

BASIS DATA LANJUT

TUGAS 11



ADHE WIDYA GALIH KARTIKA

SIB 2C_03

244107060067

Jurusan Teknologi Informasi

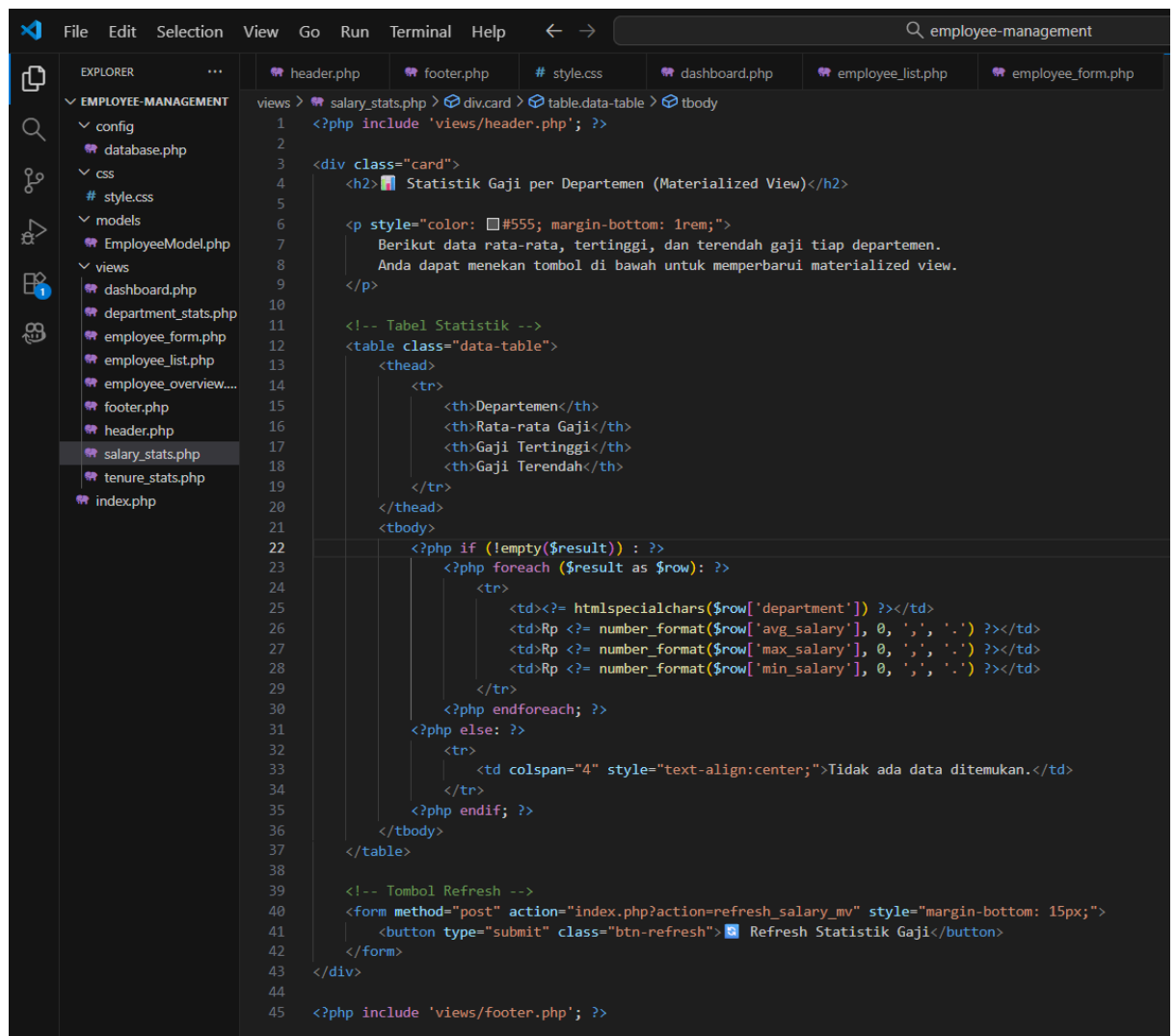
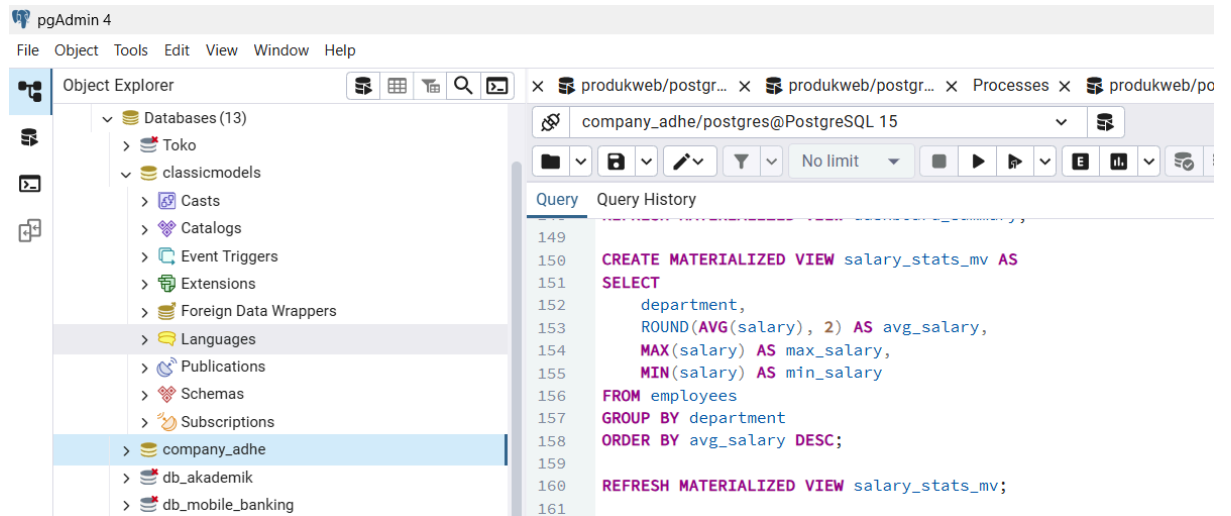
POLITEKNIK NEGERI MALANG

2025

TUGAS 1: IMPLEMENTASI FUNGSI AGREGAT PADA WEB

1. Halaman Statistik Gaji (salary_stats.php)

- Tampilkan: Rata-rata gaji, gaji tertinggi, gaji terendah per departemen
- Gunakan fungsi: AVG(), MAX(), MIN(), GROUP BY



Sistem Manajemen Karyawan

Aplikasi CRUD Sederhana dengan PostgreSQL & PHP

Dashboard

Data Karyawan

Tambah Karyawan

Statistik Departemen

Refresh Dashboard

Statistik Gaji

Statistik Gaji per Departemen (Materialized View)

Berikut data rata-rata, tertinggi, dan terendah gaji tiap departemen. Anda dapat menekan tombol di bawah untuk memperbarui materialized view.

Departemen	Rata-rata Gaji	Gaji Tertinggi	Gaji Terendah
IT	Rp 8.283.333	Rp 9.000.000	Rp 7.800.000
Operations	Rp 8.100.000	Rp 11.000.000	Rp 6.000.000
Finance	Rp 7.650.000	Rp 8.200.000	Rp 7.200.000
HR	Rp 7.400.000	Rp 9.500.000	Rp 6.200.000
Marketing	Rp 6.375.000	Rp 7.000.000	Rp 5.800.000

Refresh Statistik Gaji

2. Halaman Masa Kerja (tenure_stats.php)

- Tampilkan: Karyawan berdasarkan masa kerja (Junior: <1 tahun, Middle: 1-3 tahun, Senior: >3 tahun)
- Gunakan fungsi: COUNT(), CASE WHEN, GROUP BY

The screenshot shows the pgAdmin 4 interface. On the left, the 'Object Explorer' pane shows a tree of databases, with 'company_adhe' selected. The main pane displays a SQL query in the 'Query' editor. The query is as follows:

```

191 CREATE MATERIALIZED VIEW tenure_stats_mv AS
192 SELECT
193     CASE
194         WHEN AGE(CURRENT_DATE, hire_date) < INTERVAL '1 year' THEN 'Junior'
195         WHEN AGE(CURRENT_DATE, hire_date) BETWEEN INTERVAL '1 year' AND INTERVAL '3 years' THEN 'Middle'
196         ELSE 'Senior'
197     END AS tenure_level,
198     COUNT(*) AS total_employees,
199     STRING_AGG(first_name || ' ' || last_name, ', ') AS employee_names
200 FROM employees
201 GROUP BY tenure_level
202 ORDER BY total_employees DESC;
203

```

```

1  <?php
2  /**
3   * FILE: views/tenure_stats.php
4   * FUNGSI: Menampilkan statistik masa kerja karyawan (Junior, Middle, Senior)
5   */
6  include 'views/header.php';
7  ?>
8
9  <h2>Statistik Masa Kerja Karyawan</h2>
10
11  <p style="margin-bottom: 2rem; color: #666;">
12      Data berikut diambil dari <code>tenure_stats_mv</code> (Materialized View) di database PostgreSQL.
13  </p>
14
15  <?php if (!empty($result)): ?>
16      <!-- Hitung total -->
17      <?php
18          $total_employees = array_sum(array_column($result, 'total_employees'));
19      ?>
20
21      <!-- Cards Summary -->
22      <div class="dashboard-cards">
23          <div class="card">
24              <h3>Total Karyawan</h3>
25              <div class="number"><?php echo $total_employees; ?></div>
26          </div>
27          <div class="card">
28              <h3>Level Masa Kerja</h3>
29              <div class="number"><?php echo count($result); ?> Level</div>
30          </div>
31      </div>
32
33      <!-- Tabel Statistik Detail -->
34      <table class="data-table">
35          <thead>
36              <tr>
37                  <th>Tingkat Masa Kerja</th>
38                  <th>Jumlah Karyawan</th>
39                  <th>Nama Karyawan</th>
40              </tr>
41          </thead>
42          <tbody>
43              <?php foreach ($result as $row): ?>
44                  <tr>
45                      <td>
46                          <strong><?php echo htmlspecialchars($row['tenure_level']); ?></strong>
47                      </td>
48                      <td style="text-align: center;">
49                          <span style="padding: 0.25rem 0.75rem; background: #667eea; color: white; border-radius: 20px;">
50                              <?php echo $row['total_employees']; ?>
51                          </span>
52                      </td>
53                      <td>
54                          <?php
55                              $names = explode(',', $row['employee_names']);
56                              echo "<ul style='margin: 0; padding-left: 20px;'>";
57                              foreach ($names as $name) {
58                                  echo "<li>" . htmlspecialchars(trim($name)) . "</li>";
59                              }
60                              echo "</ul>";
61                          ?>
62                      </td>
63                  </tr>
64              <?php endforeach; ?>
65          </tbody>
66      </table>
67  </div>

```

```
68 <!-- Tombol Refresh -->
69 <div style="text-align:center; margin-top: 30px;">
70   <a href="index.php?action=refresh_tenure_mv"
71     style="
72       display: inline-block;
73       background-color: #4a6cf7;
74       color: #fff;
75       padding: 10px 25px;
76       border-radius: 8px;
77       text-decoration: none;
78       font-weight: 500;
79       transition: background-color 0.3s ease;
80     "
81     onmouseover="this.style.backgroundColor='#3651c9'"
82     onmouseout="this.style.backgroundColor='#4a6cf7'">
83     Refresh Data
84   </a>
85 </div>
86
87 <?php else: ?>
88   <div style="text-align: center; padding: 3rem; background: #f8f9fa; border-radius: 8px;">
89     <p style="font-size: 1.2rem; color: #666;">Tidak ada data statistik masa kerja.</p>
90     <p style="color: #999;">Pastikan materialized view <code>tenure_stats_mv</code> sudah dibuat dan di-refresh.</p>
91     <a href="index.php?action=create" class="btn btn-primary" style="margin-top: 1rem;">Tambah Data Karyawan</a>
92   </div>
93 <?php endif; ?>
94
95 <div style="margin-top: 2rem; padding: 1rem; background: #e7f3ff; border-radius: 5px;">
96   <strong>Informasi:</strong>
97   Data ini diambil dari Materialized View PostgreSQL dengan fungsi agregat
98   <code>COUNT()</code> dan <code>STRING_AGG()</code> untuk mengelompokkan karyawan berdasarkan masa kerja.
99 </div>
100
101 <?php include 'views/footer.php'; ?>
```

Dashboard

Data Karyawan

Tambah Karyawan

Statistik Departemen

Refresh Dashboard

Statistik Gaji

Statistik Masa Kerja

Statistik Masa Kerja Karyawan

Data berikut diambil dari tenure_stats_mv (Materialized View) di database PostgreSQL

Total Karyawan

20

Level Masa Kerja

2 Level

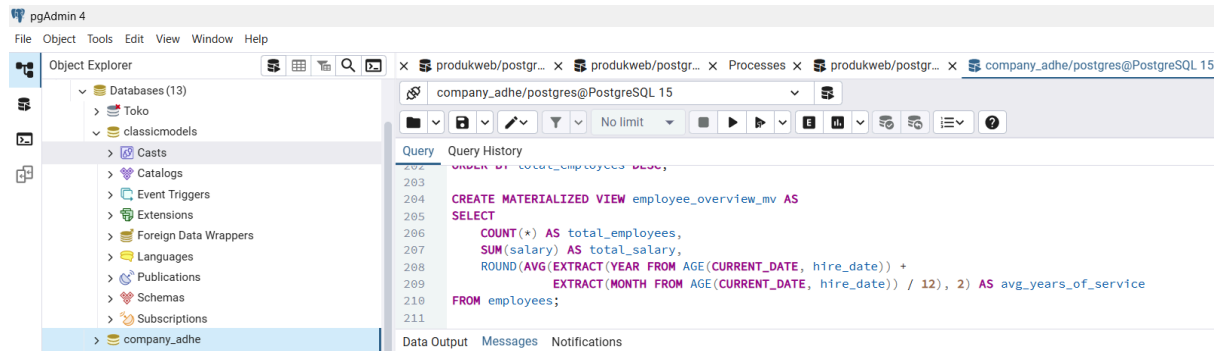
Tingkat Masa Kerja	Jumlah Karyawan	Nama Karyawan
Middle	7	<ul style="list-style-type: none">Dewi AnggrainiLinda PermataRina WulandariEko SupriyantoFitri HandayaniGita MaharaniNina Zahra
Senior	13	<ul style="list-style-type: none">Ahmad HidayatSiti RahayuBudi SantosoRizki PratamaMaya SariJoko WidodoFajar NugrohoHendra KurniawanDian PuspitaIrfan SetiawanKartika DewiLukman HakimOscar Fernando

Refresh Data

Informasi: Data ini diambil dari Materialized View PostgreSQL dengan fungsi agregat COUNT() dan STRING_AGG() untuk mengelompokkan karyawan berdasarkan masa kerja.

3. Halaman Ringkasan Karyawan (employee_overview.php)

- Tampilkan: Total karyawan, total gaji per bulan, rata-rata masa kerja
- Gunakan fungsi: COUNT(), SUM(), AVG()



```
1 <?php
2 /**
3  * FILE: views/employee_overview.php
4  * FUNGSI: Menampilkan ringkasan data keseluruhan karyawan
5  */
6 include 'views/header.php';
7 ?>
8
9 <h2> Ringkasan Karyawan</h2>
10
11 <p style="margin-bottom: 2rem; color: #666;">
12     Data berikut diambil dari <code>employee_overview_mv</code> (Materialized View) PostgreSQL.
13 </p>
14
15 <?php if (!empty($overview)): ?>
16     <!-- Cards Summary -->
17     <div class="dashboard-cards">
18         <div class="card">
19             <h3>Total Karyawan</h3>
20             <div class="number"><?php echo $overview['total_employees']; ?></div>
21         </div>
22         <div class="card">
23             <h3>Total Gaji per Bulan</h3>
24             <div class="number">Rp <?php echo number_format($overview['total_salary'], 0, ',', '.'); ?></div>
25         </div>
26         <div class="card">
27             <h3>Rata-rata Masa Kerja</h3>
28             <div class="number"><?php echo $overview['avg_years_of_service']; ?> tahun</div>
29         </div>
30     </div>
```

```
32 <!-- Tombol Refresh -->
33 <div style="text-align:center; margin-top: 30px;">
34 <a href="index.php?action=refresh_employee_mv"
35 style="
36 display: inline-block;
37 background-color: #4a6cf7;
38 color: #ffff;
39 padding: 10px 25px;
40 border-radius: 8px;
41 text-decoration: none;
42 font-weight: 500;
43 transition: background-color 0.3s ease;
44 "
45 onmouseover="this.style.backgroundColor='#3651c9'"
46 onmouseout="this.style.backgroundColor='#4a6cf7'">
47 Refresh Data
48 </a>
49 </div>
50
51 <?php else: ?>
52 <div style="text-align: center; padding: 3rem; background: #f8f9fa; border-radius: 8px;">
53 <p style="font-size: 1.2rem; color: #666;">Tidak ada data ringkasan karyawan.</p>
54 <p style="color: #999;">Pastikan materialized view <code>employee_overview_mv</code> sudah dibuat dan di-refresh.</p>
55 </div>
56 <?php endif; ?>
57
58 <div style="margin-top: 2rem; padding: 1rem; background: #e7f3ff; border-radius: 5px;">
59 <strong>Informasi:</strong>
60 Data ini dihitung menggunakan fungsi agregat <code>COUNT()</code>, <code>SUM()</code>, dan <code>AVG()</code> untuk menampilkan total karyawan,
61 </div>
62
63 <?php include 'views/footer.php'; ?>
```

Sistem Manajemen Karyawan

Aplikasi CRUD Sederhana dengan PostgreSQL & PHP

Dashboard

Data Karyawan

Tambah Karyawan


Statistik Departemen

Refresh Dashboard

Statistik Gaji

Statistik Masa Kerja

Ringkasan Karyawan

 **Ringkasan Karyawan**

Data berikut diambil dari employee_overview_mv (Materialized View) PostgreSQL.

Total Karyawan

20

Total Gaji per Bulan

Rp 152.300.000

Rata-rata Masa Kerja

3.42 tahun

Refresh Data

Informasi: Data ini dihitung menggunakan fungsi agregat COUNT(), SUM(), dan AVG() untuk menampilkan total karyawan, total gaji, serta rata-rata masa kerja.

© 2025 Sistem Manajemen Karyawan. All rights reserved.

TUGAS 2: BUAT TABLE FUNCTION POSTGRESQL

1. Function: get_employees_by_salary_range(min_salary DECIMAL, max_salary DECIMAL)

- Mengembalikan: karyawan dengan gaji dalam range tertentu
- Kolom: id, full_name, department, position, salary

The screenshot shows the pgAdmin 4 interface. On the left, the 'Object Explorer' pane displays a tree of databases, with 'company_adhe' selected under the 'postgres' database. The main pane shows a SQL query editor with the following code:

```
CREATE OR REPLACE FUNCTION get_employees_by_salary_range(  
    min_salary DECIMAL,  
    max_salary DECIMAL  
)  
RETURNS TABLE (  
    id INT,  
    full_name VARCHAR(100),  
    department VARCHAR(50),  
    "position" VARCHAR(50),  
    salary DECIMAL  
)  
AS $$  
BEGIN  
    RETURN QUERY  
    SELECT  
        e.id,  
        (e.first_name || ' ' || e.last_name)::VARCHAR(100) AS full_name,  
        e.department,  
        e."position",  
        e.salary  
    FROM employees e  
    WHERE e.salary BETWEEN min_salary AND max_salary  
    ORDER BY e.salary;  
END;  
$$ LANGUAGE plpgsql;
```

The screenshot shows the pgAdmin 4 interface with the same SQL query editor. The query has been executed, and the results are displayed in the 'Data Output' pane. The results are as follows:

	id	full_name	department	position	salary
1	14	Fitri Handayani	Marketing	Content	5800000.00
2	19	Nina Zahra	Marketing	Social Media	5900000.00
3	13	Eko Supriyanto	Operations	Logistics	6000000.00
4	15	Gita Maharani	HR	Training Officer	6200000.00
5	10	Rina Wulandari	HR	Recruitment	6500000.00
6	8	Linda Permata	Marketing	Digital	6800000.00
7	4	Dewi Anggraini	Marketing	Marketing	7000000.00
8	11	Hendra Kurniawan	Finance	Tax	7200000.00
9	18	Lukman Hakim	Operations	Warehouse	7300000.00
10	3	Budi Santoso	Finance	Accountant	7500000.00
11	17	Kartika Dewi	Finance	Auditor	7700000.00
12	9	Fajar Nugroho	IT	Web Developer	7800000.00
13	5	Rizki Pratama	IT	System Administrator	8000000.00

2. Function: get_department_summary()

- Mengembalikan: ringkasan statistik per departemen
- Kolom: department, employee_count, avg_salary, total_budget

The screenshot shows the pgAdmin 4 interface. On the left, the Object Explorer displays a tree of databases, with 'company_adhe' selected. The main query editor on the right shows the following SQL code:

```
318 CREATE OR REPLACE FUNCTION get_department_summary()
319 RETURNS TABLE (
320     department VARCHAR(50),
321     employee_count BIGINT,
322     avg_salary NUMERIC(15,2),
323     total_budget NUMERIC(15,2)
324 )
325 AS $$
326 BEGIN
327     RETURN QUERY
328     SELECT
329         e.department,
330         COUNT(*)::BIGINT AS employee_count,
331         ROUND(AVG(e.salary), 2) AS avg_salary,
332         SUM(e.salary) AS total_budget
333     FROM employees e
334     GROUP BY e.department
335     ORDER BY e.department;
336 END;
337 $$ LANGUAGE plpgsql;
```

The screenshot shows the pgAdmin 4 interface after executing the function. The query editor displays the following SQL code:

```
338 SELECT * FROM get_department_summary();
339
340
```

Below the query editor, the 'Data Output' tab shows the results of the function call in a table format:

	department character varying	employee_count bigint	avg_salary numeric	total_budget numeric
1	Finance	4	7650000.00	30600000.00
2	HR	3	7400000.00	22200000.00
3	IT	6	8283333.33	49700000.00
4	Marketing	4	6375000.00	25500000.00
5	Operations	3	8100000.00	24300000.00