

## LAMPIRAN

### 1. Source Code Data Buku

```
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Linq;
using System.Net;
using System.Web;
using System.Web.Mvc;
using Perpustakaan.Models;

namespace Perpustakaan.Controllers
{
    public class BukuController : Controller
    {
        private perpusEntities1 db = new perpusEntities1();

        // GET: Buku
        public ActionResult Index()
        {
```

```
        var buku = db.Buku.Include(b => b.Buku_Bahasa1).Include(b =>
b.Buku_Kategori1).Include(b => b.Buku_Penerbit1).Include(b =>
b.Buku_Penulis1).Include(b => b.Buku_Rak1);

        return View(buku.ToList());
    }

    public ActionResult loaddata()
    {
        using (perpusEntities1 dc = new perpusEntities1())
        {
            // dc.Configuration.LazyLoadingEnabled = false; // if your table is relational,
contain foreign key

            var data = dc.Buku.OrderBy(a => a.Buku_Judul).ToList();

            return Json(new { data = data }, JsonRequestBehavior.AllowGet);
        }
    }

    // GET: Buku/Details/5
    public ActionResult Details(int? id)
    {
        if (id == null)
        {
            return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
        }

        Buku buku = db.Buku.Find(id);

        if (buku == null)
        {
```

```
        return HttpNotFound();
    }
    return View(buku);
}

// GET: Buku/Create
public ActionResult Create()
{
    ViewBag.Buku_Bahasa = new SelectList(db.Buku_Bahasa, "Bahasa_Id",
    "Bahasa_Nama");
    ViewBag.Buku_Kategori = new SelectList(db.Buku_Kategori, "Katrgori_Id",
    "Kategori_Nama");
    ViewBag.Buku_Penerbit = new SelectList(db.Buku_Penerbit, "Penerbit_Id",
    "Penerbit_Nama");
    ViewBag.Buku_Penulis = new SelectList(db.Buku_Penulis, "Penulis_Id",
    "Penulis_Nama");
    ViewBag.Buku_Rak = new SelectList(db.Buku_Rak, "Rak_Id", "Rak_Nama");
    return View();
}

// POST: Buku/Create
// To protect from overposting attacks, please enable the specific properties you
want to bind to, for
// more details see http://go.microsoft.com/fwlink/?LinkId=317598.
[HttpPost]
[ValidateAntiForgeryToken]
```

```
public ActionResult Create([Bind(Include =
"Buku_Id,Buku_Bcd,Buku_Judul,Buku_Kategori,Buku_Penulis,Buku_Penerbit,Buku_Ra
k,Buku_Bahasa,Buku_Jumlah,Buku_Tahun_Terbit")] Buku buku)

{

    if (ModelState.IsValid)

    {

        db.Buku.Add(buku);

        db.SaveChanges();

        return RedirectToAction("Index");

    }

    ViewBag.Buku_Bahasa = new SelectList(db.Buku_Bahasa, "Bahasa_Id",
    "Bahasa_Nama", buku.Buku_Bahasa);

    ViewBag.Buku_Kategori = new SelectList(db.Buku_Kategori, "Katrgori_Id",
    "Kategori_Nama", buku.Buku_Kategori);

    ViewBag.Buku_Penerbit = new SelectList(db.Buku_Penerbit, "Penerbit_Id",
    "Penerbit_Nama", buku.Buku_Penerbit);

    ViewBag.Buku_Penulis = new SelectList(db.Buku_Penulis, "Penulis_Id",
    "Penulis_Nama", buku.Buku_Penulis);

    ViewBag.Buku_Rak = new SelectList(db.Buku_Rak, "Rak_Id", "Rak_Nama",
    buku.Buku_Rak);

    return View(buku);

}

// GET: Buku/Edit/5

public ActionResult Edit(int? id)

{

    if (id == null)
```

```
{
    return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
}

Buku buku = db.Buku.Find(id);

if (buku == null)
{
    return HttpNotFound();
}

ViewBag.Buku_Bahasa = new SelectList(db.Buku_Bahasa, "Bahasa_Id",
"Bahasa_Nama", buku.Buku_Bahasa);

ViewBag.Buku_Kategori = new SelectList(db.Buku_Kategori, "Katrgori_Id",
"Kategori_Nama", buku.Buku_Kategori);

ViewBag.Buku_Penerbit = new SelectList(db.Buku_Penerbit, "Penerbit_Id",
"Penerbit_Nama", buku.Buku_Penerbit);

ViewBag.Buku_Penulis = new SelectList(db.Buku_Penulis, "Penulis_Id",
"Penulis_Nama", buku.Buku_Penulis);

ViewBag.Buku_Rak = new SelectList(db.Buku_Rak, "Rak_Id", "Rak_Nama",
buku.Buku_Rak);

return View(buku);
}

// POST: Buku/Edit/5

// To protect from overposting attacks, please enable the specific properties you
want to bind to, for

// more details see http://go.microsoft.com/fwlink/?LinkId=317598.

[HttpPost]
[ValidateAntiForgeryToken]
```

```

    public ActionResult Edit([Bind(Include =
"Buku_Id,Buku_Bcd,Buku_Judul,Buku_Kategori,Buku_Penulis,Buku_Penerbit,Buku_Ra
k,Buku_Bahasa,Buku_Jumlah,Buku_Tahun_Terbit")] Buku buku)

    {

        if (ModelState.IsValid)

        {

            db.Entry(buku).State = EntityState.Modified;

            db.SaveChanges();

            return RedirectToAction("Index");

        }

        ViewBag.Buku_Bahasa = new SelectList(db.Buku_Bahasa, "Bahasa_Id",
"Bahasa_Nama", buku.Buku_Bahasa);

        ViewBag.Buku_Kategori = new SelectList(db.Buku_Kategori, "Katrgori_Id",
"Kategori_Nama", buku.Buku_Kategori);

        ViewBag.Buku_Penerbit = new SelectList(db.Buku_Penerbit, "Penerbit_Id",
"Penerbit_Nama", buku.Buku_Penerbit);

        ViewBag.Buku_Penulis = new SelectList(db.Buku_Penulis, "Penulis_Id",
"Penulis_Nama", buku.Buku_Penulis);

        ViewBag.Buku_Rak = new SelectList(db.Buku_Rak, "Rak_Id", "Rak_Nama",
buku.Buku_Rak);

        return View(buku);

    }

    // GET: Buku/Delete/5

    public ActionResult Delete(int? id)

    {

        if (id == null)

        {

```

```
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    Buku buku = db.Buku.Find(id);
    if (buku == null)
    {
        return HttpNotFound();
    }
    return View(buku);
}

// POST: Buku/Delete/5
[HttpPost, ActionName("Delete")]
[ValidateAntiForgeryToken]
public ActionResult DeleteConfirmed(int id)
{
    Buku buku = db.Buku.Find(id);
    db.Buku.Remove(buku);
    db.SaveChanges();
    return RedirectToAction("Index");
}

protected override void Dispose(bool disposing)
{
    if (disposing)
    {
```

```
        db.Dispose();
    }
    base.Dispose(disposing);
}
}
```

## 2. Source Code Data Anggota

```
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Linq;
using System.Net;
using System.Web;
using System.Web.Mvc;
using Perpustakaan.Models;

namespace Perpustakaan.Controllers
{
    public class AnggotaController : Controller
    {
```



```
private perpusEntities1 db = new perpusEntities1();

// GET: Anggota

public ActionResult Index()
{
    var anggota = db.Anggota.Include(a => a.Jenis_Kelamin);
    return View(anggota.ToList());
}

// GET: Anggota/Details/5

public ActionResult Details(int? id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    Anggota anggota = db.Anggota.Find(id);
    if (anggota == null)
    {
        return HttpNotFound();
    }
    return View(anggota);
}
```

```
// GET: Anggota/Create

public ActionResult Create()

{

    ViewBag.Anggota_Jk = new SelectList(db.Jenis_Kelamin, "JK_Id", "JK_Nama");

    return View();

}

// POST: Anggota/Create

// To protect from overposting attacks, please enable the specific properties you
want to bind to, for

// more details see http://go.microsoft.com/fwlink/?LinkId=317598.

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Create([Bind(Include =
"Anggota_Id,Anggota_Bcd,Anggota_Nim,Anggota_Nama,Anggota_TTL,Anggota_Tgl,A
nggota_Jk,Anggota_Alamat,Anggota_Email")] Anggota anggota)

{

    if (ModelState.IsValid)

    {

        db.Anggota.Add(anggota);

        db.SaveChanges();

        return RedirectToAction("Index");

    }

}
```

```
        ViewBag.Anggota_Jk = new SelectList(db.Jenis_Kelamin, "JK_Id", "JK_Nama",
anggota.Anggota_Jk);

        return View(anggota);
    }

// GET: Anggota/Edit/5
public ActionResult Edit(int? id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    Anggota anggota = db.Anggota.Find(id);

    if (anggota == null)
    {
        return HttpNotFound();
    }

    ViewBag.Anggota_Jk = new SelectList(db.Jenis_Kelamin, "JK_Id", "JK_Nama",
anggota.Anggota_Jk);

    return View(anggota);
}

// POST: Anggota/Edit/5
```

```

// To protect from overposting attacks, please enable the specific properties you
want to bind to, for

// more details see http://go.microsoft.com/fwlink/?LinkId=317598.

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Edit([Bind(Include =
"Anggota_Id,Anggota_Bcd,Anggota_Nim,Anggota_Nama,Anggota_TTL,Anggota_Tgl,A
nggota_Jk,Anggota_Alamat,Anggota_Email")] Anggota anggota)

{

    if (ModelState.IsValid)

    {

        db.Entry(anggota).State = EntityState.Modified;

        db.SaveChanges();

        return RedirectToAction("Index");

    }

    ViewBag.Anggota_Jk = new SelectList(db.Jenis_Kelamin, "JK_Id", "JK_Nama",
anggota.Anggota_Jk);

    return View(anggota);

}

// GET: Anggota/Delete/5

public ActionResult Delete(int? id)

{

    if (id == null)

```

```
{
    return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
}

Anggota anggota = db.Anggota.Find(id);

if (anggota == null)
{
    return HttpNotFound();
}

return View(anggota);
}

// POST: Anggota/Delete/5
[HttpPost, ActionName("Delete")]
[ValidateAntiForgeryToken]
public ActionResult DeleteConfirmed(int id)
{
    Anggota anggota = db.Anggota.Find(id);
    db.Anggota.Remove(anggota);
    db.SaveChanges();
    return RedirectToAction("Index");
}

protected override void Dispose(bool disposing)
```

```
{  
    if (disposing)  
    {  
        db.Dispose();  
    }  
    base.Dispose(disposing);  
}  
}
```

## 2. Source Code Peminjaman Buku

```
using System;  
using System.Collections.Generic;  
using System.Data;  
using System.Data.Entity;  
using System.Linq;  
using System.Net;  
using System.Web;  
using System.Web.Mvc;  
using Perpustakaan.Models;  
  
namespace Perpustakaan.Controllers  
{
```

```
public class PeminjamanController : Controller
{
    private perpusEntities1 db = new perpusEntities1();

    // GET: Peminjaman
    public ActionResult Index()
    {
        var peminjaman = db.Peminjaman.Include(p => p.Anggota).Include(p =>
p.Batas_Hari).Include(p => p.Denda);
        return View(peminjaman.ToList());
    }

    // GET: Peminjaman/Details/5
    public ActionResult Details(int? id)
    {
        if (id == null)
        {
            return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
        }
        Peminjaman peminjaman = db.Peminjaman.Find(id);
        if (peminjaman == null)
        {
            return HttpNotFound();
        }
    }
}
```

```
    }

    return View(peminjaman);
}

// GET: Peminjaman/Create
public ActionResult Create(int id_pinjaman)
{
    Peminjaman pem = db.Peminjaman.Find(id_pinjaman);

    ViewBag.ID_Peminjaman = pem.Peminjaman_Id;

    ViewData["DaftarPinjaman"] = db.Peminjaman_Detail.Where(e =>
e.Peminjaman_Id == id_pinjaman).ToList();

    return View();
}

public ActionResult PeminjamanBuku(int id_pinjaman, string barcodeBuku)
{
    int idBuku = db.Buku.Where(e =>
e.Buku_Bcd.Contains(barcodeBuku)).First().Buku_Id;

    Peminjaman_Detail pem = new Peminjaman_Detail();

    pem.Peminjaman_Id = id_pinjaman;
```



```
pem.Buku_Id = idBuku;

pem.Tgl_Kembali = DateTime.Now;

db.Peminjaman_Detail.Add(pem);

db.SaveChanges();

return RedirectToAction("Create", new { id_pinjaman = pem.Peminjaman_Id });
}

public ActionResult Pinjam(string id_anggota)
{
    int idAnggota = db.Anggota.Where(e =>
e.Anggota_Bcd.Contains(id_anggota)).First().Anggota_Id;

    Peminjaman baru = new Peminjaman();

    baru.Anggota_Id = idAnggota;

    baru.Peminjaman_Tanggal = DateTime.Now;

    baru.Denda_Id = 1;

    baru.Batas_Hari_Id = 1;

    db.Peminjaman.Add(bar);

    db.SaveChanges();

    return RedirectToAction("Create", new { id_pinjaman = baru.Peminjaman_Id });
```

```
}

// GET: Peminjaman/Edit/5

public ActionResult Edit(int? id)

{

    if (id == null)

    {

        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

    }

    Peminjaman peminjaman = db.Peminjaman.Find(id);

    if (peminjaman == null)

    {

        return HttpNotFound();

    }

    ViewBag.Anggota_Id = new SelectList(db.Anggota, "Anggota_Id",

"Anggota_Bcd", peminjaman.Anggota_Id);

    ViewBag.Batas_Hari_Id = new SelectList(db.Batas_Hari, "Batas_Hari_Id",

"Hari", peminjaman.Batas_Hari_Id);

    ViewBag.Denda_Id = new SelectList(db.Denda, "Denda_Id", "Nominal",

peminjaman.Denda_Id);

    return View(peminjaman);

}

// POST: Peminjaman/Edit/5
```

```
// To protect from overposting attacks, please enable the specific properties you
want to bind to, for

// more details see http://go.microsoft.com/fwlink/?LinkId=317598.

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Edit([Bind(Include =
"Peminjaman_Id,Anggota_Id,Peminjaman_Tanggal,Denda_Id,Batas_Hari_Id")]
Peminjaman peminjaman)

{

    if (ModelState.IsValid)

    {

        db.Entry(peminjaman).State = EntityState.Modified;

        db.SaveChanges();

        return RedirectToAction("Index");

    }

    ViewBag.Anggota_Id = new SelectList(db.Anggota, "Anggota_Id",
"Anggota_Bcd", peminjaman.Anggota_Id);

    ViewBag.Batas_Hari_Id = new SelectList(db.Batas_Hari, "Batas_Hari_Id",
"Hari", peminjaman.Batas_Hari_Id);

    ViewBag.Denda_Id = new SelectList(db.Denda, "Denda_Id", "Nominal",
peminjaman.Denda_Id);

    return View(peminjaman);

}

// GET: Peminjaman/Delete/5
```

```
public ActionResult Delete(int? id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    Peminjaman peminjaman = db.Peminjaman.Find(id);
    if (peminjaman == null)
    {
        return HttpNotFound();
    }
    return View(peminjaman);
}

// POST: Peminjaman/Delete/5
[HttpPost, ActionName("Delete")]
[ValidateAntiForgeryToken]
public ActionResult DeleteConfirmed(int id)
{
    Peminjaman peminjaman = db.Peminjaman.Find(id);
    db.Peminjaman.Remove(peminjaman);
    db.SaveChanges();
    return RedirectToAction("Index");
}
```

```
}  
}}
```