

Web Dinamis

Pada pertemuan-pertemuan sebelumnya kita telah belajar bagaimana membuat sebuah web statis dengan HTML dan PHP. Pada web statis, konten-konten disisipkan di kode-kode PHP atau HTML sehingga bila terjadi perubahan maka akan berpengaruh pada kode yang telah ditulis. Dari kelemahan tersebut muncullah konsep web dinamis. Konsep web dinamis adalah memisahkan konten dengan kode. Sehingga bila ada perubahan pada konten, tidak serta merta merubah kode yang telah ditulis.

Untuk menerapkan konsep web dinamis, kita membutuhkan sebuah Basisdata (DBMS). basisdata inilah yang digunakan untuk menyimpan konten-konten web. Terdapat banyak DBMS yang dapat kita pakai untuk membangun web dinamis, dari RDBMS hingga NO-SQL. Walaupun pada prakteknya kode yang nanti kita tulis dapat digunakan pada sebagian besar DBMS, namun pada kelas ini, kita akan membatasi penggunaan database hanya pada database MySQL.

Agar dapat memahami mekanisme penggunaan DBMS pada PHP, Anda diharuskan memahami terlebih dahulu konsep basisdata, atau lebih spesifik pada DBMS MySQL. Pada modul ini tidak akan dijelaskan secara detail, silahkan buka kembali referensi matakuliah Basisdata atau dari sumber-sumber lain di internet atau perpustakaan.

Query-query yang nantinya sering kita gunakan untuk membangun web dinamis adalah query dengan jenis DML (*Data Manipulation Language*), yang meliputi SELECT, INSERT, UPDATE, DELETE. Silahkan pelajari dan praktekkan ulang query-query tersebut untuk mengingat kembali.

Terdapat beberapa cara untuk menghubungkan PHP dengan DBMS (MySQL), antara lain:

1. Dengan fungsi mysql_connect()
2. Dengan fungsi mysqli_connect()
3. PDO (PHP Data Object)
4. ORM / Active Record

Khusus fungsi mysql_connect() sudah tidak didukung lagi pada PHP versi 7 dan tidak direkomendasikan lagi untuk digunakan karena banyak kerentanan pada fungsi tersebut. Sebagai alternatif, kita dapat menggunakan fungsi mysqli_connect(). Namun pada kelas ini, kita akan fokus pada penggunaan PDO yang lebih *object-oriented* daripada mysqli_connect(). Keuntungan utama menggunakan PDO adalah, dengan PDO kita dapat menghubungkan PHP dengan banyak DBMS tidak hanya MySQL. Jadi, misalkan kita telah membuat aplikasi web dengan database MySQL, dan di kemudian hari kita ingin mengganti MySQL dengan database PostgreSQL, kita tidak perlu mengganti kode-kode aplikasi kita dan hanya cukup mengganti pada beberapa parameter objek PDO. Hal ini berbeda jika kita menggunakan mysqli_connect(), bila dikemudian terjadi pergantian basisdata, maka kita harus mengganti sebagian besar kode.

Berikut ini beberapa format cara akses PDO yang akan kita gunakan :

1 CONNECT TO THE DATABASE <pre>\$dsn = "mysql:host=" . DB_HOST . ";charset=" . DB_CHARSET . ";dbname=" . DB_NAME; \$pdo = new PDO(\$dsn, DB_USER, DB_PASSWORD);</pre>	2 UPDATE, INSERT, REPLACE, DELETE <pre>\$stmt = \$pdo->prepare(SQL); \$stmt->execute([DATA ARRAY]); // Last affected ID \$lastID = \$pdo->lastInsertId();</pre>
3 SELECT - ALL AT ONCE <pre>\$stmt = \$pdo->prepare(SQL); \$stmt->execute([DATA ARRAY]); \$result = \$stmt->fetchAll();</pre>	4 SELECT - ROW BY ROW <pre>\$stmt = \$pdo->prepare(SQL); \$stmt->execute([DATA ARRAY]); while (\$row = \$stmt->fetch(PDO::FETCH_NAMED)) // ... }</pre>

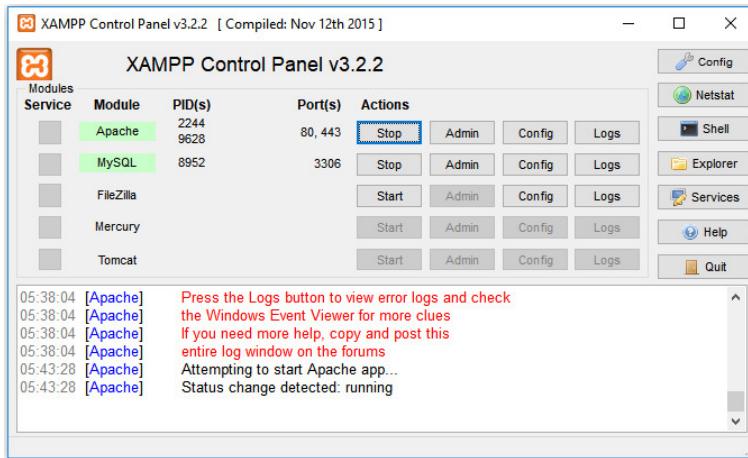


A. MEMBUAT BASISDATA DAN TABEL

Langkah pertama untuk membuat web dinamis adalah membuat basisdata dan tabel-tabel. Untuk mulai membuat basisdata, Anda memerlukan sebuah editor SQL, ada banyak editor yang dapat digunakan, antara lain SQLFront, SQLYog, HeidiSQL, PHPMyAdmin, dll.

Berikut ini langkah-langkahnya.

1. Jalankan Apache dan MySQL di XAMPP



2. Buatlah basisdata dengan nama “pdocrud”.
3. Buatlah tabel “contacts” dengan kriteria sesuai query dibawah ini

```
1 CREATE TABLE `contacts` (
2   `id` INT(11) NOT NULL AUTO_INCREMENT,
3   `name` VARCHAR(255) NOT NULL,
4   `email` VARCHAR(255) NOT NULL,
5   `phone` VARCHAR(255) NOT NULL,
6   `title` VARCHAR(255) NOT NULL,
7   `created` DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP,
8   PRIMARY KEY (`id`)
9 )
```

4. Tambahkan beberapa record pada tabel “contacts”

#	id	name	email	phone	title	created
1	1	John Doe	johndoe@example.com	2026550143	Judge	2019-05-08 17:32:00
2	2	David Deacon	daviddeacon@example.com	2025550121	Security	2019-05-08 17:28:44
3	3	Sam White	samwhite@example.com	2004550121	Employee	2019-05-08 17:29:27
4	4	Colin Chaplin	colinchaplin@example.com	2022550178	Supervisor	2019-05-08 17:29:27
5	5	Ricky Waltz	rickywaltz@example.com	7862342390	Employee	2019-05-09 19:16:00
6	6	Arnold Hall	arnoldhall@example.com	5089573579	Manager	2019-05-09 19:17:00
7	7	Toni Adams	alvah1981@example.com	2603668738	Employee	2019-05-09 19:19:00
8	8	Donald Perry	donald1983@example.com	7019007916	Employee	2019-05-09 19:20:00



B. MENAMPILKAN DATA DI WEB BROWSER

Seperti yang telah kita bahas sebelumnya, kita akan menghubungkan PHP dengan MySQL menggunakan PDO. Dibawah ini adalah tampilan yang menjadi goal kita.

Read Contacts					
Create Contact					
#	Name	Email	Phone	Title	Created
1	John Doe	johndoe@example.com	2026550143	Judge	2019-05-08 17:32:00
2	David Deacon	daviddeacon@example.com	2025550121	Security	2019-05-08 17:28:44
3	Sam White	samwhite@example.com	2004550121	Employee	2019-05-08 17:29:27
4	Colin Chaplin	colinchaplin@example.com	2022550178	Supervisor	2019-05-08 17:29:27
5	Ricky Waltz	rickywaltz@example.com	7862342390		2019-05-09 19:16:00
6	Arnold Hall	arnoldhall@example.com	5089573579	Manager	2019-05-09 19:17:00
7	Toni Adams	alvah1981@example.com	2603668738		2019-05-09 19:19:00

Berikut ini langkah-langkahnya :

1. Membuat file functions.php

File “functions.php” berisi fungsi untuk menghubungkan PHP ke MySQL menggunakan PDO. File ini nantinya akan selalu di “include” di setiap file php yang membutuhkan koneksi basisdata. Dibawah ini kode functions.php.

```
● ● ●
1 <?php
2
3     function pdo_connect(){
4         $DATABASE_HOST = 'localhost';
5         $DATABASE_USER = 'root';
6         $DATABASE_PASS = 'password_anda';
7         $DATABASE_NAME = 'pdocrud';
8         try {
9             return new PDO( 'mysql:host=' . $DATABASE_HOST .
10                         ;dbname=' . $DATABASE_NAME ,
11                         $DATABASE_USER,
12                         $DATABASE_PASS);
13         } catch (PDOException $exception) {
14             die ('Failed to connect to database!');
15         }
16     }
17
18 ?>
19
20
```

Penjelasan kode :

- Baris 5-6, sesuaikan dengan username dan password DBMS di komputer Anda
- Baris 9 adalah kode untuk menghubungkan PHP dengan MySQL. Sesuaikan parameter didalamnya mulai dari jenis DBMS, host, nama basisdata, user dan password DBMS.
- Baris 13 adalah eksepsi yang akan dieksekusi apabila PDO gagal menghubungi basisdata



2. Membuat file index.php

```
1 <?php
2
3     include "functions.php";
4
5     $pdo = pdo_connect();
6     $stmt = $pdo->prepare('SELECT * FROM contacts');
7     $stmt->execute();
8     $contacts = $stmt->fetchAll(PDO::FETCH_ASSOC);
9
10 ?>
11
12 <h2>Read Contacts</h2>
13 <a href="create.php" class="create-contact">Create Contact</a>
14 <table>
15     <thead>
16         <tr>
17             <th>#</th>
18             <th>Name</th>
19             <th>Email</th>
20             <th>Phone</th>
21             <th>Title</th>
22             <th>Created</th>
23             <th></th>
24         </tr>
25     </thead>
26     <tbody>
27         <?php foreach ($contacts as $contact): ?>
28         <tr>
29             <td><?=$contact['id']?></td>
30             <td><?=$contact['name']?></td>
31             <td><?=$contact['email']?></td>
32             <td><?=$contact['phone']?></td>
33             <td><?=$contact['title']?></td>
34             <td><?=$contact['created']?></td>
35             <td>
36                 <a href="update.php?id=<?=$contact['id']?>" class="edit">edit</a>
37                 <a href="delete.php?id=<?=$contact['id']?>" class="trash">delete</a>
38             </td>
39         </tr>
40     <?php endforeach; ?>
41     </tbody>
42 </table>
```

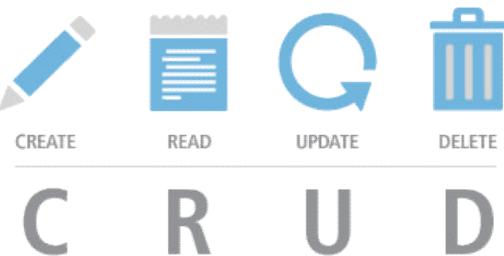
Penjelasan kode :

- Baris 3 adalah kode untuk menyisipkan file “functions.php” yang telah kita buat sebelumnya.
- Baris 4 kita memanggil fungsi pdo_connect() yang terdapat pada file “functions.php”.
- Baris 6-7 PDO mengeksekusi query SQL
- Baris 8 kode untuk memperoleh semua data dari hasil eksekusi query dan menetapkannya ke sebuah variabel bernama contacts. PDO:FETCH_ASSOC adalah method untuk memperoleh data lama bentuk assosiative array. Selain FETCH_ASSOC terdapat method lain yang dapat digunakan (<https://www.php.net/manual/en/pdostatement.fetch.php>)
- Baris 27-40 adalah kode perulangan untuk menampilkan semua data
- Baris 36-37 adalah kode HTML untuk membuat tautan ke form edit dan hapus data.



C. CRUD

CRUD adalah akronim untuk Create, Read, Update, dan Delete. Operasi CRUD adalah manipulasi data dasar untuk database. Pada bahasan sebelumnya kita telah mencoba menampilkan data record pada web browser, dan secara tidak langsung kita telah mempraktekkan konsep CRUD itu sendiri khususnya pada bagian Read. Pada subbab ini akan dibahas bagaimana membuat aplikasi yang dapat menambahkan data baru, mengubah serta menghapus data.



Berikut ini langkah-langkahnya:

1. Membuat File Create.php

```
1 <?php
2
3     include 'functions.php';
4     $pdo = pdo_connect();
5
6     if (!empty($_POST)) {
7         $name = $_POST['name'];
8         $email = $_POST['email'];
9         $phone = $_POST['phone'];
10        $title = $_POST['title'];
11        $created = date('Y-m-d H:i:s');
12        $stmt = $pdo->prepare('INSERT INTO contacts VALUES (?, ?, ?, ?, ?, ?)');
13        $stmt->execute([$id, $name, $email, $phone, $title, $created]);
14        header("location:index.php");
15    }
16
17 ?>
18
19 <h2>Create Contact</h2>
20 <form action="create.php" method="post">
21     <label>Name</label>
22     <input type="text" name="name" id="name" required><br>
23     <label>Email</label>
24     <input type="text" name="email" id="email" required><br>
25     <label>Phone</label>
26     <input type="text" name="phone" id="phone" required><br>
27     <label>Title</label>
28     <input type="text" name="title" id="title" required><br>
29     <input type="submit" value="Create">
30     <a href="index.php">Cancel</a>
31 </form>
```

Penjelasan kode :

- Baris 3 adalah kode untuk menyisipkan file “functions.php”.
- Baris 4 kita memanggil fungsi pdo_connect() yang terdapat pada file “functions.php”.
- Baris 6 adalah kondisi apabila terdapat kiriman dari form maka blok 7-14 akan dieksekusi.
- Baris 7-10 adalah kode untuk menetapkan kiriman-kiriman dari form.
- Baris 11 adalah kode untuk membuat tanggal dan jam sekarang secara otomatis untuk masukan record database.
- Baris 12-13 PDO mengeksekusi query SQL untuk menambahkan data.
- Baris 13 merupakan parameter berupa array yang akan mengisi value-value sesuai simbol tanda tanya pada query SQL di baris 12.
- Baris 14 adalah kode untuk mengarahkan web browser ke halaman index.php setelah semua kode pada baris sebelumnya dieksekusi.
- Baris 20-31 merupakan kode HTML untuk menampilkan form.

The screenshot shows a web browser window with the URL "localhost/pdocrud/create.php". The page title is "Create Contact". There are four input fields labeled "Name", "Email", "Phone", and "Title", each containing a placeholder text ("Name", "Email", "Phone", and "Title" respectively). At the bottom of the form are two buttons: "Create" and "Cancel".

2. Membuat File Update.php

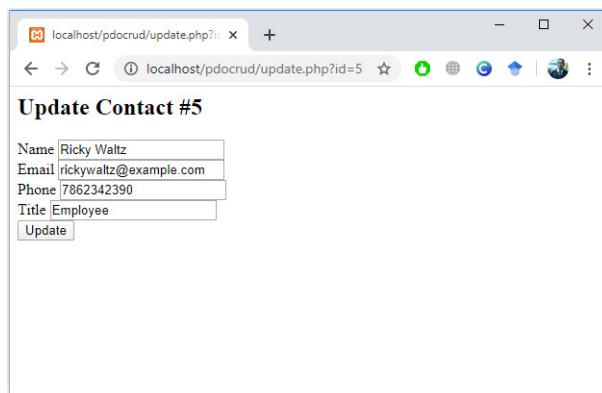
```
1 <?php
2
3     include 'functions.php';
4     $pdo = pdo_connect();
5
6     if (isset($_GET['id'])) {
7         if (!empty($_POST)) {
8             $name = $_POST['name'];
9             $email = $_POST['email'];
10            $phone = $_POST['phone'];
11            $title = $_POST['title'];
12            $stmt = $pdo->prepare('UPDATE contacts
13                                SET name = ?, email = ?, phone = ?, title = ?
14                                WHERE id = ?');
15            $stmt->execute([$name, $email, $phone, $title, $_GET['id']]);
16            header("location:index.php");
17        }
18
19        $stmt = $pdo->prepare('SELECT * FROM contacts WHERE id = ?');
20        $stmt->execute([$_GET['id']]);
21        $contact = $stmt->fetch(PDO::FETCH_ASSOC);
22        if (!$contact) {
23            die ('Contact doesn\'t exist!');
24        }
25    } else {
26        die ('No ID specified!');
27    }
28
29 ?>
30
31
32 <h2>Update Contact #<?=$contact['id']?></h2>
33 <form action="update.php?id=<?=$contact['id']?>" method="post">
34     <label>Name</label>
35     <input type="text" name="name" value="<?=$contact['name']?>" id="name">
36     <br>
37     <label>Email</label>
38     <input type="text" name="email" value="<?=$contact['email']?>" id="email">
39     <br>
40     <label>Phone</label>
41     <input type="text" name="phone" value="<?=$contact['phone']?>" id="phone">
42     <br>
43     <label>Title</label>
44     <input type="text" name="title" value="<?=$contact['title']?>" id="title">
45     <br>
46     <input type="submit" value="Update">
47 </form>
```

Penjelasan kode :

- Baris 3 adalah kode untuk menyiapkan file “functions.php”.
- Baris 4 kita memanggil fungsi pdo_connect() yang terdapat pada file “functions.php”.
- Baris 6 adalah kondisi apabila properti “id” pada URL tidak kosong maka blok 7-24 akan dieksekusi.
- Baris 7 adalah kondisi apabila terdapat kiriman dari form maka blok 8-16 akan dieksekusi.
- Baris 8-11 adalah kode untuk menetapkan kiriman-kiriman dari form.
- Baris 12-15 PDO mengeksekusi query SQL untuk update data berdasarkan “id”.



- Baris 15 merupakan parameter berupa array yang akan mengisi value-value sesuai simbol tanda tanya pada query SQL di baris 12.
- Baris 16 adalah kode untuk mengarahkan web browser ke halaman index.php setelah semua kode pada baris sebelumnya dieksekusi.
- Baris 19-20 PDO mengeksekusi query SQL untuk mendapatkan sebuah record berdasarkan “id”-nya.
- Baris 21 adalah kode untuk memperoleh data dari hasil eksekusi query pada baris 19 dan menetapkannya ke sebuah variabel bernama contact.
- Baris 33-47 merupakan kode HTML untuk menampilkan form. Perhatikan pada setiap elemen input form (baris 35, 38, 41, 44) kita menetapkan value untuk tiap-tiap elemen input berdasarkan variabel contact yang telah kita tetapkan sebelumnya. Perhatikan pula pada atribut action form, action mengarah kepada file “update.php” dan dengan tambahan properti “id” (mis. update.php?id=5).



3. Membuat File Delete.php

```

1 <?php
2
3     include 'functions.php';
4     $pdo = pdo_connect();
5
6     if (isset($_GET['id'])) {
7         $stmt = $pdo->prepare('DELETE FROM contacts WHERE id = ?');
8         $stmt->execute([$_GET['id']]);
9         header("location:index.php");
10    } else {
11        die ('No ID specified!');
12    }
13
14 ?>
15
16

```

Penjelasan kode :

- Baris 3 adalah kode untuk menyisipkan file “functions.php”.
- Baris 4 kita memanggil fungsi pdo_connect() yang terdapat pada file “functions.php”.
- Baris 6 adalah kondisi apabila terdapat kiriman dari form maka blok 7-9 akan dieksekusi.
- Baris 7 PDO mengeksekusi query SQL untuk menghapus data berdasarkan “id”.
- Baris 8 merupakan parameter berupa array yang akan mengisi value-value sesuai simbol tanda tanya pada query SQL di baris 7.
- Baris 9 adalah kode untuk mengarahkan web browser ke halaman index.php setelah semua kode pada baris sebelumnya dieksekusi.



D. STYLING

Pada subbab ini kita akan mempercantik tampilan aplikasi, nantinya aplikasi akan nampak seperti gambar dibawah ini.

Read Contacts						
Create Contact						
Show 10 entries			Search:			
#	Name	Email	Phone	Title	Created	
1	John Doe	johndoe@example.com	2026550143	Judge	2019-05-08 17:32:00	edit delete
2	David Deacon	daviddeacon@example.com	2025550121	Security	2019-05-08 17:28:44	edit delete
3	Sam White	samwhite@example.com	2004550121	Employee	2019-05-08 17:29:27	edit delete
4	Colin Chaplin	colinchaplin@example.com	2022550178	Supervisor	2019-05-08 17:29:27	edit delete
5	Ricky Waltz	rickywaltz@example.com	7862342390	Employee	2019-05-09 19:16:00	edit delete
6	Arnold Hall	arnoldhall@example.com	5089573579	Manager	2019-05-09 19:17:00	edit delete
7	Toni Adams	alvah1981@example.com	2603668738	Employee	2019-05-09 19:19:00	edit delete
8	Donald Perry	donald1983@example.com	7019007916	Employee	2019-05-09 19:20:00	edit delete
9	Joe McKinney	nadia.dooole0@example.com	6153353674	Employee	2019-05-09 19:20:00	edit delete
10	Angela Horst	angela1977@example.com	3094234980	Assistant	2019-05-09 19:21:00	edit delete
#	Name	Email	Phone	Title	Created	
Showing 1 to 10 of 15 entries						
					Previous 1 2 Next	

Tampilan diatas dibangun dengan menggunakan framework Bootstrap dan library datatables. Framework bootstrap dapat diperoleh di alamat <https://getbootstrap.com>, sedangkan library datatables dapat diperoleh di alamat <https://datatables.net>.



Berikut ini langkah-langkah untuk memberi style :

1. Tambahkan kode-kode styling pada “functions.php”. Berikut ini kode secara lengkap.

```
● ● ●
1 <?php
2
3 function pdo_connect(){
4     $DATABASE_HOST = 'localhost';
5     $DATABASE_USER = 'root';
6     $DATABASE_PASS = 'password_anda';
7     $DATABASE_NAME = 'pdocrud';
8     try {
9         return new PDO( 'mysql:host=' . $DATABASE_HOST .
10                     ';dbname=' . $DATABASE_NAME,
11                     $DATABASE_USER,
12                     $DATABASE_PASS);
13     } catch (PDOException $exception) {
14         die ('Failed to connect to database!');
15     }
16 }
17
18 function style_script(){
19     return
20     '<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css">
21     <link rel="stylesheet" href="https://cdn.datatables.net/1.10.20/css/dataTables.bootstrap4.min.css">
22     <script src="https://code.jquery.com/jquery-3.4.1.slim.js"></script>
23     <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js"></script>
24     <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/bootstrap.min.js"></script>
25     <script src="https://cdn.datatables.net/1.10.20/js/jquery.dataTables.min.js"></script>
26     <script src="https://cdn.datatables.net/1.10.20/js/dataTables.bootstrap4.min.js"></script>';
27 }
28 ?>
```

Kode-kode javascript dan css diatas dapat disalin atau diunduh pada tautan berikut :

- <https://getbootstrap.com/docs/4.4/getting-started/introduction/>
- <https://cdn.datatables.net/> (Pilih “Bootstrap 4” pada bagian “Styling options”)

Penjelasan kode:

Menambahkan fungsi baru yaitu style_script() yang berisi tautan-tautan file css dan javascript dari *cloud storage*. Anda dapat mengunduhnya satu-persatu lalu menyimpannya di komputer Anda sehingga aplikasi tidak membutuhkan koneksi ke sumber eksternal. Dibawah ini adalah kode style_script() jika Anda memilih untuk mengunduh semua kode javascript dan css ke penyimpanan lokal.

```
● ● ●
1 function style_script(){
2     return '<link rel="stylesheet" href="css/bootstrap.min.css">
3     <link rel="stylesheet" href="css/dataTables.bootstrap4.min.css">
4     <script src="js/jquery-3.4.1.slim.js"></script>
5     <script src="js/popper.min.js"></script>
6     <script src="js/bootstrap.min.js"></script>
7     <script src="js/jquery.dataTables.min.js"></script>
8     <script src="js/dataTables.bootstrap4.min.js"></script>';
9 }
```

2. Modifikasi file “index.php” sehingga menjadi seperti dibawah ini.

```
● ● ●
1 <?php include "functions.php"; ?>
2 <!DOCTYPE html>
3 <html lang="en">
4 <head>
5     <meta charset="UTF-8">
6     <meta name="viewport" content="width=device-width, initial-scale=1.0">
7     <meta http-equiv="X-UA-Compatible" content="ie=edge">
8     <?= style_script() ?>
9     <script>
10         $(document).ready(function() {
11             $('#employee').DataTable();
12         });
13     </script>
14
15     <title>Simple CRUD with PDO</title>
16 </head>
17 <body>
18     <?php
19         $pdo = pdo_connect();
20         $stmt = $pdo->prepare('SELECT * FROM contacts');
21         $stmt->execute();
22         $contacts = $stmt->fetchAll(PDO::FETCH_ASSOC);
23     ?>
24     <div class="container">
25         <h2>Read Contacts</h2>
26         <a type="button" class="btn btn-primary" href="create.php"
27             class="create-contact">Create Contact</a>
28         <br><br>
29         <div class="row">
30             <div class="col">
31                 <table class="table table-striped" id="employee">
32                     <thead>
33                         <tr>
34                             <th>#</th>
35                             <th>Name</th>
36                             <th>Email</th>
37                             <th>Phone</th>
38                             <th>Title</th>
39                             <th>Created</th>
40                             <th></th>
41                         </tr>
42                     </thead>
43                     <tbody>
44                         <?php foreach ($contacts as $contact) : ?>
45                         <tr>
46                             <td><?= $contact['id'] ?></td>
47                             <td><?= $contact['name'] ?></td>
48                             <td><?= $contact['email'] ?></td>
49                             <td><?= $contact['phone'] ?></td>
50                             <td><?= $contact['title'] ?></td>
51                             <td><?= $contact['created'] ?></td>
52                             <td class="actions">
53                                 <a type="button"
54                                     class="btn btn-sm btn-outline btn-success edit"
55                                     href="update.php?id=<?= $contact['id'] ?>">edit</a>
56                                 <a type="button"
57                                     class="btn btn-sm btn-outline btn-danger trash"
58                                     href="delete.php?id=<?= $contact['id'] ?>">delete</a>
59                             </td>
60                         </tr>
61                     <?php endforeach; ?>
62                 </tbody>
63                 <tfoot>
64                     <tr>
65                         <th>#</th>
66                         <th>Name</th>
67                         <th>Email</th>
68                         <th>Phone</th>
69                         <th>Title</th>
70                         <th>Created</th>
71                         <th></th>
72                     </tr>
73                 </tfoot>
74             </table>
75         </div>
76     </div>
77     </body>
78 </html>
```



3. Modifikasi file “create.php” sehingga menjadi seperti dibawah ini.

```
● ● ●
1 <?php
2     include 'functions.php';
3     $pdo = pdo_connect();
4
5     if (!empty($_POST)) {
6         $name = $_POST['name'];
7         $email = $_POST['email'];
8         $phone = $_POST['phone'];
9         $title = $_POST['title'];
10        $created = date('Y-m-d H:i:s');
11        $stmt = $pdo->prepare('INSERT INTO contacts VALUES (?, ?, ?, ?, ?, ?)');
12        $stmt->execute([$id, $name, $email, $phone, $title, $created]);
13        header("location:index.php");
14    }
15 ?>
16 <!DOCTYPE html>
17 <html lang="en">
18 <head>
19     <meta charset="UTF-8">
20     <meta name="viewport" content="width=device-width, initial-scale=1.0">
21     <meta http-equiv="X-UA-Compatible" content="ie=edge">
22     <?= style_script() ?>
23     <title>Input New Record</title>
24 </head>
25 <body>
26     <div class="container" style="margin-top:50px">
27         <div class="row">
28             <div class="col-md-5 col-sm-12 col-xs-12">
29                 <div class="card">
30                     <div class="card-body">
31                         <h5 class="card-title">Create contact</h5>
32                         <form action="create.php" method="post">
33                             <input class="form-control form-control-sm"
34                                   placeholder="Type name" type="text"
35                                   name="name" id="name" required><br>
36                             <input class="form-control form-control-sm"
37                                   placeholder="Email" type="text"
38                                   name="email" id="email" required><br>
39                             <input class="form-control form-control-sm"
40                                   placeholder="Phone number" type="text"
41                                   name="phone" id="phone" required><br>
42                             <input class="form-control form-control-sm"
43                                   placeholder="Title" type="text"
44                                   name="title" id="title" required><br>
45                             <input class="btn btn-primary btn-sm"
46                                   type="submit" value="Create">
47                             <a href="index.php" type="button"
48                                   class="btn btn-warning btn-sm">Cancel</a>
49                         </form>
50                     </div>
51                     <div class="col-md-7 col-sm-12 col-xs-12"></div>
52                 </div>
53             </div>
54         </div>
55     </body>
56 </html>
```



Berikut ini tampilan dari file create.php.

The screenshot shows a web browser window with a title bar "Input New Record". The address bar displays "localhost/pdocrud/styliz...". The main content area contains a form titled "Create contact". The form includes four input fields: "Type name", "Email", "Phone number", and "Title". At the bottom of the form are two buttons: "Create" (blue) and "Cancel" (yellow). The entire form is enclosed in a light gray border.

4. Modifikasi file “update.php” sehingga menjadi seperti dibawah ini.

```
1 <?php
2     include 'functions.php';
3     $pdo = pdo_connect();
4     if (isset($_GET['id'])) {
5         if (!empty($_POST)) {
6             $name = $_POST['name'];
7             $email = $_POST['email'];
8             $phone = $_POST['phone'];
9             $title = $_POST['title'];
10            $stmt = $pdo->prepare('UPDATE contacts SET name = ?, email = ?, phone = ?, title = ? WHERE id = ?');
11            $stmt->execute([$name, $email, $phone, $title, $_GET['id']]);
12            header("location:index.php");
13        }
14
15        $stmt = $pdo->prepare('SELECT * FROM contacts WHERE id = ?');
16        $stmt->execute([$_GET['id']]);
17        $contact = $stmt->fetch(PDO::FETCH_ASSOC);
18        if (!$contact) {
19            die ('Contact doesn\'t exist!');
20        }
21    } else {
22        die ('No ID specified!');
23    }
24 ?>
25 <!DOCTYPE html>
26 <html lang="en">
27 <head>
28     <meta charset="UTF-8">
29     <meta name="viewport" content="width=device-width, initial-scale=1.0">
30     <meta http-equiv="X-UA-Compatible" content="ie=edge">
31     <?=style_script()?>
32     <title>Update Contact # <?=$contact['id']?></title>
33 </head>
34 <body>
35 <div class="container" style="margin-top:50px">
36     <div class="row">
37         <div class="col-md-5 col-sm-12 col-xs-12">
38             <div class="card">
39                 <div class="card-body">
40                     <h5 class="card-title">Update contact # <?=$contact['id']?></h5>
41                     <form action="update.php?id=<?=$contact['id']?>" method="post">
42                         <input class="form-control form-control-sm"
43                             placeholder="Type name" type="text"
44                             name="name" value="<?=$contact['name']?>" id="name" required><br>
45                         <input class="form-control form-control-sm"
46                             placeholder="Email" type="text"
47                             name="email" value="<?=$contact['email']?>" id="email" required><br>
48                         <input class="form-control form-control-sm"
49                             placeholder="Phone number" type="text"
50                             name="phone" value="<?=$contact['phone']?>" id="phone"><br>
51                         <input class="form-control form-control-sm"
52                             placeholder="Title" type="text"
53                             name="title" value="<?=$contact['title']?>" id="title"><br>
54                         <input class="btn btn-primary btn-sm" type="submit" value="Update">
55                         <a href="index.php" type="button"
56                             class="btn btn-warning btn-sm">Cancel</a>
57                     </form>
58                 </div>
59             </div>
60         </div>
61         <div class="col-md-7 col-sm-12 col-xs-12"></div>
62     </div>
63 </div>
64 </body>
65 </html>
```



Berikut ini tampilan dari file update.php.

The screenshot shows a web browser window titled "Update Contact # 1". The address bar displays "localhost/pdocrud/styliz...". The main content is a form titled "Update contact # 1" with four input fields:

- First Name: John Doe
- Email: johndoe@example.com
- Phone: 2026550143
- Role: Judge

Below the form are two buttons: "Update" (blue) and "Cancel" (yellow).

5. File "delete.php" tidak perlu dimodifikasi lagi karena tidak membutuhkan *styling*.

E. TUGAS

Kerjakan soal-soal dibawah ini.

1. Buatlah tabel baru (pada database yang sama seperti sebelumnya) dengan field username dan password.
2. Tambahkan beberapa record pada tabel tersebut. Gunakan enkripsi hash MD5 atau SHA khusus pada field password.
3. Buatlah halaman login untuk mengamankan halaman CRUD pada latihan sebelumnya.
4. Data autentikasi halaman login diambil dari tabel database pada poin 1.

Tugas tidak perlu dikumpulkan, namun akan diperiksa pada pertemuan selanjutnya.

