

Untuk melakukan profiling kode PHP agar tahu mana yang lambat atau lebih sering dipanggil, Anda bisa menggunakan alat profiling seperti Xdebug atau Tideways. Namun, jika Anda ingin melakukan profiling sederhana secara manual dalam kode Anda, Anda bisa menggunakan fungsi `microtime(true)` untuk mengukur waktu eksekusi dan menulis log setiap kali fungsi atau bagian kode tertentu dipanggil.

Berikut adalah cara sederhana untuk melakukan profiling dalam kode PHP Anda:

1. Buat Fungsi Logging untuk Profiling

Tambahkan fungsi `log_message` dan `profile` untuk mencatat waktu eksekusi:

php

Copy code

```
<?php

/**
 * Log message to a file
 *
 * This function writes a message to the specified log file with a timestamp.
 *
 * @param string $message The message to log
 * @param string $file The log file path
 */
function log_message($message, $file) {
    // Get the current timestamp
    $timestamp = date("Y-m-d H:i:s");

    // Format the log message
    $log_message = "[" . $timestamp . "] " . $message . "\n";

    // Write the log message to the file
    file_put_contents($file, $log_message, FILE_APPEND);
}

/**
```

```

* Profile a piece of code
*
* This function profiles the execution time of a piece of code.
*
* @param string $label The label for the profiled code
* @param callable $code The piece of code to profile
*/
function profile($label, $code) {
    $start_time = microtime(true);
    $result = $code();
    $end_time = microtime(true);
    $execution_time = $end_time - $start_time;
    log_message("Profile - $label: $execution_time seconds", "profile_log.txt");
    return $result;
}
?>

```

2. Profiling di db.php

Profiling waktu koneksi ke database:

php

Copy code

```
<?php
```

```
include("profiler.php"); // Include the profiler functions
```

```
$servername = "localhost";
```

```
$username = "root";
```

```
$password = "";
```

```
$dbname = "simple_crud";
```

```
// Create connection with profiling
```

```

$conn = profile("Database Connection", function() use ($servername, $username, $password, $dbname)
{
    return new mysqli($servername, $username, $password, $dbname);
});

if ($conn->connect_error) {
    log_message("Connection failed: " . $conn->connect_error, "db_log.txt");
    die("Connection failed: " . $conn->connect_error);
} else {
    log_message("Connection successful", "db_log.txt");
}
?>

```

3. Profiling di add.php

Profiling proses penambahan item:

php

Copy code

```

<?php
include("auth.php");
include("db.php");
include("profiler.php"); // Include the profiler functions

check_login();

if ($_SERVER['REQUEST_METHOD'] == "POST") {
    $name = $_POST['name'];
    $description = $_POST['description'];
    log_message("Received input - Name: $name, Description: $description", "add_log.txt");

    if (!empty($name) && !empty($description)) {

```

```

profile("Insert Item", function() use ($conn, $name, $description) {
    $query = "INSERT INTO items (name, description) VALUES ('$name', '$description')";
    if (mysqli_query($conn, $query)) {
        log_message("Successfully inserted - Name: $name, Description: $description", "add_log.txt");
        header("Location: index.php");
        die;
    } else {
        log_message("Failed to insert - Error: " . mysqli_error($conn), "add_log.txt");
        echo "Error: " . mysqli_error($conn);
    }
});
} else {
    log_message("Invalid input - Name: $name, Description: $description", "add_log.txt");
    echo "Please enter valid information!";
}
}
?>
<!DOCTYPE html>
<html>
<head>
    <title>Add Item</title>
    <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
    <form method="post">
        <div>Add Item</div>
        <input type="text" name="name" placeholder="Name"><br><br>
        <textarea name="description" placeholder="Description"></textarea><br><br>
        <input type="submit" value="Add"><br><br>

```

```
</form>
</body>
</html>
```

4. Profiling di edit.php

Profiling proses pengeditan item:

php

Copy code

```
<?php
include("auth.php");
include("db.php");
include("profiler.php"); // Include the profiler functions

check_login();

$id = $_GET['id'];
if ($_SERVER['REQUEST_METHOD'] == "POST") {
    $name = $_POST['name'];
    $description = $_POST['description'];

    log_message("Received input for update - ID: $id, Name: $name, Description: $description",
"edit_log.txt");

    if (!empty($name) && !empty($description)) {
        profile("Update Item", function() use ($conn, $id, $name, $description) {
            $query = "UPDATE items SET name='$name', description='$description' WHERE id=$id";
            if (mysqli_query($conn, $query)) {
                log_message("Successfully updated - ID: $id, Name: $name, Description: $description",
"edit_log.txt");
                header("Location: index.php");
                die;
            }
        });
    }
}
```

```

    } else {
        log_message("Failed to update - ID: $id, Error: " . mysqli_error($conn), "edit_log.txt");
        echo "Error: " . mysqli_error($conn);
    }
});
} else {
    log_message("Invalid input for update - ID: $id, Name: $name, Description: $description",
"edit_log.txt");
    echo "Please enter valid information!";
}
} else {
    $query = "SELECT * FROM items WHERE id=$id LIMIT 1";
    $result = mysqli_query($conn, $query);
    $row = mysqli_fetch_assoc($result);
}
?>
<!DOCTYPE html>
<html>
<head>
    <title>Edit Item</title>
    <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
    <form method="post">
        <div>Edit Item</div>
        <input type="text" name="name" value="<?php echo $row['name']; ?>"><br><br>
        <textarea name="description"><?php echo $row['description']; ?></textarea><br><br>
        <input type="submit" value="Save"><br><br>
    </form>

```

```
</body>
```

```
</html>
```

5. Profiling di delete.php

Profiling proses penghapusan item:

php

Copy code

```
<?php
```

```
include("auth.php");
```

```
include("db.php");
```

```
include("profiler.php"); // Include the profiler functions
```

```
check_login();
```

```
$id = $_GET['id'];
```

```
if (!empty($id) && is_numeric($id)) {
```

```
    profile("Delete Item", function() use ($conn, $id) {
```

```
        $query = "DELETE FROM items WHERE id=$id";
```

```
        if (mysqli_query($conn, $query)) {
```

```
            log_message("Successfully deleted - ID: $id", "delete_log.txt");
```

```
            header("Location: index.php");
```

```
            die;
```

```
        } else {
```

```
            log_message("Failed to delete - ID: $id, Error: " . mysqli_error($conn), "delete_log.txt");
```

```
            echo "Error: " . mysqli_error($conn);
```

```
        }
```

```
    });
```

```
} else {
```

```
    log_message("Invalid item ID - ID: $id", "delete_log.txt");
```

```
    echo "Invalid item ID!";
```

```
}
```

```
?>
```

6. Profiling di index.php

Profiling waktu pengambilan data:

php

Copy code

```
<?php
```

```
include("auth.php");
```

```
include("db.php");
```

```
include("profiler.php"); // Include the profiler functions
```

```
check_login();
```

```
$items = profile("Fetch Items", function() use ($conn) {
```

```
    $query = "SELECT * FROM items";
```

```
    return mysqli_query($conn, $query);
```

```
});
```

```
?>
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
    <title>Items</title>
```

```
    <link rel="stylesheet" type="text/css" href="style.css">
```

```
</head>
```

```
<body>
```

```
    <a href="add.php">Add New Item</a>
```

```
    <table>
```

```
        <tr>
```

```
            <th>ID</th>
```



```

        <th>Name</th>

        <th>Description</th>

        <th>Actions</th>
    </tr>
    <?php while($row = mysqli_fetch_assoc($items)): ?>
        <tr>
            <td><?php echo $row['id']; ?></td>
            <td><?php echo $row['name']; ?></td>
            <td><?php echo $row['description']; ?></td>
            <td>
                <a href="edit.php?id=<?php echo $row['id']; ?>">Edit</a>
                <a href="delete.php?id=<?php echo $row['id']; ?>">Delete</a>
            </td>
        </tr>
    <?php endwhile; ?>
</table>
</body>
</html>

```

Dengan menggunakan fungsi profile, Anda dapat mencatat waktu eksekusi setiap bagian kode yang penting dalam file profile_log.txt. Ini memberikan pandangan yang lebih jelas tentang bagian mana dari kode Anda yang mungkin memerlukan optimasi.