Kedua query SQL yang Anda berikan melakukan tugas yang mirip, tetapi mereka menggunakan sintaks dan fungsi yang berbeda karena tampaknya ditujukan untuk dua jenis sistem manajemen basis data (DBMS) yang berbeda. Berikut adalah perbedaan utama antara query-query tersebut:

**1. Sintaks dan Fungsi SQL**

* **Query Pertama: MySQL**

sql

Copy code

SELECT

c.\*,

b.buyer,

b.no\_order,

b.no\_mesin,

b.no\_resep,

b.warna,

b.proses,

b.target,

IF(c.status='selesai', a.lama\_proses, TIME\_FORMAT(TIMEDIFF(NOW(), c.tgl\_buat), '%H:%i')) AS lama,

a.operator\_keluar,

c.id AS idm,

b.id AS ids

FROM

tbl\_montemp c

LEFT JOIN tbl\_schedule b ON c.id\_schedule = b.id

LEFT JOIN tbl\_hasilcelup a ON a.id\_montemp = c.id

WHERE

$shft

DATE\_FORMAT(c.tgl\_buat, '%Y-%m-%d') BETWEEN '$Awal' AND '$Akhir'

ORDER BY

b.no\_mesin ASC;

* + **Fungsi IF**: Digunakan untuk melakukan percabangan logika di MySQL.
  + **Fungsi TIME\_FORMAT dan TIMEDIFF**: Digunakan untuk mengformat dan menghitung selisih waktu di MySQL.
  + **Fungsi DATE\_FORMAT**: Digunakan untuk memformat tanggal di MySQL.
* **Query Kedua: SQL Server (T-SQL)**

sql

Copy code

SELECT

c.\*,

b.buyer,

b.no\_order,

b.no\_mesin,

b.no\_resep,

b.warna,

b.proses,

b.target,

CASE

WHEN c.status = 'selesai' THEN a.lama\_proses

ELSE FORMAT(DATEDIFF(MINUTE, c.tgl\_buat, GETDATE()), '00') + ':' + RIGHT('00' + CAST(DATEDIFF(SECOND, DATEADD(MINUTE, DATEDIFF(MINUTE, 0, c.tgl\_buat), 0), GETDATE()) % 60 AS VARCHAR(2)), 2)

END AS lama,

a.operator\_keluar,

c.id AS idm,

b.id AS ids

FROM

db\_dying.tbl\_montemp c

LEFT JOIN db\_dying.tbl\_schedule b ON c.id\_schedule = b.id

LEFT JOIN db\_dying.tbl\_hasilcelup a ON a.id\_montemp = c.id

WHERE

$shft

FORMAT(c.tgl\_buat, 'yyyy-MM-dd') BETWEEN '$Awal' AND '$Akhir'

ORDER BY

b.no\_mesin ASC;

* + **Fungsi CASE**: Digunakan untuk melakukan percabangan logika di SQL Server.
  + **Fungsi DATEDIFF dan DATEADD**: Digunakan untuk menghitung selisih waktu dan menambahkan waktu di SQL Server.
  + **Fungsi FORMAT**: Digunakan untuk memformat tanggal dan waktu di SQL Server.

**2. Perhitungan Lama Proses**

* **MySQL**:
  + Menggunakan IF untuk menentukan nilai lama berdasarkan status.
  + Jika status adalah 'selesai', lama diambil dari kolom a.lama\_proses.
  + Jika tidak, lama dihitung menggunakan TIMEDIFF antara waktu saat ini (NOW()) dan c.tgl\_buat, dan di-format menggunakan TIME\_FORMAT.
* **SQL Server**:
  + Menggunakan CASE untuk menentukan nilai lama.
  + Jika status adalah 'selesai', lama diambil dari kolom a.lama\_proses.
  + Jika tidak, lama dihitung dengan DATEDIFF untuk menit dan detik, dan di-format secara manual dengan FORMAT dan RIGHT.

**3. Format Tanggal**

* **MySQL**: Menggunakan DATE\_FORMAT(c.tgl\_buat, '%Y-%m-%d') untuk format tanggal.
* **SQL Server**: Menggunakan FORMAT(c.tgl\_buat, 'yyyy-MM-dd') untuk format tanggal.

**4. Fungsi dan Syntax Khusus**

* **MySQL**: Menggunakan IF, TIME\_FORMAT, dan TIMEDIFF.
* **SQL Server**: Menggunakan CASE, DATEDIFF, DATEADD, FORMAT, dan RIGHT.

**Ringkasan**

* Query pertama adalah untuk MySQL, menggunakan sintaks dan fungsi yang spesifik untuk MySQL.
* Query kedua adalah untuk SQL Server, menggunakan sintaks dan fungsi yang spesifik untuk SQL Server.

Untuk menggunakan query ini, Anda harus memastikan bahwa sistem manajemen basis data Anda sesuai dengan sintaks dan fungsi yang digunakan.

### Perbedaan antara Query MySQL dan SQL Server

#### 1. **Fungsi dan Sintaks**

* **MySQL Query:**

sql

Copy code

SELECT

a.kd\_stop,

a.mulai\_stop,

a.selesai\_stop,

a.ket,

IF(ISNULL(TIMEDIFF(c.tgl\_mulai, c.tgl\_stop)), a.lama\_proses,

CONCAT(LPAD(FLOOR((((HOUR(a.lama\_proses) \* 60) + MINUTE(a.lama\_proses)) - ((HOUR(TIMEDIFF(c.tgl\_mulai, c.tgl\_stop)) \* 60) + MINUTE(TIMEDIFF(c.tgl\_mulai, c.tgl\_stop)))) / 60), 2, 0), ':',

LPAD(((((HOUR(a.lama\_proses) \* 60) + MINUTE(a.lama\_proses)) - ((HOUR(TIMEDIFF(c.tgl\_mulai, c.tgl\_stop)) \* 60) + MINUTE(TIMEDIFF(c.tgl\_mulai, c.tgl\_stop)))) % 60), 2, 0))) AS lama\_proses,

a.status AS sts,

TIME\_FORMAT(IF(ISNULL(TIMEDIFF(c.tgl\_mulai, c.tgl\_stop)), a.lama\_proses,

CONCAT(LPAD(FLOOR((((HOUR(a.lama\_proses) \* 60) + MINUTE(a.lama\_proses)) - ((HOUR(TIMEDIFF(c.tgl\_mulai, c.tgl\_stop)) \* 60) + MINUTE(TIMEDIFF(c.tgl\_mulai, c.tgl\_stop)))) / 60), 2, 0), ':',

LPAD(((((HOUR(a.lama\_proses) \* 60) + MINUTE(a.lama\_proses)) - ((HOUR(TIMEDIFF(c.tgl\_mulai, c.tgl\_stop)) \* 60) + MINUTE(TIMEDIFF(c.tgl\_mulai, c.tgl\_stop)))) % 60), 2, 0))), '%H') AS jam,

TIME\_FORMAT(IF(ISNULL(TIMEDIFF(c.tgl\_mulai, c.tgl\_stop)), a.lama\_proses,

CONCAT(LPAD(FLOOR((((HOUR(a.lama\_proses) \* 60) + MINUTE(a.lama\_proses)) - ((HOUR(TIMEDIFF(c.tgl\_mulai, c.tgl\_stop)) \* 60) + MINUTE(TIMEDIFF(c.tgl\_mulai, c.tgl\_stop)))) / 60), 2, 0), ':',

LPAD(((((HOUR(a.lama\_proses) \* 60) + MINUTE(a.lama\_proses)) - ((HOUR(TIMEDIFF(c.tgl\_mulai, c.tgl\_stop)) \* 60) + MINUTE(TIMEDIFF(c.tgl\_mulai, c.tgl\_stop)))) % 60), 2, 0))), '%i') AS menit,

a.point,

DATE\_FORMAT(a.mulai\_stop, '%Y-%m-%d') AS t\_mulai,

DATE\_FORMAT(a.selesai\_stop, '%Y-%m-%d') AS t\_selesai,

TIME\_FORMAT(a.mulai\_stop, '%H:%i') AS j\_mulai,

TIME\_FORMAT(a.selesai\_stop, '%H:%i') AS j\_selesai,

TIMESTAMPDIFF(MINUTE, a.mulai\_stop, a.selesai\_stop) AS lama\_stop\_menit,

a.acc\_keluar,

b.proses,

b.buyer,

b.langganan,

b.no\_order,

b.jenis\_kain,

b.no\_mesin,

b.warna,

b.lot,

b.energi,

b.dyestuff,

b.ket\_status,

b.kapasitas,

b.loading,

b.resep,

b.kategori\_warna,

c.l\_r,

c.rol,

c.bruto,

c.pakai\_air,

DATE\_FORMAT(c.tgl\_buat, '%Y-%m-%d') AS tgl\_in,

DATE\_FORMAT(a.tgl\_buat, '%Y-%m-%d') AS tgl\_out,

DATE\_FORMAT(c.tgl\_buat, '%H:%i') AS jam\_in,

DATE\_FORMAT(a.tgl\_buat, '%H:%i') AS jam\_out,

IF(ISNULL(a.g\_shift), b.g\_shift, a.g\_shift) AS shft,

a.operator\_keluar,

b.nokk,

b.no\_warna,

b.lebar,

b.gramasi,

c.carry\_over,

c.operator,

c.no\_program,

c.pjng\_kain,

c.tekanan,

c.rpm,

c.cycle\_time,

c.nozzle,

b.no\_hanger,

b.no\_item,

b.po,

b.tgl\_delivery

FROM

tbl\_schedule b

LEFT JOIN tbl\_montemp c ON c.id\_schedule = b.id

LEFT JOIN tbl\_hasilcelup a ON a.id\_montemp = c.id

WHERE

$shft

DATE\_FORMAT(c.tgl\_buat, '%Y-%m-%d') BETWEEN '$Awal' AND '$Akhir'

ORDER BY

b.no\_mesin ASC;

* **SQL Server Query:**

sql

Copy code

SELECT

a.kd\_stop,

a.mulai\_stop,

a.selesai\_stop,

a.ket,

CASE

WHEN c.tgl\_mulai IS NULL OR c.tgl\_stop IS NULL THEN a.lama\_proses

ELSE

FORMAT(DATEPART(HOUR, DATEADD(MINUTE, DATEDIFF(MINUTE, c.tgl\_stop, c.tgl\_mulai), 0)), '00') + ':' +

FORMAT(DATEPART(MINUTE, DATEADD(MINUTE, DATEDIFF(MINUTE, c.tgl\_stop, c.tgl\_mulai), 0)), '00')

END AS lama\_proses,

a.status AS sts,

DATEPART(HOUR, DATEADD(MINUTE, DATEDIFF(MINUTE, c.tgl\_stop, c.tgl\_mulai), 0)) AS jam,

DATEPART(MINUTE, DATEADD(MINUTE, DATEDIFF(MINUTE, c.tgl\_stop, c.tgl\_mulai), 0)) AS menit,

a.point,

FORMAT(a.mulai\_stop, 'yyyy-MM-dd') AS t\_mulai,

FORMAT(a.selesai\_stop, 'yyyy-MM-dd') AS t\_selesai,

FORMAT(a.mulai\_stop, 'HH:mm') AS j\_mulai,

FORMAT(a.selesai\_stop, 'HH:mm') AS j\_selesai,

DATEDIFF(MINUTE, a.mulai\_stop, a.selesai\_stop) AS lama\_stop\_menit,

a.acc\_keluar,

b.proses,

b.buyer,

b.langganan,

b.no\_order,

b.jenis\_kain,

b.no\_mesin,

b.warna,

b.lot,

b.energi,

b.dyestuff,

b.ket\_status,

b.kapasitas,

b.loading,

b.resep,

b.kategori\_warna,

c.l\_r,

c.rol,

c.bruto,

c.pakai\_air,

FORMAT(c.tgl\_buat, 'yyyy-MM-dd') AS tgl\_in,

FORMAT(a.tgl\_buat, 'yyyy-MM-dd') AS tgl\_out,

FORMAT(c.tgl\_buat, 'HH:mm') AS jam\_in,

FORMAT(a.tgl\_buat, 'HH:mm') AS jam\_out,

COALESCE(a.g\_shift, b.g\_shift) AS shft,

a.operator\_keluar,

b.nokk,

b.no\_warna,

b.lebar,

b.gramasi,

c.carry\_over,

c.operator,

c.no\_program,

c.pjng\_kain,

c.tekanan,

c.rpm,

c.cycle\_time,

c.nozzle,

b.no\_hanger,

b.no\_item,

b.po,

b.tgl\_delivery

FROM

db\_dying.tbl\_schedule b

LEFT JOIN db\_dying.tbl\_montemp c ON c.id\_schedule = b.id

LEFT JOIN db\_dying.tbl\_hasilcelup a ON a.id\_montemp = c.id

WHERE

$shft

FORMAT(c.tgl\_buat, 'yyyy-MM-dd') BETWEEN '$Awal' AND '$Akhir'

ORDER BY

b.no\_mesin ASC;

#### 2. **Perhitungan** lama\_proses

* **MySQL:**
  + Menggunakan IF dan TIMEDIFF untuk menangani perhitungan waktu.
  + Menggunakan CONCAT, LPAD, dan FLOOR untuk membentuk format waktu.
  + TIME\_FORMAT digunakan untuk format waktu dalam jam dan menit.
* **SQL Server:**
  + Menggunakan CASE dan DATEADD untuk menangani perhitungan waktu.
  + Menggunakan FORMAT dan DATEPART untuk menghasilkan jam dan menit.
  + DATEDIFF digunakan untuk menghitung selisih waktu dalam menit.

#### 3. **Format Tanggal dan Waktu**

* **MySQL:**
  + DATE\_FORMAT digunakan untuk memformat tanggal.
  + TIME\_FORMAT digunakan untuk memformat waktu.
* **SQL Server:**
  + FORMAT digunakan untuk memformat tanggal dan waktu.

#### 4. **Penanganan NULL**

* **MySQL:**
  + Menggunakan ISNULL untuk memeriksa nilai NULL.
* **SQL Server:**
  + Menggunakan COALESCE untuk memeriksa dan menangani nilai NULL.

#### 5. **Fungsi yang Digunakan**

* **MySQL:**
  + IF, ISNULL, TIMEDIFF, HOUR, MINUTE, CONCAT, LPAD, FLOOR, TIME\_FORMAT, TIMESTAMPDIFF
* **SQL Server:**
  + CASE, DATEADD, DATEDIFF, DATEPART, FORMAT, COALESCE

### Ringkasan

* **Query MySQL** menggunakan fungsi dan sintaks spesifik untuk MySQL, seperti IF, TIME\_FORMAT, dan TIMESTAMPDIFF.
* **Query SQL Server** menggunakan fungsi dan sintaks spesifik untuk SQL Server, seperti CASE, FORMAT, dan DATEDIFF.

Kedua query ini bertujuan untuk memberikan hasil yang serupa tetapi menyesuaikan dengan DBMS yang digunakan, yang mempengaruhi fungsi dan sintaks yang dipilih.

Kedua potongan kode PHP yang Anda berikan digunakan untuk menampilkan informasi mengenai lama\_proses dalam format jam dan menit di dalam elemen <td>. Namun, ada perbedaan signifikan dalam cara mereka memproses dan menampilkan data. Mari kita uraikan perbedaannya:

**1. Kode Pertama:**

php

Copy code

<td><?php if($rowd['lama\_proses']!=""){echo $rowd['jam']." Jam ".$rowd['menit']." Menit";}?></td>

**Penjelasan:**

* **Cek Kondisi:** Kode ini memeriksa apakah lama\_proses tidak kosong (tidak sama dengan "").
* **Menampilkan Jam dan Menit:** Jika lama\_proses tidak kosong, ia menampilkan nilai dari $rowd['jam'] dan $rowd['menit'] dengan format "Jam" dan "Menit".

**Kekurangan:**

* **Ketergantungan pada Variabel Lain:** Kode ini mengandalkan variabel $rowd['jam'] dan $rowd['menit'] yang harus sudah diisi sebelumnya di tempat lain dalam kode. Kode ini tidak mengolah atau memformat lama\_proses secara langsung.
* **Format Output:** Tidak ada penjaminan format jam dan menit (misalnya, dua digit untuk jam dan menit) karena jam dan menit harus sudah dalam format yang benar.

**2. Kode Kedua:**

php

Copy code

<td><?php echo !empty($rowd['lama\_proses']) ? sprintf('%02d Jam %02d Menit', intval(explode(':', $rowd['lama\_proses'])[0]), intval(explode(':', $rowd['lama\_proses'])[1])) : ""; ?></td>

**Penjelasan:**

* **Cek Kondisi:** Menggunakan !empty() untuk memeriksa apakah lama\_proses tidak kosong atau tidak null.
* **Format Jam dan Menit:**
  + **explode(':', $rowd['lama\_proses']):** Memecah string lama\_proses yang diasumsikan dalam format HH:MM menjadi array dengan dua elemen.
  + **intval():** Mengkonversi elemen array ke integer.
  + **sprintf('%02d Jam %02d Menit', ...):** Memformat jam dan menit dengan dua digit, memastikan bahwa angka akan selalu memiliki dua digit (misalnya, "01 Jam 05 Menit").

**Kelebihan:**

* **Pemrosesan dan Format:** Kode ini secara langsung memproses lama\_proses, memecahnya menjadi jam dan menit, dan memformat output dengan cara yang konsisten.
* **Penjaminan Format:** Menjamin format dua digit untuk jam dan menit, yang membuat tampilan lebih rapi dan konsisten.

**Ringkasan Perbedaan:**

* **Pengolahan Data:** Kode pertama mengandalkan variabel lain yang harus diisi sebelumnya, sementara kode kedua memproses dan memformat data lama\_proses secara langsung.
* **Format Output:** Kode kedua menjamin format output dengan dua digit untuk jam dan menit, sementara kode pertama tidak memberikan jaminan format.
* **Ketergantungan pada Format:** Kode pertama memerlukan format lama\_proses sudah sesuai dengan format yang diharapkan, sedangkan kode kedua melakukan pemrosesan dan format secara internal.

Jika lama\_proses dalam format HH:MM, kode kedua adalah pilihan yang lebih baik karena secara otomatis menangani format dan penampilan output.