

# Modul Tutorial CRUD Dasar Codeigniter 3 dan Bootstrap

Create : Istia Ningrum & Rani Oktariani NZ

---

## 1. Persiapan

Sebelum memulai tutorial ini, pastikan anda sudah memiliki:

- XAMPP atau software web server lainnya
- Text editor untuk menulis kode, seperti Sublime Text, Notepad++, atau Visual Studio Code (Tapi Kali ini saya menggunakan Visual Studi Code)
- Penguasaan dasar PHP dan MySQL

Berikut Penjelasan Website yang akan dibuat :

Perpusda merupakan aplikasi reservasi atau booking peminjaman buku pada sebuah perpustakaan Daerah Kabupaten Brebes yang dilakukan secara online dengan melibatkan anggota, admin dan aplikasi Perpusda.

Aplikasi Perpusda adalah aplikasi berbasis web yang dibuat untuk booking sebuah buku untuk dipinjam. Tujuan dibuatkannya aplikasi ini agar memudahkan para pengguna dalam mencari buku dan ingin meminjam buku tersebut. Jadi sebelum di datang ke perpustakaan dan meminjam sebuah buku. Pengguna bisa mencari terlebih dahulu buku yang akan dipinjam, untuk memastikan bahwa di perpustakaan yang akan dia kunjungi benar bahwa buku tersebut tersedia.

## 2. Project Starter

- 1) **Download Codeigniter 3 dari situs resminya.**
- 2) **Ekstrak file Codeigniter 3 ke dalam folder htdocs di XAMPP.**
- 3) **Ubah nama folder Codeigniter 3 menjadi nama Perpusda.**
- 4) **Buka folder project Perpusda di Vscope.**

## 3. Pengembangan dan Perancangan Website

### 1) Setting Base URL

Pengaturan base url adalah pengaturan URL dasar dalam sebuah project aplikasi yang dibuat dengan menggunakan codeigniter. Pengaturan base url di codeigniter terdapat di dalam file config.php di dalam folder project codeigniter. **perpusda/application/config/config.php**

Helper dan Library bisa dibilang seperti perpustakaan coding dengan fungsi-fungsi yang berbeda dan memiliki kegunaan masing-masing. Tinggal memanggil dan menggunakannya saja. Kita juga bisa membuat helper atau library kita sendiri di codeigniter untuk memudahkan pekerjaan kita.

Di tahap ini kita akan men-load atau membuka beberapa library dan helper codeigniter yang kita butuhkan dalam pembuatan aplikasi perpustakaan ini. Untuk menjalankan atau membuka library dan helper secara otomatis di codeigniter, kita bisa mengaturnya dalam file autoload.php dalam folder *application/config/autoload.php*.

### **perpusda/application/config/autoload.php**

Library yang akan kita load secara otomatis adalah *database, session dan form\_validation*.

Cari sintak

```
$autoload['libraries'] = array();
```

Kemudian tambahkan database, session dan form\_validation. Sehingga menjadi seperti berikut:

```
$autoload['libraries'] = array('form_validation', 'session',  
'database');
```

Dan helper yang akan kita load secara otomatis adalah *url*:

```
$autoload['helper'] = array();
```

Kemudian tambahkan “url” (*karena kita hanya membutuhkan helper url disini*). Maka akan menjadi seperti berikut.

```
$autoload['helper'] = array('url');
```

Proses autoload library dan helper selesai. Selanjutnya kita akan membuat database untuk aplikasi perpustakaan dan melakukan konfigurasi database pada codeigniter.

## **2) Merancang & Membuat Database Aplikasi perpustakaan dan Konfigurasi database di Codeigniter**

Berikut adalah gambaran struktur database yang akan kita buat untuk aplikasi perpustakaan ini. Buat sebuah database baru dengan nama “pustaka” kemudian buatlah table-table seperti berikut:

### a) Tabel User (user)

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	<b>id</b>	int(11)			Tidak	Tidak ada		AUTO_INCREMENT	Ubah  Hapus Lainnya
2	<b>nama</b>	varchar(128)	latin1_swedish_ci		Tidak	Tidak ada			Ubah  Hapus Lainnya
3	<b>email</b>	varchar(128)	latin1_swedish_ci		Tidak	Tidak ada			Ubah  Hapus Lainnya
4	<b>image</b>	varchar(128)	latin1_swedish_ci		Tidak	Tidak ada			Ubah  Hapus Lainnya
5	<b>password</b>	varchar(256)	latin1_swedish_ci		Tidak	Tidak ada			Ubah  Hapus Lainnya
6	<b>role_id</b>	int(11)			Tidak	Tidak ada			Ubah  Hapus Lainnya
7	<b>is_active</b>	int(1)			Tidak	Tidak ada			Ubah  Hapus Lainnya
8	<b>tanggal_input</b>	int(11)			Tidak	Tidak ada			Ubah  Hapus Lainnya
9	<b>alamat</b>	varchar(255)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah  Hapus Lainnya

### b) Tabel Role (role)

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	<b>id</b>	int(11)			Tidak	Tidak ada		AUTO_INCREMENT	Ubah  Hapus Lainnya
2	<b>role</b>	varchar(128)	latin1_swedish_ci		Tidak	Tidak ada			Ubah  Hapus Lainnya

### Isi Tabel role

	id	role
<input type="checkbox"/> Ubah  Salin  Hapus	1	administrator
<input type="checkbox"/> Ubah  Salin  Hapus	2	member

### c) Tabel Kategori (kategori)

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	<b>id</b>	int(5)			Tidak	Tidak ada		AUTO_INCREMENT	Ubah  Hapus Lainnya
2	<b>kategori</b>	varchar(45)	latin1_swedish_ci		Tidak	Tidak ada			Ubah  Hapus Lainnya

## Isi Tabel kategori

				id	kategori
<input type="checkbox"/>				1	Komputer
<input type="checkbox"/>				2	Bahasa
<input type="checkbox"/>				3	Sains
<input type="checkbox"/>				4	Hobby
<input type="checkbox"/>				5	Komunikasi
<input type="checkbox"/>				6	Hukum
<input type="checkbox"/>				7	Agama
<input type="checkbox"/>				8	Populer
<input type="checkbox"/>				9	Komik
<input type="checkbox"/>				11	fiksi

## d) Tabel Buku (buku)

				id	judul_buku	id_kategori	pengarang	penerbit	tahun_terbit	isbn	stok	dipinjam	dibooking	image
<input type="checkbox"/>				2	Hujan	11	Tere Liye	Gramedia Pustaka Utama	2019	9786020324784	2	0	2	img28446637.jpg
<input type="checkbox"/>				5	Belajar Coding Itu Penting di Era Revolusi Industr...	1	YENI MULYANI	Pustaka Baru Press	2020	9786023763672	6	1	0	img1705707313.jpg
<input type="checkbox"/>				14	Morfologi Bahasa Indonesia	2	Jusrin Efendi Pohan	Rineka Cipta	2019	9786232091825	5	1	-3	img12345.jpg
<input type="checkbox"/>				15	Buku Ajar Pengantar Ilmu Komunikasi	5	Didik Hariyanto	Pustaka Ilmu Group	2015	9786237066866	4	0	1	img1250.jpg
<input type="checkbox"/>				16	Buku Dasar-Dasar Teknik Informatika	1	Novega Pratama Adiputra	Deepublish	2020	9786230208195	5	0	0	img7654.jpg
<input type="checkbox"/>				17	20 Bisnis IT dari Hobi Menjadi Rezeki	4	Jubilee Enterprise	PT Elex Media Komputindo	2010	87968686787879	5	0	0	img1705711256.jpeg
<input type="checkbox"/>				18	Atomic Habits	8	James Clear	Penguin Books	2018	97868687978796	5	0	1	img34890.jpg
<input type="checkbox"/>				19	Bersandar pada Sains	3	Joko Priyono	Bhuana Ilmu Populer	2022	565756565768868	5	0	0	img2345.jpeg
<input type="checkbox"/>				20	One Piece 98	9	Eiichiro Oda	Elex Media Komputindo	2022	9786230030185	10	0	0	img1705757280.jpg

## Isi tabel buku

				id	judul_buku	id_kategori	pengarang	penerbit	tahun_terbit	isbn	stok	dipinjam	dibooking	image
<input type="checkbox"/>				2	Hujan	11	Tere Liye	Gramedia Pustaka Utama	2019	9786020324784	2	0	2	img28446637.jpg
<input type="checkbox"/>				5	Belajar Coding Itu Penting di Era Revolusi Industr...	1	YENI MULYANI	Pustaka Baru Press	2020	9786023763672	6	1	0	img1705707313.jpg
<input type="checkbox"/>				14	Morfologi Bahasa Indonesia	2	Jusrin Efendi Pohan	Rineka Cipta	2019	9786232091825	5	1	-3	img12345.jpg
<input type="checkbox"/>				15	Buku Ajar Pengantar Ilmu Komunikasi	5	Didik Hariyanto	Pustaka Ilmu Group	2015	9786237066866	4	0	1	img1250.jpg
<input type="checkbox"/>				16	Buku Dasar-Dasar Teknik Informatika	1	Novega Pratama Adiputra	Deepublish	2020	9786230208195	5	0	0	img7654.jpg
<input type="checkbox"/>				17	20 Bisnis IT dari Hobi Menjadi Rezeki	4	Jubilee Enterprise	PT Elex Media Komputindo	2010	87968686787879	5	0	0	img1705711256.jpeg
<input type="checkbox"/>				18	Atomic Habits	8	James Clear	Penguin Books	2018	97868687978796	5	0	1	img34890.jpg
<input type="checkbox"/>				19	Bersandar pada Sains	3	Joko Priyono	Bhuana Ilmu Populer	2022	565756565768868	5	0	0	img2345.jpeg
<input type="checkbox"/>				20	One Piece 98	9	Eiichiro Oda	Elex Media Komputindo	2022	9786230030185	10	0	0	img1705757280.jpg

e) Tabel temp

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	<b>id</b>	int(11)			Tidak	<i>Tidak ada</i>		AUTO_INCREMENT	Ubah  Hapus Lainnya
2	<b>tgl_booking</b>	datetime			Ya	<i>NULL</i>			Ubah  Hapus Lainnya
3	<b>id_user</b>	varchar(12)	latin1_swedish_ci		Ya	<i>NULL</i>			Ubah  Hapus Lainnya
4	<b>email_user</b>	varchar(128)	latin1_swedish_ci		Ya	<i>NULL</i>			Ubah  Hapus Lainnya
5	<b>id_buku</b>	int(11)			Ya	<i>NULL</i>			Ubah  Hapus Lainnya
6	<b>judul_buku</b>	varchar(255)	latin1_swedish_ci		Tidak	<i>Tidak ada</i>			Ubah  Hapus Lainnya
7	<b>image</b>	varchar(255)	latin1_swedish_ci		Tidak	<i>Tidak ada</i>			Ubah  Hapus Lainnya
8	<b>penulis</b>	varchar(128)	latin1_swedish_ci		Tidak	<i>Tidak ada</i>			Ubah  Hapus Lainnya
9	<b>penerbit</b>	varchar(128)	latin1_swedish_ci		Tidak	<i>Tidak ada</i>			Ubah  Hapus Lainnya
10	<b>tahun_terbit</b>	year(4)			Tidak	<i>Tidak ada</i>			Ubah  Hapus Lainnya

f) Tabel booking

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	<b>id_booking</b>	varchar(12)	latin1_swedish_ci		Tidak	<i>Tidak ada</i>			Ubah  Hapus Lainnya
2	<b>tgl_booking</b>	date			Tidak	<i>Tidak ada</i>			Ubah  Hapus Lainnya
3	<b>batas_ambil</b>	date			Tidak	<i>Tidak ada</i>			Ubah  Hapus Lainnya
4	<b>id_user</b>	int(11)			Tidak	<i>Tidak ada</i>			Ubah  Hapus Lainnya

g) Tabel booking\_detail

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	<b>id</b>	int(11)			Tidak	<i>Tidak ada</i>		AUTO_INCREMENT	Ubah  Hapus Lainnya
2	<b>id_booking</b>	varchar(12)	latin1_swedish_ci		Tidak	<i>Tidak ada</i>			Ubah  Hapus Lainnya
3	<b>id_buku</b>	int(11)			Tidak	<i>Tidak ada</i>			Ubah  Hapus Lainnya

## Isi Tabel booking\_detail

		id	id_booking	id_buku
<input type="checkbox"/>	Ubah  Salin  Hapus	5	20012024002	2

## h) Tabel pinjam

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	no_pinjam	varchar(12)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah  Hapus Lainnya
2	tgl_pinjam	date			Tidak	Tidak ada			Ubah  Hapus Lainnya
3	id_booking	varchar(12)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah  Hapus Lainnya
4	id_user	int(11)			Tidak	Tidak ada			Ubah  Hapus Lainnya
5	tgl_kembali	date			Tidak	Tidak ada			Ubah  Hapus Lainnya
6	tgl_pengembalian	date			Tidak	Tidak ada			Ubah  Hapus Lainnya
7	status	enum('Pinjam', 'Kembali')	utf8mb4_general_ci		Tidak	Pinjam			Ubah  Hapus Lainnya
8	total_denda	int(11)			Tidak	Tidak ada			Ubah  Hapus Lainnya

## Isi Tabel pinjam

		no_pinjam	tgl_pinjam	id_booking	id_user	tgl_kembali	tgl_pengembalian	status	total_denda
<input type="checkbox"/>	Ubah  Salin  Hapus	20012024004	2024-01-20	20012024001	4	2024-01-23	0000-00-00	Pinjam	0
<input type="checkbox"/>	Ubah  Salin  Hapus	21012024004	2024-01-21	20012024001	4	2024-01-24	2024-01-21	Kembali	0
<input type="checkbox"/>	Ubah  Salin  Hapus	22012024004	2024-01-22	22012024003	6	2024-01-25	0000-00-00	Pinjam	0
<input type="checkbox"/>	Ubah  Salin  Hapus	29052023001	2023-05-29	29052023001	4	2023-06-01	0000-00-00	Pinjam	0
<input type="checkbox"/>	Ubah  Salin  Hapus	29052023002	2023-05-29	29052023001	4	2023-06-01	2024-01-19	Kembali	0
<input type="checkbox"/>	Ubah  Salin  Hapus	29052023003	2023-05-29	29052023001	4	2023-06-01	2024-01-19	Kembali	0

## i) Tabel detail\_pinjam

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	no_pinjam	varchar(12)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah  Hapus Lainnya
2	id_buku	int(11)			Tidak	Tidak ada			Ubah  Hapus Lainnya
3	denda	int(11)			Tidak	Tidak ada			Ubah  Hapus Lainnya

Isi Tabel detail\_pinjam

no_pinjam	id_buku	denda
29052023002	14	5000
29052023003	5	5000
21012024004	16	5000

### 3) Membuat Koneksi Database Codeigniter dengan MySQL

Sebelum melakukan seting konfigurasi pada file database, terlebih dahulu harus disiapkan databasenya. Agar bisa langsung dipraktekkan. Selanjutnya untuk menghubungkan codeigniter dengan database buka file *config database* codeigniter yang terletak di **application/config/database.php** kemudian cari baris sintak seperti berikut:

```
database.php
application > config > database.php
1  <?php
2  defined('BASEPATH') OR exit('No direct script access allowed');
3  $active_group = 'default';
4  $query_builder = TRUE;
5
6  $db['default'] = array(
7      'dsn' => '',
8      'hostname' => 'localhost',
9      'username' => '',
10     'password' => '',
11     'database' => '',
12     'dbdriver' => 'mysqli',
13     'dbprefix' => '',
14     'pconnect' => FALSE,
15     'db_debug' => (ENVIRONMENT !== 'production'),
16     'cache_on' => TRUE,
17     'cachedir' => '',
18     'char_set' => 'utf8',
19     'dbcollat' => 'utf8_general_ci',
20     'swap_pre' => '',
21     'encrypt' => FALSE,
22     'compress' => FALSE,
23     'stricton' => FALSE,
24     'failover' => array(),
25     'save_queries' => TRUE
26 );
```

Kemudian masukkan username, password, dan nama database, untuk password jika tidak ada maka bisa dibiarkan kosong. Sehingga akan menjadi.

```

database.php
application > config > database.php
1  <?php
2  defined('BASEPATH') OR exit('No direct script access allowed');
3  $active_group = 'default';
4  $query_builder = TRUE;
5
6  $db['default'] = array(
7      'dsn' => '',
8      'hostname' => 'localhost',
9      'username' => 'root',
10     'password' => '',
11     'database' => 'pustaka3',
12     'dbdriver' => 'mysqli',
13     'dbprefix' => '',
14     'pconnect' => FALSE,
15     'db_debug' => (ENVIRONMENT !== 'production'),
16     'cache_on' => TRUE,
17     'cachedir' => '',
18     'char_set' => 'utf8',
19     'dbcollat' => 'utf8_general_ci',
20     'swap_pre' => '',
21     'encrypt' => FALSE,
22     'compress' => FALSE,
23     'stricton' => FALSE,
24     'failover' => array(),
25     'save_queries' => TRUE
26 );

```

#### 4) Membuat Model untuk CRUD

Pada tahap ini akan dibuat fungsi-fungsi yang bisa digunakan berulang kali, tanpa harus repot banyak-banyak membuat fungsi pada model. contohnya untuk insert data ke database, hanya buat satu fungsi untuk insert data ke database. Dan bisa kita gunakan secara berulang-ulang kali. Begitu juga dengan fungsi edit, hapus dan update data pada database.

Buat 2 buah file model dengan nama **ModelUser.php** dan **ModelBuku.php**. perlu diperhatikan untuk Penamaan Controller dan Model harus diawali dengan huruf besar di codeigniter 3. Memang tidak masalah jika dibuat dengan nama kecil, tetapi akan menyalahi aturan penggunaan codeigniter yang benar dan akan terjadi trouble saat diupload dihosting.



## ModelUser.php

```
<?php
defined('BASEPATH') or exit('No direct script access allowed');

class ModelUser extends CI_Model
{
    public function simpanData($data = null)
    {
        $this->db->insert('user', $data);
    }

    public function cekData($where = null)
    {
        return $this->db->get_where('user', $where);
    }

    public function getUserWhere($where = null)
    {
        return $this->db->get_where('user', $where);
    }

    public function cekUserAccess($where = null)
    {
        $this->db->select('*');
        $this->db->from('access_menu');
        $this->db->where($where);
        return $this->db->get();
    }

    public function getUserLimit()
    {
        $this->db->select('*');
        $this->db->from('user');
        $this->db->limit(10, 0);
        return $this->db->get();
    }
}
```

## ModelBuku.php

```
<?php
defined('BASEPATH') or exit('No direct script access allowed');

class ModelBuku extends CI_Model
{
    //manajemen buku
    public function getBuku()
    {
        return $this->db->get('buku');
    }

    public function bukuWhere($where)
    {
        return $this->db->get_where('buku', $where);
    }

    public function simpanBuku($data = null)
    {
        $this->db->insert('buku', $data);
    }

    public function updateBuku($data = null, $where = null)
    {
        $this->db->update('buku', $data, $where);
    }

    public function hapusBuku($where = null)
    {
        $this->db->delete('buku', $where);
    }

    public function total($field, $where)
    {
        $this->db->select_sum($field);
        if(!empty($where) && count($where) > 0){
            $this->db->where($where);
        }
    }
}
```

Dengan model seperti di atas, Untuk menginput data ke database kita bisa menggunakan function *simpan()*, untuk menampilkan data atau mengambil data dari database kita bisa menggunakan function *tampil()*, untuk menghapus data dari database kita bisa menggunakan function *hapus()*, dan untuk mengupdate data pada database kita bisa menggunakan function *ubah()*.

Kemudian kita harus mengatur autoload model di application/config/autoload.php. agar model-model yang sudah dibuat bisa digunakan dan di load secara otomatis. Jadi tidak perlu lagi kita load dari controller.

### Application/config/autoload.php

Cari baris sintak berikut:

```
$autoload['model'] = array();
```

Kemudian ubah menjadi

```
$autoload['model'] = array('ModelUser', 'ModelBuku');
```

## 5) Membuat Tampilan Katalog Buku dan Detail Buku

### A. Membuat Controller Home

Buatlah sebuah controller baru dengan nama **Home.php**. Dengan script sebagai berikut:

```
<?php

class Home extends CI_Controller
{
    function __construct()
    {
        parent::__construct();
    }

    public function index()
    {
        $data = [
            'judul' => "Katalog Buku",
            'buku' => $this->ModelBuku->getBuku()->result(),
        ];

        //jika sudah login dan jika belum login

        if ($this->session->userdata('email')) {
            $user = $this->ModelUser->cekData(['email' => $this->session->userdata('email')])->row_array();

            $data['user'] = $user['nama'];

            $this->load->view('templates/templates-user/header', $data);
            $this->load->view('buku/daftarbuku', $data);

            $this->load->view('templates/templates-user/footer', $data);
        } else {
            $data['user'] = 'Pengunjung';
            $this->load->view('templates/templates-user/header', $data);
            $this->load->view('buku/daftarbuku', $data);

            $this->load->view('templates/templates-user/footer', $data);
        }
    }
}
```

Pada controller Home dilakukan pemanggilan view header, daftarbuku, modal, dan footer. Oleh karena itu pastikan view-view tersebut telah dibuat sebelum memanggil controller Home. Pada controller ini juga dilakukan pengecekan terhadap user. Jika sudah login, maka akan muncul nama user pada menu header, dan jika belum login maka akan muncul kata pengunjung pada header.

## B. Membuat View untuk Template User

### 1. Membuat view header

Buat sebuah file dengan nama **header.php** pada folder **Application/views/templates/templates-user**. Tulis script berikut:

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Pustaka-Booking | <?= $judul; ?></title>
    <link rel="icon" type="image/png" href="<?= base_url('assets/img/logo/'); ?>logo-pb.png">
    <link rel="stylesheet" href="<?= base_url('assets/'); ?>user/css/bootstrap.css">
</head>
```

```

<link href="<?= base_url('assets/'); ?>vendor/fontawesome-
free/css/all.min.css" rel="stylesheet" type="text/css">
<link href="<?= base_url('assets/'); ?>datatable/datatables.css" rel="stylesh
eet" type="text/css">
</head>

<body>
    <nav class="navbar navbar-expand-lg navbar-light bg-light">
        <div class="container">
            <a class="navbar-brand" href="<?= base_url(); ?>">Pustaka</a>
            <button class="navbar-toggler" type="button" data-
toggle="collapse" data-target="#navbarNavAltMarkup" aria-
controls="navbarNavAltMarkup" aria-expanded="false" aria-
label="Toggle navigation">
                <span class="navbar-toggler-icon"></span>
            </button>

            <div class="collapse navbar-collapse" id="navbarNavAltMarkup">
                <div class="navbar-nav">
                    <a class="nav-item nav-
link active" href="<?= base_url(); ?>">Beranda <span class="sr-
only">(current)</span></a>
                    <?php
                        if (!empty($this->session->userdata('email'))){ ?>
                            <a class="nav-item nav-link" href="#">Booking Buku</a>

                            <a class="nav-item nav-
link" href="<?= base_url('member/myprofil'); ?>">Profil Saya</a>
                            <a class="nav-item nav-
link" href="<?= base_url('member/logout'); ?>"><i class="fas fw fa-
login"></i> Log out</a>

                            <?php } else { ?>

                                <a class="nav-item nav-link" data-toggle="modal" data-
target="#daftarModal" href="#"><i class="fas fw fa-login"></i> Daftar</a>

                                <a class="nav-item nav-link" data-toggle="modal" data-
target="#loginModal" href="#"><i class="fas fw fa-login"></i> Log in</a>

                                <?php } ?>
                                <span class="nav-item nav-link nav-
right" style="display:block; margin-
left:20px;">Selamat Datang <b><?= $user; ?></b></span>
                            </div>
                        </div>
                    </div>
                </nav>
                <div class="container mt-5">

```

## 2. Membuat view footer

Buat sebuah file dengan nama footer.php pada folder Application/views/templates/templates-user. Tulis script berikut:

```
</div>
<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965Dz00rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo" crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-U02eT0CphQ6hJty5KVphtPhzWj9WO1c1HTMga3JDZwrnQq4sF86dIHNDz0W1" crossorigin="anonymous"></script>

<script src="<?= base_url(); ?>assets/user/js/bootstrap.js"></script>

<script src="<?= base_url('assets/'); ?>vendor/jquery/jquery.min.js"></script>
<script src="<?= base_url('assets/'); ?>vendor/bootstrap/js/bootstrap.bundle.min.js"></script>

<!-- Core plugin JavaScript-->
<script src="<?= base_url('assets/'); ?>vendor/jquery-easing/jquery.easing.min.js"></script>

<!-- Custom scripts for all pages-->
<script src="<?= base_url('assets/'); ?>js/sb-admin-2.min.js"></script>
<script>
    $($('.alert').alert().delay(3000).slideUp('slow'));
</script>

</body>

</html>
```

## C. Membuat View daftarbuku

Buatlah sebuah file view baru dengan nama **daftarbuku.php** pada folder Application/views/buku. Ketikkan script berikut:

```

<?= $this->session->flashdata('pesan'); ?>

<div style="padding: 25px;">
    <div class="x_panel">

        <div class="x_content">

            <!-- Tampilkan semua produk -->
            <div class="row">
                <!-- looping products -->
                <?php foreach ($buku as $buku) { ?>
                    <div class="col-md-2 col-md-3">
                        <div class="thumbnail" style="height: 370px;">
                            

                                <div class="caption">
                                    <h5 style="min-height:30px;"><?= $buku-
>pengarang ?></h5>

                                    <h5><?= $buku->penerbit ?></h5>
                                    <h5><?= substr($buku-
>tahun_terbit, 0, 4) ?></h5>

                                    <p>
                                        <?php
                                        if ($buku->stok < 1) {
                                            echo "<i class='btn btn-outline-
primary fas fw fa-shopping-cart'> Booking&nbsp;&nbsp;  0</i>";
                                        } else {
                                            echo "<a class='btn btn-outline-
primary fas fw fa-shopping-
cart' href='\" . base_url('booking/tambahBooking/' . $buku-
>id) . \"'> Booking</a>";
                                        }
                                    }>

                                    <a class="btn btn-outline-
warning fas fw fa-search" href="<?= base_url('home/detailBuku/' . $buku-
>id); ?>"> Detail</a></p>
                                </div>
                            </div>
                        </div> <?php } ?>
                    <!-- end looping -->
                </div>
            </div>
        </div>
    </div>
</div>

```

#### D. Mengatur controller default

Pada file **routes.php**, ubah controller default menjadi home.

```

$route['default_controller'] = 'home';
$route['404_override'] = '';
$route['translate_uri_dashes'] = FALSE;

```

PERPUSDA Beranda Daftar Log in Selamat Datang **Pengunjung**

[illegible]



Tambahkan sebuah fungsi baru dengan nama detailBuku pada controller Home

```
public function detailBuku()
{
    $id = $this->uri->segment(3);
    $buku = $this->ModelBuku->joinKategoriBuku(['buku.id' => $id])->result();

    $data['user'] = "Pengunjung";
    $data['title'] = "Detail Buku";

    foreach ($buku as $fields) {
        $data['judul'] = $fields->judul_buku;
        $data['pengarang'] = $fields->pengarang;
        $data['penerbit'] = $fields->penerbit;
        $data['kategori'] = $fields->kategori;
        $data['tahun'] = $fields->tahun_terbit;
        $data['isbn'] = $fields->isbn;
        $data['gambar'] = $fields->image;
        $data['dipinjam'] = $fields->dipinjam;
        $data['dibooking'] = $fields->dibooking;
        $data['stok'] = $fields->stok;
        $data['id'] = $id;
    }
    $this->load->view('templates/templates-user/header', $data);
    $this->load->view('buku/detail-buku', $data);
    $this->load->view('templates/templates-user/footer');
}
```

Buka kembali **ModelUser.php** yang pernah dibuat, kemudian edit pada bagian fungsi **JoinKategoriBuku()**, dengan skrip sebagai berikut:

```
public function joinKategoriBuku($where)
{
    $this->db->from('buku');
    $this->db->join('kategori', 'kategori.id = buku.id_kategori');
    $this->db->where($where);
    return $this->db->get();
}
```

## 2. Membuat view detail-buku

Buat file detail-buku.php pada folder Application/views/buku untuk menampilkan halaman detail buku.

```
<div class="x_panel" align="center">

    <div class="x_content">
        <div class="row">
            <div class="col-sm-3 col-md-3">
                <div class="thumbnail" style="height: auto; position: relative; left: 100%; width: 200%;">
                    

                <div class="caption">
                    <h5 style="min-height:40px;" align="center"><?= $pengarang ?></h5>
                    <center>
                        <table class="table table-striped">
                            <tr>
                                <th nowrap>Judul Buku: </th>
                                <td nowrap><?= $judul; ?></td>
                                <td>&nbsp;</td>
                                <th>Kategori: </th>
                                <td><?= $kategori ?></td>
                            </tr>
                            <tr>
                                <th nowrap>Penerbit: </th>
                                <td><?= $penerbit ?></td>
                                <td>&nbsp;</td>
                                <th>Dipinjam: </th>
                                <td><?= $dipinjam ?></td>
```

```

        </tr>
        <tr>
            <th nowrap>Tahun Terbit: </th>
            <td><?= substr($tahun, 0, 4) ?></td>
            <td>&nbsp;</td>
            <th>Dibooking: </th>
            <td><?= $dibooking ?></td>
        </tr>
        <tr>
            <th>ISBN: </th>
            <td><?= $isbn ?></td>
            <td>&nbsp;</td>
            <th>Tersedia: </th>
            <td><?= $stok ?></td>
        </tr>
    </table>
</center>
<p>
    <a class="btn btn-outline-primary fas fw fa-shopping-
    cart" href="<?= base_url('booking/tambahBooking/' . $id); ?>"> Booking</a>
    <span class="btn btn-outline-secondary fas fw fa-
    reply" onclick="window.history.go(-1)"> Kembali</span>
</p>
</div>
</div>
</div>
</div>
</div>
</div>

```

Hasil Akhir:

PERPUSDA Beranda Daftar Log in Selamat DatangPengunjung



YENI MULYANI

<b>Judul Buku:</b>	Belajar Coding Itu Penting di Era Revolusi Industri 4.0	<b>Kategori:</b>	Komputer
<b>Penerbit:</b>	Pustaka Baru Press	<b>Dipinjam:</b>	2
<b>Tahun Terbit:</b>	2020	<b>Dibooking:</b>	0
<b>ISBN:</b>	9786023763672	<b>Tersedia:</b>	5

Booking

Kembali

## 6) Membuat Halaman Login dan Pendaftaran Anggota

Secara default jika sistem belum melakukan login maka menu pada header berisi **Daftar** dan **Login**. Sedangkan jika sudah melakukan login menu akan berubah menjadi **Booking**, **Profile saya** dan **Logout**. Pada pertemuan ini akan dibahas mengenai fungsi-fungsi tersebut. Untuk menu Booking akan dibahas dipertemuan berikutnya.

### Kondisi menu sebelum login

PERPUSDA Beranda Daftar Log in Selamat Datang **Pengunjung**

### kondisi menu setelah login

PERPUSDA Beranda Booking0 Buku Profil Saya Log out Selamat Datang **Raanni**

Untuk membuat halaman Login dan daftar, kita menggunakan modal. Pemanggilan modal ada pada halaman **header.php** yang telah kita buat.

```
<?php } else { ?>

<a class="nav-item nav-link" data-toggle="modal" data-target="#daftarModal" href="#"><i class="fas fw fa-login"></i> Daftar</a>

<a class="nav-item nav-link" data-toggle="modal" data-target="#loginModal" href="#"><i class="fas fw fa-login"></i> Log in</a>

<?php } ?>

<span class="nav-item nav-link nav-right" style="display:block; margin-left:20px;">Selamat Datang <b><? = $user; ?></b></span>

</div>
```

## A. Membuat Halaman Login dan Daftar Anggota

Buat Buat sebuah file dengan nama **modal.php** pada folder **Application/views/templates/templates-user**. view modal berfungsi untuk membuat tampilan halaman login dan daftar member. Ketikkan script berikut:

```
<!-- login modal -->
<div class="modal fade" tabindex="-1" id="loginModal" role="dialog">
  <div class="modal-dialog" role="document">
    <div class="modal-content">
      <div class="modal-header">
        <h5 class="modal-title">Login Member</h5>
        <button type="button" class="close" data-dismiss="modal" aria-label="Close">
          <span aria-hidden="true">&times;</span></button>
      </div>
      <form action="<? = base_url('member'); ?>" method="post">
```

```

        <div class="modal-body">
            <div class="form-group row">
                <label for="email" class="col-sm-2 col-form-
label">Email</label>
                <div class="col-sm-10">
                    <input type="text" class="form-
control" id="email" name="email" placeholder="Alamat Email">
                </div>
            </div>
            <div class="form-group row">
                <label for="inputPassword" class="col-sm-2 col-form-
label">Password</label>
                <div class="col-sm-10">
                    <input type="password" class="form-
control" id="password" name="password" placeholder="Password">
                </div>
            </div>
        </div>
        <div class="modal-footer">
            <button type="button" class="btn btn-outline-secondary" data-
dismiss="modal">Close</button>
            <button type="submit" class="btn btn-outline-
primary">Log in</button>
        </div>
    </form>
</div>
</div>
</div>
<!--/login modal -->

<!-- daftar modal -->
<div class="modal fade" tabindex="-1" id="daftarModal" role="dialog">
    <div class="modal-dialog" role="document">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title">Daftar Anggota</h5>
                <button type="button" class="close" data-dismiss="modal" aria-
label="Close">
                    <span aria-hidden="true">&times;</span>
                </button>
            </div>
            <form action="{<?= base_url('member/daftar'); ?>" method="post">
                <div class="modal-body">
                    <div class="form-group">

```

```

        <input type="text" class="form-control form-control-
user" id="nama" name="nama" placeholder="Nama Lengkap">
    </div>
    <div class="form-group">
        <input type="text" class="form-control form-control-
user" id="alamat" name="alamat" placeholder="Alamat Lengkap">
    </div>
    <div class="form-group">
        <input type="text" class="form-control form-control-
user" id="email" name="email" placeholder="Alamat Email">
    </div>
    <div class="form-group">
        <input type="password" class="form-control form-control-
user" id="password1" name="password1" placeholder="Password">
    </div>
    <div class="form-group">
        <input type="password" class="form-control form-control-
user" id="password2" name="password2" placeholder="Ulangi Password">
    </div>
</div>

    <div class="modal-footer">
        <button type="button" class="btn btn-outline-secondary" data-
dismiss="modal">Close</button>
        <button type="submit" class="btn btn-outline-
primary">Simpan</button>
    </div>
</form>
</div>
</div>
</div>
<!--/end of Modal Daftar -->

<!-- modal info-->
<div class="modal fade" tabindex="-1" id="modalinfo" role="dialog">
    <div class="modal-dialog" role="document">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title">Informasi</h5>
                <button type="button" class="close" data-dismiss="modal" aria-
label="Close">
                    <span aria-hidden="true">&times;</span>
                </button>
            </div>
            <div class="modal-body">

```

```

        <span class="alert alert-message alert-
success">Waktu Pengambilan Buku 1x24 jam dari Booking!!!</span>
    </div>
    <div class="modal-footer">
        <a class="btn btn-outline-info" href="{?= base_url(); ?}">Ok</a>
    </div>
</div>
</div>
<!--/modal info -->

```

The image shows two modal forms side-by-side. The 'Login Member' modal on the left has a title bar with a close button (X), a label 'Login Member', and two input fields: 'Email' (with placeholder 'Alamat Email') and 'Password' (with placeholder 'Password'). It has 'Close' and 'Log in' buttons at the bottom. The 'Daftar Anggota' modal on the right has a title bar with a close button (X), a label 'Daftar Anggota', and four input fields: 'Nama Lengkap', 'Alamat tempat Tinggal', 'Alamat Email', and 'Password' (with placeholder 'Ulangi Password'). It has 'Close' and 'Simpan' buttons at the bottom.

Untuk memanggil halaman login, sistem memanggil id **loginModal** yang ada pada file modal.php. sedangkan untuk memanggil halaman daftar sistem memanggil id **daftarModal**

```

<!-- login modal -->
<div class="modal fade" tabindex="-1" id="loginModal" role="dialog">
    <div class="modal-dialog" role="document">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title">Login Member</h5>

```

```

<!-- daftar modal -->
<div class="modal fade" tabindex="-1" id="daftarModal" role="dialog">
    <div class="modal-dialog" role="document">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title">Daftar Anggota</h5>

```

Untuk dapat memanggil view modal.php ini, kita akan lakukan pengeditan pada controller **Home.php** (fungsi **index** dan **detailBuku**) yang telah kita buat sebelumnya, dengan menambahkan pemanggilan file modal.php tersebut.

```

application > controllers > Home.php
16     ];
17     //jika sudah login dan jika belum login
18     if ($this->session->userdata('email')) {
19         $user = $this->ModelUser->cekData(['email' => $this->session->
20
21         $data['user'] = $user['nama'];
22
23         $this->load->view('templates/templates-user/header', $data);
24         $this->load->view('buku/daftarbuku', $data);
25         $this->load->view('templates/templates-user/modal');
26         $this->load->view('templates/templates-user/footer', $data);
27     } else {
28         $data['user'] = 'Pengunjung';
29         $this->load->view('templates/templates-user/header', $data);
30         $this->load->view('buku/daftarbuku', $data);
31         $this->load->view('templates/templates-user/modal');
32         $this->load->view('templates/templates-user/footer', $data);
33     }
34 }

    $data['gambar'] = $fields->image;
    $data['dipinjam'] = $fields->dipinjam;
    $data['dibooking'] = $fields->dibooking;
    $data['stok'] = $fields->stok;
    $data['id'] = $id;
}

    $this->load->view('templates/templates-user/header', $data);
    $this->load->view('buku/detail-buku', $data);
    $this->load->view('templates/templates-user/modal');
    $this->load->view('templates/templates-user/footer');
}

```

## B. Membuat Controller Member

Controller member berisi fungsi-fungsi seperti pengecekan data login, pendaftaran anggota, management profile, dan fungsi logout. Buatlah sebuah controller dengan nama **Member.php**

### 1. Membuat fungsi login

Buatlah fungsi `_login` pada controller Member yang berisi coding untuk melakukan pengecekan login. Sistem akan mengarahkan ke halaman home setelah proses login dilakukan.



```

<?php

class Member extends CI_Controller
{
    function __construct()
    {
        parent::__construct();
    }

    public function index()
    {
        $this->_login();
    }

    private function _login()
    {
        $email = htmlspecialchars($this->input->post('email', true));
        $password = $this->input->post('password', true);

        $user = $this->ModelUser->cekData(['email' => $email])->row_array();

        //jika usernya ada
        if ($user) {
            //jika user sudah aktif
            if ($user['is_active'] == 1) {
                //cek password
                if (password_verify($password, $user['password'])) {
                    $data = [
                        'email' => $user['email'],
                        'role_id' => $user['role_id'],
                        'id_user' => $user['id'],
                        'nama' => $user['nama']
                    ];

                    $this->session->set_userdata($data);
                    redirect('home');
                } else {
                    $this->session->set_flashdata('pesan', '<div class="alert alert-danger alert-message" role="alert">Password salah!!</div>');
                    redirect('home');
                }
            } else {
                $this->session->set_flashdata('pesan', '<div class="alert alert-danger alert-message" role="alert">User belum diaktifasi!!</div>');
                redirect('home');
            }
        } else {
            $this->session->set_flashdata('pesan', '<div class="alert alert-danger alert-message" role="alert">Email tidak terdaftar!!</div>');
            redirect('home');
        }
    }
}

```

## 2. Membuat fungsi daftar

Fungsi daftar member berisi script tentang proses penyimpanan data pendaftar. Silakan tambahkan fungsi dengan nama **daftar** pada controller **Member**, dan ketikkan coding berikut:

```
public function daftar()
{
    $this->form_validation->set_rules('nama', 'Nama Lengkap', 'required', [
        'required' => 'Nama Belum diis!!'
    ]);
    $this->form_validation->
    set_rules('alamat', 'Alamat Lengkap', 'required', [
        'required' => 'Alamat Belum diis!!'
    ]);
    $this->form_validation->
    set_rules('email', 'Alamat Email', 'required|trim|valid_email|is_unique[user.email]', [
        'valid_email' => 'Email Tidak Benar!!',
        'required' => 'Email Belum diisi!!',
        'is_unique' => 'Email Sudah Terdaftar!'
    ]);
    $this->form_validation->
    set_rules('password1', 'Password', 'required|trim|min_length[3]|matches[password2]', [
        'matches' => 'Password Tidak Sama!!',
        'min_length' => 'Password Terlalu Pendek'
    ]);
    $this->form_validation->
    set_rules('password2', 'Repeat Password', 'required|trim|matches[password1]');

    $email = $this->input->post('email', true);
    $data = [
        'nama' => htmlspecialchars($this->input->post('nama', true)),
        'alamat' => $this->input->post('alamat', true),
        'email' => htmlspecialchars($email),
        'image' => 'default.jpg',
        'password' => password_hash($this->input->
    post('password1'), PASSWORD_DEFAULT),
        'role_id' => 2,
        'is_active' => 1,
        'tanggal_input' => time()
    ];
    $this->ModelUser->simpanData($data);
    $this->session->set_flashdata('pesan', '<div class="alert alert-
success alert-
message" role="alert">Selamat!! akun anggota anda sudah dibuat.</div>');
    redirect(base_url());
}
```

### 3. Membuat fungsi myProfil

Fungsi myProfil digunakan untuk menampilkan data profil anggota yang sedang aktif/login. Dengan mengirimkan data ke dalam view index. Berikut scrip fungsi myProfil

```
public function myProfil()
{
    $data['judul'] = 'Profil Saya';
    $user = $this->ModelUser->cekData(['email' => $this->session->userdata('email')])>row_array();

    foreach ($user as $a) {
        $data = [
            'image' => $user['image'],

            'user' => $user['nama'],
            'email' => $user['email'],
            'tanggal_input' => $user['tanggal_input'],
        ];
    }

    $this->load->view('templates/templates-user/header', $data);
    $this->load->view('member/index', $data);
    $this->load->view('templates/templates-user/modal');
    $this->load->view('templates/templates-user/footer', $data);
}
```

Pada script diatas, sistem memanggil view **index** yang ada di folder member. Oleh karena itu kita akan membuat sebuah file **index.php** di folder **Application/views/member**. Isi dari script index sebagai berikut:

```
<!-- Begin Page Content -->
<div class="container-fluid">

    <div class="row">
        <div class="col-lg-6 justify-content-x">
            <?=$this->session->flashdata('pesan'); ?>
        </div>
    </div>

    <div class="card mb-3" style="max-width: 540px;">
        <div class="row no-gutters">
            <div class="col-md-4">
                
            </div>
            <div class="col-md-8">
                <div class="card-body">
                    <h5 class="card-title"><?=$user; ?></h5>
                    <p class="card-text"><?=$email; ?></p>
                    <p class="card-text"><small class="text-muted">Jadi member sejak: <br><b><?=$date('d F Y', $tanggal_input); ?></b></small>
                </p>
                </div>
                <div class="btn btn-info ml-3 my-3">
                    <a href="<?=$base_url('member/ubahprofil'); ?>" class="text text-white"><i class="fas fa-user-edit"></i> Ubah Profil</a>
                </div>
            </div>
        </div>
    </div>
</div>
```

```

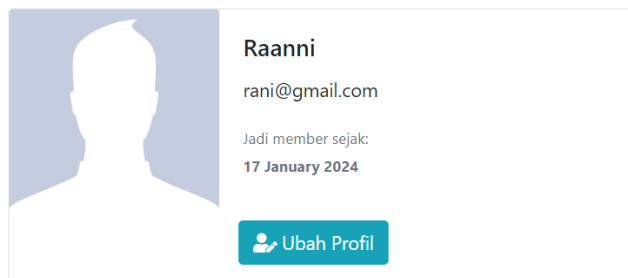
</div>
</div>
<!-- /.container-fluid -->

</div>
<!-- End of Main Content -->

```

Hasil dari Script tersebut adalah :

PERPUSDA Beranda Booking0 Buku Profil Saya Log out Selamat DatangRaanni



#### 4. Membuat fungsi ubahProfil

Fungsi ubahProfil digunakan untuk melakukan perubahan terhadap data member. Berikut script program fungsi ubahProfil yang ada pada controller Member

```

public function ubahProfil()
{
    $data['judul'] = 'Profil Saya';
    $user = $this->ModelUser->cekData(['email' => $this->session->userdata('email')])>row_array();

    foreach ($user as $a) {
        $data = [
            'image' => $user['image'],
            'user' => $user['nama'],
            'email' => $user['email'],
            'tanggal_input' => $user['tanggal_input'],
        ];
    }
}

```

```

        $this->form_validation->
set_rules('nama', 'Nama Lengkap', 'required|trim', [
    'required' => 'Nama tidak Boleh Kosong'
]);

if ($this->form_validation->run() == false) {
    $this->load->view('templates/templates-user/header', $data);
    $this->load->view('member/ubah-anggota', $data);
    $this->load->view('templates/templates-user/modal');
    $this->load->view('templates/templates-user/footer', $data);
} else {
    $nama = $this->input->post('nama', true);
    $email = $this->input->post('email', true);

    //jika ada gambar yang akan diupload
    $upload_image = $_FILES['image']['name'];

    if ($upload_image) {
        $config['upload_path'] = './assets/img/profile/';
        $config['allowed_types'] = 'gif|jpg|png';
        $config['max_size'] = '3000';
        $config['max_width'] = '1024';
        $config['max_height'] = '1000';
        $config['file_name'] = 'pro' . time();

        $this->load->library('upload', $config);

        if ($this->upload->do_upload('image')) {
            $gambar_lama = $data['user']['image'];
            if ($gambar_lama != 'default.jpg') {
                unlink(FCPATH . 'assets/img/profile/' . $gambar_lama);
            }

            $gambar_baru = $this->upload->data('file_name');
            $this->db->set('image', $gambar_baru);
        } else {
        }
    }

    $this->db->set('nama', $nama);
    $this->db->where('email', $email);
    $this->db->update('user');

    $this->session->set_flashdata('pesan', '<div class="alert alert-
success alert-message" role="alert">Profil Berhasil diubah </div>');
    redirect('member/myprofil');
}
}

```

Pada saat tombol ubah profil diklik, sistem akan mengarahkan ke view ubah-anggota.php, dimana file view tersebut menampilkan data profil anggota untuk diedit. Adapun data yang bisa

diedit hanya nama anggota dan gambar. Berikut script **ubah-anggota.php** yang ditulis di folder **Application/views/member**.

```
<!-- Begin Page Content -->
<div class="container-fluid">

    <div class="row">
        <div class="col-lg-9">
            <?= form_open_multipart('member/ubahprofil'); ?>
            <div class="form-group row">
                <label for="email" class="col-sm-2 col-form-label">Email</label>
                <div class="col-sm-10">
                    <input type="text" class="form-
control" id="email" name="email" value="<?= $email; ?>" readonly>
                </div>
            </div>
            <div class="form-group row">
                <label for="nama" class="col-sm-2 col-form-
label">Nama Lengkap</label>
                <div class="col-sm-10">
                    <input type="text" class="form-
control" id="nama" name="nama" value="<?= $user; ?>">
                    <?= form_error('nama', '<small class="text-danger pl-
3">', '</small>'); ?>
                </div>
            </div>
            <div class="form-group row">
                <div class="col-sm-2">Gambar</div>
                <div class="col-sm-10">
                    <div class="row">
                        <div class="col-sm-3">
                            
                        </div>
                        <div class="col-sm-9">
```



## 5. Membuat fungsi Logout

Berikut script untuk fungsi logout, script berada di controller **Member.php**

```
public function logout()
{
    $this->session->unset_userdata('email');
    $this->session->unset_userdata('role_id');

    $this->session->set_flashdata('pesan', '<div class="alert alert-
success alert-message" role="alert">Anda telah logout!!</div>');
    redirect('home');
}
```

## 6. Merubah Pustaka Helper

Buka kembali pustaka helper yang pernah dibuat (pustaka\_helper.php). edit file tersebut seperti script berikut:

```
<?php

function cek_login()
{
    $ci = get_instance();

    if (!$ci->session->userdata('email')) {
        $ci->session->set_flashdata('pesan', '<div class="alert alert-
danger" role="alert">Akses ditolak. Anda belum login!! </div>');
        if ($ci->session->userdata('role_id') == 1) {
            redirect('autentifikasi');
        } else {
            redirect('home');
        }
    } else {
        $role_id = $ci->session->userdata('role_id');
        $id_user = $ci->session->userdata('id_user');
    }
}

function cek_user()
{
    $ci = get_instance();

    $role_id = $ci->session->userdata('role_id');
    if ($role_id != 1) {
        $ci->session->set_flashdata('pesan', '<div class="alert alert-
danger" role="alert">Akses tidak diizinkan </div>');
        redirect('home');
    }
}
```

## 7) Membuat Halaman Transaksi booking peminjaman

### A. Membuat Model Booking

Model booking berisi script mengenai fungsi-fungsi pengaksesan database. Seperti perintah simpan, edit, hapus, update, dll. Pada model booking, perintah hanya untuk proses booking buku.



```

<?php if (!defined('BASEPATH')) exit('No direct script access allowed');

class ModelBooking extends CI_Model
{
    public function getData($table)
    {
        return $this->db->get($table)->row();
    }

    public function getDataWhere($table, $where)
    {
        $this->db->where($where); return
        $this->db->get($table);
    }

    public function getOrderById($table, $order, $limit)
    {
        $this->db->order_by($order, 'desc');
        $this->db->limit($limit); return
        $this->db->get($table);
    }

    public function joinOrder($where)

```

Buatlah sebuah model baru dengan nama **ModelBooking.php** lalu ketikkan perintah berikut:

```

{
    $this->db->select('*');
    $this->db->from('booking bo');
    $this->db->join('booking_detail d', 'd.id_booking=bo.id_booking');
    $this->db->join('buku bu ', 'bu.id=d.id_buku');
    $this->db->where($where);

    return $this->db->get();
}

public function simpanDetail($where = null)
{
    $sql = "INSERT INTO booking_detail (id_booking,id_buku) SELECT booking.id
_booking,temp.id_buku FROM booking, temp WHERE temp.id_user=booking.id_user AND b
ooking.id_user='$where'";
    $this->db->query($sql);
}

public function insertData($table, $data)
{
    $this->db->insert($table, $data);
}

public function updateData($table, $data, $where)
{
    $this->db->update($table, $data, $where);
}

public function deleteData($where, $table)
{
    $this->db->where($where);
    $this->db->delete($table);
}

public function find($where)
{
    //Query mencari record berdasarkan ID-nya
    $this->db->limit(1);
    return $this->db->get('buku', $where);
}

```

```

    public function createTemp()
    {
        $this->db->
>query('CREATE TABLE IF NOT EXISTS temp(id_booking varchar(12), tgl_booking DATETIME, email_user varchar(128), id_buku int)');
    }

    public function selectJoin()
    {
        $this->db->select('*');
        $this->db->from('booking');
        $this->db->
>join('booking_detail', 'booking_detail.id_booking=booking.id_booking');
        $this->db->join('buku', 'booking_detail.id_buku=buku.id');return
        $this->db->get();
    }

    public function showtemp($where)
    {
        return $this->db->get('temp', $where);
    }

    public function kodeOtomatis($tabel, $key)
    {
        $this->db->select('right(' . $key . ',3) as kode', false);
        $this->db->order_by($key, 'desc');
        $this->db->limit(1);
        $query = $this->db->get($tabel);if
        ($query->num_rows() <> 0) {
            $data = $query->row();
            $kode = intval($data->kode) + 1;
        } else {
            $kode = 1;
        }

        $kodemax = str_pad($kode, 3, "0", STR_PAD_LEFT);
        $kodejadi = date('dmY') . $kodemax;

        return $kodejadi;
    }

```

## B. Membuat Model Pinjam

Model pinjam pada dasarnya memiliki fungsi yang sama seperti model booking. Yang berbeda hanya tabel yang terlibat didalamnya. Model pinjam khusus untuk menangani proses yang berhubungan dengan tabel pinjam.

Buatlah sebuah model baru dengan nama **ModelPinjam.php** lalu ketikkan perintah berikut:

```
<?php if (!defined('BASEPATH')) exit('No Direct Script Access Allowed');

class ModelPinjam extends CI_Model
{
    //manip table pinjam
    public function simpanPinjam($data)
    {
        $this->db->insert('pinjam', $data);
    }

    public function selectData($table, $where)
    {
        return $this->db->get($table, $where);
    }

    public function updateData($data, $where)
    {
        $this->db->update('pinjam', $data, $where);
    }

    public function deleteData($tabel, $where)
    {
        $this->db->delete($tabel, $where);
    }

    public function joinData()
    {
        $this->db->select('*');
        $this->db->from('pinjam');
        $this->db->join('detail_pinjam', 'detail_pinjam.no_pinjam=pinjam.no_pinjam', 'Right');

        return $this->db->get()->result_array();
    }
}
```

```

}

//manip tabel detai pinjam
public function simpanDetail($idbooking, $nopinjam)
{
    $sql = "INSERT INTO detail_pinjam (no_pinjam,id_buku) SELECT pinjam.no_pi
njam,booking_detail.id_buku FROM pinjam, booking_detail WHERE booking_detail.id_b
ooking=$idbooking AND pinjam.no_pinjam='$nopinjam'";
    $this->db->query($sql);
}
}

```

Setelah kita membuat ModelBooking.php dan ModelPinjam.php, selanjutnya edit file Autoload.php bagian model, dengan menambahkan ModelBooking dan ModelPinjam.

```

application > config > autoload.php
133 | $autoload['model'] = array('first_model' => 'first');|
134 */
135 $autoload['model'] = array('ModelUser', 'ModelBuku', 'ModelBooking', 'ModelPinjam');
136

```

Lakukan pengeditan pada file header.php, pada bagian menu booking buku. Seperti pada script berikut:

```

application > views > templates > templates-user > header.php
25 <a class="nav-item nav-link active" href="= base_url(); ?/beranda
26 <span class="sr-only">(current)</span></a>
27 <?php
28 if (empty($this->session->userdata('email'))) { ?>
29 <a class="nav-item nav-link" href="= base_url('booking'); ?
30 &gt;"&gt;Booking &lt;b&gt;<?= $this-&gt;ModelBooking-&gt;getDataWhere('temp',
31 ['email_user' =&gt; $this-&gt;session-&gt;userdata('email')])-&gt;num_rows(); ?
32 &gt;&lt;/b&gt; Buku&lt;/a&gt;
33
34 &lt;a class="nav-item nav-link" href="<?= base_url('member/myprofil');
35 ?&gt;"&gt;Profil Saya&lt;/a&gt;
36 &lt;a class="nav-item nav-link" href="<?= base_url('member/logout'); ?
37 &gt;"&gt;&lt;i class="fas fa-fw fa-logout"&gt;&lt;/i&gt; Log out&lt;/a&gt;
38
39 &lt;?php } else { ?&gt;
</pre

```

Sehingga ketika tampil halaman home, maka data buku yang diboeking akan muncul seperti berikut:

Pustaka Beranda **Booking 0 Buku** Profil Saya Log out Selamat Datang Maruloh

### C. Membuat Controller Booking

Controller Booking.php berisi fungsi-fungsi yang berhubungan dengan proses booking buku. Didalamnya terdapat fungsi tambahBuku, hapusbooking, bookingSelesai, info, dan exportToPdf. Pada pertemuan ini akan kita buat fungsi index, tambahBuku, viewDataBooking, dan hapusbooking.

Buatlah sebuah controller baru dengan nama **Booking.php**. ketikkan pada controller script berikut:

```

<?php
defined('BASEPATH') or exit('No Direct Script Access Allowed');
date_default_timezone_set('Asia/Jakarta');

class Booking extends CI_Controller
{
    public function __construct()
    {
        parent::__construct();
        cek_login();
        $this->load->model(['ModelBooking', 'ModelUser']);
    }

    public function index()
    {
        $id = ['bo.id_user' => $this->uri->segment(3)];
        $id_user = $this->session->userdata('id_user');
        $data['booking'] = $this->ModelBooking->joinOrder($id)->result();

        $user = $this->ModelUser->cekData(['email' => $this->session->userdata('email')])->row_array();
        foreach ($user as $a) {
            $data = [
                'image' => $user['image'],
                'user' => $user['nama'],
                'email' => $user['email'],
                'tanggal_input' => $user['tanggal_input']
            ];
        }
        $dtb = $this->ModelBooking->showtemp(['id_user' => $id_user])->num_rows();

        if ($dtb < 1) {

```

```

        $this->session->set_flashdata('pesan', '<div class="alert alert-
massege alert-danger" role="alert">Tidak Ada Buku dikeranjang</div>');
        redirect(base_url());
    } else {
        $data['temp'] = $this->db-
>query("select image, judul_buku, penulis, penerbit, tahun_terbit,id_buku from te
mp where id_user='$id_user'")->result_array();
    }
    $data['judul'] = "Data Booking";

    $this->load->view('templates/templates-user/header', $data);
    $this->load->view('booking/data-booking', $data);
    $this->load->view('templates/templates-user/modal');
    $this->load->view('templates/templates-user/footer');
}

```

Fungsi index pada controller Booking adalah menampilkan data booking yang telah dilakukan oleh member.

```

$this->load->view('booking/data-booking', $data);

```

Jika sistem memeriksa tidak ditemukan adanya buku yang dipinjam, maka sistem melakukan feedback berupa pesan.

```

if ($dtb < 1) {
    $this->session->set_flashdata('pesan', '<div class="alert alert-
massege alert-danger" role="alert">Tidak Ada Buku dikeranjang</div>');
    redirect(base_url());
}

```

## 1. Membuat fungsi tambahBooking

Fungsi tambahBooking berisi script proses penyimpanan data booking buku. Dimana penyimpanan pada proses ini dilakukan di table **temp**, yang melibatkan table buku sebagai objek yang dipinjam.



Pada controller **Booking.php**, Tambahkan sebuah fungsi baru dengan nama **tambahBooking**, dan ketikkan script berikut:

```
public function tambahBooking()
{
    $id_buku = $this->uri->segment(3);
```

```

//memilih data buku yang untuk dimasukkan ke tabel temp/keranjang melalui variabel
$isi

$d = $this->db->query("Select*from buku where id='$id_buku'")->row();

//berupa data2 yang akan disimpan ke dalam tabel temp/keranjang
$isi = [
    'id_buku' => $id_buku, 'judul_buku' =>
    $d->judul_buku,
    'id_user' => $this->session->userdata('id_user'), 'email_user'
    => $this->session->userdata('email'), 'tgl_booking' =>
    date('Y-m-d H:i:s'),
    'image' => $d->image, 'penulis'
    => $d->pengarang, 'penerbit' =>
    $d->penerbit,
    'tahun_terbit' => $d->tahun_terbit
];

//cek apakah buku yang di klik booking sudah ada di keranjang
$temp = $this->ModelBooking-
>getDataWhere('temp', ['id_buku' => $id_buku])->num_rows();

$userid = $this->session->userdata('id_user');

//cek jika sudah memasukan 3 buku untuk dibooking dalam keranjang
$tempuser = $this->db-
>query("select*from temp where id_user='$userid'")->num_rows();

//cek jika masih ada booking buku yang belum diambil
$databooking = $this->db-
>query("select*from booking where id_user='$userid'")->num_rows();if
($databooking > 0) {
    $this->session->set_flashdata('pesan', '<div class="alert alert-danger alert-
message" role="alert">Masih Ada booking buku sebelumnya yang belum diambil.<br> Abmil
Buku yang dibooking atau tunggu 1x24 Jam untuk bisa booking kembali </div>');
;
    redirect(base_url());
}

//jika buku yang diklik booking sudah ada di keranjangif ($temp
> 0)

```

```

    }

    //jika buku yang akan diboeking sudah mencapai 3 item
    if ($tempuser == 3) {
        $this->session->set_flashdata('pesan', '<div class="alert alert-
danger alert-message" role="alert">Booking Buku Tidak Boleh Lebih dari 3</div>');
        redirect(base_url() . 'home');
    }

    //membuat tabel temp jika belum ada
    $this->ModelBooking->createTemp();
    $this->ModelBooking->insertData('temp', $isi);

    //pesan ketika berhasil memasukkan buku ke keranjang
    $this->session->set_flashdata('pesan', '<div class="alert alert-
success alert-
message" role="alert">Buku berhasil ditambahkan ke keranjang </div>');
    redirect(base_url() . 'home');
}

```

Pada fungsi tambahBooking, dilakukan pengecekan jumlah buku yang telah diboeking. Jika buku yang diboeking sudah 3, maka member tidak bias melakukan proses booking kembali

```

if ($tempuser == 3) {
    $this->session->set_flashdata('pesan', '<div class="alert alert-
danger alert-message" role="alert">Booking Buku Tidak Boleh Lebih dari 3</div>');
    redirect(base_url() . 'home');
}

```

System akan membuat table temp jika pada database belum ditemukan table temp, lalu menyimpan data kedalam table tersebut

```

$this->ModelBooking->createTemp();
$this->ModelBooking->insertData('temp', $isi);

```

Hasil dari fungsi tambahBooking

Pustaka Beranda **Booking 1 Buku** Profil Saya Log out Selamat Datang Maruloh, M.Kom

Saat menu booking diklik, akan menampilkan data booking seperti berikut:

Pustaka	Beranda	Booking 1 Buku	Profil Saya	Log out	Selamat Datang <b>Maruloh, M.Kom</b>
No.	Buku	Penulis	penerbit	Tahun	Pilihan
1		Fintri Indriyani, Yunita, Dinda AM, Artika S, Sriyadi	Graha Ilmu	2020	

[▶ Lanjutkan Booking Buku](#) [■ Selesaikan Booking](#)

Untuk membuat halaman tersebut, buatlah sebuah view baru dengan nama **data-booking.php**, letakkan pada folder **Application/views/booking**.

Berikut script **data-booking.php**

```

<div class="container">
  <center>
    <table>
      <tr>
        <td>
          <div class="table-responsive full-width">
            <table class="table table-bordered table-striped table-
hover" id="table-datatable">
              <tr>
                <th>No.</th>
                <th>Buku</th>
                <th>Penulis</th>
                <th>penerbit</th>
                <th>Tahun</th>
                <th>Pilihan</th>
              </tr>
              <?php
                $no = 1;
                foreach ($temp as $t) {
                  ?>
                    <tr>
                      <td><?= $no; ?></td>
                      <td>
                        
                      </td>

```

```
 <?= $t['penulis']; ?></td>  <?= $t['penerbit']; ?></td>  <?= substr($t['tahun_terbit'], 0, 4); ?></td>    | | | </td> </tr> <tr>   | | | | | |
```

Pada view data-booking, terdapat 2 tombol. **Lanjutkan Booking Buku** dan **Selesaikan Booking**. Untuk tombol Lanjutkan Booking Buku, system akan mengarahkan ke controller awal (Home.php) untuk melanjutkan booking buku

```

<a class="btn btn-sm btn-outline-
primary" href="<?php echo base_url(); ?>"><span class="fas fw fa-
play"></span> Lanjutkan Booking Buku</a>

```

Sedangkan tombol Selesaikan Booking, system mengarahkan ke fungsi bookingSelesai yang ada di controller Booking.

```
<a class="btn btn-sm btn-outline-success" href="php echo base_url() . 'booking/bookingSelesai/' . $this-&gt;session-&gt;userdata('id_user'); ?"><span class="fas fw fa-stop"></span> Selesaikan Booking</a>
```

## 2. Membuat fungsi hapusbooking

function ini akan di jalankan ketika member melakukan interaksi pada data booking dengan mengklik tombol hapus.

Pustaka	Beranda	Booking 1 Buku	Profil Saya	Log out	Selamat Datang Maruloh, M.Kom
No.	Buku	Penulis	penerbit	Tahun	Pilihan
1		Fintri Indriyani, Yunita, Dinda AM, Artika S, Sriyadi	Graha Ilmu	2020	
<div><a href="#">▶ Lanjutkan Booking Buku</a> <a href="#">■ Selesaikan Booking</a></div>					

Dalam fungsi ini terdapat script untuk menghapus data booking buku yang ada pada tabel temp. buatlah sebuah fungsi baru pada controller Booking.php. ketikkan script berikut:

```

public function hapusbooking()
{
    $id_buku = $this->uri->segment(3);
    $id_user = $this->session->userdata('id_user');

    $this->ModelBooking->deleteData(['id_buku' => $id_buku], 'temp');
    $kosong = $this->db->query("select*from temp where id_user='$id_user'")-
>num_rows();

    if ($kosong < 1) {
        $this->session->set_flashdata('pesan', '<div class="alert alert-
massege alert-danger" role="alert">Tidak Ada Buku dikeranjang</div>');
        redirect(base_url());
    } else {
        redirect(base_url() . 'booking');
    }
}

```



Pada script tersebut terdapat perintah untuk menghapus data booking yang terdapat pada tabel temp berdasarkan id buku yang dipilih. Dan menampilkan kembali data terbaru (setelah dihapus). Jika data terbaru ternyata buku sudah tidak ada lagi (kosong) maka sistem akan menampilkan pesan. Setelah proses hapus booking, sistem secara otomatis mengarahkan ke controller Booking.php

```
$this->ModelBooking->deleteData(['id_buku' => $id_buku], 'temp');
    $kosong = $this->db->query("select * from temp where id_user='$id_user'")->num_rows();

    if ($kosong < 1) {
        $this->session->set_flashdata('pesan', '<div class="alert alert-massege alert-danger" role="alert">Tidak Ada Buku dikeranjang</div>');
        redirect(base_url());
    } else {
        redirect(base_url() . 'booking');
    }
}
```



Hasil setelah proses hapus booking, booking buku kembali kosong

Pustaka Beranda **Booking 0 Buku** Profil Saya Log out Selamat Datang Maruloh, M.Kom

## 8) Membuat Halaman Lihat Keranjang Peminjaman dan Selesai Booking

### A. Membuat Fungsi Selesai Booking

Pada tahap ini. untuk menyelesaikan booking buku, maka member harus melakukan interaksi pada web dengan melakukan klik pada tombol **selesaikan booking** yang ada pada tampilan keranjang.

No.	Buku	Penulis	penerbit	Tahun	Pilihan
1		Fintri Indriyani, Yunita, Dinda AM, Artika S, Sriyadi	Graha Ilmu	2020	



Agar tombol selesaikan booking bisa berjalan, maka kita akan menambahkan function baru pada file controller **Booking.php** yaitu function **bookingSelesai**. Berikut kode program fungsi bookingSelesai

```

public function bookingSelesai($where)
{
    //mengupdate stok dan dibooking di tabel buku saat proses booking diselesaikan
    $this->db->query("UPDATE buku, temp SET buku.dibooking=buku.dibooking+1, buku.stok=buku.stok-1 WHERE buku.id=temp.id_buku");

    $tglsekarang = date('Y-m-d');
    $isibooking = [
        'id_booking' => $this->ModelBooking->kodeOtomatis('booking', 'id_booking'),
        'tgl_booking' => date('Y-m-d H:m:s'),
        'batas_ambil' => date('Y-m-d', strtotime('+2 days', strtotime($tglsekarang))),
        'id_user' => $where
    ];
    //menyimpan ke tabel booking dan detail booking, dan mengosongkan tabel temporeri
    $this->ModelBooking->insertData('booking', $isibooking);
    $this->ModelBooking->simpanDetail($where);
    $this->ModelBooking->kosongkanData('temp');

    redirect(base_url() . 'booking/info');
}

```

Pada fungsi bookingSelesai terdapat beberapa perintah yang dijalankan, diantaranya pengupdate-an stok dan info buku yang diboeking.

```
$this->db->query("UPDATE buku, temp SET buku.dibooking=buku.dibooking+1, buku.stok=buku.stok-1 WHERE buku.id=temp.id_buku");
```

Terdapat juga fungsi untuk melakukan proses penyimpanan pada tabel booking, detail booking, dan penghapusan data pada tabel temp. sistem akan secara otomatis mengarahkan ke fungsi info pada controller Booking.php

```
$this->ModelBooking->insertData('booking', $isibooking);  
$this->ModelBooking->simpanDetail($where);  
$this->ModelBooking->kosongkanData('temp');  
  
redirect(base_url() . 'booking/info');
```

## **B. Membuat Fungsi Info**

Fungsi info dipanggil saat sistem mengeksekusi fungsi booking selesai. Pada fungsi ini dilakukan pemanggilan terhadap view info-booking. Dimana info booking ini akan menampilkan data akhir dari proses booking buku. Buatlah fungsi baru dengan nama **info** didalam controller Booking.php dan ketikkan script berikut:

```

public function info()
{
    $where = $this->session->userdata('id_user');
    $data['user'] = $this->session->userdata('nama');
    $data['judul'] = "Selesai Booking";
    $data['useraktif'] = $this->ModelUser->cekData(['id' => $this->session->userdata('id_user')])->result();
    $data['items'] = $this->db->query("select*from booking bo, booking_detail d, buku bu where d.id_booking=bo.id_booking and d.id_buku=bu.id and bo.id_user='$where'")->result_array();

    $this->load->view('templates/templates-user/header', $data);
    $this->load->view('booking/info-booking', $data);
    $this->load->view('templates/templates-user/modal');
    $this->load->view('templates/templates-user/footer');
}

```

### C. Membuat View Info Booking

Seperti dijelaskan sebelumnya, view info booking digunakan untuk menampilkan data akhir dari proses booking buku. Buatlah file dengan nama **info-booking.php** dalam folder **Application/views/booking**. Ketikkan script berikut:

```
<div class="container">
  <center>
    <table>
      <?php
      foreach ($useraktif as $u) {
        ?>
        <tr>
          <td nowrap>Terima Kasih <b><?= $u->nama; ?></b>
          </td>
        </tr>
        <tr>
          <td>Buku Yang ingin Anda Pinjam Adalah Sebagai berikut:</td>
        </tr>
      <?php } ?>
      <tr>
        <td>
          <div class="table-responsive">
            <table class="table table-bordered table-striped table-
hover" id="table-datatable">
              <tr>
                <th>No.</th>
                <th>Buku</th>
                <th>Penulis</th>
                <th>Penerbit</th>
                <th>Tahun</th>
              </tr>
              <?php
              $no = 1;
              foreach ($items as $i) {
                ?>
                <tr>
                  <td><?= $no; ?></td>
                  <td>
                    
                  </td>
                  <td nowrap><?= $i['pengarang']; ?></td>
                  <td nowrap><?= $i['penerbit']; ?></td>
```

```

<div class="container">
  <center>
    <table>
      <?php
        foreach ($useraktif as $u) {
          ?>
            <tr>
              <td nowrap>Terima Kasih <b><?=$u->nama; ?></b>
              </td>
            </tr>
            <tr>
              <td>Buku Yang ingin Anda Pinjam Adalah Sebagai berikut:</td>
            </tr>
          <?php } ?>
          <tr>
            <td>
              <div class="table-responsive">
                <table class="table table-bordered table-striped table-
hover" id="table-datatable">
                  <tr>
                    <th>No.</th>
                    <th>Buku</th>
                    <th>Penulis</th>
                    <th>Penerbit</th>
                    <th>Tahun</th>
                  </tr>
                  <?php
                    $no = 1;
                    foreach ($items as $i) {
                      ?>
                        <tr>
                          <td><?=$no; ?></td>
                          <td>
                            
                          </td>
                          <td nowrap><?=$i['pengarang']; ?></td>
                          <td nowrap><?=$i['penerbit']; ?></td>

```

```


        <td nowrap><?= $i['tahun_terbit']; ?></td>
    </tr>
    <?php $no++;
    } ?>
</table>
</div>
</td>
</tr>
<tr>
    <td>
        <hr>
    </td>
</tr>
<tr>
    <td>
        <a class="btn btn-sm btn-outline-
danger" onclick="information('Waktu Pengambilan Buku 1x24 jam dari Booking!!!')"
href="<?php echo base_url() . 'booking/exportToPdf/' . $this->session-
>userdata('id_user'); ?>"><span class="far fa-lg fa-fw fa-file-
pdf"></span> Pdf</a>
    </td>
</tr>
</table>
</center>
</div>


```

Hasil dari view **info-booking.php** adalah sebagai berikut:

Pustaka Beranda Booking 0 Buku Profil Saya Log out Selamat Datang **Maruloh, M.Kom**

Terima Kasih **Maruloh, M.Kom**  
Buku Yang ingin Anda Pinjam Adalah Sebagai berikut:

No.	Buku	Penulis	Penerbit	Tahun
1		Fintri Indriyani, Yunita, Dinda AM, Artika S, Sriyadi	Graha Ilmu	2020

 Pdf

Dapat dilihat data booking kembali menjadi 0. Dan terdapat tombol cetak pdf sebagaibukti booking saat pengambilan buku.



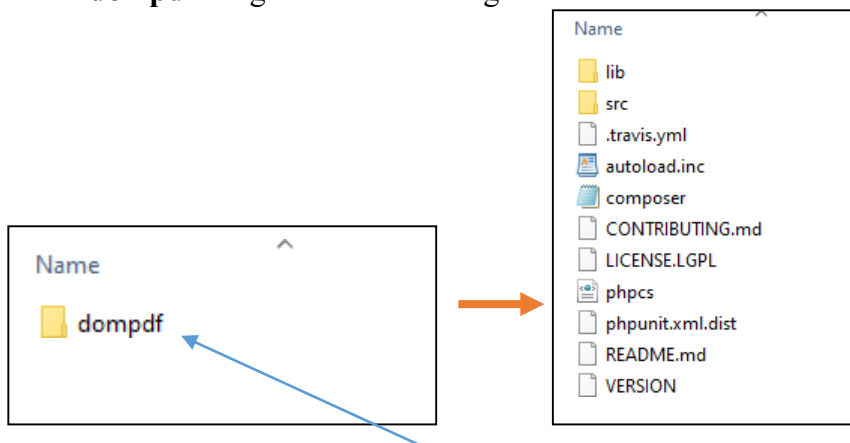
## D. Mencetak Bukti Booking

Untuk membuat cetak laporan PDF pada PHP, kita bisa menggunakan library DOMPDPF. DOMPDPF salah satu library PHP yang bisa di gunakan untuk membuat cetak laporan PDF dengan sangat mudah.

### 1. Penggunaan DOMPDPF

Langkah-langkah menggunakan DOMPDPF adalah sebagai berikut:

- pertama kita download terlebih dahulu librarynya dompdf for codeigniter, untuk download silakan klik link [disini](#)
- selanjutnya, ekstrak file dompdf\_1-0-2 yang sudah kita download. hasil ekstrak folder **dompdf** dengan isi folder sebagai berikut



- pindahkan atau copy folder **dompdf** ke folder application/third\_party pada aplikasi pustaka\_booking kita.

### 2. Membuat fungsi exportToPdf

Tambahkan sebuah fungsi baru pada controller **Booking.php** dengan nama **exportToPdf**, ketikkan script berikut:

```
public function exportToPdf()
{
    $id_user = $this->session->userdata('id_user');
    $data['user'] = $this->session->userdata('nama');
    $data['judul'] = "Cetak Bukti Booking";
    $data['useraktif'] = $this->ModelUser->cekData(['id' => $this->session->userdata('id_user')])>result();
}
```

```

        $data['items'] = $this->db->query("select*from booking bo, booking_detail
d, buku bu where d.id_booking=bo.id_booking and d.id_buku=bu.id and
bo.id_user='$id_user'")->result_array();

```

```

        $sroot      = $_SERVER['DOCUMENT_ROOT'];
        include $sroot."/pustaka-
booking/application/third_party/dompdf/autoload.inc.php";

        $dompdf = new Dompdf\Dompdf();

```

```

        $this->load->view('booking/bukti-pdf', $data);

```

```

        $paper_size  = 'A4'; // ukuran kertas
        $orientation = 'landscape'; //tipe format kertas potrait atau landscape
        $html = $this->output->get_output();

```

```

        $dompdf->set_paper($paper_size,$orientation);

```

Ukuran kertas dapat diatur sesuai dengan kebutuhan, begitupun nama file pdf yang akan dihasilkan dapat kita rubah sesuai keinginan

```

        $paper_size  = 'A4'; // ukuran kertas

```

```

        $dompdf->stream("bukti-booking-$id_user.pdf", array('Attachment' => 0));

```

### 3. Membuat view bukti-pdf

View bukti-pdf berisi data hasil cetakan dari pdf, kita bisa mengatur seperti apa tampilannya.

Berikut scrip **bukti-pdf.php** yang dibuat pada folder **Application-views-booking**

```
<table border=1>
    <?php
    foreach ($useraktif as $u) {
    ?>
```

```

        <tr>
            <th>Nama Anggota : <?= $u->nama; ?></th>
        </tr>
        <tr>
            <th>Buku Yang diboeking:</th>
        </tr>
    <?php } ?>
    <tr>
        <td>
            <div class="table-responsive">
                <table border=1>
                    <tr>
                        <th>No.</th>
                        <th>Buku</th>
                        <th>Penulis</th>
                        <th>penerbit</th>
                        <th>Tahun</th>
                    </tr>
                    <?php
                    $no = 1;
                    foreach ($items as $i) {
                        ?>
                        <tr>
                            <td><?= $no; ?></td>
                            <td>
                                <?= $i['judul_buku']; ?>
                            </td>
                            <td><?= $i['pengarang']; ?></td>
                            <td><?= $i['penerbit']; ?></td>
                            <td><?= $i['tahun_terbit']; ?></td>
                        </tr>
                        <?php $no++;
                    } ?>
                </table>
            </div>
        </td>
    </tr>
    <tr>
        <td>
            <hr>

```

```

        </td>
    </tr>
</table>

```

hasil dari cetak bukti pdf:

Nama Anggota : Maruloh, M.Kom				
Buku Yang dibooking:				
No.	Buku	Penulis	penerbit	Tahun
1	Analisa Perancangan Sistem Informasi	Fintri Indriyani, Yunita, Dinda AM, Artika S, Sriyadi	Graha Ilmu	2020
57ab56750233425dbdf5a30aa2fe2306				

## 9) Membuat Halaman Pengambilan dan Pengembalian Buku

### A. Merubah Index Controller Admin

Sebelum kita memulai proses pengambilan dan pengembalian buku, terlebih dahulu kita edit controller Admin. Jika sebelumnya index controller admin berisi tampilan data yang ada di database, seperti data buku, data anggota, buku diboeking dll. Sekarang kita akan modifikasi controller dengan menambahkan script update data buku setelah diboeking, serta hapus otomatis data booking saat buku tidak diambil/dipinjam oleh anggota.

Oleh karena silakan buka kembali controller admin, dan sesuaikan dengan script berikut:

```
<?php
defined('BASEPATH') or exit('No direct script access allowed');

class Admin extends CI_Controller
{
    public function __construct()
    {
        parent::__construct();
        cek_login();
        cek_user();
    }
}
```

```

public function index()
{
    $data['judul'] = 'Dashboard';
    $data['user'] = $this->ModelUser->cekData(['email' => $this->session-
>userdata('email')])>row_array();
    $data['anggota'] = $this->ModelUser->getUserLimit()->result_array();
    $data['buku'] = $this->ModelBuku->getLimitBuku()->result_array();

    //mengupdate stok dan dibooking pada tabel buku
    $detail = $this->db-
>query("SELECT*FROM booking,booking_detail WHERE DAY(curdate()) < DAY(batas_ambila
) AND booking.id_booking=booking_detail.id_booking")>result_array();foreach
($detail as $key) {
    $id_buku = $key['id_buku'];
    $batas = $key['tgl_booking'];
    $tglawal = date_create($batas);
    $tglskrg = date_create();
    $beda = date_diff($tglawal, $tglskrg);

    if ($beda->days > 2) {
        $this->db-
>query("UPDATE buku SET stok=stok+1, dibooking=dibooking-1 WHERE id='$id_buku'");
    }
}

//menghapus otomatis data booking yang sudah lewat dari 2 hari
$booking = $this->ModelBooking->getData('booking');if
(!empty($booking)) {
    foreach ($booking as $bo) {
        $id_booking = $booking->id_booking;
        $tglbooking = $booking->tgl_booking;
        $tglawal = date_create($tglbooking);
        $tglskrg = date_create();
        $beda = date_diff($tglawal, $tglskrg);

        if ($beda->days > 2) {
            $this->db-
>query("DELETE FROM booking WHERE id_booking='$id_booking'");
            $this->db-

```

```
$this->load->view('templates/header', $data);  
$this->load->view('templates/sidebar', $data);  
$this->load->view('templates/topbar', $data);  
$this->load->view('admin/index', $data);  
$this->load->view('templates/footer');  
}  
}
```

Pada **ModelBuku** silakan tambahkan fungsi **getLimitBuku()** sengan script berikut:

```
public function getLimitBuku()  
{  
    $this->db->limit(5);  
    return $this->db->get('buku');  
}
```

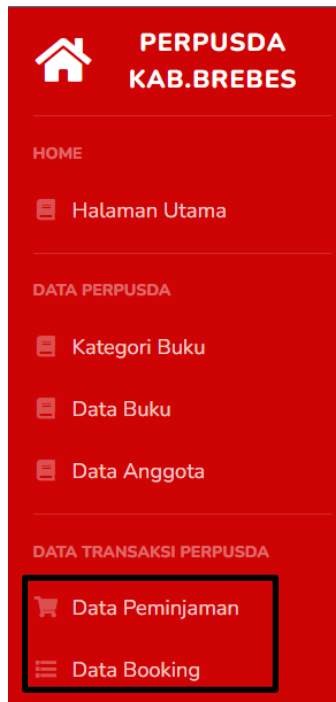


## B. Membuat Menu Transaksi

Untuk membuat menu laporan buku, kita harus membuka kembali project pustaka- booking yang pernah kita kerjakan pada Pemrograman Web II. Silakan buka file sidebar.php yang berada di folder **application/views/templates/sidebar.php**, silakan tambahkan coding berikut:

```
<!-- Divider -->
<hr class="sidebar-divider mt-3">
<!--Heading-->
<div class="sidebar-heading">
    Data Transaksi Perpustakaan
</div>
<!--Nav Item -Dashboard-->
<li class="nav-item active">
    <!--Nav Item - Dashboard-->
<li class="nav-item">
    <a class="nav-link pb-0" href="{base_url('pinjam')}">
        <i class="fa fa-fw fa-shopping-cart"></i>
        <span>Data Peminjaman</span></a>
</li>
<li class="nav-item">
    <a class="nav-link pb-0" href="{base_url('pinjam/daftarBooking')}">
        <i class="fa fa-fw fa-list"></i>
        <span>Data Booking</span></a>
</li>
</li>
<!-- Divider -->
<hr class="sidebar-divider mt-3">
```

Hasil dari script tersebut adalah :



## A. Membuat Halaman Data Booking

Pada sidebar.php yang telah kita edit diatas. sebelum menampilkan halaman databooking buku, terlebih dahulu kita buat fungsi daftar booking yang ada di controller pinjam

```
<li class="nav-item">
    <a class="nav-link pb-0" href="<?= base_url('pinjam/daftarBooking'); ?>">
```

Oleh karena itu diawal kita buat terlebih dahulu controller **Pinjam.php** dan menambahkan fungsi daftarBooking di controller ters

```
<?php if (!defined('BASEPATH')) exit('No Direct Script Access Allowed');

class Pinjam extends CI_Controller
{
    public function __construct()
    {
        parent::__construct();
        cek_login();
        cek_user();
    }

    public function index()
    {
    }

    public function daftarBooking()
    {
        $data['judul'] = "Daftar Booking";
        $data['user'] = $this->ModelUser->cekData(['email' => $this->session->userdata('email')])>row_array();
        $data['pinjam'] = $this->db->query("select*from booking")>result_array();

        $this->load->view('templates/header', $data);
        $this->load->view('templates/sidebar', $data);
        $this->load->view('templates/topbar', $data);
        $this->load->view('booking/daftar-booking', $data);
        $this->load->view('templates/footer');
    }
}
```

Pada controller Pinjam.php, model yang digunakan adalah ModelBuku, ModelUser, dan ModelPinjam. Fungsi daftarBooking digunakan untuk mengirim data judul, user, dan data booking buku, serta memanggil view daftar-booking yang ada pada folder booking.

Langkah berikutnya adalah kita membuat view dengan nama **daftar-booking.php** didalam folder **Application/views/booking**. Ketikkan script berikut:

```
<div class="container">
    <center>
        <table>
            <tr>
                <td>
                    <div class="table-responsive full-width">
                        <table class="table table-bordered table-striped table-
hover" id="table-datatable">
                            <tr>
                                <th>No.</th>
                                <th>ID Booking</th>
                                <th>Tanggal Booking</th>
                                <th>ID User</th>
                                <th>Aksi</th>
                                <th>Denda / Buku /Hari</th>
                                <th>Lama Pinjam</th>
                            </tr>
                            <?php
$no = 1;
foreach ($pinjam as $p) {
    ?>
                        <tr>
                            <td><?= $no; ?></td>
                            <td><a class="btn btn-
link" href="<?= base_url('pinjam/bookingDetail/' . $p['id_booking']); ?>"><?= $p[
'id_booking']; ?></a></td>
                            <td><?= $p['tgl_booking']; ?></td>
                            <td><?= $p['id_user']; ?></td>
                            <form action="<?= base_url('pinjam/pinjamAct/
' . $p['id_booking']); ?>" method="post">
                                <td nowrap>
                                    <button type="submit" class="btn btn-
sm btn-outline-info"><i class="fas fa-fw fa-cart-plus"></i> Pinjam</button>
                                </td>
                            <td>
```


```

                                <input class="form-check-
user rounded-
sm" style="width:100px" type="text" name="denda" id="denda" value="5000">
                                <?= form_error(); ?>
                                </td>
                                <td>
                                <input class="form-check-
user rounded-sm" style="width:100px" type="text" name="lama" id="lama" value="3">
                                <?= form_error(); ?>
                                </td>
                                </form>
                                </tr>
                                <?php $no++;
                                } ?>
                                </table>
                                </div>
                                </td>
                                </tr>
                                <tr>
                                <td align="center"><a href="<?= base_url('pinjam/daftarBooking');
?>" class="btn btn-link"><i class="fas fa-fw fa-refresh"></i> Refresh</a></td>
                                </tr>
                                </table>
                                </center>
                                </div>

```

Hasil dari view daftar-booking.php adalah sebagai berikut:

## Daftar Booking

No.	ID Booking	Tanggal Booking	ID User	Aksi	Denda / Buku /Hari	Lama Pinjam
1	23042020001	2020-04-23	2	 Pinjam	5000	3
Refresh						

Pada tampilan diatas, terlihat 3 tombol (link) yang bisa dipilih oleh admin. Yaitu **tombol Pinjam** saat member ingin mengambil buku, **link ID Booking** untuk melihat detail buku yang diboeking, dan **Link Refresh** untuk merefresh halaman daftar-booking.

Untuk link ID Booking sistem mengarahkan ke fungsi **bookingDetail** yang ada di controller Pinjam.php (harus buat fungsi bookingDetail)

```
<td><a class="btn btn-link" href="<?= base_url('pinjam/bookingDetail/' . $p['id_booking']); ?>"><?= $p['id_booking']; ?></a></td>
```

Untuk tombol Pinjam, sistem akan mengarahkan ke fungsi pinjamAct yang juga ada di controller Pinjam.php (harus buat fungsi pinjamAct)

```
<form action="<?= base_url('pinjam/pinjamAct/' . $p['id_booking']); ?>" method="post">
```

Sedangkan untuk link refresh, sistem hanya akan mengarahkan ke fungsi daftarBooking yang ada di controller Pinjam (tidak perlu membuat fungsi daftarBooking, karena sudah dibuat)

```
<td align="center"><a href="<?= base_url('pinjam/daftarBooking'); ?>" class="btn btn-link"><i class="fas fa-fw fa-refresh"></i> Refresh</a></td>
```

## 1. Membuat Fungsi bookingDetail

Seperti penjelasan sebelumnya, untuk melihat detail booking, kita akan membuat fungsi **bookingDetail** dalam controller Pinjam.php. Ketikkan scrip berikut :

```

public function bookingDetail()
{
    $id_booking = $this->uri->segment(3);
    $data['judul'] = "Booking Detail";
    $data['user'] = $this->ModelUser->cekData(['email' => $this->session->userdata('email')])->row_array();
    $data['agt_booking'] = $this->db->query("select*from booking b, user u where b.id_user=u.id and b.id_booking='$id_booking'")->result_array();
    $data['detail'] = $this->db->query("select id_buku,judul_buku,pengarang,penerbit,tahun_terbit from booking_detail d, buku b where d.id_buku=b.id and d.id_booking='$id_booking'")->result_array();

    $this->load->view('templates/header', $data);
    $this->load->view('templates/sidebar', $data);
    $this->load->view('templates/topbar', $data);
    $this->load->view('booking/booking-detail', $data);
    $this->load->view('templates/footer');
}

```

Script diatas, akan memanggil view booking-detail yang ada di folder booking. Oleh karena itu kita akan membuat sebuah view baru di folder Application/views/booking dengan nama booking-detail.php. berikut scriptnya:

```
<div class="container">
    <center>
        <table>
            <?php foreach ($agt_booking as $ab) { ?>
                <tr>
                    <td>Data Anggota</td>
                    <td>:</td>
                    <th><?= $ab['nama']; ?></th>
                </tr>
                <tr>
                    <td>ID Booking</td>
                    <td>:</td>
                    <th><?= $ab['id_booking']; ?></th>
                </tr>
            <?php } ?>
            <tr>
                <td colspan="3">&nbsp;</td>
            </tr>
            <tr>
                <td colspan="3">
                    <hr>
                </td>
            </tr>
            <tr>
                <td colspan="3">
                    <div class="table-responsive full-width">
                        <table class="table table-bordered table-striped table-hover"
id="table-datatable">
                            <tr>
                                <th>No.</th>
                                <th>ID Buku</th>
                                <th>Judul Buku</th>
                                <th>Pengarang</th>
                                <th>Penerbit</th>
                                <th>Tahun</th>
                            </tr>
                        </table>
                    </div>
                </td>
            </tr>
        </table>
    </center>
</div>
```



```

        <tr>
            <td><?= $no; ?></td>
            <td><?= $d['id_buku']; ?></td>
            <td><?= $d['judul_buku']; ?></td>
            <td><?= $d['pengarang']; ?></td>
            <td><?= $d['penerbit']; ?></td>
            <td><?= $d['tahun_terbit']; ?></td>
        </tr>
        <?php $no++;
        } ?>
    </table>
</div>
</td>
</tr>
<tr>
    <td align="center" colspan="3"><a href="#" onclick="window.histor
y.go(-1)" class="btn btn-outline-dark"><i class="fas fa-fw fa-
reply"></i> Kembali</a></td>
</tr>
</table>
</center>
</div>

```

Hasil dari detail booking:

## Booking Detail

Imam Nawawi 

Data Anggota : Maruloh, M.Kom  
ID Booking : 23042020001

No.	ID Buku	Judul Buku	Pengarang	Penerbit	Tahun
1	4	Analisa Perancangan Sistem Informasi	Fintri Indriyani, Yunita, Dinda AM, Artika S, Sriyadi	Graha Ilmu	2020

[↩ Kembali](#)

## 2. Membuat Fungsi pinjamAct

Untuk menjalankan mengeksekusi tombol pinjam, kita harus membuat terlebih dahulu

**fungsi pinjamAct** di dalam controller **Pinjam.php**. Adapun scriptnya adalah sebagai berikut:

```

public function pinjamAct()
{
    $id_booking = $this->uri->segment(3);
    $lama = $this->input->post('lama', TRUE);

    $bo = $this->db->query("SELECT*FROM booking WHERE id_booking='$id_booking'")->row();

    $tglsekarang = date('Y-m-d');
    $no_pinjam = $this->ModelBooking->kodeOtomatis('pinjam', 'no_pinjam');
    $databooking = [
        'no_pinjam' => $no_pinjam,
        'id_booking' => $id_booking,
        'tgl_pinjam' => $tglsekarang,
        'id_user' => $bo->id_user,
        'tgl_kembali' => date('Y-m-
d', strtotime('+ ' . $lama . ' days', strtotime($tglsekarang))), 'tgl_pengembalian' => '0000-00-00',
        'status' => 'Pinjam',
        'total_denda' => 0
    ];
    $this->ModelPinjam->simpanPinjam($databooking);
    $this->ModelPinjam->simpanDetail($id_booking, $no_pinjam);
    $denda = $this->input->post('denda', TRUE);
    $this->db->query("update detail_pinjam set denda='$denda'");

    //hapus Data booking yang bukunya diambil untuk dipinjam
    $this->ModelPinjam->deleteData('booking', ['id_booking' => $id_booking]);
    $this->ModelPinjam->deleteData('booking_detail', ['id_booking' => $id_booking]);
    //$this->db->query("DELETE FROM booking WHERE id_booking='$id_booking'");

    //update dibooking dan dipinjam pada tabel buku saat buku yang dibookingdiambil
    untuk dipinjam
    $this->db->query("UPDATE buku, detail_pinjam SET buku.dipinjam=buku.dipinjam+1, buku.dibooking=buku.dibooking-1 WHERE buku.id=detail_pinjam.id_buku");

    $this->session->set_flashdata('pesan', '<div class="alert alert- message alert-
success" role="alert">Data Peminjaman Berhasil Disimpan</div>');
    redirect(base_url() . 'pinjam');
}

```

Pada fungsi tambahAct, tanggal kembali akan ditentukan dengan menambahkan lama pinjam dengan tanggal yang berlaku di sistem.

```
$lama = $this->input->post('lama', TRUE);
```

```
$tglsekarang = date('Y-m-d');
```

```
'tgl_kembali' => date('Y-m-d', strtotime('+ ' . $lama . ' days', strtotime($tglsekarang))),
```

Data yang dikirim akan disimpan kedalam tabel pinjam dan detail pinjam, dan data booking serta data detail booking otomatis dihapus (pindah ke tabel pinjam dan detail)

```
$this->ModelPinjam->simpanPinjam($databooking);  
$this->ModelPinjam->simpanDetail($id_booking);
```

```
$this->ModelPinjam->deleteData('booking', ['id_booking' => $id_booking]);  
$this->ModelPinjam->deleteData('booking_detail', ['id_booking' => $id_booking]);
```

Data buku dipinjam akan ditambah 1 dan data buku booking akan dihapus 1

```
$this->db->query("UPDATE buku, detail_pinjam SET buku.dipinjam=buku.dipinjam+1, buku.dibooking=buku.dibooking-1 WHERE buku.id=detail_pinjam.id_buku");
```

Sistem secara otomatis akan mengalihkan ke controller Pinjam.php

```
$this->session->set_flashdata('pesan', '<div class="alert alert-message alert-success" role="alert">Data Peminjaman Berhasil Disimpan</div>');  
redirect(base_url() . 'pinjam');
```

## B. Membuat Halaman Data Peminjaman

### 1. Membuat Fungsi Index pada Controller Pinjam

fungsi index ini akan dijalankan pertama kali saat sistem memanggil controller Pinjam.

Tambahkan scrip index pada controller pinjam sebagai berikut:

```
public function index()
{
    $data['judul'] = "Data Pinjam";
    $data['user'] = $this->ModelUser->cekData(['email' => $this->session->userdata('email')])->row_array();
    $data['pinjam'] = $this->ModelPinjam->joinData();

    $this->load->view('templates/header', $data);
}
```

```

$this->load->view('templates/sidebar', $data);
$this->load->view('templates/topbar', $data);
$this->load->view('pinjam/data-pinjam', $data);
$this->load->view('templates/footer');
}

```

Dari script diatas, terlihat bahwa saat controller Pinjam dipanggil, maka dia akan meload halaman view data-pinjam yang ada di folder pinjam. Termasuk mengirimkan data-data seperti user, termasuk isian data yang akan ditampilkan. Silakan tambahkan sebuah file viewbaru dengan nama **data-pinjam.php** didalam folder **Application/views/pinjam** dan ketikkan script berikut:

```

<div class="container">
    <center>
        <table>
            <tr>
                <td>
                    <div class="table-responsive full-width">
                        <table class="table table-bordered table-striped table-
hover" id="table-datatable">
                            <tr>
                                <th>No Pinjam</th>
                                <th>Tanggal Pinjam</th>
                                <th>ID User</th>
                                <th>ID Buku</th>
                                <th>Tanggal Kembali</th>
                                <th>Tanggal Pengembalian</th>
                                <th>Terlambat</th>
                                <th>Denda</th>
                                <th>Status</th>
                                <th>Total Denda</th>
                                <th>Pilihan</th>
                            </tr>
                            <?php

                            foreach ($pinjam as $p) {
                                <tr>
                                    <td><?= $p['no_pinjam']; ?></td>
                                    <td><?= $p['tgl_pinjam']; ?></td>
                                    <td><?= $p['id_user']; ?></td>
                                    <td><?= $p['id_buku']; ?></td>

```

```

        <td><?= $p['tgl_kembali']; ?></td>
        <td>
            <?= date('Y-m-d'); ?>
            <input type="hidden" name="tgl_pengembalian"
id="tgl_pengembalian" value="<?= date('Y-m-d'); ?>">
        </td>
        <td>
            <?php
            $tgl1 = new DateTime($p['tgl_kembali']);
            $tgl2 = new DateTime();
            $selisih = $tgl2->diff($tgl1)-
>format("%a");
            echo $selisih;
            ?> Hari
        </td>
        <td><?= $p['denda']; ?></td>

        <?php if ($p['status'] == "Pinjam") {
            $status = "warning";
        } else {
            $status = "secondary";
        } ?>
        <td><i class="btn btn-outline-
<?= $status; ?> btn-sm"><?= $p['status']; ?></i></td>

        <?php
        if ($selisih < 0) {
            $total_denda = $p['denda'] * 0;
        } else {
            $total_denda = $p['denda'] * $selisih;
        }
        ?>

        <td><?= $total_denda; ?>
            <input type="hidden" name="totaldenda" id
="totaldenda" value="<?= $total_denda; ?>">
        </td>
        <td nowrap>
            <?php if ($p['status'] == "Kembali") { ?>
                <i class="btn btn-sm btn-outline-
secondary"><i class="fas fa-fw fa-edit"></i>Ubah Status</i>

```

```

        <?php } ?>
    </td>
</tr>
<?php
} ?>
</table>
</div>
</td>
</tr>

</table>
</center>
</div>

```

Hasil dari script diatas adalah :

#### Data Pinjam

Imam Nawawi 

No Pinjam	Tanggal Pinjam	ID User	ID Buku	Tanggal Kembali	Tanggal Pengembalian	Terlambat	Denda	Status	Total Denda	Pilihan
23042020001	2020-04-23	2	4	2020-04-26	2020-04-23	2 Hari	5000	<span>Pinjam</span>	10000	<span>Ubah</span>

Didalam halaman view data-pinjam.php, terdapat sebuah link untuk merubah status peminjaman dari pinjam menjadi kembali. Link tersebut mengarahkan ke fungsi ubahStatus dengan meminta data id\_buku dan no\_pinjam yang ada di controller Pinjam.php

```

<td nowrap>
    <?php if ($p['status'] == "Kembali") { ?>
        <i class="btn btn-sm btn-outline-secondary"><i class="fas fa-fw fa-
edit"></i>Ubah Status</i>
    <?php } else { ?>
        <a class="btn btn-sm btn-outline-
info" href="<?= base_ur ('pinjam/ubahStatus/' . $p['id_buku'] . '/' . $p['no_pinj
am']); ?>"><i class="fas fa-fw fa-edit"></i>Ubah Status</a>
    <?php } ?>
</td>

```

Karena itu langkah berikutnya kita akan buat sebuah fungsi dengan nama **ubahStatus** di dalam controller **Pinjam.php**. adapun scriptnya adalah sebagai berikut:

```

public function ubahStatus()
{
    $id_buku = $this->uri->segment(3);
    $no_pinjam = $this->uri->segment(4);
    $where = ['id_buku' => $this->uri->segment(3),];

    $tgl = date('Y-m-d');
    $status = 'Kembali';
    //update status menjadi kembali pada saat buku dikembalikan
    $this->db-
>query("UPDATE pinjam, detail_pinjam SET pinjam.status='$status', pinjam.tgl_peng
embalian='$tgl' WHERE detail_pinjam.id_buku='$id_buku' AND pinjam.no_pinjam='$no_
pinjam'");

    //update stok dan dipinjam pada tabel buku
    $this->db-
>query("UPDATE buku, detail_pinjam SET buku.dipinjam=buku.dipinjam-
1, buku.stok=buku.stok+1 WHERE buku.id=detail_pinjam.id_buku");

    $this->session->set_flashdata('pesan', '<div class="laert alert-
message alert-success" role="alert"></div>');
    redirect(base_url('pinjam'));
}

```

Pada script fungsi ubah status, akan diambil data berupa id buku dan no.pinjam, kemudian merubah status menjadi kembali. Sistem juga akan mengupdate data buku dipinjam menambah stok buku yang dikembalikan.

```

$id_buku = $this->uri->segment(3);
$no_pinjam = $this->uri->segment(4);

```

```

$this->db-
>query("UPDATE pinjam, detail_pinjam SET pinjam.status='$status', pinjam.tgl_peng
embalian='$tgl' WHERE detail_pinjam.id_buku='$id_buku' AND pinjam.no_pinjam='$no_
pinjam'");
$this->db->query("UPDATE buku, detail_pinjam SET buku.dipinjam=buku.dipinjam-
1, buku.stok=buku.stok+1 WHERE buku.id=detail_pinjam.id_buku");


```



Hasil dari script :

## Data Pinjam

Imam Nawawi 

No Pinjam	Tanggal Pinjam	ID User	ID Buku	Tanggal Kembali	Tanggal Pengembalian	Terlambat	Denda	Status	Total Denda	Pilihan
23042020001	2020-04-23	2	4	2020-04-26	2020-04-23	2 Hari	5000	Kembali	10000	 Ubah Status

## 10) Membuat dan Mencetak Laporan Data Buku ke Printer

### A. Membuat Menu Laporan

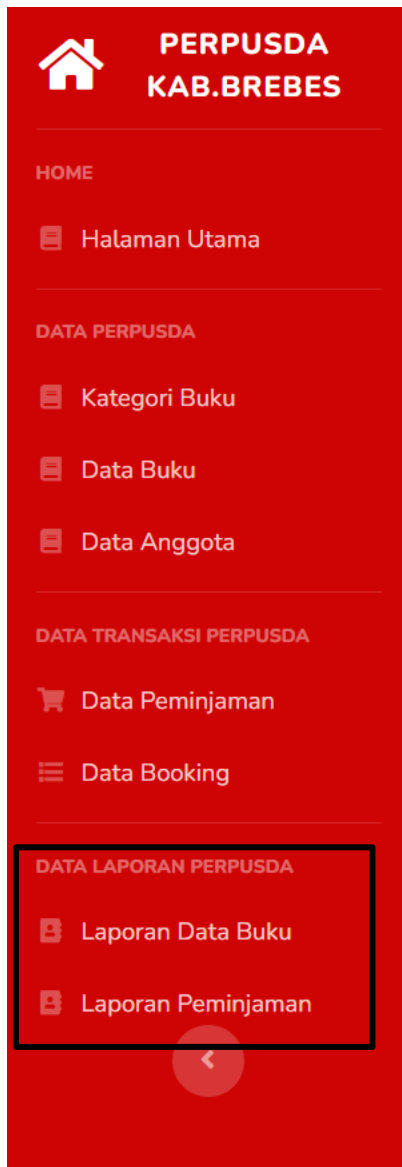
Untuk membuat menu laporan buku, kita harus membuka kembali project pustaka- booking yang pernah kita kerjakan pada Pemrograman Web II. Silakan buka file sidebar.php yang berada di folder **application/views/templates/sidebar.php**, silakan tambahkan coding berikut:

```
<!-- Heading -->
<div class="sidebar-heading">
    Data Laporan Perpustakaan
</div>
<!-- Nav Item - Dashboard -->
<li class="nav-item active">
    <!-- Nav Item - Dashboard -->
<li class="nav-item">
    <a class="nav-link pb-0" href="{?= base_url('laporan/laporan_buku'); ?}">
        <i class="fa fa-fw fa-address-book"></i>
        <span>Laporan Data Buku</span></a>
</li>
<li class="nav-item">
    <a class="nav-link pb-0" href="{?= base_url('laporan/laporan_pinjam'); ?}">
        <i class="fa fa-fw fa-address-book"></i>
        <span>Laporan Peminjaman</span></a>
</li>
</li>

<!-- Sidebar Toggler (Sidebar) -->
<div class="text-center d-none d-md-inline">
    <button class="rounded-circle border-0" id="sidebarToggle"></button>
</div>

</ul>
<!-- End of Sidebar -->
```

Hasil dari coding diatas adalah sebagai berikut:



## B. Membuat halaman Laporan Buku

Langkah-langkah untuk membuat halaman laporan buku adalah sebagai berikut:

1. Buatlah Controller baru dengan nama **Laporan**.
2. Buat fungsi **laporan\_buku** pada controller Laporan.

```
<?php
defined('BASEPATH') or exit('No Direct script access allowed');

class Laporan extends CI_Controller
{
    function __construct()
    {
        parent::__construct();
    }
    public function laporan_buku()
    {
        $data['judul'] = 'Laporan Data Buku';
        $data['user'] = $this->ModelUser->cekData(['email' => $this->session->userdata('email')])>row_array();
        $data['buku'] = $this->ModelBuku->getBuku()->result_array();
        $data['kategori'] = $this->ModelBuku->getKategori()->result_array();

        $this->load->view('templates/header', $data);
        $this->load->view('templates/sidebar', $data);
        $this->load->view('templates/topbar', $data);
        $this->load->view('buku/laporan_buku', $data);
        $this->load->view('templates/footer');
    }
}
```

Fungsi tersebut digunakan untuk memanggil view **laporan\_buku** yang ada didalam folder **buku**, dengan mengirimkan data buku kedalamnya.

3. Setelah itu kita buat file laporan\_buku.php untuk menampilkan data buku di dalam folder **application/views/buku/laporan\_buku.php**

```
<!-- Begin Page Content -->
<div class="container-fluid">

    <?= $this->session->flashdata('pesan'); ?>
    <div class="row">
        <div class="col-lg-12">
            <?php if(validation_errors()){?>
```

```

        <div class="alert alert-danger" role="alert">
        <?= validation_errors();?>
        </div>

    <?php }?>
    <?= $this->session->flashdata('pesan'); ?>
    <a href="<?= base_url('laporan/cetak_laporan_buku'); ?>" class="btn btn-primary
mb-3"><i class="fas fa-print"></i> Print</a>
    <a href="<?= base_url('laporan/laporan_buku_pdf'); ?>" class="btn btn
-warning mb-3"><i class="far fa-file-pdf"></i> Download Pdf</a>
    <a href="<?= base_url('laporan/export_excel'); ?>" class="btn btn-success mb-
3"><i class="far fa-file-excel"></i> Export ke Excel</a>

    <table class="table table-hover">
        <thead>
            <tr>
                <th scope="col">#</th>
                <th scope="col">Judul</th>
                <th scope="col">Pengarang</th>
                <th scope="col">Penerbit</th>
                <th scope="col">Tahun Terbit</th>
                <th scope="col">ISBN</th>
                <th scope="col">Stok</th>
            </tr>
        </thead>
        <tbody>

            <?php
                $a = 1;
                foreach ($buku as $b) { ?>
                    <tr>
                        <th scope="row"><?= $a++; ?></th>
                        <td><?= $b['judul_buku']; ?></td>
                        <td><?= $b['pengarang']; ?></td>
                        <td><?= $b['penerbit']; ?></td>
                        <td><?= $b['tahun_terbit']; ?></td>
                        <td><?= $b['isbn']; ?></td>
                        <td><?= $b['stok']; ?></td>
                    </tr>
                <?php } ?>
            </tbody>
        </table>
    </div>

```

<!-- End of Main Content -->

4. Hasil dari coding di atas adalah sebagai berikut:



Laporan Data Buku					
#	Judul	Pengarang	Penerbit	Tahun Terbit	ISBN
1	Hujan	Tere Liye	Gramedia Pustaka Utama	2019	9786020324784
2	Belajar Coding Itu Penting di Era Revolusi Industri 4.0	YENI MULYANI	Pustaka Baru Press	2020	9786023763672
3	Morfologi Bahasa Indonesia	Jusrin Efendi Pohan	Rineka Cipta	2019	9786232091825
4	Buku Ajar Pengantar Ilmu Komunikasi	Didik Hariyanto	Pustaka Ilmu Group	2015	9786237066866
5	Buku Dasar-Dasar Teknik Informatika	Novega Pratama Adiputra	Deepublish	2020	9786230208195
6	20 Bisnis IT dari Hobi Menjadi Rezeki	Jubilee Enterprise	PT Elex Media Komputindo	2010	87968686787879
7	Atomic Habits	James Clear	Penguin Books	2018	97868687978796

### C. Membuat Print Laporan Data Buku

Berdasarkan view laporan buku yang sudah kita buat sebelumnya, untuk membuat print laporan data buku, kita diharuskan menambahkan fungsi **cetak\_laporan\_buku** di dalam controller Laporan.

```
<?= $this->session->flashdata('pesan'); ?>
<a href="<?= base_url('laporan/cetak_laporan_buku'); ?>" class="btn btn-primary mb-3"><i class="fas fa-print"></i> Print</a>
<a href="<?= base_url('laporan/laporan_buku_pdf'); ?>" class="btn btn-warning mb-3"><i class="far fa-file-pdf"></i> Download Pdf</a>
<a href="<?= base_url('laporan/export_excel'); ?>" class="btn btn-success mb-3"><i class="far fa-file-excel"></i> Export ke Excel</a>
<table class="table table-hover">
  <thead>
    <tr>
```

Adapun langkah-langkah detail dalam membuat Print Laporan Data Buku adalah sebagai berikut:

1. Tambahkan fungsi **cetak\_laporan\_buku** pada **controller Laporan**

```
public function cetak_laporan_buku()  
{  
    $data['buku'] = $this->ModelBuku->getBuku()->result_array();  
    $data['kategori'] = $this->ModelBuku->getKategori()->result_array();  
    $this->load->view('buku/laporan_print_buku', $data);  
}
```

2. Membuat file laporan\_print\_buku.php pada folder



```

<!DOCTYPE html>

<html>
<head>
  <title></title>
</head>
<body>
  <style type="text/css">
    .table-data{
      width: 100%;
      border-collapse: collapse;
    }

    .table-data tr th,
    .table-data tr td{
      border:1px solid black;
      font-size: 11pt;
      font-family:Verdana; padding:
      10px 10px 10px 10px;
    }
    h3{
      font-family:Verdana;
    }
  </style>

  <h3><center>Laporan Data Buku Perputakaan Online</center></h3>
  <br/>
  <table class="table-data">
    <thead>
      <tr>
        <th>No</th>
        <th>Judul Buku</th>
        <th>Pengarang</th>
        <th>Terbit</th>

```

**application/views/buku/laporan\_print\_buku.php**

```

    <th>Tahun Penerbit</th>
    <th>ISBN</th>
    <th>Stok</th>
  </tr>
</thead>
<tbody>
  <?php
    $no = 1;
    foreach($buku as $b){
      ?>
      <tr>
        <th scope="row"><?= $no++; ?></th>
        <td><?= $b['judul_buku']; ?></td>
        <td><?= $b['pengarang']; ?></td>
        <td><?= $b['penerbit']; ?></td>
        <td><?= $b['tahun_terbit']; ?></td>
        <td><?= $b['isbn']; ?></td>
        <td><?= $b['stok']; ?></td>
      </tr>
    <?php
  }
  ?>
</tbody>
</table>

<script type="text/javascript">
  window.print();
</script>

</body>
</html>

```

3. Hasil dari codingan tersebut adalah :

Laporan Data Buku Perputakaan Online

No	Judul Buku	Pengarang	Terbit	Tahun Penerbit	ISBN	Stok
1	Hujan	Tere Liye	Gramedia Pustaka Utama	2019	9786020324784	2
2	Belajar Coding Itu Penting di Era Revolusi Industri 4.0	YENI MULYANI	Pustaka Baru Press	2020	9786023763672	5
3	Morfologi Bahasa Indonesia	Jusrin Efendi Pohan	Rineka Cipta	2019	9786232091825	5
4	Buku Ajar Pengantar Ilmu Komunikasi	Didik Hariyanto	Pustaka Ilmu Group	2015	9786237066866	3
5	Buku Dasar-Dasar Teknik Informatika	Novega Pratama Adiputra	Deepublish	2020	9786230208195	5
6	20 Bisnis IT dari Hobi Menjadi Rezeki	Jubilee Enterprise	PT Elex Media Komputindo	2010	87968686787879	5
7	Atomic Habits	James Clear	Penguin Books	2018	97868687978796	5
8	Bersandar pada Sains	Joko Priyono	Bhuana Ilmu Populer	2022	565756565768868	5
9	One Piece 98	Eiichiro Oda	Elex Media Komputindo	2022	9786230030185	10

## 11) Membuat dan Mencetak Laporan Data Buku ke Tampilan Pdf dan Excel

### A. Membuat Cetak Laporan Data Buku Pdf

Langkah-langkah untuk melakukan cetak laporan data buku pdf adalah sebagaiberikut:

1. Pastikan DOMPdF sudah diatur seperti pada langkah sebelumnya
2. Menambahkan fungsi `laporan_buku_pdf` pada **controller Laporan**

```
<?= $this->session->flashdata('pesan'); ?>
<a href="<?= base_url('laporan/cetak_laporan_buku'); ?>" class="btn btn-primary mb-3"><i class="fas fa-print"></i> Print</a>
<a href="<?= base_url('laporan/laporan_buku_pdf'); ?>" class="btn btn-warning mb-3"><i class="far fa-file-pdf"></i> Download Pdf</a>
<a href="<?= base_url('laporan/export_excel'); ?>" class="btn btn-success mb-3"><i class="far fa-file-excel"></i> Export ke Excel</a>
<table class="table table-hover">
  <thead>
    <tr>
```

Menambahkan fungsi `laporan_buku_pdf`

```
public function laporan_buku_pdf()
{
    $data['buku'] = $this->ModelBuku->getBuku()->result_array();
    // $this->load->library('dompdf_gen');
    $sroot = $_SERVER['DOCUMENT_ROOT'];
    include $sroot . "/pustaka-
booking/application/third_party/dompdf/autoload.inc.php";
```

```

        $dompdf = new Dompdf\Dompdf();
        $this->load->view('buku/laporan_pdf_buku', $data);
        $paper_size = 'A4'; // ukuran kertas
        $orientation = 'landscape'; //tipe format kertas potrait atau
landscape
        $html = $this->output->get_output();
        $dompdf->set_paper($paper_size, $orientation);
        //Convert to PDF
        $dompdf->load_html($html);
        $dompdf->render();
        $dompdf->stream("laporan_data_buku.pdf", array('Attachment' =>
0));
        // nama file pdf yang di hasilkan
    }

```

Teman-teman bisa mengatur ukuran kertas, format kertas serta nama file pdf yang akan di hasilkan nantinya

### 3. Buat view baru dengan nama **laporan\_pdf\_buku**

**Application/views/buku/laporan\_pdf\_buku.php**

```

<!DOCTYPE html>
<html>
<head>
    <title></title>
</head>
<body>
    <style type="text/css">
        .table-data{
            width: 100%;
            border-collapse: collapse;
        }

        .table-data tr th,
        .table-data tr td{
            border:1px solid black;
            font-size: 11pt;
            padding: 10px 10px 10px 10px;
        }
    </style>

```

```

<h3><center>Laporan Data Buku Perputakaan Online</center></h3>
<br/>
<table class="table-data">
  <thead>
    <tr>
      <th>No</th>
      <th>Judul Buku</th>
      <th>Pengarang</th>
      <th>Terbit</th>
      <th>Tahun Penerbit</th>
      <th>ISBN</th>
      <th>Stok</th>
    </tr>
  </thead>

  <tbody>
    <?php
    $no = 1;
    foreach($buku as $b){
    ?>
    <tr>
      <th scope="row"><?= $no++; ?></th>
      <td><?= $b['judul_buku']; ?></td>
      <td><?= $b['pengarang']; ?></td>
      <td><?= $b['penerbit']; ?></td>
      <td><?= $b['tahun_terbit']; ?></td>
      <td><?= $b['isbn']; ?></td>
      <td><?= $b['stok']; ?></td>
    </tr>
    <?php
  }
  ?>
</tbody>
</table>
</body>
</html>

```

4. Hasil dari codingan tersebut adalah sebagai berikut:

#### Laporan Data Buku Perputakaan Online

No	Judul Buku	Pengarang	Terbit	Tahun Penerbit	ISBN	Stok
1	Hujan	Tere Liye	Gramedia Pustaka Utama	2019	9786020324784	2
2	Belajar Coding Itu Penting di Era Revolusi Industri 4.0	YENI MULYANI	Pustaka Baru Press	2020	9786023763672	5
3	Morfologi Bahasa Indonesia	Jusrin Efendi Pohan	Rineka Cipta	2019	9786232091825	5
4	Buku Ajar Pengantar Ilmu Komunikasi	Didik Hariyanto	Pustaka Ilmu Group	2015	9786237066866	3
5	Buku Dasar-Dasar Teknik Informatika	Novega Pratama Adiputra	Deepublish	2020	9786230208195	5
6	20 Bisnis IT dari Hobi Menjadi Rezeki	Jubilee Enterprise	PT Elex Media Komputindo	2010	87968686787879	5
7	Atomic Habits	James Clear	Penguin Books	2018	97868687978796	5
8	Bersandar pada Sains	Joko Priyono	Bhuana Ilmu Populer	2022	565756565768868	5
9	One Piece 98	Eiichiro Oda	Elex Media Komputindo	2022	9786230030185	10

### B. Mengeksport Data kedalam File Excel

Langkah-langkah untuk melakukan cetak laporan data buku pdf adalah sebagai berikut:

1. Menambahkan fungsi **export\_excel** pada controller **Laporan**

```
public function export_excel()
{
    $data = array( 'title' => 'Laporan Buku',
                  'buku' => $this->ModelBuku->getBuku()->result_array());
    $this->load->view('buku/export_excel_buku', $data);
}
```

2. Membuat file **export\_excel\_buku.php** pada folder **Application/views/buku/**  
**export\_excel\_buku.php**

```

<?php
header("Content-type: application/vnd-ms-excel");
header("Content-Disposition: attachment; filename=$title.xls");
header("Pragma: no-cache");
header("Expires: 0");
?>

<h3><center>Laporan Data Buku Perputakaan Online</center></h3>
<br/>
<table class="table-data">
  <thead>
    <tr>
      <th>No</th>
      <th>Judul Buku</th>
      <th>Pengarang</th>
      <th>Terbit</th>
      <th>Tahun Penerbit</th>
      <th>ISBN</th>
      <th>Stok</th>
    </tr>
  </thead>

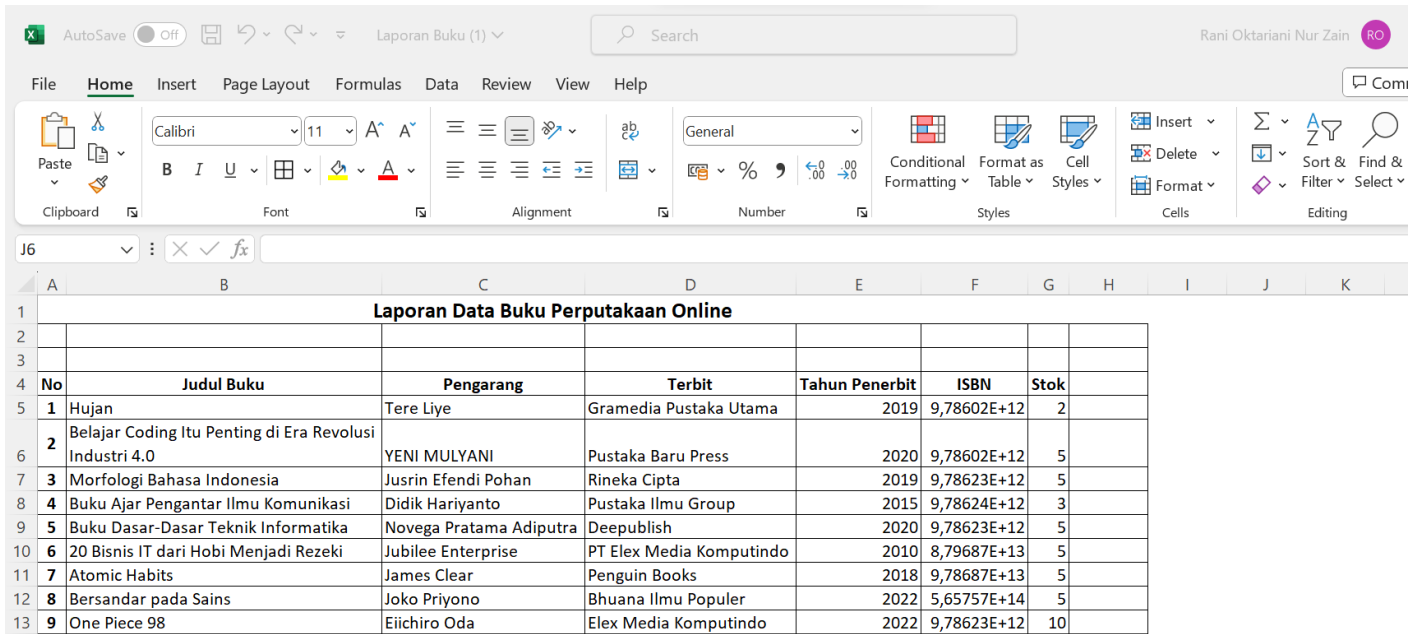
```

```

  <tbody>
    <?php
    $no = 1;
    foreach($buku as $b){
    ?>
    <tr>
      <th scope="row"><?= $no++; ?></th>
      <td><?= $b['judul_buku']; ?></td>
      <td><?= $b['pengarang']; ?></td>
      <td><?= $b['penerbit']; ?></td>
      <td><?= $b['tahun_terbit']; ?></td>
      <td><?= $b['isbn']; ?></td>
      <td><?= $b['stok']; ?></td>
    </tr>
    <?php
    }
    ?>
  </tbody>
</table>

```

### 3. Hasil dari ekspor data:



Laporan Data Buku Perpustakaan Online						
No	Judul Buku	Pengarang	Terbit	Tahun Penerbit	ISBN	Stok
1	Hujan	Tere Liye	Gramedia Pustaka Utama	2019	9,78602E+12	2
2	Belajar Coding Itu Penting di Era Revolusi Industri 4.0	YENI MULYANI	Pustaka Baru Press	2020	9,78602E+12	5
3	Morfologi Bahasa Indonesia	Jusrin Efendi Pohan	Rineka Cipta	2019	9,78623E+12	5
4	Buku Ajar Pengantar Ilmu Komunikasi	Didik Hariyanto	Pustaka Ilmu Group	2015	9,78624E+12	3
5	Buku Dasar-Dasar Teknik Informatika	Novega Pratama Adiputra	Deepublish	2020	9,78623E+12	5
6	20 Bisnis IT dari Hobi Menjadi Rezeki	Jubilee Enterprise	PT Elex Media Komputindo	2010	8,79687E+13	5
7	Atomic Habits	James Clear	Penguin Books	2018	9,78687E+13	5
8	Bersandar pada Sains	Joko Priyono	Bhuana Ilmu Populer	2022	5,65757E+14	5
9	One Piece 98	Eiichiro Oda	Elex Media Komputindo	2022	9,78623E+12	10



## 12) Membuat Laporan Peminjaman

### A. Membuat Halaman Laporan Peminjaman

Yang terakhir yang akan kita buat adalah fitur laporan transaksi. Pada dasarnya pembuatan laporan peminjaman sama seperti pembuatan laporan buku, hanya saja pada laporan peminjaman ada beberapa tabel yang terlibat selain tabel pinjam itu sendiri. Untuk outputnya masih sama yaitu cetak printer, cetak pdf, dan ekspor excel

Coba teman-teman perhatikan pada menu laporan yang ada di file sidebar.php. Kita telah membuat link nya menuju ke fungsi laporan\_pinjam() di controller Laporan.php

```
<!-- Heading -->
<div class="sidebar-heading">
    Laporan
</div>
<!-- Nav Item - Dashboard -->
<li class="nav-item active">
    <!-- Nav Item - Dashboard -->
<li class="nav-item">
    <a class="nav-link pb-0" href="{?= base_url('laporan/laporan_buku'); ?}">
        <i class="fa fa-fw fa-address-book"></i>
        <span>Laporan Data Buku</span></a>
</li>
<li class="nav-item">
    <a class="nav-link pb-0" href="{?= base_url('laporan/laporan_anggota'); ?}">
        <i class="fa fa-fw fa-address-book"></i>
        <span>Laporan Data Anggota</span></a>
</li>
<li class="nav-item">
    <a class="nav-link pb-0" href="{?= base_url('laporan/laporan_pinjam'); ?}">
        <i class="fa fa-fw fa-address-book"></i>
        <span>Laporan Peminjaman</span></a>
</li>
</li>

<!-- Divider -->
<hr class="sidebar-divider mt-3">

<!-- Sidebar Toggler (Sidebar) -->
<div class="text-center d-none d-md-inline">
    <button class="rounded-circle border-0" id="sidebarToggle"></button>
</div>
```

Oleh karena itu kita akan membuat fungsi baru lagi di controller Laporan.php. yaitu fungsi laporan\_pinjam().

#### Application/controllers/Laporan.php

```
public function laporan_pinjam()
{
    $data['judul'] = 'Laporan Data Peminjaman';
    $data['user'] = $this->ModelUser->cekData(['email' => $this->session->userdata('email')])>row_array();
    $data['laporan'] = $this->db->query("select * from pinjam p,detail_pinjam d,
        buku b,user u where d.id_buku=b.id and p.id_user=u.id
        and p.no_pinjam=d.no_pinjam")>result_array();

    $this->load->view('templates/header', $data);
    $this->load->view('templates/sidebar');
    $this->load->view('templates/topbar', $data);
    $this->load->view('pinjam/laporan-pinjam', $data);

    $this->load->view('templates/footer');
}
```

Pada script diatas, data laporan tidak hanya memanggil data pada tabel pinjam saja,tetapi juga melibatkan tabel detail pinjam, user, dan buku.

```
$data['laporan'] = $this->db->query("select * from pinjam p,detail_pinjam d,
    buku b,user u where d.id_buku=b.id and p.id_user=u.id
    and p.no_pinjam=d.no_pinjam")>result_array();
```

berdasarkan script diatas, kita akan membuat sebuah view baru dalam folder pinjam.

Dengan nama **laporan-pinjam.php**

**Application/views/pinjam/laporan-pinjam.php**

```
<!-- Begin Page Content -->
<div class="container-fluid">

    <?=$this->session->flashdata('pesan'); ?>
    <div class="row">
        <div class="col-lg-12">
            <?php if(validation_errors()){?>
                <div class="alert alert-danger" role="alert">
                    <?=$validation_errors();?>
                </div>
            <?php }?>
            <?=$this->session->flashdata('pesan'); ?>
            <a href="<?=$base_url('laporan/cetak_laporan_pinjam'); ?>" class="btn btn-primary mb-3"><i class="fas fa-print"></i> </a>
            <a href="<?=$base_url('laporan/laporan_pinjam_pdf'); ?>" class="btn btn-warning mb-3"><i class="far fa-file-pdf"></i> </a>
            <a href="<?=$base_url('laporan/export_excel_pinjam'); ?>" class="btn btn-success mb-3"><i class="far fa-file-excel"></i> </a>
            <table class="table table-hover">
                <thead>
                    <tr>
                        <th scope="col">#</th>
                        <th scope="col">Nama Anggota</th>
                        <th scope="col">Judul Buku</th>
                        <th scope="col">Tanggal Pinjam</th>
                        <th scope="col">Tanggal Kembali </th>
                        <th scope="col">Tanggal Dikembalikan</th>
                        <th scope="col">Total Denda</th>
                        <th scope="col">Status</th>
                    </tr>
                </thead>
            </table>
        </div>
    </div>
</div>
```

```

        </tr>
    </thead>
    <tbody>

        <?php
            $a = 1;
            foreach ($laporan as $l) { ?>
                <tr>
                    <th scope="row"><?= $a++; ?></th>
                    <td><?= $l['nama']; ?></td>
                    <td><?= $l['judul_buku']; ?></td>
                    <td><?= $l['tgl_pinjam']; ?></td>
                    <td><?= $l['tgl_kembali']; ?></td>
                    <td><?= $l['tgl_pengembalian']; ?></td>
                    <td><?= $l['total_denda']; ?></td>
                    <td><?= $l['status']; ?></td>
                </tr>
                <?php } ?>
            </tbody>
        </table>
    </div>
</div>
</div>
<!-- /.container-fluid -->
</div>
<!-- End of Main Content -->

```

Tetap ada 3 aksi yang bisa kita pilih untuk laporan pinjam, yaitu print, cetak pdf, dan ekspor ke excel

```

<a href="<?= base_url('laporan/cetak_laporan_pinjam'); ?>" class="btn btn-
primary mb-3"><i class="fas fa-print"></i> </a>
    <a href="<?= base_url('laporan/laporan_pinjam_pdf'); ?>" class="btn b
tn-warning mb-3"><i class="far fa-file-pdf"></i> </a>
    <a href="<?= base_url('laporan/export_excel_pinjam'); ?>" class="btn
btn-success mb-3"><i class="far fa-file-excel"></i> </a>

```

```





<td><?= $l['nama']; ?></td>
<td><?= $l['judul_buku']; ?></td>
<td><?= $l['tgl_pinjam']; ?></td>
<td><?= $l['tgl_kembali']; ?></td>
<td><?= $l['tgl_pengembalian']; ?></td>
<td><?= $l['total_denda']; ?></td>
<td><?= $l['status']; ?></td>

```

Data yang dikirim adalah kombinasi antara tabel pinjam, detail pinjam, user, dan buku.

Sekarang kita coba klik pada menu “laporan Transaksi”

Maka akan di arahkan ke fungsi laporan\_pinjam() yang menampilkan data peminjaman buku.

Laporan Data Peminjaman							Rani Oktariani Nur Zain 
  							
#	Nama Anggota	Judul Buku	Tanggal Pinjam	Tanggal Kembali	Tanggal Dikembalikan	Total Denda	Status
1	Rani Oktariani Nur Zain	Morfologi Bahasa Indonesia	2023-05-29	2023-06-01	2024-01-19	0	Kembali

### C. Membuat Print Laporan Pinjam

Perhatikan pada view laporan-pinjam.php, pada icon print akan diarahkan ke controllerLaporan.php dan fungsi cetak\_laporan\_pinjam.

```
<a href="<?= base_url('laporan/cetak_laporan_pinjam'); ?>" class="btn btn-primary mb-3"><i class="fas fa-print"></i> </a>
```

```
<a href="<?= base_url('laporan/laporan_pinjam_pdf'); ?>" class="btn btn-warning mb-3"><i class="far fa-file-pdf"></i> </a>
```

```
<a href="<?= base_url('laporan/export_excel_pinjam'); ?>" class="btn btn-info mb-3"><i class="fas fa-file-excel"></i> </a>
```

Oleh karena itu kita harus menambahkan sebuah fungsi baru pada controller Laporan.php dengan nama cetak\_laporan\_pinjam().

```
public function cetak_laporan_pinjam()
{
    $data['laporan'] = $this->db->query("select * from pinjam p,detail_pinjam d,
    buku b,user u where d.id_buku=b.id and p.id_user=u.id
    and p.no_pinjam=d.no_pinjam")->result_array();

    $this->load->view('pinjam/laporan-print-pinjam', $data);
}
```

Fungsi diatas digunakan untuk memanggil data dan mengirimkannya ke view laporanprint-pinjam yang ada di folder pinjam.

Langkah berikutnya kita akan membuat sebuah view dengan nama **laporan-print-pinjam.php** di dalam folder pinjam

#### **Application/views/pinjam/laporan-print-pinjam.php**

```
<!DOCTYPE html>
<html>
<head>
  <title></title>
</head>
<body>
  <style type="text/css">
    .table-data{
      width: 100%;
      border-collapse: collapse;
    }

    .table-data tr th,
    .table-data tr td{ border: 1px
      solid black;font-size: 11pt;
      font-family: Verdana; padding:
      10px 10px 10px 10px;
    }
    .table-data th{ background-
      color: grey;
    }
    h3{
      font-family: Verdana;
    }
  </style>

  <h3><center>LAPORAN DATA PEMINJAMAN BUKU</center></h3>
  <br/>
  <table class="table-data">
    <thead>
      <tr>
        <th>No</th>
        <th>Nama Anggota</th>
        <th>Judul Buku</th>
        <th>Tanggal Pinjam</th>
        <th>Tanggal Kembali</th>
```

```

    </tr>
</thead>
<tbody>
    <?php
    $no = 1;
    foreach($laporan as $l){
    ?>
    <tr>
        <td scope="row"><?= $no++; ?></td>
        <td><?= $l['nama']; ?></td>
        <td><?= $l['judul_buku']; ?></td>
        <td><?= $l['tgl_pinjam']; ?></td>
        <td><?= $l['tgl_kembali']; ?></td>
        <td><?= $l['tgl_pengembalian']; ?></td>
        <td><?= $l['total_denda']; ?></td>
        <td><?= $l['status']; ?></td>
    </tr>
    <?php
    }
    ?>
</tbody>
</table>
<script type="text/javascript">
    window.print();
</script>

</body>
</html>

```

Hasil akhir dari script diatas adalah :

#### LAPORAN DATA PEMINJAMAN BUKU

No	Nama Anggota	Judul Buku	Tanggal Pinjam	Tanggal Kembali	Tanggal Pengembalian	Total Denda	Status
1	Rani Oktariani Nur Zain	Morfologi Bahasa Indonesia	2023-05-29	2023-06-01	2024-01-19	0	Kembali
2	Rani Oktariani Nur Zain	Belajar Coding Itu Penting di Era Revolusi Industri 4.0	2023-05-29	2023-06-01	2024-01-19	0	Kembali
3	Rani Oktariani Nur Zain	Buku Dasar-Dasar Teknik Informatika	2024-01-21	2024-01-24	2024-01-21	0	Kembali

#### D. Membuat Cetak Pdf Laporan Peminjaman

Untuk membuat laporan peminjaman cetak pdf kita buka kembali view laporan-pinjam.php untuk melihat controller dan fungsi apa yang dipanggil.

```
<a href="= base_url('laporan/cetak_laporan_pinjam'); ?" class="btn btn-primary mb-3"><i class="fas fa-print"></i> </a>  
<a href="= base_url('laporan/laporan_pinjam_pdf'); ?" class="btn btn-warning mb-3"><i class="far fa-file-pdf"></i> </a>
```

Dari script view laporan-pinjam.php dapat kita lihat bahwa sistem mengarahkan ke controller Laporan.php dan memanggil fungsi laporan\_pinjam\_pdf. Oleh karena itu langkah berikutnya kita akan menambahkan fungsi baru pada controller **Laporan.php** yang diberi nama **laporan\_pinjam\_pdf()**

Berikut script fungsi laporan\_pinjam\_pdf()

```
public function laporan_pinjam_pdf()  
{  
    $data['laporan'] = $this->db->query("select * from pinjam p,detail_pinjam  
d,  
    buku b,user u where d.id_buku=b.id and p.id_user=u.id and  
    p.no_pinjam=d.no_pinjam")->result_array();  
    // $this->load->library('dompdf_gen');  
    $sroot = $_SERVER['DOCUMENT_ROOT'];  
    include $sroot . "/pustaka-  
booking/application/third_party/dompdf/autoload.inc.php";  
    $dompdf = new Dompdf\Dompdf();  
    $this->load->view('pinjam/laporan_pdf_pinjam', $data);  
    $paper_size = 'A4'; // ukuran kertas  
    $orientation = 'landscape'; //tipe format kertas potrait atau landscape  
    $html = $this->output->get_output();  
  
    $dompdf->set_paper($paper_size, $orientation);  
    //Convert to PDF  
    $dompdf->load_html($html);  
    $dompdf->render();  
    $dompdf->stream("laporan data peminjaman.pdf", array('Attachment' => 0));  
}
```



Fungsi `laporan_pinjam_pdf()` berisi script untuk mengatur tampilan file pdf (orientation, size, name, dll) sedangkan isi dari file pdf sistem akan memanggil view `laporan-pdf-pinjam.php` yang ada di folder `pinjam`.

Jadi langkah berikutnya adalah membuat view **laporan-pdf-pinjam.php** untuk menampilkan data dari laporan.

#### Application/views/pinjam/laporan-pdf-pinjam.php

```
<!DOCTYPE html>
<html>
<head>
  <title></title>
</head>
<body>
  <style type="text/css">
    .table-data{
      width: 100%;
      border-collapse: collapse;
    }

    .table-data tr th,
    .table-data tr td{
      border:1px solid black;
      font-size: 11pt;
      font-family:Verdana;
      padding: 10px 10px 10px 10px;
    }
    .table-data th{
      background-color:grey;
    }
    h3{
      font-family:Verdana;
    }
  </style>

  <h3><center>LAPORAN DATA PEMINJAMAN BUKU</center></h3>
  <br/>
  <table class="table-data">
    <thead>
      <tr>
        <th>No</th>
        <th>Nama Anggota</th>
        <th>Judul Buku</th>
```

```

        <th>Tanggal Pinjam</th>
        <th>Tanggal Kembali</th>
        <th>Tanggal Pengembalian</th>
        <th>Total Denda</th>
        <th>Status</th>
    </tr>
</thead>
<tbody>
    <?php
    $no = 1;
    foreach($laporan as $l){
    ?>
    <tr>
        <td scope="row"><?= $no++; ?></td>
        <td><?= $l['nama']; ?></td>
        <td><?= $l['judul_buku']; ?></td>
        <td><?= $l['tgl_pinjam']; ?></td>
        <td><?= $l['tgl_kembali']; ?></td>
        <td><?= $l['tgl_pengembalian']; ?></td>
        <td><?= $l['total_denda']; ?></td>
        <td><?= $l['status']; ?></td>
    </tr>
    <?php
    }
    ?>
</tbody>
</table>

</body>
</html>

```

Hasil dari file pdf yang telah didownload:

#### LAPORAN DATA PEMINJAMAN BUKU

No	Nama Anggota	Judul Buku	Tanggal Pinjam	Tanggal Kembali	Tanggal Pengembalian	Total Denda	Status
1	Rani Oktariani Nur Zain	Morfologi Bahasa Indonesia	2023-05-29	2023-06-01	2024-01-19	0	Kembali
2	Rani Oktariani Nur Zain	Belajar Coding Itu Penting di Era Revolusi Industri 4.0	2023-05-29	2023-06-01	2024-01-19	0	Kembali
3	Rani Oktariani Nur Zain	Buku Dasar-Dasar Teknik Informatika	2024-01-21	2024-01-24	2024-01-21	0	Kembali

### E. Mengexport Laporan Peminjaman ke Bentuk Excel

Sama seperti cetak laporan sebelumnya, kita lihat kembali view laporan-pinjam.php untuk melihat controller yang dipanggil

```
<a href="<?= base_url('laporan/cetak_laporan_pinjam'); ?>" class="btn btn-primary mb-3"><i class="fas fa-print"></i> </a>
```

```
<a href="<?= base_url('laporan/laporan_pinjam_pdf'); ?>" class="btn btn-warning mb-3"><i class="far fa-file-pdf"></i> </a>
```

```
<a href="<?= base_url('laporan/export_excel_pinjam'); ?>" class="btn btn-success mb-3"><i class="far fa-file-excel"></i> </a>
```

Untuk proses export data menjadi excel fungsi yang digunakan adalah `export_excel_pinjam` yang masih berada di controller `Laporan.php`

Berikut script fungsi `export_excel_pinjam()`

```
public function export_excel_pinjam()
{
    $data = array( 'title' => 'Laporan Data Peminjaman Buku',
    'laporan' => $this->db->query("select * from pinjam p,detail_pinjam d,
    buku b,user u where d.id_buku=b.id and p.id_user=u.id
    and p.no_pinjam=d.no_pinjam")->result_array());
    $this->load->view('pinjam/export-excel-pinjam', $data);
}
```

Pada script fungsi `export_excel_pinjam()` diatur judul untuk laporan yang dibuat, serta

mengirim nilai dari tabel yang terlibat didalamnya. Data-data tersebut akan dikirim kedalam file view export-excel-pinjam.php

Langkah terakhir dari pembuatan export data ke excel adalah membuat view dengan nama export-excel-pinjam.php yang ada di folder pinjam

#### **Application/views/pinjam/export-excel-pinjam.php**

```
<?php
header("Content-type: application/vnd-ms-excel");
header("Content-Disposition: attachment; filename=$title.xls");
header("Pragma: no-cache");
header("Expires: 0");
?>
<style type="text/css">
.table-data{
    width: 100%;
    border-collapse: collapse;
}

.table-data tr th,
.table-data tr td{
    border:1px solid black;
```

```
font-size: 11pt; font-  
family: Verdana;  
padding: 10px 10px 10px 10px;  
}  
.table-data th{ background-  
color: grey;  
}  
h3{  
font-family: Verdana;  
}  
</style>
```

```
<h3><center>LAPORAN DATA PEMINJAMAN BUKU</center></h3>
```

```
<br/>
```

```
<table class="table-data" border=1>
```

```
<thead>
```

```
<tr>
```

```
<th>No</th>
```

```
<th>Nama Anggota</th>
```

```
<th>Judul Buku</th>
```

```
<th>Tanggal Pinjam</th>
```

```
<th>Tanggal Kembali</th>
```

```
<th>Tanggal Pengembalian</th>
```

```
<th>Total Denda</th>
```

```
<th>Status</th>
```

```
</tr>
```

```
</thead>
```

```
<tbody>
```

```
<?php
```

```
$no = 1; foreach($laporan as
```

```
$l){
```

```
?>
```

```
<tr>
```

```
<td scope="row"><?= $no++; ?></td>
```

```
<td><?= $l['nama']; ?></td>
```

```
<td><?= $l['judul_buku']; ?></td>
```

```
<td><?= $l['tgl_pinjam']; ?></td>
```

```
<td><?= $l['tgl_kembali']; ?></td>
```

```
<td><?= $l['tgl_pengembalian']; ?></td>
```

```
?>
</tbody>
</table>
```

Hasil akhir dari file yang telah diubah kedalam bentuk excel adalah :

A1		:	✕	✓	<i>fx</i>	LAPORAN DATA PEMINJAMAN BUKU		
	A	B	C	D	E	F	G	H
1	LAPORAN DATA PEMINJAMAN BUKU							
2								
3								
4	No	Nama Anggota	Judul Buku	Tanggal Pinjam	Tanggal Kembali	Tanggal Pengembalian	Total Denda	Status
5	1	Rani Oktariani Nur Zain	Morfologi Bahasa Indonesia	29/05/2023	01/06/2023	19/01/2024	0	Kembali
6	2	Rani Oktariani Nur Zain	Belajar Coding Itu Penting di Era Revolusi Industri 4.0	29/05/2023	01/06/2023	19/01/2024	0	Kembali
7	3	Rani Oktariani Nur Zain	Buku Dasar-Dasar Teknik Informatika	21/01/2024	24/01/2024	21/01/2024	0	Kembali
8								