

Edit Data (Update) dan Hapus Data (Delete)



Edit Data (Update) dan Hapus Data (Delete)

Pada praktikum kali ini kita akan membuat fitur **Edit (Update)** serta **Hapus (Delete) Data yang ada di database** melalui **form modal Bootstrap**. Fitur ini ditempatkan pada **Project Laravel 11 yang telah dibuat pada praktikum sebelumnya**. Database yang digunakan adalah database MySQL **tokobuku** yang digunakan pada praktikum sebelumnya. Skenarionya antara lain:

- Mekanisme **edit** dipicu oleh tombol **update** yang **ada ditiap data**. Ketika **tombol update ditekan**, maka sistem akan menampilkan keseluruhan detail data dari data yang dipilih dan ditampilkan ke dalam form dalam bentuk modal.
- Untuk mekanisme **hapus** dipicu oleh tombol **delete** yang **ada ditiap data**. Ketika **tombol delete ditekan**, maka sistem akan menampilkan semacam notifikasi berupa konfirmasi untuk menanyakan apakah yakin ingin menghapus atau tidak.

Untuk merealisasikan hal tersebut, maka kita membutuhkan beberapa **pengaturan dasar**, **pengaturan routing**, **function controller** serta **function model**. Project yang digunakan adalah **lav11**. Setelah memahami skenario yang akan dilakukan, maka langkah-langkahnya sebagai berikut

1. Sebagai langkah awal kita harus **membuat form update berbentuk modal**. Form ini ditempatkan pada view **buku.blade.php**. Syntax untuk modal ini diletakkan setelah blok `<div id="modaltambah" class="modal fade" role="dialog"></div>`. Syntaxnya sebagai berikut

```
<div id="modalupdate" class="modal fade" role="dialog">
  <div class="modal-dialog modal-lg">
    <div class="modal-content">
      <div class="modal-header">
        <h4 class="modal-title">Form Update Data</h4>
      </div>
      <div class="modal-body">
        <div class="row">
          <input type="hidden" id="txtkodee" readonly>
          <div class="form-group col-4">
            <label>Tahun Terbit*</label>
            <input type="text" class="form-control forme" id="txttahune" placeholder="Masukkan Tahun Terbit" maxlength="4"
              " autocomplete="off">
          </div>
          <div class="form-group col-4">
            <label>ISBN*</label>
            <input type="text" class="form-control forme" id="txtisbne" placeholder="Masukkan ISBN" autocomplete="off">
          </div>
          <div class="form-group col-9">
            <label>Judul Buku*</label>
            <input type="text" class="form-control forme" id="txtjudule" placeholder="Masukkan Judul Buku" autocomplete="
              off">
          </div>
          <div class="form-group col-3">
            <label>Rak*</label>
            <input type="text" class="form-control forme" id="txtrake" placeholder="Masukkan Rak" maxlength="3"
              autocomplete="off">
          </div>
        </div>
      </div>
    </div>
  </div>
</div>
```

```

<div class="form-group col-6">
    <label>Pengarang*</label>
    <select class="form-control forme" id="cbopengarange" style="width: 100%;">
        <option value=''>-- Pilih Salah Satu --</option>
        @php
            if(is_array($dtpn)){
                if(count($dtpn)>0){
                    foreach($dtpn as $k){
                        $id = $k->ID_Pengarang;
                        $nama = $k->Nama_Pengarang;
                        echo "<option value='$id'>$nama</option>";
                    }
                }
            }
        @endphp
    </select>
</div>
<div class="form-group col-6">
    <label>Penerbit*</label>
    <select class="form-control forme" id="cbopenerbite" style="width: 100%;">
        <option value=''>-- Pilih Salah Satu --</option>
        @php
            if(is_array($dtpr)){
                if(count($dtpr)>0){
                    foreach($dtpr as $k){
                        $id = $k->ID_Penerbit;
                        $nama = $k->Nama_Penerbit;
                        echo "<option value='$id'>$nama</option>";
                    }
                }
            }
        @endphp
    </select>
</div>
</div>
<div class="modal-footer">
    <button type="button" class="btn btn-secondary waves-effect waves-light" data-dismiss="modal">Batal</button>
    <button type="button" class="btn btn-primary waves-effect waves-light" onclick="update()">Simpan Perubahan</button>
</div>
</div>
</div>

```

- Langkah berikutnya adalah **membuat pemacu untuk menjalankan function yang berfungsi menampilkan data pada form modal update**. Untuk merealisasikannya, kita harus **memodifikasi variabel tombolkelola yang ada didalam function ambilBuku pada controller Utama.php**. Ubah variabel tombolkelola dari

```
$tombolkelola = "";
```

Menjadi

```
$tombolkelola = "<button type='button' class='btn btn-icon btn-round btn-primary btn-sm mr-1 mb-1' data-kode='".$kode."' onclick='filter(this)'><i class='icon-pencil'></i></button> <button type='button' class='btn btn-icon btn-round btn-danger btn-sm mr-1 mb-1' data-kode='".$kode."' onclick='hapus(this)'><i class='icon-trash'></i></button>";
```

Jika menu buku dijalankan, maka tampilan jika berhasil seperti gambar berikut (terdapat 2 buah tombol di masing-masing data)

Atlantis





















Data Buku

Halaman Pengelolaan Buku

Tambah Refresh Data

Show 10 entries

Search:

Operasi	Kode	Judul	Pengarang	Penerbit	Tahun	ISBN
 	165311	Membangun Aplikasi Handphone dengan FBUS & Visual Basic	X-Oerang	Andi Publisher	2004	979-731-075-2
 	165312	Pemrograman Visual Basic 6.0	X-Oerang	Andi Publisher	2005	979-731-828-1
 	165313	Membuat Menu Cantik Untuk Aplikasi Visual Basic 6.0	Ridwan Sanjaya, SE, S.Kom	Andi Publisher	2005	979-731-647-5
 	165314	Pemrograman Tingkat Lanjut dengan Visual Basic 6.0 dan Crystal Report	Madcoms	Andi Publisher	2005	979-763-031-5
 	165315	Visual Basic .NET 2005	Widodo Budiharto, M.Kom	Andi Publisher	2006	979-763-380-2
 	165316	Panduan Pemrograman dan Referensi Kamus Visual Basic 6.0	Madcoms	Andi Publisher	2006	979-763-336-5
 	165317	Kumpulan Tip Trik Pemrograman Visual Basic	Jaja Jamaludin Malik	Andi Publisher	2007	979-763-654-2
 	165318	Mudah & Cepat Menguasai Visual Basic	Ir. Suryanto Thabrani, MM	Mediakita	2008	978-979-794-067-6
 	165319	Pemrograman Web Dinamis dengan Kolaborasi PHP & JAVA	Afriyudi, M.Kom	Andi Publisher	2008	978-979-29-0262-4
 	165320	Mengotak-Atik Windows dengan VB Script	VygoriViva CR	Gaya Media	2008	978-979-1078-34-4

3. Langkah berikutnya adalah **membuat function bernama filterData**. Function ini ditempatkan sejajar dengan function lain di model **Mdata**. Function ini digunakan untuk **mendapatkan data yang akan diedit nantinya**. Syntax-nya sebagai berikut

```
public function filterData($kode){  
    $dt = DB::select("SELECT * FROM buku WHERE Kode_Buku = '$kode'");  
    return $dt;  
}
```

4. Langkah berikutnya adalah **membuat function yang bernama filterBuku**. Function ini ditempatkan di controller **Utama.php** sejajar dengan function lainnya. Function ini berguna untuk **mengatur data yang dihasilkan oleh function filterData yang ada di model Mdata**. Syntax-nya sebagai berikut

```
public function filterBuku(Request $z){  
    $status = false;  
    $kode = $z->input("kode");  
    $mx = new Mdata();  
    $dt = $mx->filterData($kode);  
    if(is_array($dt)){  
        if(count($dt)>0){  
            $status = true;  
            foreach($dt as $x){  
                $judul = $x->Judul;  
                $idpenerbit = $x->ID_Penerbit;  
                $idpengarang = $x->ID_Pengarang;  
                $isbn = $x->ISBN;  
                $tahunterbit = $x->Tahun_Terbit;  
                $rak = $x->Rak;  
            }  
        }  
    }  
    if($status){  
        return sprintf('{"kode":"1","pesan":"Data Tersedia","judul":"%s","id_penerbit":"%s",  
            "id_pengarang":"%s","isbn":"%s","tahun_terbit":"%s","rak":"%s"}', $judul, $idpenerbit, $  
            idpengarang, $isbn, $tahunterbit, $rak);  
    }else{  
        return '{"kode":"0","pesan":"Data Tidak di Temukan"}';  
    }  
}
```

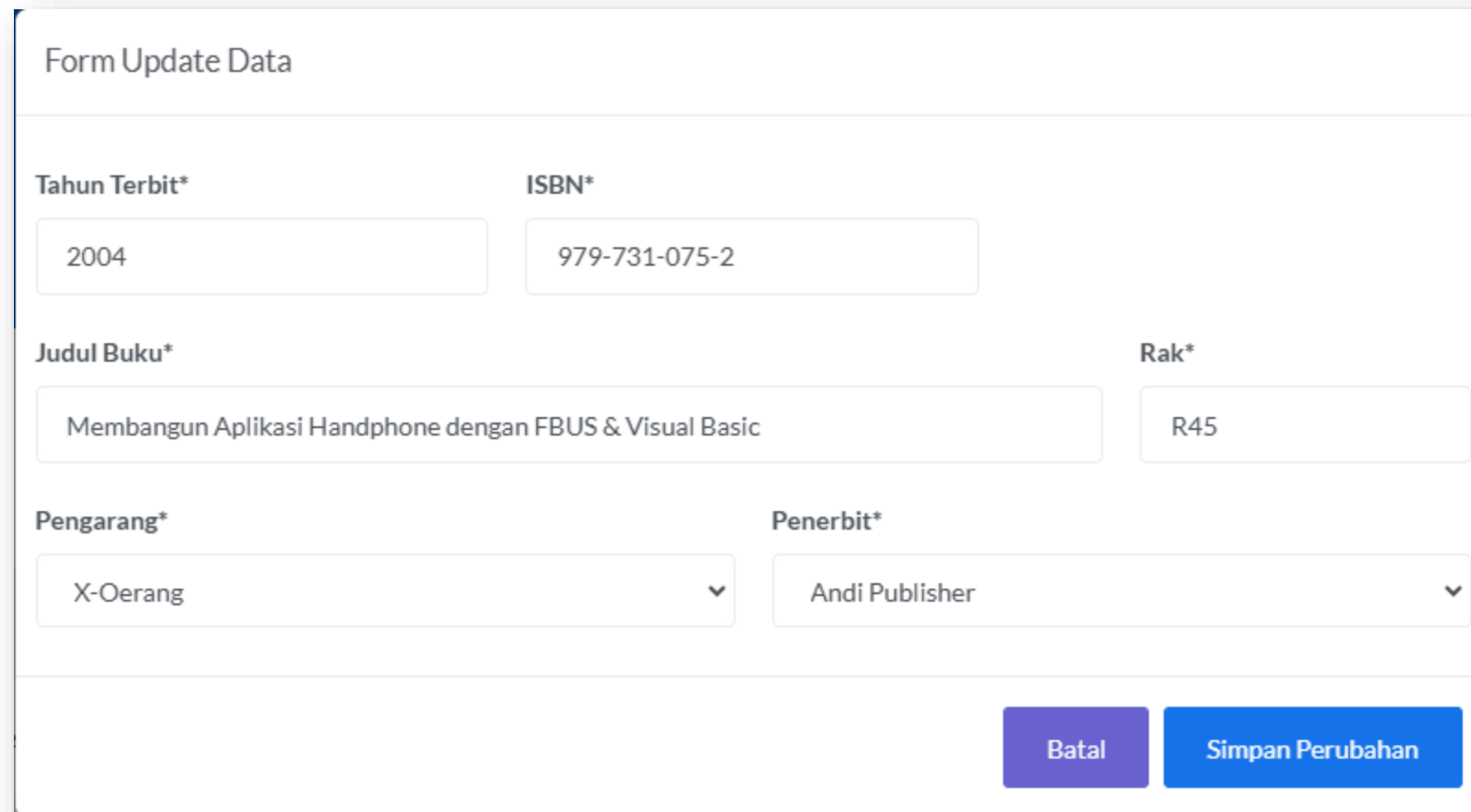

5. Langkah berikutnya adalah **membuat pengaturan routing untuk mengakses function filterBuku yang ada di controller Utama.php**. Tambahkan pengaturan routing dibawah ini serta tempatkan sejajar dengan pengaturan lainnya di file **web.php**

```
Route::post('/filterbuku', [Utama::class, 'filterBuku']);
```

6. Langkah berikutnya adalah **membuat function yang bernama filter**. Function ini ditempatkan di view **buku.blade.php** sejajar dengan function javascript lainnya. Function ini berguna untuk **memfilter data buku yang akan diedit untuk dimunculkan ke form edit berbentuk modal**. Syntax-nya sebagai berikut

```
function filter(e1){
    let kode = $(e1).data("kode");
    $.ajax({
        url: "{{ URL::to('/filterbuku') }}",
        method: "POST",
        data: {"_token": "{{ csrf_token() }}", kode: kode},
        cache: "false",
        success: function(respon){
            let data = JSON.parse(respon);
            if(data.kode == "1"){
                $("#txtkodee").val(kode);
                $("#txtjudule").val(data.judul);
                $("#cbopengarang").val(data.id_pengarang).change();
                $("#cbopenerbite").val(data.id_penerbit).change();
                $("#txtisbne").val(data.isbn);
                $("#txttahune").val(data.tahun_terbit);
                $("#txtrake").val(data.rak);
                $("#modalupdate").modal({backdrop: "static", keyboard: false});
            }else{
                swal({title: "Gagal", text: data.pesan, icon: "error"});
            }
        },
        error: function(){
            swal({title: 'Gagal', text: 'Mekanisme Filter Ada Yang Bermasalah', icon: 'error'});
        }
    });
}
```

Jika benar, maka saat tombol **update** ditekan akan **memunculkan form modal ditengah layar beserta detail data yang dipilih** seperti gambar dibawah ini

A screenshot of a 'Form Update Data' modal. The form has a title bar 'Form Update Data'. It contains several input fields: 'Tahun Terbit*' with the value '2004', 'ISBN*' with the value '979-731-075-2', 'Judul Buku*' with the value 'Membangun Aplikasi Handphone dengan FBUS & Visual Basic', and 'Rak*' with the value 'R45'. There are two dropdown menus: 'Pengarang*' with the selected value 'X-Oerang' and 'Penerbit*' with the selected value 'Andi Publisher'. At the bottom right, there are two buttons: 'Batal' (purple) and 'Simpan Perubahan' (blue).

Form Update Data	
Tahun Terbit*	ISBN*
2004	979-731-075-2
Judul Buku*	Rak*
Membangun Aplikasi Handphone dengan FBUS & Visual Basic	R45
Pengarang*	Penerbit*
X-Oerang	Andi Publisher
<div>Batal Simpan Perubahan</div>	

7. Langkah berikutnya adalah **membuat function bernama updateData**. Function ini ditempatkan sejajar dengan function lain di model **Mdata**. Function ini digunakan untuk **menyimpan data (update) yang telah diedit**. Syntax-nya sebagai berikut

```
public function updateData($kode, $judul, $pengarang, $penerbit, $isbn, $tahun, $rak){  
    $dt = DB::update("UPDATE buku SET Judul='$judul', ID_Pengarang='$pengarang', ID_Penerbit='$penerbit', ISBN='$isbn',  
        Tahun_Terbit='$tahun', Rak='$rak' WHERE Kode_Buku='$kode'");  
    return $dt;  
}
```

8. Langkah berikutnya adalah **membuat function yang bernama updateBuku**. Function ini ditempatkan di controller **Utama.php** sejajar dengan function lainnya. Function ini berguna untuk **menerima data yang dikirimkan oleh view pada saat proses update data**. Data tersebut kemudian dikirimkan ke function **updateData** yang ada di model **Mdata** untuk disimpan kembali ke database. Syntax-nya sebagai berikut

```

public function updateBuku(Request $z){
    $kode = $z->input("kode");
    $judul = $z->input("judul");
    $pengarang = $z->input("pengarang");
    $penerbit = $z->input("penerbit");
    $tahun = $z->input("tahun");
    $rak = $z->input("rak");
    $isbn = $z->input("isbn");
    $mx = new Mdata();
    $proses = $mx->updateData($kode, $judul, $pengarang, $penerbit, $isbn, $tahun, $rak);
    if($proses == "1"){
        return '{"kode":"1","pesan":"Data Berhasil di Update"}';
    }else{
        return '{"kode":"0","pesan":"Data Gagal di Update, Periksa Kembali Isian Anda"}';
    }
}

```

9. Langkah berikutnya adalah **membuat pengaturan routing untuk mengakses function updateBuku yang ada di controller Utama.php**. Tambahkan pengaturan routing berikut sejajar dengan pengaturan lainnya di file **web.php**

```

Route::post('/updatebuku', [Utama::class, 'updateBuku']);

```

10. Langkah berikutnya adalah **membuat function yang bernama update**. Function ini ditempatkan di view **buku.blade.php** sejajar dengan function javascript lainnya. Function ini berguna untuk **menyimpan data buku yang telah diedit**. Syntax-nya sebagai berikut


```

function update(){
    let kode = $("#txtkodee").val();
    let judul = $("#txtjudule").val();
    let pengarang = $("#cbopengarange").val();
    let penerbit = $("#cbopenerbite").val();
    let isbn = $("#txtisbne").val();
    let tahun = $("#txttahune").val();
    let rak = $("#txtrake").val();
    if(kode == "" || judul == "" || isbn == "" || pengarang == "" || penerbit == "" || tahun == "" || rak == ""){
        swal({title: "Gagal", text: "Ada Isian Yang Masih Kosong", icon: "error"});
        return;
    }
    $.ajax({
        url: "{ URL::to('/updatebuku'); }",
        method: "POST",
        data: {"_token": "{ csrf_token() }", kode: kode, judul: judul, isbn: isbn, pengarang: pengarang, penerbit: penerbit, tahun: tahun, rak: rak},
        cache: "false",
        success: function(response){
            let data = JSON.parse(response);
            if(data.kode == "1"){
                swal({title: "Berhasil", text: data.pesan, icon: "success"});
                $(".forme").val("");
                tabel.ajax.reload(null, false);
                $("#modalupdate").modal("hide");
            }else{
                swal({title: "Gagal", text: data.pesan, icon: "error"});
            }
        },
        error: function(){
            swal({title: "Gagal", text: "Mekanisme Update Ada Yang Bermasalah", icon: "error"});
        }
    })
}

```

11. Langkah berikutnya adalah **mencoba untuk mekanisme update data**. Pilih salah satu data kemudian edit salah satu item dari data tersebut kemudian tekan tombol **Simpan Perubahan**. Apakah data sudah berubah? Jika sudah maka mekanisme sudah sesuai. Contoh gambar dibawah ini

Atlantis

Data Buku

Halaman Pengelolaan Buku

TambahRefresh Data

Search:

Operasi

↑↓

Kode

↑↓

Judul

↑↓

Tahun

↑↓

ISBN

↑↓

165342

Mudah Membuat Aplikasi dengan CodeIgniter

165343

From Zero to a Professional PHP Developer

165326

Learning By Sample Visual Basic 2008

Soetam Rizky

Prestasi Pustaka

165340

Aplikasi Bisnis dengan PHP dan MySQL

Muhammad Sadeli

Maxikom

999999

Augmented Reality for Dummy

VygoriViva CR

Lokomedia

Showing 31 to 35 of 35 entries

Previous

1

2

3

4

Next

Form Update Data

Tahun Terbit*

2024

ISBN*

979-777-233-8

Judul Buku*

Augmented Reality for Dummy Rev 1

Rak*

G59

Pengarang*

VygoriViva CR

Penerbit*

Lokomedia

Batal

Simpan Perubahan

Atlantis

Data Buku











Halaman Pengelolaan Buku

Tambah

Refresh Data

Show 10 entries

Search:

Operasi	Kode	Judul	Pengarang	Penerbit	Tahun	ISBN
 	165342	Mudah Membuat Aplikasi SMS Gateway dengan CodeIgniter	Wahana Komputer	Elex Media Komputindo	2014	978-602-02-4272-9
 	165343	From Zero to a Pro JavaScript & jQuery	Abdul Kadir	Andi Publisher	2014	978-979-29-3987-3
 	165326	Learning By Sample Visual Basic 2008	Soetam Rizky	Prestasi Pustaka	2020	979-602-8470-64-3
 	165340	Aplikasi Bisnis dengan PHP dan MySQL	Muhammad Sadeli	Maxikom	2020	978-602-7680-69-2
 	999999	Augmented Reality for Dummy Rev 1	VygoriViva CR	Lokomedia	2024	979-777-233-8

Showing 31 to 35 of 35 entries

Previous

1

2

3

4

Next

12. Langkah berikutnya adalah **membuat function bernama deleteData**. Function ini ditempatkan sejajar dengan function lain di model **Mdata**. Function ini digunakan untuk **menghapus data (delete) yang ada di database**. Syntax-nya sebagai berikut

```
public function deleteData($kode){
    $dt = DB::delete("DELETE FROM buku WHERE Kode_Buku='$kode'");
    return $dt;
}
```

13. Langkah berikutnya adalah **membuat function yang bernama deleteBuku**. Function ini ditempatkan di controller **Utama.php** sejajar dengan function lainnya. Function ini berguna untuk **memerintahkan function deleteData yang ada di model Mdata.php untuk menghapus data yang ada di database**. Syntax-nya sebagai berikut

```

public function deleteBuku(Request $z){
    $kode = $z->input("kode");
    $mx = new Mdata();
    $proses = $mx->deleteData($kode);
    if($proses == "1"){
        return '{"kode":"1","pesan":"Data Berhasil di Hapus"}';
    }else{
        return '{"kode":"0","pesan":"Data Gagal di Hapus, Periksa Kembali Isian Anda"}';
    }
}

```

14. Langkah berikutnya adalah **membuat pengaturan routing untuk mengakses function deleteBuku yang ada di controller Utama.php**. Tambahkan pengaturan routing berikut sejajar dengan pengaturan lainnya di file **web.php**

```

Route::post('/deletebuku', [Utama::class, 'deleteBuku']);

```

15. Langkah berikutnya adalah **membuat function yang bernama delete**. Function ini ditempatkan di view **buku.blade.php** sejajar dengan function javascript lainnya. Function ini berguna untuk **menghapus data buku yang telah dipilih**. Syntax-nya sebagai berikut

```

function hapus(el){
  let kode = $(el).data("kode");
  swal({
    title: 'Hapus Buku',
    text: "Anda Yakin Ingin Menghapus Buku Ini ?",
    icon: 'warning',
    buttons:{
      confirm: {text : 'Yakin', className : 'btn btn-success'},
      cancel: {visible: true, text: 'Tidak', className: 'btn btn-danger'}
    }
  }).then((Hapuss)=>{
    if(Hapuss){
      $.ajax({
        url: "{{ URL::to('/deletebuku') }}",
        method: "POST",
        data: {"_token": "{{ csrf_token() }}", kode: kode},
        cache: "false",
        success: function(respon){
          let data = JSON.parse(respon);
          if(data.kode == "1"){
            swal({title: "Berhasil", text: data.pesan, icon: "success"});
            tabel.ajax.reload(null, false);
          }else{
            swal({title: "Gagal", text: data.pesan, icon: "error"});
          }
        },
        error: function(){
          swal.close();
          swal({title: 'Gagal', text: 'Mekanisme Hapus Ada Yang Bermasalah', icon: 'error'});
        }
      });
    }
  });
}

```

16. Langkah berikutnya adalah **mencoba untuk mekanisme delete data**. Pilih salah satu data kemudian **tekan tombol hapus**. Kemudian harusnya akan **muncul semacam konfirmasi apakah yakin ingin dihapus atau tidak**. Jika **ditekan yakin maka data akan menghapus dan sebaliknya**. Apakah mekanismenya sudah sesuai? Jika sudah maka konsep sudah sesuai. Contoh gambar dibawah ini

Atlantis

Data Buku

Halaman Pengelolaan Buku

TambahRefresh Data

Show 10 entries

Search:

Operasi	Kode	Judul	Pengarang	Penerbit	Tahun	ISBN
<div><div></div><div></div></div>	165342	Mudah Membuat Aplikasi SMS Gateway dengan CodeIgniter	2014	978-602-02-4272-9
<div><div></div><div></div></div>	165343	From Zero to a Pro JavaScript & JQuery	2014	978-979-29-3987-3
<div><div></div><div></div></div>	165326	Learning By Sample Visual Basic 2008	2020	979-602-8470-64-3
<div><div></div><div></div></div>	165340	Aplikasi Bisnis dengan PHP dan MySQL	2020	978-602-7680-69-2
<div><div></div><div></div></div>	999999	Augmented Reality for Dummy Rev 1	2024	979-777-233-8

Showing 31 to 35 of 35 entries

Previous1234Next

Hapus Buku

Anda Yakin Ingin Menghapus Buku Ini ?

YakinTidak

Atlantis

Data Buku

Halaman Pengelolaan Buku

TambahRefresh Data

Show 10 entries

Search:

Operasi	Kode	Judul	Pengarang	Penerbit	Tahun	ISBN
<div><div></div><div></div></div>	165342	Mudah Membuat Aplikasi SMS Gateway dengan CodeIgniter	2014	978-602-02-4272-9
<div><div></div><div></div></div>	165343	From Zero to a Pro JavaScript & JQuery	2014	978-979-29-3987-3
<div><div></div><div></div></div>	165326	Learning By Sample Visual Basic 2008	2020	979-602-8470-64-3
<div><div></div><div></div></div>	165340	Aplikasi Bisnis dengan PHP dan MySQL	2020	978-602-7680-69-2

Showing 31 to 34 of 34 entries

Previous1234Next

Berhasil

Data Berhasil di Hapus

OK

17. Sampai langkah ini, praktikum dianggap berhasil karena sudah bisa untuk mengupdate **data serta menghapus data dengan segala mekanismenya**. Silahkan tanya pengajar jika terdapat masalah atau hal lainnya.