

LAPORAN TUGAS PEMROGRAMAN WEB

TUGAS KEEMPAT—*Praktikum Modul Keempat: MongoDB*



Nama : Theo Jesen Naftali Sirait

NRP : 3125600106

Dosen Pengajar : Prof. M. Udin Harun Al Rasyid S. Kom, Ph. D

PROGRAM STUDI S. Tr. TEKNIK INFORMATIKA

POLITEKNIK ELEKTRONIKA NEGERI SURABAYA (PENS)

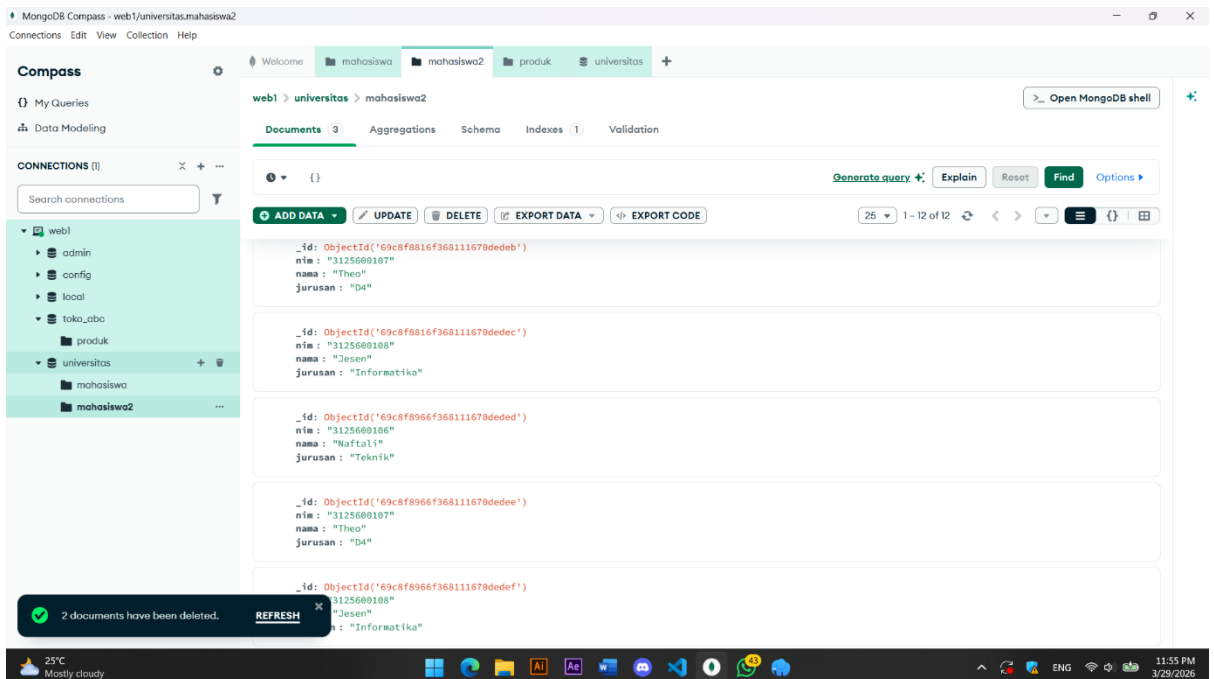
TAHUN 2026

BAB I

PERCOBAAN

1.1. Percobaan Pertama—Ketiga: Instalasi MongoDB.

1.1.1. Output:



1.1.2. Penjelasan:

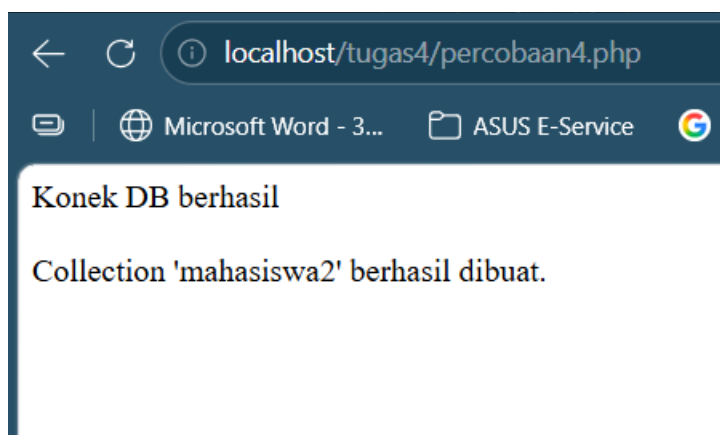
Percobaan ini bertujuan untuk memahami *Database* MongoDB

1.2. Percobaan Keempat—Menghubungkan dan *Collection* ke MongoDB.

1.2.1. *Source Code*:

```
<?php
require 'index.php';
$result = $db->createCollection('mahasiswa2');
echo "Col
?>
```

1.2.2. *Output*:



1.2.3. Penjelasan:

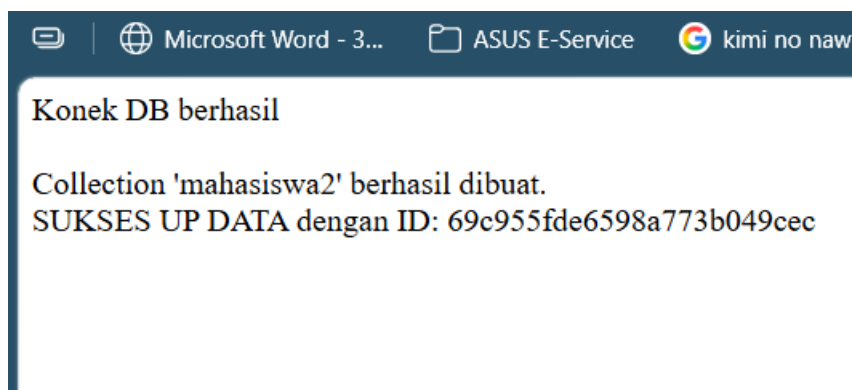
Pada percobaan ini, dibuat sebuah collection bernama mahasiswa di dalam database universitas menggunakan fungsi `createCollection()`.

1.3. Percobaan Kelima—Memasukkan Data ke Tabel (Insert).

1.3.1. *Source Code:*

```
<?php
require 'percobaan4.php';
$collection = $client->universitas->mahasiswa2;
$data = [
    'nim'=>'3125600106',
    'nama'=>'Theo Jesen Naftali Sirait',
    'jurusan'=>'Teknik Informatika'
];
$result = $collection->insertOne($data);
echo "<br>SUKSES UP DATA dengan ID: ".$result->getInsertedId();
?>
```

1.3.2. *Output:*



1.3.3. Penjelasan:

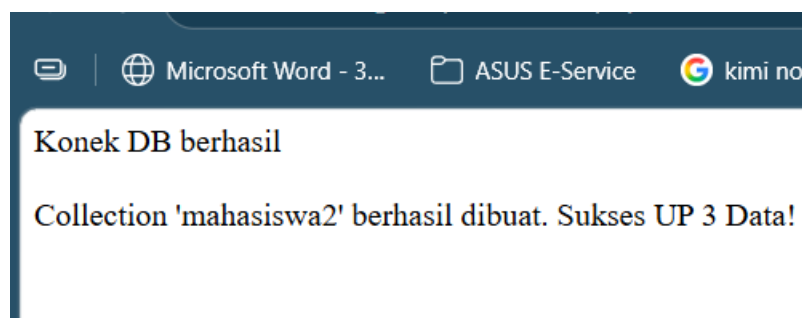
Percobaan ini menunjukkan cara memasukkan satu dokumen ke dalam collection menggunakan `insertOne()`, di mana data dikirim dalam bentuk array asosiatif PHP yang langsung dipetakan menjadi dokumen BSON di MongoDB. Setiap data yang dimasukkan otomatis mendapatkan `_id` unik yang bisa diakses melalui `getInsertedId()`.

1.4. Percobaan Keenam—Memasukkan Data Banyak.

1.4.1. Source Code:

```
<?php
require 'percobaan4.php';
$collection = $client->universitas->mahasiswa2;
$data = [['nim'=>'3125600106', 'nama'=>'Naftali', 'jurusan'=>'Teknik'], ['nim'=>'3125600107', 'nama'=>'Theo',
'jurusan'=>'D4'], ['nim'=>'3125600108', 'nama'=>'Jesen', 'jurusan'=>'Informatika']];
$result = $collection->insertMany($data);
echo "\nSukses UP ".$result->getInsertedCount()." Data!\n";
?>
```

1.4.2. Output:



1.4.3. Penjelasan:

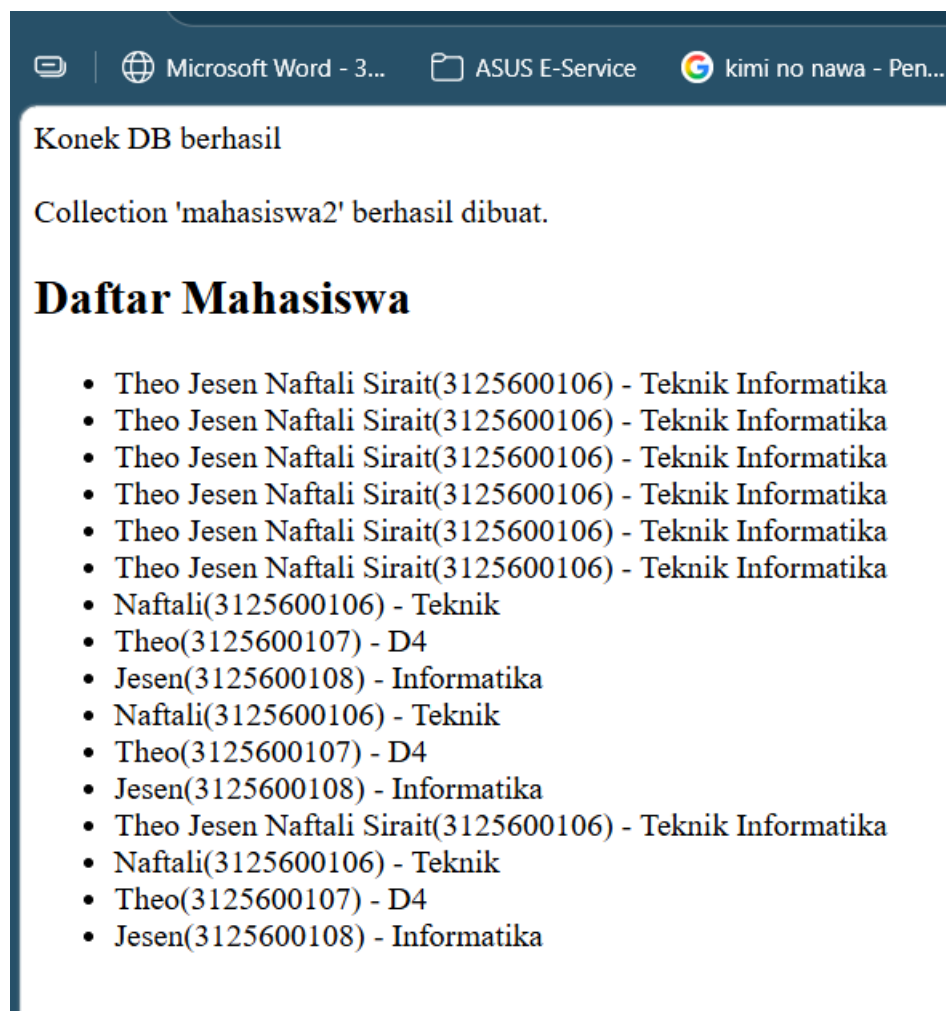
Pada percobaan ini digunakan `insertMany()` untuk memasukkan beberapa dokumen sekaligus dalam satu operasi, yang jauh lebih efisien dibandingkan melakukan `insert` berulang satu per satu. Data dikirim sebagai array berisi beberapa array asosiatif, lalu MongoDB memprosesnya dalam satu batch.

1.5. Percobaan Ketujuh—Menampilkan Data dalam HTML.

1.5.1. *Source Code:*

```
<?php
require 'percobaan4.php';
$collection = $client->universitas->mahasiswa2;
$cursor = $collection->find();
echo "<h2>Daftar Mahasiswa</h2>";
echo "<ul>";
foreach ($cursor as $document) {
    echo "<li>".$document['nama'].("".$document['nim'].") - ".$document['jurusan'], "</li>";
}
echo "</ul>";
?>
```

1.5.2. *Output:*



Konek DB berhasil

Collection 'mahasiswa2' berhasil dibuat.

Daftar Mahasiswa

- Theo Jesen Naftali Sirait(3125600106) - Teknik Informatika
- Theo Jesen Naftali Sirait(3125600106) - Teknik Informatika
- Theo Jesen Naftali Sirait(3125600106) - Teknik Informatika
- Theo Jesen Naftali Sirait(3125600106) - Teknik Informatika
- Theo Jesen Naftali Sirait(3125600106) - Teknik Informatika
- Theo Jesen Naftali Sirait(3125600106) - Teknik Informatika
- Naftali(3125600106) - Teknik
- Theo(3125600107) - D4
- Jesen(3125600108) - Informatika
- Naftali(3125600106) - Teknik
- Theo(3125600107) - D4
- Jesen(3125600108) - Informatika
- Theo Jesen Naftali Sirait(3125600106) - Teknik Informatika
- Naftali(3125600106) - Teknik
- Theo(3125600107) - D4
- Jesen(3125600108) - Informatika

1.5.3. Penjelasan:

Percobaan ini berfokus pada pengambilan dan penampilan data menggunakan fungsi `find()`, yang setara dengan `SELECT *` pada SQL jika tanpa parameter. Hasilnya berupa objek cursor yang bisa diiterasi menggunakan loop untuk mengakses setiap dokumen.

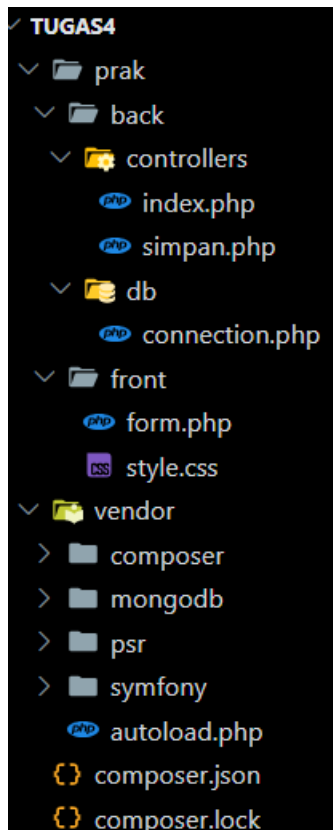
BAB II

TUGAS PRAKTIKUM

2.1. Aplikasi pengadaan barang dengan MongoDB

2.1.1. Source Code:

Struktur File:



form.php:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="/style.css">
  <title>Toko ABC</title>
</head>
<body>
  <h1>Tambah Barang</h1>
  <h2>PT. Google Jaya Makmur</h2>
  <form action="../back/controllers/simpan.php" method="POST">
    <label for="nama">Nama Barang:<br/></label>
    <input name="nama" id="nama" type="text"/>
```

```

<label for="kategori"><br>Kategori:<br></label>
<select name="kategori" id="kategori">
  <option value="1">Barang</option>
  <option value="2">Jasa</option>
</select>
<label for="stok"><br>stok:<br></label>
<input type="range" name="stok"min="0" max="100" value="50" oninput="output.value = this.value">
<output name="output"></output>
<label for="harga"><br>Harga:<br></label>
<input name="harga" id="harga" type="number" min="1000" step="500" value="1000"/>
<button type="submit" name="dataBarang">Simpan Barang</button>
</body>
</html>

```

connection.php:

```

<?php
require '../././vendor/autoload.php';
try {
  $client = new MongoDB\Client("mongodb://localhost:27017");
  $collection = $client->toko_abc->produk;
} catch ( Exception $e) {
  echo "Error: ". $e->getMessage();
}
?>

```

simpan.php:

```

<?php
require './db/connection.php';
echo '<link rel="stylesheet" href=".././front/style.css">';
if (isset($_POST['dataBarang'])) {
  $nama = $_POST['nama'];
  $kategori = $_POST['kategori'];
  $stok = (int) $_POST['stok'];
  $harga = $_POST['harga'];
  $data = ['nama'=> $nama, 'kategori'=> $kategori, 'stok'=> $stok, 'harga'=> $harga];
  $result = $collection->insertOne($data);
  if ($result) {
    echo "<h3>Data Berhasil Disimpan dengan ID: </h3>". $result->getInsertedId();
    echo "<a href='.././front/form.php'>Kembali ke Form</a> ----- <a href='index.php'>Lihat
Tabel</a>";
  } else {
    echo "Error: ";
  }
}
?>

```

index.php:

```

<?php
require __DIR__ . './db/connection.php';
echo '<link rel="stylesheet" href=".././front/style.css">';

```

```

$filter = [];
if (isset($_GET['lowstock'])) {
    $filter = ['stok' => ['$lt' => 5]];
}
$cursor = $collection->find($filter);
?>
<h2>Daftar Mahasiswa</h2>
<form method="GET" id="filterForm">
    <label class="switch">
        <input type="checkbox" name="lowstock" value="1"
            onchange="document.getElementById('filterForm').submit()"
            <?php if (isset($_GET['lowstock'])) echo 'checked'; ?>>
        <span class="slider"></span>
    </label>
    <span>Tampilkan stok <5</span>
</form>
<br>
<table border="2">
<tr>
    <th>Nama Barang</th>
    <th>Kategori</th>
    <th>Stok</th>
    <th>Harga</th>
</tr>
<?php foreach ($cursor as $document): ?>
<tr>
    <td><?= $document['nama'] ?></td>
    <td><?= $document['kategori'] ?></td>
    <td><?= $document['stok'] ?></td>
    <td>Rp. <?= $document['harga'] ?></td>
</tr>
<?php endforeach; ?>
</table><br><br>

```

Figure 1Form Hapus

2.1.2. Output:

Tambah Barang

PT. Google Jaya Makmur

Nama Barang:

Kategori:

stok:

Harga:

Figure 2-form.php

Data Berhasil Disimpan dengan ID:

69c956cfe6598a773b049cf0

[Kembali ke Form](#)

[Lihat Tabel](#)

Figure 3-simpan.php

Daftar Mahasiswa

Tampilkan stok < 5

Nama Barang	Kategori	Stok	Harga
Gir	1	4	Rp. 1000
karbon	1	50	Rp. 1000
Transportasi	2	50	Rp. 3000
Transportasi	2	50	Rp. 3000
Kain	1	58	Rp. 7000

Figure 4-index tanpa filter

Daftar Mahasiswa

Tampilkan stok < 5

Nama Barang	Kategori	Stok	Harga
Gir	1	4	Rp. 1000

Figure 5-index dengan filter stok kurang daripada 5

2.1.3. Penjelasan

Aplikasi yang dibuat merupakan sistem sederhana untuk mengelola data inventaris barang menggunakan PHP dan database MongoDB. Pada tahap awal, dibuat database bernama toko_abc dengan collection produk sebagai tempat penyimpanan data barang. Selanjutnya, dikembangkan file input.php yang berfungsi sebagai antarmuka untuk memasukkan data barang berupa nama, kategori, stok, dan harga. Data yang diinput kemudian diproses oleh simpan.php, di mana data tersebut ditangkap melalui metode POST, dikonversi ke tipe data yang sesuai (khususnya stok dan harga sebagai numerik), lalu disimpan ke dalam collection menggunakan fungsi insertOne() dari MongoDB.

Untuk menampilkan data, dibuat file index.php yang mengambil seluruh dokumen dari collection menggunakan fungsi find() dan menampilkannya dalam bentuk tabel HTML agar mudah dibaca. Selain itu, ditambahkan fitur opsional berupa filter untuk menampilkan hanya barang dengan stok di bawah 5, yang diimplementasikan menggunakan query MongoDB dengan operator \$lt. Fitur ini dikontrol melalui input dari pengguna (checkbox/switch), yang kemudian mempengaruhi query saat pengambilan data.

DOKUMENTASI

naftali.it.student.pens.ac.id