

Copyright © 2008 Alex Budiyanto

Seluruh dokumen ini dapat digunakan, dimodifikasi dan disebarakan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen.

Web Service Hands on Lab

May 3

2008

Alex Budiyanto

Enterprise Application
Development using
.NET Framework

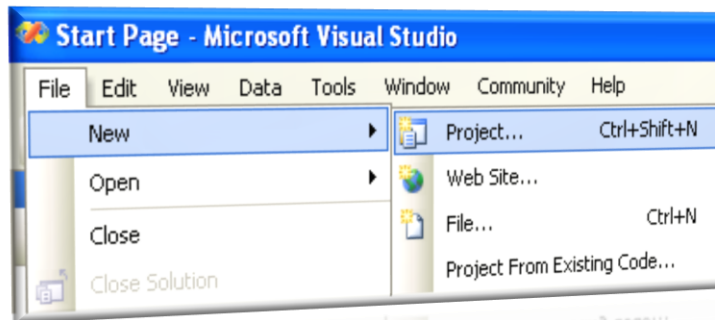
HOL SQL Web Service

1. Tujuan Hands on Lab

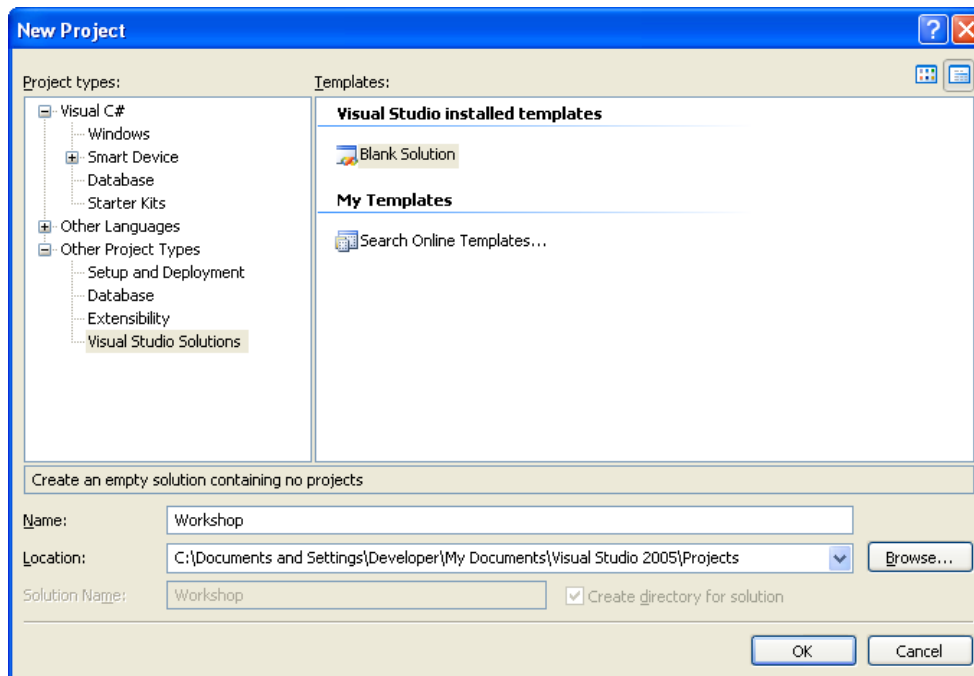
- a. Memahami konsep layering dalam pembangunan aplikasi.
- b. Memahami konsep *object oriented programming*.
- c. Memahami cara pemakaian stored procedure yang sudah dibuat sebelumnya.
- d. Memahami cara pembuatan Web Service.

2. Pembuatan Solution dan Project

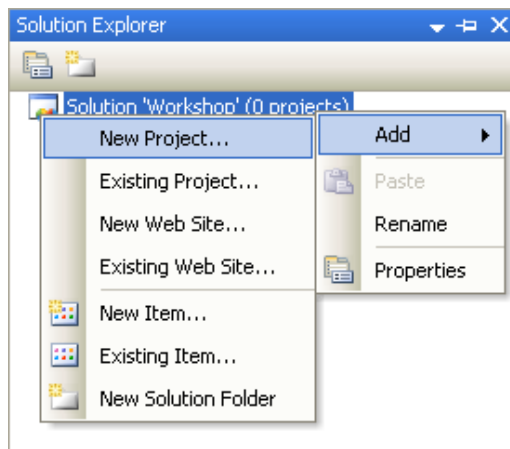
- Tarik nafas dalam-dalam, dan berdoa ☺.
- Buka program Visual Studio 2005 anda, kemudian pilih menu *File* → *New* → *Project*. Seperti pada gambar dibawah ini:



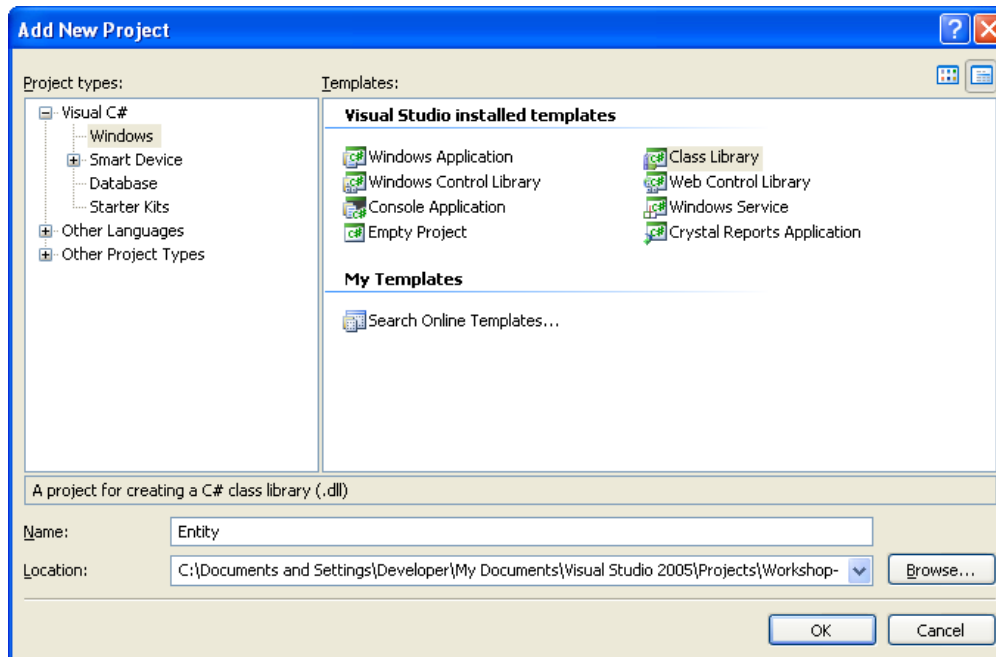
- Pada jendela **New Project**. Expand node "Other Project Type" dan pilih "Visual Studio Solution". Pada bagian "Visual Studio Installed Templates" pilih "Blank Solution". Isikan nama Solutin dengan "Workshop" kemudian tekan tombol OK. Untuk lebih jelas-nya lihat gambar dibawah ini:



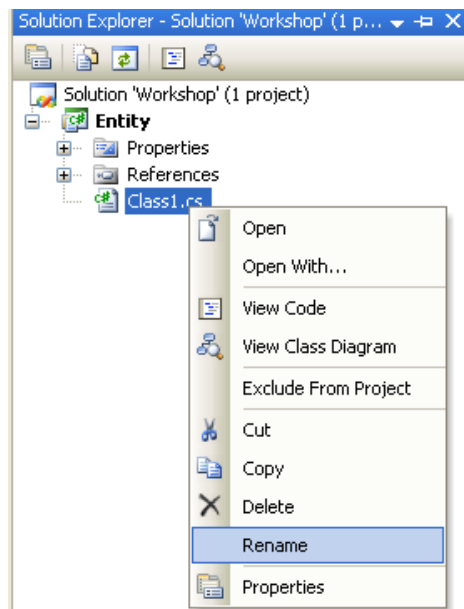
- Pilih Menu **View** → **Solution Explorer** untuk menampilkan jendela “*Solution Explorer*”. Setelah jendela **Solution Explorer** terbuka, klik kanan pada nama Solution “*Workshop*”, kemudian pilih menu **Add** → **New Project**. Untuk lebih jelas-nya lihat gambar dibawah ini:



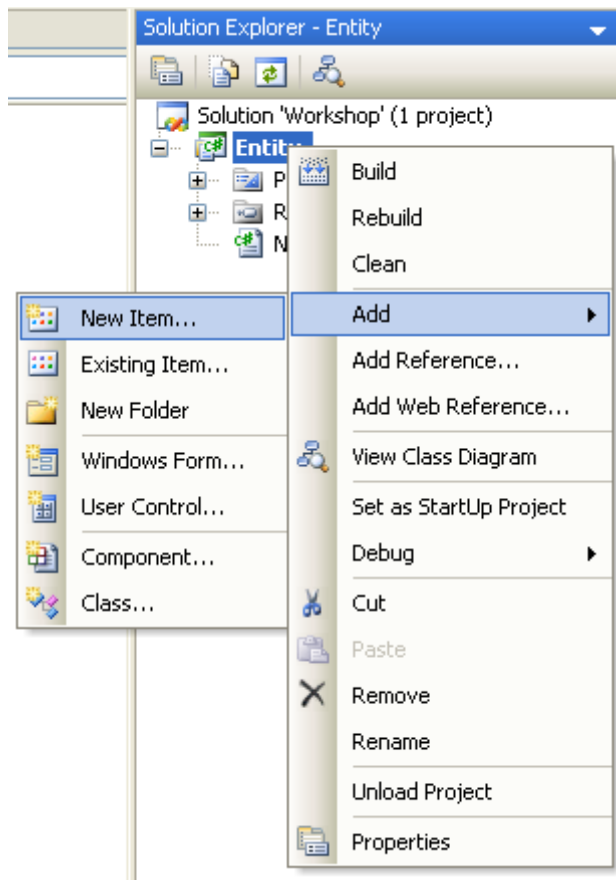
- Pada jendela “**Add New Project**” dalam section “**Project Type**” expand node “**Visual C#**” kemudian pilih node **Windows**. Kemudian pada section “**Visual Studio installed templates**” pilih “**Class Library**”. Kemudian isikan “**Entity**” sebagai nama project. Untuk lebih jelas-nya lihat gambar dibawah ini:



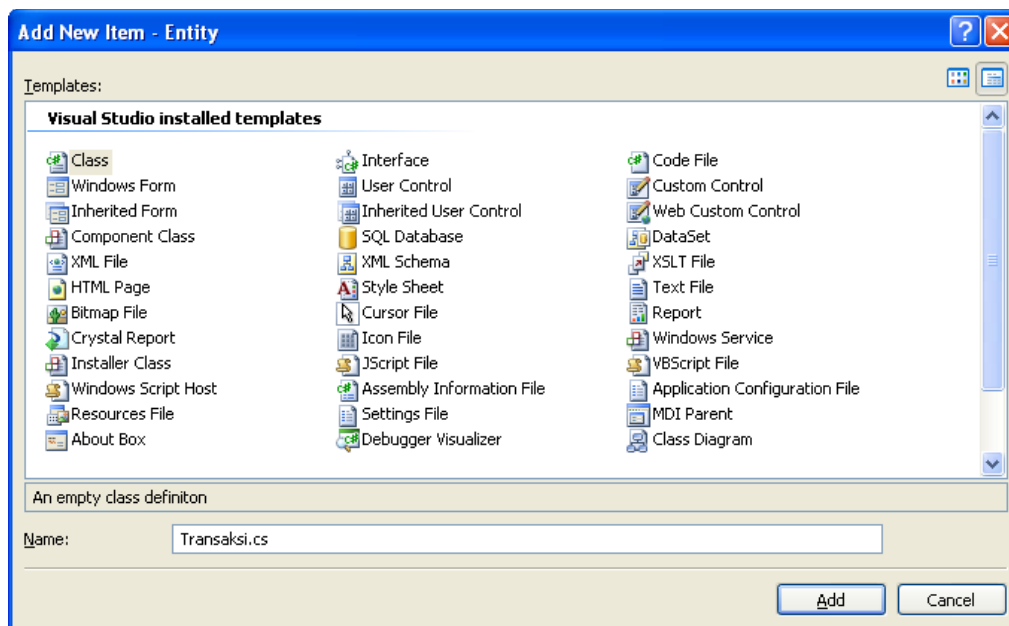
- Pada **Solution Explorer**, klik kanan Node “**Class1.cs**” dan pilih menu “**Rename**” kemudian ganti namanya menjadi “**Nasabah.cs**”. Untuk lebih jelas-nya lihat gambar dibawah ini:



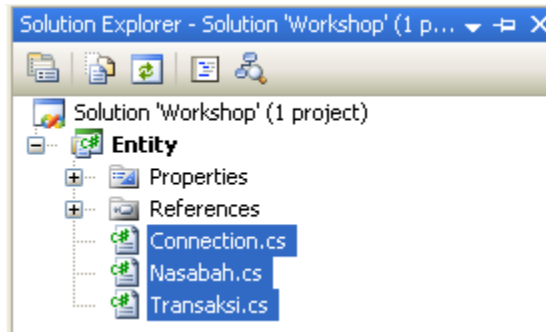
- Masih dalam **Solution Explorer**, klik kanan node “**Entity**” kemudian pilih menu Add → New **Item**. Untuk lebih jelas-nya lihat gambar dibawah ini:



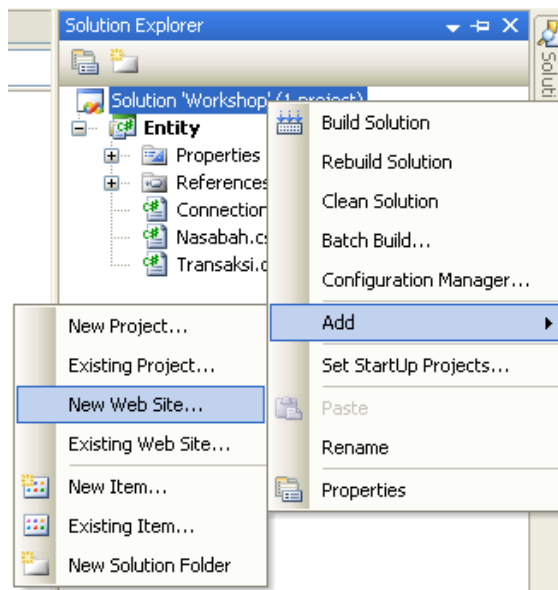
- Pada jendela “Add New Item – Entity” pada section “Visual Studio installed templates” pilih object **Class** dan isikan **Transaksi.cs** sebagai nama class. Untuk lebih jelas-nya lihat gambar dibawah ini:



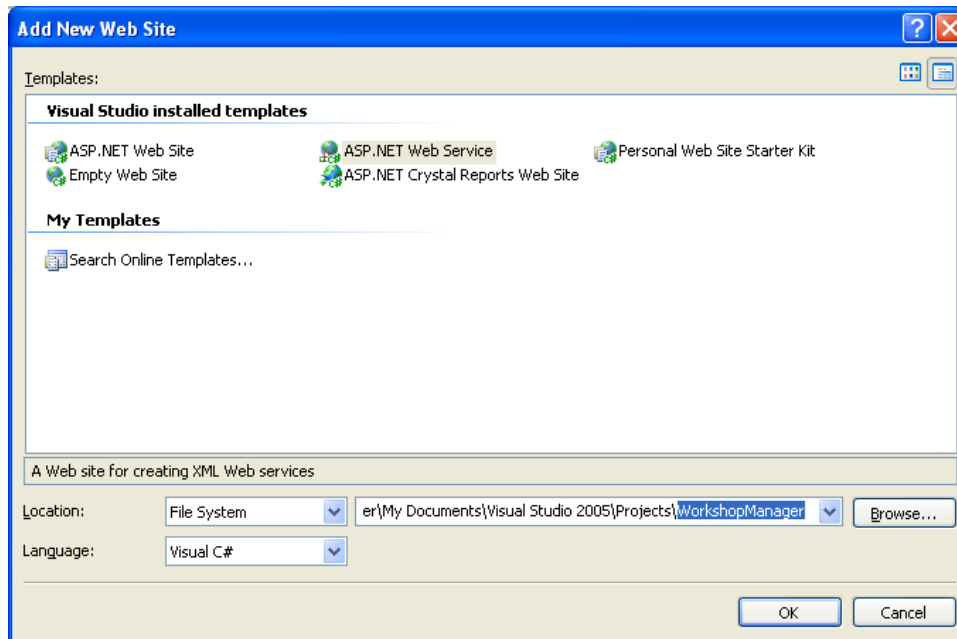
- Lakukan hal yang sama untuk menambahkan class “**Connection.cs**”. Sehingga hasil akhirnya seperti gambar dibawah ini.



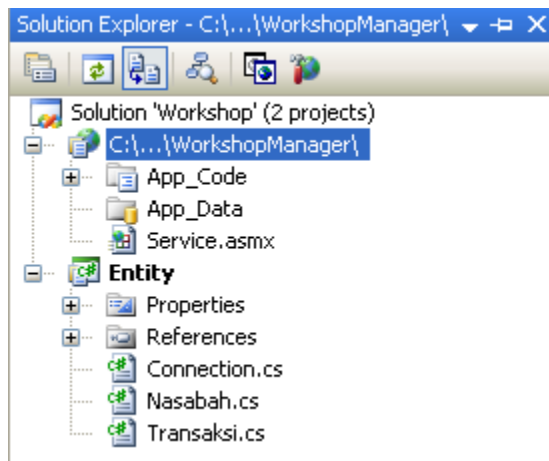
- Masih dalam Solution Explorer, klik kanan Solution “Workshop” kemudian pilih menu Add → New Website seperti pada gambar dibawah ini:



- Pada jendela “Add New Website” dalam section “Visual Studio installed templates” pilih object **ASPNET Web Service**. Pilih location “File System” dan isikan nama project dengan “**Workshop Manager**”. Pilih **Visual C#** sebagai pilihan bahasa. Untuk lebih jelasnya lihat gambar dibawah ini:



- Hasil akhir dari langkah diatas, dalam **Solution Explorer** terdapat nama project “**Workshop Manager**” seperti dalam gambar dibawah ini:



3. Pembuatan Class Library

- Dalam **Solution Explorer** di dalam project “**Entity**”. Double klik object **Connection.cs** kemudian ketikkan isi kode, sehingga seluruh kode-nya seperti dibawah ini:

```
using System.Data.SqlClient;
namespace Workshop.DBConnection
{
    public class Connection
    {
        private string connectionString = @"Data Source=
        .\sqlexpress;Initial Catalog=Workshop;
        Integrated Security=True;Pooling=False";
    }
}
```

```

        public SqlConnection GetConnection()
        {
            return new SqlConnection(connectionString);
        }
    }
}

```

- Dalam **Solution Explorer** di dalam project “**Entity**”. Double klik object **Nasabah.cs** kemudian ketikkan isi kode, sehingga seluruh kode-nya seperti dibawah ini:

```

using System;
using System.Collections.Generic;
using System.Text;
using System.Data.SqlClient;
using System.Data;
using Workshop.DBConnection;

namespace Workshop.Entity
{
    [Serializable]
    public class Nasabah
    {
        #region member of Nasabah
        private string noRekening;
        private string nama;
        private string alamat;
        private string pekerjaan;
        private decimal saldo;
        private Transaksi transaksi;
        #endregion

        #region property of Nasabah
        public string NoRekening
        {
            get { return noRekening; }
        }
        public string Nama
        {
            get { return nama; }
            set { nama = value; }
        }
        public string Alamat
        {
            get { return alamat; }
            set { alamat = value; }
        }
        public string Pekerjaan
        {
            get { return pekerjaan; }
            set { pekerjaan = value; }
        }
        public decimal Saldo
        {
            get { return saldo; }
        }
        public Transaksi Transaksi
        {
            get
            {
                return transaksi;
            }
        }
        #endregion
    }
}

```

```
#region constructor of Nasabah
public Nasabah()
{
    this.noRekening = String.Empty;
    this.nama = String.Empty;
    this.alamat = String.Empty;
    this.pekerjaan = String.Empty;
    this.saldo = Decimal.MinValue;
    this.transaksi = new Transaksi();
}
public Nasabah(string _noRekening)
{
    this.noRekening = _noRekening;
    this.transaksi = new Transaksi(_noRekening);
}
public Nasabah(string _noRekening, string _nama, string _alamat,
    string _pekerjaan, decimal _setoranAwal)
{
    this.noRekening = _noRekening;
    this.nama = _nama;
    this.alamat = _alamat;
    this.pekerjaan = _pekerjaan;
    this.saldo = _setoranAwal;
    this.transaksi = new Transaksi(_noRekening, _setoranAwal,
        "Pembukaan Rekening");
}
public Nasabah(string _noRekening, string _nama, string _alamat,
    string _pekerjaan)
{
    this.noRekening = _noRekening;
    this.nama = _nama;
    this.alamat = _alamat;
    this.pekerjaan = _pekerjaan;
    this.saldo = 0;
    this.transaksi = new Transaksi();
}
#endregion

#region method of Nasabah
public int Add()
{
    SqlConnection server = new Connection().GetConnection();
    try
    {
        server.Open();
        SqlCommand cmd = new SqlCommand("cashier.NewNasabah", server);
        cmd.CommandType = CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("@noRekening", this.noRekening);
        cmd.Parameters.AddWithValue("@nama", this.nama);
        cmd.Parameters.AddWithValue("@alamat", this.alamat);
        cmd.Parameters.AddWithValue("@pekerjaan", this.pekerjaan);
        cmd.Parameters.AddWithValue("@setoranAwal", this.saldo);
        return cmd.ExecuteNonQuery();
    }
    catch (Exception)
    {
        return -1;
    }
    finally
    {
        server.Close();
    }
}
public int Update()
{
    SqlConnection server = new Connection().GetConnection();
    try
    {
        server.Open();
        SqlCommand cmd = new SqlCommand("cashier.UpdateNasabah", server);
        cmd.CommandType = CommandType.StoredProcedure;
    }
}
```

```
        cmd.Parameters.AddWithValue("@noRekening", this.noRekening);
        cmd.Parameters.AddWithValue("@nama", this.nama);
        cmd.Parameters.AddWithValue("@alamat", this.alamat);
        cmd.Parameters.AddWithValue("@pekerjaan", this.pekerjaan);
        return cmd.ExecuteNonQuery();
    }
    catch (Exception)
    {
        return -1;
    }
    finally
    {
        server.Close();
    }
}
public int Delete()
{
    SqlConnection server = new Connection().GetConnection();
    try
    {
        server.Open();
        SqlCommand cmd = new SqlCommand("cashier.DeleteNasabah", server);
        cmd.CommandType = CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("@noRekening", this.noRekening);
        return cmd.ExecuteNonQuery();
    }
    catch (Exception)
    {
        return -1;
    }
    finally
    {
        server.Close();
    }
}
public decimal GetSaldo()
{
    SqlConnection server = new Connection().GetConnection();
    try
    {
        server.Open();
        SqlCommand cmd = new SqlCommand("cashier.GetSaldoByNasabah", server);
        cmd.CommandType = CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("@noRekening", this.noRekening);
        SqlDataReader reader = cmd.ExecuteReader();
        if (reader.HasRows)
        {
            reader.Read();
            return Convert.ToDecimal(reader["saldo"]);
        }
        else
            return -2;
    }
    catch (Exception)
    {
        return -1;
    }
    finally
    {
        server.Close();
    }
}
public Nasabah ShowDetail()
{
    SqlConnection server = new Connection().GetConnection();
    try
    {
        server.Open();
        SqlCommand cmd = new SqlCommand("cashier.ShowNasabahByNoRekening",
            server);
        cmd.CommandType = CommandType.StoredProcedure;
```

```

        cmd.Parameters.AddWithValue("@noRekening", this.noRekening);
        SqlDataReader reader = cmd.ExecuteReader();
        if (reader.HasRows)
        {
            reader.Read();
            this.nama = reader["nama"].ToString();
            this.alamat = reader["alamat"].ToString();
            this.pekerjaan = reader["pekerjaan"].ToString();
            this.saldo = Convert.ToDecimal(reader["saldo"]);
            return this;
        }
        else
            return new Nasabah();
    }
    catch (Exception)
    {
        return new Nasabah();
    }
    finally
    {
        server.Close();
    }
}
#endregion
}
}

```

- Dalam **Solution Explorer** di dalam project **"Entity"**. Double klik object **Transaksi.cs** kemudian ketikkan isi kode, sehingga seluruh kode-nya seperti dibawah ini:

```

using System;
using System.Collections.Generic;
using System.Text;
using Workshop.DBConnection;
using System.Data.SqlClient;
using System.Data;

namespace Workshop.Entity
{
    [Serializable]
    public class Transaksi
    {
        #region member of Transaksi
        private int id;
        private string jenis;
        private DateTime tanggal;
        private string noRekening;
        private decimal mutasi;
        private string keterangan;
        private decimal saldo;
        #endregion

        #region property of Transaksi
        public int Id
        {
            get { return id; }
        }

        public string Jenis
        {
            get { return jenis; }
        }
        public DateTime Tanggal
        {
            get { return tanggal; }
        }
    }
}

```

```
public string NoRekening
{
    get { return noRekening; }
}
public decimal Mutasi
{
    get { return mutasi; }
    set { mutasi = value; }
}
public string Keterangan
{
    get { return keterangan; }
    set { keterangan = value; }
}

public decimal Saldo
{
    get { return saldo; }
}
#endregion

#region custructor of Transaksi
public Transaksi()
{
    this.id = int.MinValue;
    this.jenis = string.Empty;
    this.tanggal = DateTime.MinValue;
    this.mutasi = decimal.MinValue;
    this.keterangan = string.Empty;
    this.saldo = decimal.MinValue;
    this.noRekening = string.Empty;
}

public Transaksi(string _noRekening, decimal _jumlahUang, string _keterangan)
{
    this.id = int.MinValue;
    this.jenis = string.Empty;
    this.tanggal = DateTime.MinValue;
    this.saldo = decimal.MinValue;
    this.noRekening = _noRekening;
    this.mutasi = _jumlahUang;
    this.keterangan = _keterangan;
}

public Transaksi(string _noRekening)
{
    this.id = int.MinValue;
    this.jenis = string.Empty;
    this.tanggal = DateTime.MinValue;
    this.saldo = decimal.MinValue;
    this.noRekening = _noRekening;
    this.mutasi = decimal.MinValue;
    this.keterangan = string.Empty;
}
#endregion

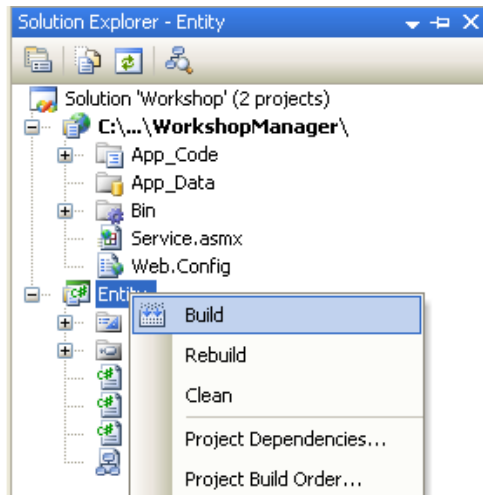
#region method of Transaksi
public int Saving()
{
    SqlConnection server = new Connection().GetConnection();
    try
    {
        server.Open();
        SqlCommand cmd = new SqlCommand("cashier.Saving", server);
        cmd.CommandType = CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("@noRekening", this.noRekening);
        cmd.Parameters.AddWithValue("@jumlahUang", this.mutasi);
        cmd.Parameters.AddWithValue("@keterangan", this.keterangan);
        return cmd.ExecuteNonQuery();
    }
    catch (Exception)
```

```

        {
            return -1;
        }
        finally
        {
            server.Close();
        }
    }
}
public int Withdraw()
{
    SqlConnection server = new Connection().GetConnection();
    try
    {
        server.Open();
        SqlCommand cmd = new SqlCommand("cashier.WithDraw", server);
        cmd.CommandType = CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("@noRekening", this.noRekening);
        cmd.Parameters.AddWithValue("@jumlahUang", this.mutasi);
        cmd.Parameters.AddWithValue("@keterangan", this.keterangan);
        return cmd.ExecuteNonQuery();
    }
    catch (Exception)
    {
        return -1;
    }
    finally
    {
        server.Close();
    }
}
public int TransferFund(Nasabah _penerima)
{
    SqlConnection server = new Connection().GetConnection();
    try
    {
        server.Open();
        SqlCommand cmd = new SqlCommand("cashier.TransferFund", server);
        cmd.CommandType = CommandType.StoredProcedure;
        cmd.Parameters.AddWithValue("@noRekeningPengirim", this.noRekening);
        cmd.Parameters.AddWithValue("@noRekeningTujuan",
            _penerima.NoRekening);
        cmd.Parameters.AddWithValue("@jumlahUang", this.mutasi);
        return cmd.ExecuteNonQuery();
    }
    catch (Exception)
    {
        return -1;
    }
    finally
    {
        server.Close();
    }
}
}
#endregion
}
}

```

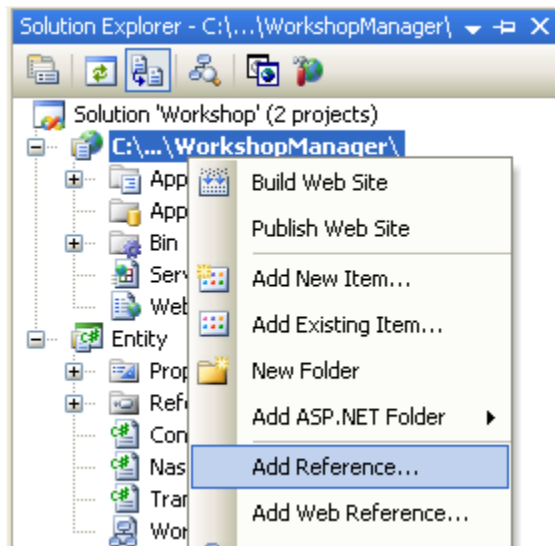
- Setelah semua kode diatas diketikan, pada **Solution Explorer**, klik kanan project **"Entity"** kemudian pilih menu **Build**. Untuk lebih jelasnya lihat gambar dibawah ini:



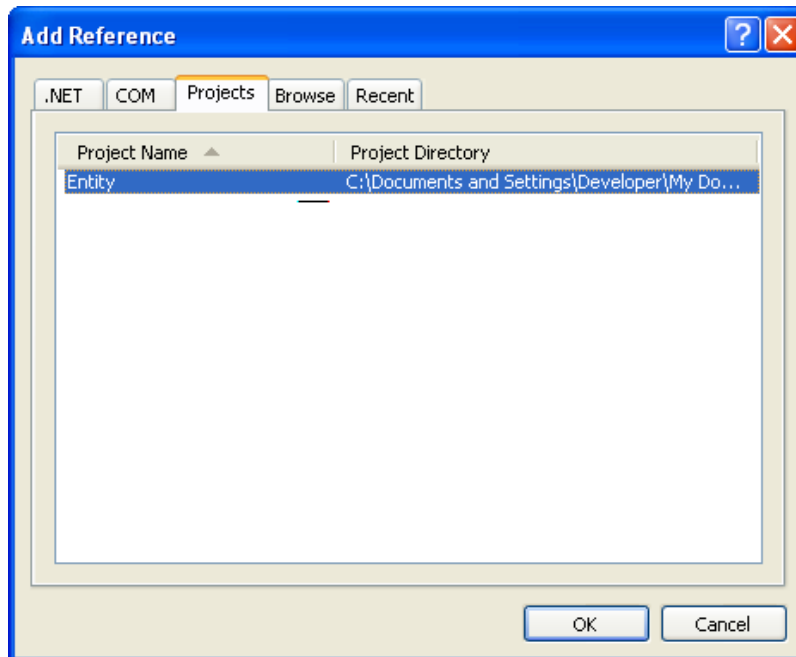
- Perbaiki kesalahan dalam code yang mungkin terjadi, dan kemudian **Build** ulang sampai kesalahan tidak muncul lagi.

4. Pembuatan Web Service

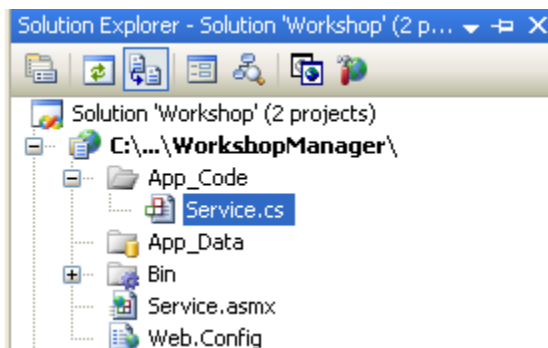
- Dalam **Solution Explorer**, klik kanan project "**Workshop Manager**" dan pilih menu **Add Reference**. Untuk lebih jelasnya lihat gambar dibawah ini:



- Dalam jendela **Add Reference**, pilih tab **Project**, kemudian dalam section Project Name, pilih project **Entity**, kemudian klik tombol **OK**. Untuk lebih jelasnya lihat gambar dibawah ini:



- Dalam **Solution Explorer**, di dalam project “**Workshop Manager**” expand node “**App_Code**” dan double klik object **Service.cs**. Untuk lebih jelasnya lihat gambar dibawah ini:



- Ketikkan isi kode, sehingga seluruh kode-nya seperti dibawah ini:

```
using System;
using System.Web;
using System.Web.Services;
using System.Web.Services.Protocols;
using Workshop.Entity;

[WebService(Namespace = "http://workshop.inf.uajy.ac.id/")]
[WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
public class Service : System.Web.Services.WebService
{
    public Service () {

        //Uncomment the following line if using designed components
        //InitializeComponent();
    }

    #region Nasabah
    [WebMethod]
    public Nasabah SearchNasabah(string _noRekening)
    {
        return new Nasabah(_noRekening).ShowDetail();
    }
}
```

```

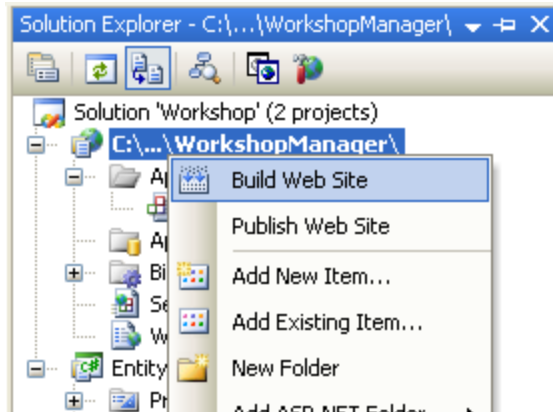
[WebMethod]
public int AddNasabah(string _noRekening, string _nama, string _alamat,
    string _pekerjaan, decimal _setoranAwal)
{
    Nasabah cust = new Nasabah(_noRekening, _nama, _alamat, _pekerjaan,
        _setoranAwal);
    return cust.Add();
}
[WebMethod]
public int UpdateNasabah(string _noRekening, string _nama, string _alamat,
    string _pekerjaan)
{
    Nasabah cust = new Nasabah(_noRekening, _nama, _alamat, _pekerjaan);
    return cust.Update();
}
[WebMethod]
public int DeleteNasabah(string _noRekening)
{
    Nasabah cust = new Nasabah(_noRekening);
    return cust.Delete();
}
[WebMethod]
public decimal GetSaldo(string _noRekening)
{
    Nasabah cust = new Nasabah(_noRekening);
    return cust.GetSaldo();
}
}
#endregion

#region Trasaction

[WebMethod]
public int Saving(string _noRekening, decimal _jumlahUang, string _keterangan)
{
    Nasabah cust = new Nasabah(_noRekening);
    cust.Transaksi.Mutasi = _jumlahUang;
    cust.Transaksi.Keterangan = _keterangan;
    return cust.Transaksi.Saving();
}
[WebMethod]
public int WithDraw(string _noRekening, decimal _jumlahUang, string _keterangan)
{
    Nasabah cust = new Nasabah(_noRekening);
    cust.Transaksi.Mutasi = _jumlahUang;
    cust.Transaksi.Keterangan = _keterangan;
    return cust.Transaksi.WithDraw();
}
[WebMethod]
public int TransferFund(string _noRekeningPengirim, string _noRekeningPenerima,
    decimal _jumlahUang)
{
    Nasabah pengirim = new Nasabah(_noRekeningPengirim);
    Nasabah penerima = new Nasabah(_noRekeningPenerima);
    pengirim.Transaksi.Mutasi = _jumlahUang;
    return pengirim.Transaksi.TransferFund(penerima);
}
}
#endregion
}

```

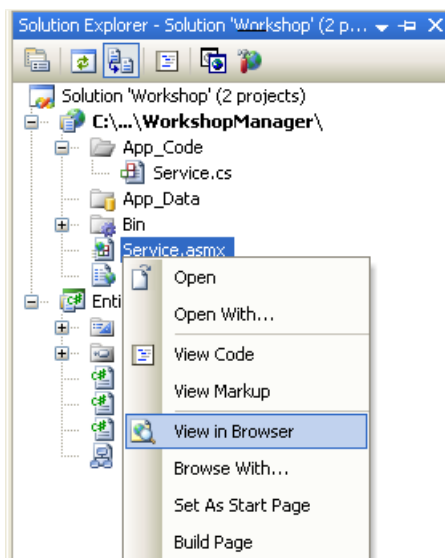
- Setelah semua kode diatas diketikan, pada **Solution Explorer**, klik kanan project **“WorkshopManager”** kemudian pilih menu **Build Web Site**. Untuk lebih jelasnya lihat gambar dibawah ini:



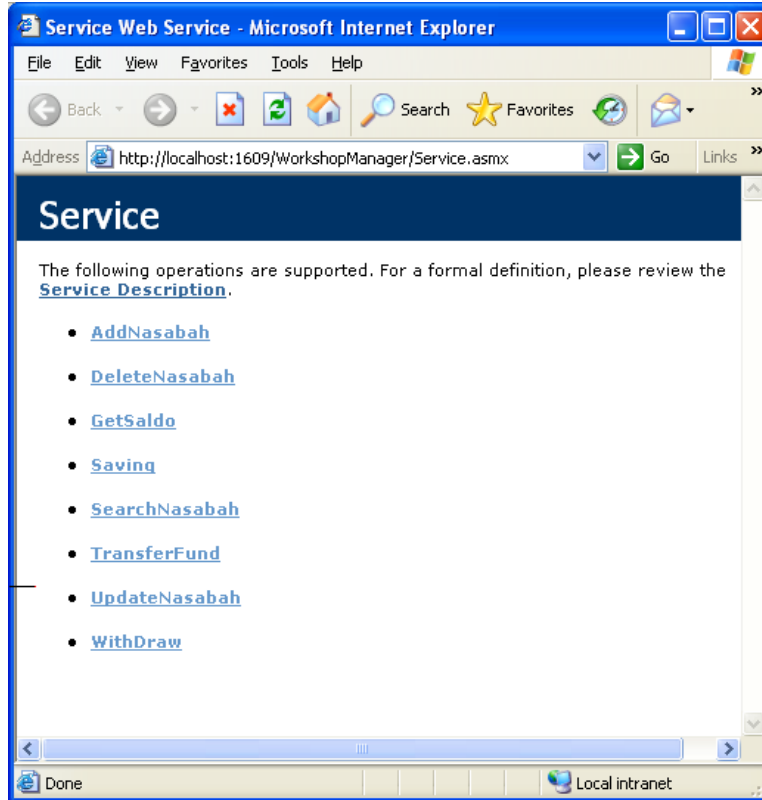
- Perbaiki kesalahan dalam code yang mungkin terjadi, dan kemudian **Build** ulang sampai kesalahan tidak muncul lagi.

5. Pengetesan Web Service

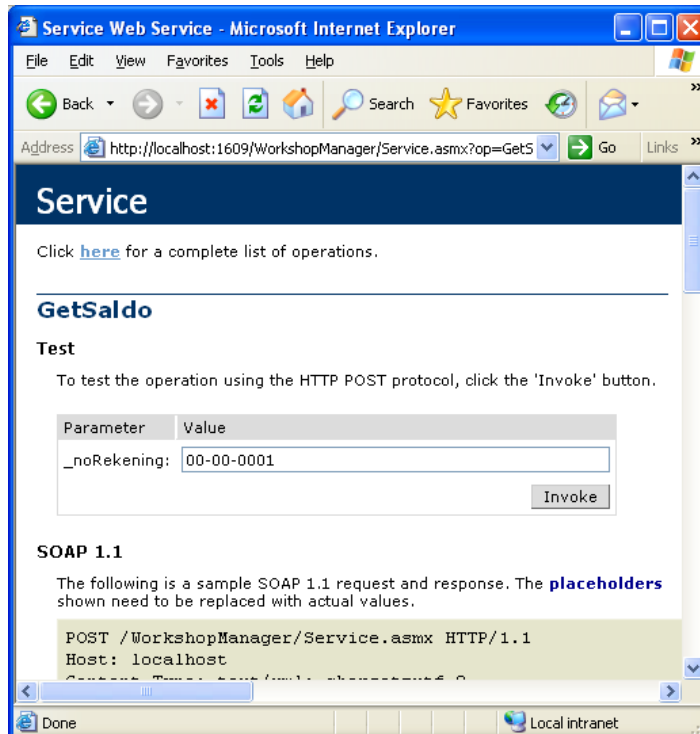
- Dalam **Solution Explorer**, di dalam project “**Workshop Manager**” klik kanan object **Service.asmx** kemudian pilih menu **View in Browser**. Untuk lebih jelasnya lihat gambar dibawah ini:



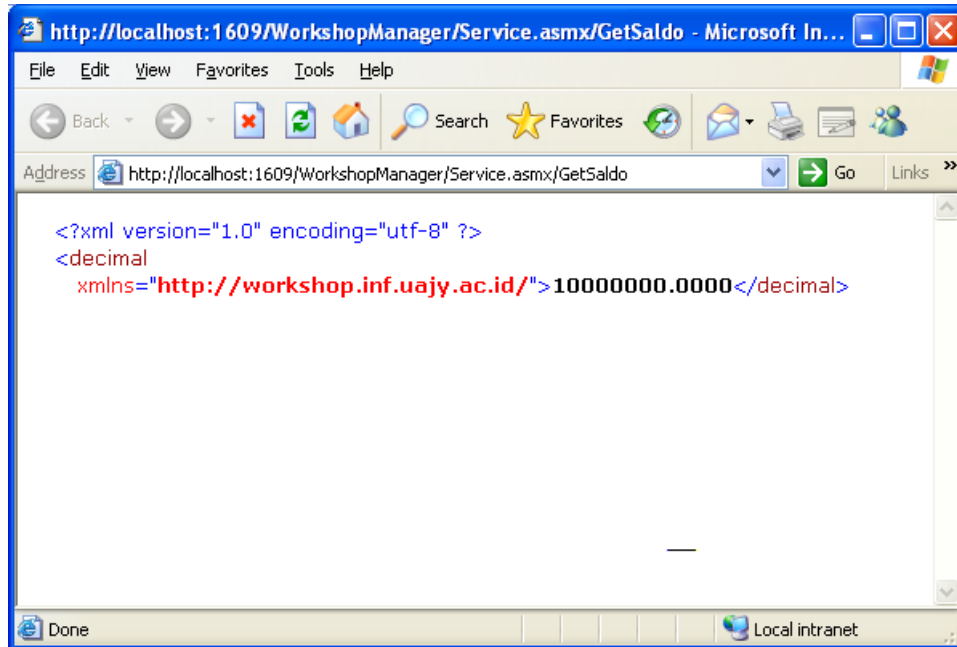
- Akan terbuka web browser, yang menampilkan daftar layanan yang bisa dipanggil, yang tampilan-nya kurang lebih seperti gambar dibawah ini:



- Klik layanan **GetSaldo**, kemudian isikan **no rekening** dari nasabah yang ingin dilihat saldonya, misalnya saja “00-00-0001”, kemudian klik tombol **Invoke** seperti pada gambar dibawah ini:



- Akan keluar hasil dari pemanggilan layanan **GetSaldo** dengan parameter no rekening dari nasabah, yang tampilan-nya kurang lebih seperti pada gambar dibawah ini:



- Ulangi langkah yang hampir sama, untuk mencoba semua layanan dari web service yang telah dibuat.

```
<session id="2" status="end" />
```

If you have questions or want to give me feedback, please contact me:

- Y!M: alex.budiyanto
- e-Mail: alex.budiyanto@student-partners.com