MySQL MAKETIME() makes and returns a time value from a given hour, minute and seconds.

```
select last_day(curdate())
+-------------------------+
| tgl_terakhir_bulan_ini |
+-------------------------+
| 2021-06-30              |
+-------------------------+
```

Fungsi Waktu -5

Functions	Description
MICROSECOND()	MySQL MICROSECOND() returns microseconds from the time or datetime expression.
MINUTE()	MySQL MINUTE() returns a minute from a time or datetime value.
MONTH()	MySQL MONTH() returns the month for the date within a range of 1 to 12 (January to December).
MONTHNAME()	MySQL MONTHNAME() returns the full name of the month for a given date.
NOW()	MySQL NOW() returns the value of current date and time in 'YYYY-MM-DD HH:MM:SS' format or YYYYMMDDHHMMSS.uuuuuu format depending on the context (numeric or string) of the function.
PERIOD_ADD()	MySQL PERIOD_ADD() adds a number of months with a period and returns the value in the format YYYYMM OR YYMM. Remember that the format YYYYMM and YYMM are not date values.
PERIOD_DIFF()	MySQL PERIOD_DIFF() returns the difference between two periods.

```
select monthname(curdate())
+----------------------+
| nama_bulan_sekarang |
+----------------------+
| June                 |
+----------------------+
```

Fungsi Waktu -6

Functions	Description
QUARTER()	MySQL QUARTER() returns the quarter of the year for a date.
SEC_TO_TIME()	MySQL SEC_TO_TIME() returns a time value by converting the seconds specified in the argument.
SECOND()	MySQL SECOND() returns the second for a time.
STR_TO_DATE()	MySQL STR_TO_DATE() returns a datetime value by taking a string and a specific format string as arguments.
SUBDATE()	MySQL SUBDATAE() subtracts a time value (as interval) from a given date.
SUBTIME()	MySQL SUBTIME() subtracts one datetime value from another.
SYSDATE()	MySQL SYSDATE() returns the current date and time in YYYY-MM-DD HH:MM:SS or YYYYMMDDHHMMSS.uuuuuu format depending on the context of the function.
TIME_FORMAT()	MySQL TIME_FORMAT() converts a time in a formatted string using the format specifiers.
TIME_TO_SEC()	MySQL TIME_TO_SEC() converts a time value in to seconds.

```
select sysdate() AS tgl_system,
quarter(sysdate()) AS sekarang_kuartil ;
+---------------------+------------------+
| tgl_system          | sekarang_kuartil |
+---------------------+------------------+
| 2021-06-17 01:05:17 |                2 |
+---------------------+------------------+
```

Fungsi Waktu -7

Functions	Description
TIME()	MySQL TIME() extracts the time part of a time or datetime expression as string format.
TIMEDIFF()	MySQL TIMEDIFF() returns the differences between two time or datetime expressions.
TIMESTAMP()	MySQL TIMESTAMP() returns a datetime value against a date or datetime expression.
TIMESTAMPADD()	MySQL TIMESTAMPADD() adds time value with a date or datetime value.
TIMESTAMPDIFF()	MySQL the TIMESTAMPDIFF() returns a value after subtracting a datetime expression from another.
TO_DAYS()	MySQL TO_DAYS() returns number of days between a given date and year 0.
UNIX_TIMESTAMP()	MySQL UNIX_TIMESTAMP() returns a Unix timestamp in seconds since '1970-01-01 00:00:00' UTC as an unsigned integer if no arguments are passed with UNIT_TIMESTAMP().
UTC_DATE()	MySQL UTC_DATE returns the current UTC (Coordinated Universal Time) date as a value in 'YYYY-MM-DD' or YYYYMMDD format depending on the context of the function i.e. in a string or numeric context.

```
select curdate(),
-> timestamp(curdate());
+------------+----------------------+
| curdate()  | timestamp(curdate()) |
+------------+----------------------+
| 2021-06-17 | 2021-06-17 00:00:00  |
+------------+----------------------+
```

Fungsi Waktu -8

Functions	Description
UTC_TIME()	MySQL UTC_TIME returns the current UTC time as a value in 'HH:MM:SS' or HHMMSS format depending on the context of the function i.e. in a string or numeric context.
UTC_TIMESTAMP()	In MySQL the UTC_TIMESTAMP returns the current UTC date and time as a value in 'YYYY-MM-DD HH:MM:SS' or YYYYMMDDHHMMSS.uuuuuu format depending on the usage of the function i.e. in a string or numeric context.
WEEK()	MySQL WEEK() returns the week number for a given date.
WEEKDAY()	MySQL WEEKDAY() returns the index of the day in a week for a given date (0 for Monday, 1 for Tuesday and .....6 for Sunday).
WEEK OF YEAR()	MySQL WEEKOFYEAR() returns the calender week (as a number) of a given date.
YEAR()	MySQL YEAR() returns the year for a given date.
YEARWEEK()	MySQL YEARWEEK() returns year and week number for a given date.

```
select curdate() AS tgl_sekarang,
-> weekday(curdate()) AS pekan_hari_ke;
+--------------+---------------+
| tgl_sekarang | pekan_hari_ke |
+--------------+---------------+
| 2021-06-17   |             3 |
+--------------+---------------+
```

Fungsi Waktu

```
SELECT NOW(),CURDATE(),CURTIME()
```

Tampilkan nama dan tahun lahir customer

- SELECT nama,year(tgl_lahir) FROM pelanggan

Tampilkan bulan saat ini

- SELECT month(current_date)

```
DATEDIFF(date1,date2)
```

```
SELECT DATEDIFF('2014-11-30','2014-11-29') AS DiffDate
```

Fungsi Waktu

```
DATE_ADD(date,INTERVAL expr type)
```

```
DATE_SUB(date,INTERVAL expr type)
```

Type Value	
MICROSECOND	YEAR
SECOND	SECOND_MICROSECOND
MINUTE	MINUTE_MICROSECOND
HOUR	MINUTE_SECOND
DAY	HOUR_MICROSECOND
WEEK	HOUR_SECOND
MONTH	HOUR_MINUTE
QUARTER	DAY_MICROSECOND
YEAR	DAY_SECOND
	DAY_MINUTE
	DAY_HOUR
	YEAR_MONTH

```
SELECT OrderId,DATE_SUB(OrderDate,INTERVAL 5 DAY) AS SubtractDate
FROM Orders
```

```
SELECT OrderId,DATE_ADD(OrderDate,INTERVAL 30 DAY) AS OrderPayDate
FROM Orders
```

Fungsi String -1

Functions	Description
ASCII	Returns the ASCII value for the specific character
CHAR_LENGTH	Returns the length of a string (in characters)
CHARACTER_LENGTH	Returns the length of a string (in characters)
CONCAT	Adds two or more expressions together
CONCAT_WS	Adds two or more expressions together with a separator
FIELD	Returns the index position of a value in a list of values
FIND_IN_SET	Returns the position of a string within a list of strings
FORMAT	Formats a number to a format like "#,###,###.##", rounded to a specified number of decimal places
INSERT	Inserts a string within a string at the specified position and for a certain number of characters
INSTR	Returns the position of the first occurrence of a string in another string

```
SELECT CONCAT("SQL ", "Tutorial ", "is ", "fun!") AS ConcatenatedString;
```

```
SELECT CONCAT_WS("-", "SQL", "Tutorial", "is", "fun!") AS ConcatenatedString;
```

ConcatenatedString

SQL Tutorial is fun!

Fungsi String -2

Functions	Description
LCASE	Converts a string to lower-case
LEFT	Extracts a number of characters from a string (starting from left)
LENGTH	Returns the length of a string (in bytes)
LOCATE	Returns the position of the first occurrence of a substring in a string
LOWER	Converts a string to lower-case
LPAD	Left-pads a string with another string, to a certain length
LTRIM	Removes leading spaces from a string
MID	Extracts a substring from a string (starting at any position)
POSITION	Returns the position of the first occurrence of a substring in a string
REPEAT	Repeats a string as many times as specified
REPLACE	Replaces all occurrences of a substring within a string, with a

```
+-----------------------+----------------------+
| UCASE(nama)           | LCASE(LEFT(nama,2)) |
+-----------------------+----------------------+
| TEH KOTAK ENAK SEKALI | te                   |
| GULA PASIR            | gu                   |
| KERTAS A4             | ke                   |
| KOMPOR GAS C800       | ko                   |
```

```
select UCASE(nama),LCASE(LEFT(nama,2)) from produk;
```

Fungsi String -3

Functions	Description
REVERSE	Reverses a string and returns the result
RIGHT	Extracts a number of characters from a string (starting from right)
RPAD	Right-pads a string with another string, to a certain length
RTRIM	Removes trailing spaces from a string
SPACE	Returns a string of the specified number of space characters
STRCMP	Compares two strings
SUBSTR	Extracts a substring from a string (starting at any position)
SUBSTRING	Extracts a substring from a string (starting at any position)
SUBSTRING_INDEX	Returns a substring of a string before a specified number of delimiter occurs
TRIM	Removes leading and trailing spaces from a string
UCASE	Converts a string to upper-case
UPPER	Converts a string to upper-case

```
+----+--------------+------+
| id | left(nama,2) | kode |
+----+--------------+------+
|  1 | Te           | TE1  |
|  2 | Gu           | GU2  |
|  3 | Ke           | KE3  |
|  4 | Ko           | KO4  |
|  5 | Bi           | BI5  |
|  6 | Mi           | MI6  |
|  7 | Ko           | KO7  |
|  9 | Bi           | BI9  |
+----+--------------+------+
```

```
select id,left(nama,2),
-> UCASE(RPAD(LEFT(nama,2),3,id)) as kode FROM produk;
```

Fungsi Control Flow

Name	Description
CASE	Case operator
IF()	If/else construct
IFNULL()	Null if/else construct
NULLIF()	Return NULL if expr1 = expr2

Fungsi Control Flow: CASE

```
CASE
    WHEN condition1 THEN result1
    WHEN condition2 THEN result2
    WHEN conditionN THEN resultN
    ELSE result
END;
```

Contoh :

```
SELECT CASE 1
  WHEN 1 THEN 'satu'
  WHEN 2 tHEN 'dua'
  ELSE 'lainya'
END ;
```

Tampilkan jenis kelamin pelanggan : L → Laki-Laki , P → Perempuan

```
SELECT nama, CASE(gender) WHEN 'L' THEN 'Laki-Laki' ELSE
'Perempuan' END as jenis_kelamin FROM pelanggan
```

Fungsi Control Flow: CASE

SQL Statement:

```
SELECT OrderID, Quantity,
CASE WHEN Quantity > 30 THEN 'The quantity is greater than 30'
WHEN Quantity = 30 THEN 'The quantity is 30'
ELSE 'The quantity is under 30'
END AS QuantityText
FROM OrderDetails;
```

OrderID	Quantity	QuantityText
10248	12	The quantity is under 30
10248	10	The quantity is under 30
10248	5	The quantity is under 30
10249	9	The quantity is under 30

Fungsi Control Flow: CASE

SQL Statement:

```
SELECT CustomerName, City, Country FROM Customers
ORDER BY (CASE
WHEN City IS NULL THEN Country
ELSE City
END);
```

Number of Records: 91

CustomerName	City	Country
Drachenblut Delikatessend	Aachen	Germany
Rattlesnake Canyon Grocery	Albuquerque	USA
Old World Delicatessen	Anchorage	USA

Fungsi Control Flow: IF

```
IF(condition, value_if_true, value_if_false)
```

Parameter	Description
condition	Required. The value to test
value_if_true	Required. The value to return if *condition* is TRUE
value_if_false	Required. The value to return if *condition* is FALSE

Contoh :

- `SELECT IF( 80 > 50 , 'Lulus','Tidak Lulus')`

Tampilkan status harga produk jika diatas 2juta mahal dan sebaliknya murah

- `SELECT id,nama,harga,IF(harga >2000000, 'Mahal','Murah') AS status FROM produk ;`

Fungsi Control Flow: IF

```
IF(condition, value_if_true, value_if_false)
```

```
SELECT OrderID, Quantity, IF(Quantity>10, "CUKUP", "KURANG")
FROM OrderDetails;
```

OrderID	Quantity	IF(Quantity>10, "CUKUP", "KURANG")
10248	12	CUKUP
10248	10	KURANG
10248	5	KURANG
10249	9	KURANG

Fungsi Control Flow: IFNULL

- Digunakan untuk menguji apakah sebuah data bernilai null

```
mysql> SELECT IFNULL(1,0);
        -> 1
mysql> SELECT IFNULL(NULL,10);
        -> 10
mysql> SELECT IFNULL(1/0,10);
        -> 10
mysql> SELECT IFNULL(1/0,'yes');
        -> 'yes'
```

- Dapat juga menggunakan keyword COALESCE untuk mengganti data yang bernilai null

```
SELECT COALESCE(NULL,1)
```

TERIMA KASIH
ATAS SEGALA PERHATIAN
SEMOGA BERMANFAAT...