

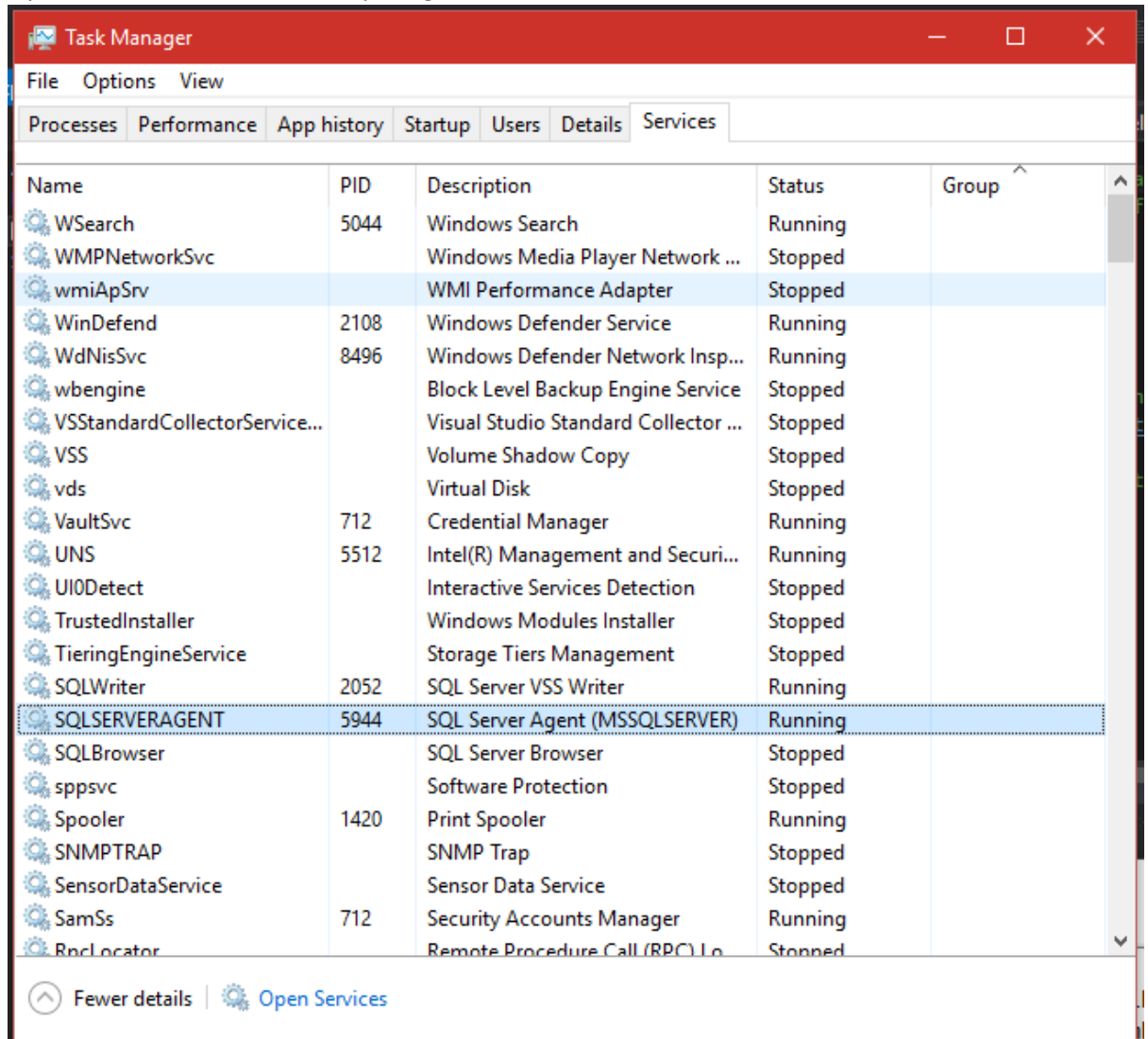
Nama : Renaldy Rizki Maulana

Nim : 151524026

Tugas : PLOO - Lab01

Langkah – Langkah Pembuatan :

1. Nyalakan service SQLSERVER seperti gambar dibawah

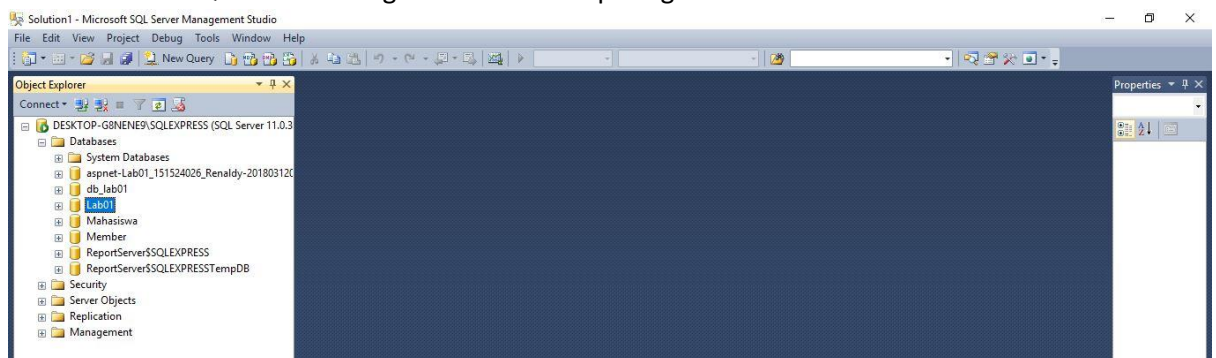


2. Buka IDE untuk mengelola database. Disini saya menggunakan SQL Server untuk IDE nya, dan pastikan bahwa database sudah bisa terhubung seperti gambar dibawah lalu buat sebuah database; disini saya membuat dengan nama **db_lab01** atau database bisa dibuat di dalam projek.



Nantinya server name diperlukan untuk membuat koneksi dengan web yang akan dibuat.

Lalu masuk ke SQL server management studio. Seperti gambar dibawah ini:



3. Buat project baru ASP Web > MVC
4. Buka file Web.config pada root project, lalu tambahkan baris code ini :
`<add name="db_lab01" connectionString="Data Source=DESKTOP-G8NENE9\SQLEXPRESS;Initial Catalog=Lab01;Integrated Security=True" providerName="System.Data.SqlClient" />`

```
<connectionStrings>
  <add name="DefaultConnection" connectionString="Data Source=(localdb)\MSSQLLocalDB;AttachDbFilename=[DataDirectory]\aspnet-Lab01_151524026_Renaldy-20180312084354.mdf" />
  <add name="db_lab01" connectionString="Data Source=DESKTOP-G8NENE9\SQLEXPRESS;Initial Catalog=Lab01;Integrated Security=True" providerName="System.Data.SqlClient" />
</connectionStrings>
```

Disini saya menggunakan windows authentication untuk credential ke SQLServer nya.

- name=db_lab01 : adalah nama dari connection string nya, bukan nama database. Disini saya menggunakan nama itu untuk mempermudah mendefinisikan nama dari connection string nya
 - Data Source=DESKTOP-G8NENE9\SQLEXPRESS : adalah nama dari lokal komputer yang digunakan untuk authentication
 - Initial Catalog=db_lab01 : adalah nama databasenya
5. Lalu Tambahkan Koneksi dari sql server.

Server Explorer → data connection → klik kanan pilih *add connection* → lalu akan tampil menu seperti pada gambar dibawah ini:

Isi server name dan tablenya lalu test connection jika success maka tekan OK, Sudah Terhubung dengan SQL Server.

- Setelah itu buat class **Member** dengan cara klik kanan pada folder **Model** pilih menu **Add > Class > Class** lalu beri nama **Member.cs**
Berikut source code dari class ini :

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;

namespace Lab01_151524026_Renaldy.Models
{
    using System;
    using System.Collections.Generic;
    using System.ComponentModel.DataAnnotations;
    using System.ComponentModel.DataAnnotations.Schema;
    using System.Data.Entity.Spatial;

    [Table("Kelas")] //nama table
```

```

public partial class Member
{
    [Key]
    [DatabaseGenerated(DatabaseGeneratedOption.Identity)]
    public int MemberId { get; set; }

    [Required]
    [Display(Name = "First Name")]
    [StringLength(255)]
    public string FirstName { get; set; }

    [Display(Name = "Last Name")]
    [StringLength(255)]
    public string LastName { get; set; }

    [Display(Name = "Date of Birth")]
    [DataType(DataType.Date)]
    [DisplayFormat(DataFormatString = "{0:yyyy-MM-dd}",
ApplyFormatInEditMode = true)]
    public DateTime DateOfBirth { get; set; }

    public Nullable<int> Height { get; set; }

    public Nullable<int> Weight { get; set; }

    [Required]
    [StringLength(20)]
    public string Email { get; set; }

    [Display(Name = "Telephone")]
    [StringLength(12)]
    public string Phone { get; set; }
}
}

```

- Setelah itu buat class **MemberModels** dengan cara klik kanan pada folder **Model** pilih menu **Add > Class > ADO.NET Entity Data Model** lalu beri nama **MemberModel.cs** setelah menambahkan, akan muncul seperti gambar dibawah, lalu pilih Empty Code First model

Berikut source code dari class ini :

```

namespace Lab01_151524026_Renaldy.Models
{
    using System;
    using System.Data.Entity;
    using System.ComponentModel.DataAnnotations.Schema;
    using System.Linq;
    public class MemberModel : DbContext
    {
        public MemberModel()
            : base("name=db_lab01") // nama connection string
        {
        }

        // Add a DbSet for each entity type that you want to include in your
        model. For more information
        // on configuring and using a Code First model, see
        http://go.microsoft.com/fwlink/?LinkId=390109.

        public virtual DbSet<Member> Member { get; set; }
    }
}

```

```

protected override void OnModelCreating(DbModelBuilder modelBuilder)
{
    modelBuilder.Entity<Member>()
        .Property(e => e.MemberId);

    modelBuilder.Entity<Member>()
        .Property(e => e.FirstName)
        .IsUnicode(false);

    modelBuilder.Entity<Member>()
        .Property(e => e.LastName)
        .IsUnicode(false);

    modelBuilder.Entity<Member>()
        .Property(e => e.DateOfBirth);

    modelBuilder.Entity<Member>()
        .Property(e => e.Height);

    modelBuilder.Entity<Member>()
        .Property(e => e.Weight);

    modelBuilder.Entity<Member>()
        .Property(e => e.Email)
        .IsUnicode(false);

    modelBuilder.Entity<Member>()
        .Property(e => e.Phone)
        .IsUnicode(false);
}
}

```

8. Setelah itu buat class controller **MemberController** dengan cara klik kanan pada folder **Controllers** pilih menu **Add > Controller... > MVC 5 Controller with views, using Entity Framework** lalu akan tampil seperti gambar di bawah dan sesuaikan, lalu beri nama controller dengan nama **MemberController**

9. Setelah itu, buka **Package Manager Console** pada menu **Tools > NuGet Package Manager > Package Manager Console**
10. Lalu, ketik perintah **Enable-Migrations** pada **Package Manager Console**. Jika muncul error seperti dibawah :

```
PM> enable-migrations
More than one context type was found in the assembly 'Lab01_151524026_Renaldy'.
To enable migrations for 'Lab01_151524026_Renaldy.Models.MemberModel', use Enable-Migrations -ContextTypeName
Lab01_151524026_Renaldy.Models.MemberModel.
To enable migrations for 'Lab01_151524026_Renaldy.Models.ApplicationDbContext', use Enable-Migrations -
ContextTypeName Lab01_151524026_Renaldy.Models.ApplicationDbContext.
PM> Enable-Migrations -ContextTypeName Lab01_151524026_Renaldy.Models.MemberModel
Checking if the context targets an existing database...
Detected database created with a database initializer. Scaffolded migration '201803150915428_InitialCreate'
corresponding to existing database. To use an automatic migration instead, delete the Migrations folder and re-
run Enable-Migrations specifying the -EnableAutomaticMigrations parameter.
Code First Migrations enabled for project Lab01_151524026_Renaldy.
```

Ketik kembali perintah :

- **Enable-Migrations -ContextTypeName**
Lab01_151524026_Renaldy.Models.MemberModel

Jika berhasil, maka ketik perintah selanjutnya yaitu :

- **add-migration**
- **Lalu masukan nama migrationnya, contoh jika sedang mengupdate table maka migrationnya bisa dinamai “updatetable”**

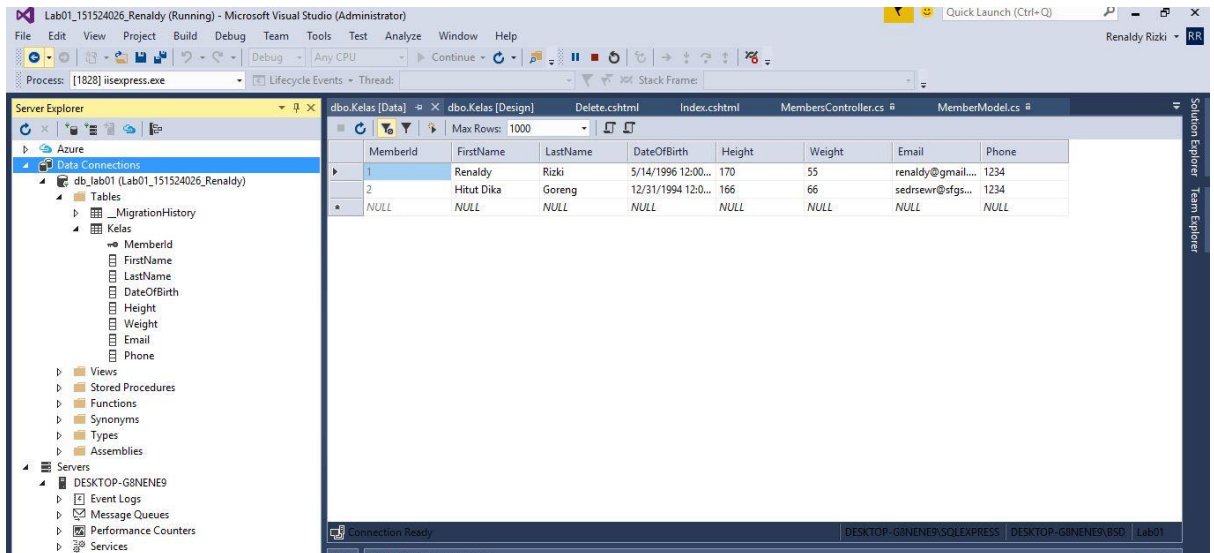
```
PM> Add-migration
cmdlet Add-Migration at command pipeline position 1
Supply values for the following parameters:
Name: lab01
Scaffolding migration 'lab01'.
The Designer Code for this migration file includes a snapshot of your current Code First model. This snapshot
is used to calculate the changes to your model when you scaffold the next migration. If you make additional
changes to your model that you want to include in this migration, then you can re-scaffold it by running 'Add-
Migration lab01' again.
```

- **update-database**

```
PM> update-database
Specify the '-Verbose' flag to view the SQL statements being applied to the target database.
Applying explicit migrations: [201803150936558_lab01].
Applying explicit migration: 201803150936558_lab01.
Caution: Changing any part of an object name could break scripts and stored procedures.
Caution: Changing any part of an object name could break scripts and stored procedures.
Running Seed method.
```

11. Setelah itu, coba running programnya. Jika berhasil, maka akan menampilkan halaman web utama (index). Untuk melihat halaman yang telah dibuat tambahkan url dengan nama controller (nama controller: MembersController) maka menjadi localhost/Members (tanpa kata controller). Lalu bisa juga melihat update data tablenya di menu data connection di Server Explorer.

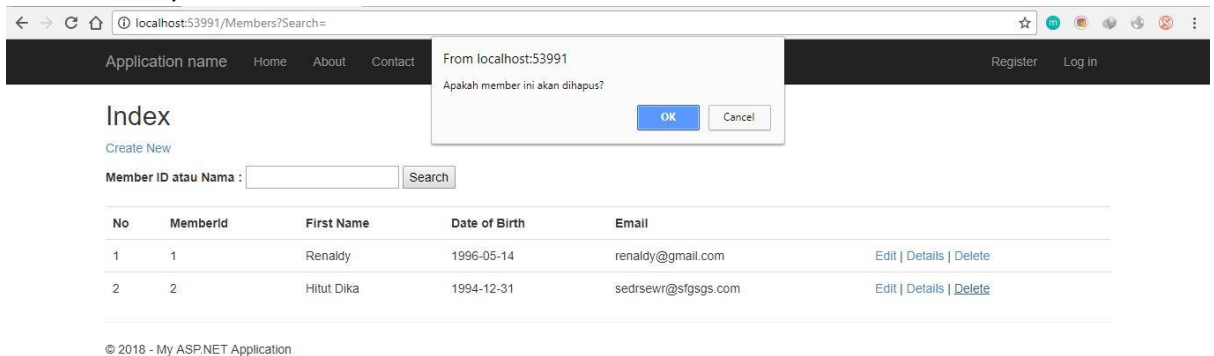
Server Explorer → Data Connection → String Connection → Tables → Lalu pada Tabel yang diupdate klik kanan lalu pilih perintah “show data table” maka akan menghasilkan tampilan seperti dibawah ini:



12. Langkah selanjutnya mengubah field index dan membuat fungsi konfirmasi dialog saat akan menghapus data member nya dengan cara mengubah file **Index.cshtml** pada folder **Views/Members/Index.cshtml**. Ubah source code seperti berikut :

Line	Sebelum	Sesudah
61	<code>@Html.ActionLink("Delete", "Delete", new { id = item.MemberId })</code>	<code>@Html.ActionLink("Delete", "Delete", new { id = item.MemberId }, new { onclick = "return confirm('Apakah member ini akan dihapus?');" })</code>

Berikut hasilnya:



13. Sebelumnya, action delete ini akan dialihkan ke halaman Delete, untuk mengatasi agar tidak dialihkan ke halaman lain dan langsung terhapus datanya maka method Delete pada controller member diubah menjadi seperti berikut :

```
public ActionResult Delete(int? id){
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    Member member = db.Member.Find(id);
    if (member == null)
```

```

    {
        return HttpNotFound();
    }

    db.Member.Remove(member);
    db.SaveChanges();
    return RedirectToAction("Index");
}

```

14. Lalu yang terakhir untuk menambahkan search pada list member, maka dilakukan perubahan pada method index dalam controller member dan view list indexnya sebagai berikut :

Controllers/MemberController.cs

```

// GET: Member
public ActionResult Index(string search)
{
    if (!String.IsNullOrEmpty(search)) {
        int memberID=0;
        Int32.TryParse(search, out memberID);

        return View(db.Member.Where(x => x.FirstName.Contains(search)
|| x.MemberId == memberID).ToList());
    }else {
        return View(db.Member.ToList());
    }
}

```

Dan untuk source code view nya menambah code sebagai berikut diatas tag html table :

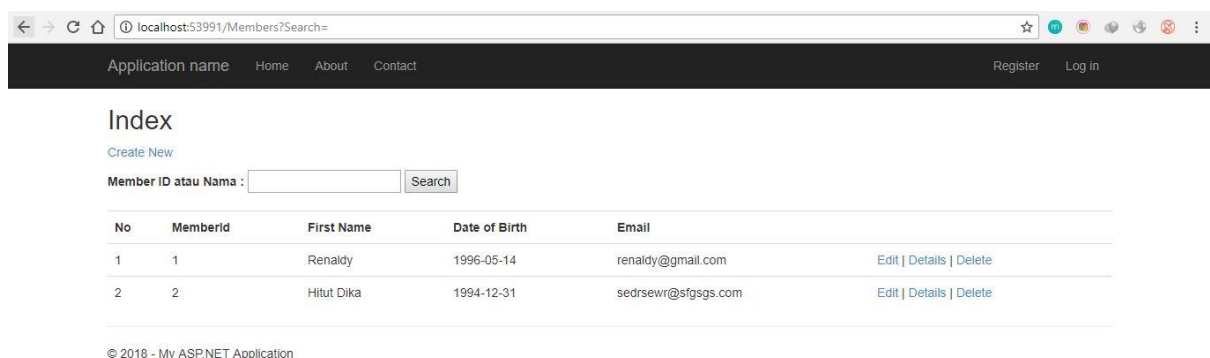
Views/Members/Index.cshtml

```

@using (Html.BeginForm("Index", "Members", FormMethod.Get))
{
    <b>Member ID atau Nama : </b>
    @Html.TextBox("Search");
    <input type="submit" value="Search" />
}
<br />

```

Screenshoot Hasil Akhir :



Hasil 1 List All Member

← → ↻ 🏠 ⓘ localhost:53991/Members?Search=Renaldy ☆ 📱 📺 🖨️ 🔍 ⋮

Application name Home About Contact Register Log in

Index

[Create New](#)

Member ID atau Nama :

No	MemberId	First Name	Date of Birth	Email	
1	1	Renaldy	1996-05-14	renaldy@gmail.com	Edit Details Delete

© 2018 - My ASP.NET Application

Hasil 2 Search by Name

← → ↻ 🏠 ⓘ localhost:53991/Members?Search=1 ☆ 📱 📺 🖨️ 🔍 ⋮

Application name Home About Contact Register Log in

Index

[Create New](#)

Member ID atau Nama :

No	MemberId	First Name	Date of Birth	Email	
1	1	Renaldy	1996-05-14	renaldy@gmail.com	Edit Details Delete

© 2018 - My ASP.NET Application

Hasil 3 Search by MemberID