

# KITAB BELAJAR PEMOGRAMMAN

# C#

Muhamad Ali

## KATA PENGANTAR

Puji dan syukur penulis ucapkan kehadirat Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya sehingga dengan niat yang tulus untuk ikut serta berdedikasi dalam dunia ilmu pengetahuan penulis dapat menyelesaikan e-book ini tanpa kendala .

Perkembangan dunia programming saat ini semakin berkembang pesat, banyak para programer pemula atau yang baru ingin belajar memulai suatu bahasa pemograman sulit mendapatkan referensi yang tepat yang dapat membantu mereka untuk belajar. E-book "KITAB BELAJAR PEMOGRAMMAN C#" ini mulanya disusun untuk melengkapi dan membantu khususnya untuk para pemula yang ingin mempelajari bahasa pemograman C#. Dengan adanya E-book ini diharapkan bisa dijadikan referensi untuk para pengembang yang ingin terjun didunia programming dan memulai belajar dengan menggunakan bahasa C#

Semoga dengan dibuatnya E-book ini selain dapat berguna untuk para pembaca dan juga dapat ikut serta dapat memajukan ilmu pengetahuan di Indonesia. Kemudian penulis juga mengharapkan kritik dan saran dari para pembaca yang akhirnya dapat berguna untuk perbaikan E-book ini kedepanya. Akhir kata penulis sampaikan terimakasih dan mohon maaf sebesar – besarnya apabila dalam E-book ini masih banyak kekurangan dan pada akhirnya penulis sampaikan Wasalamuallaikum WR.WB.

Jakarta, 15 April 2014

Penulis

**DAFTAR ISI**

Halaman

<b>Cover .....</b>	i
<b>Kata Pengantar.....</b>	.iii
<b>Daftar isi.....</b>	.iv
<b>BAB I PENGENALAN C# DAN MICROSOFT VISUAL STUDIO....</b>	1
1.1 Sejarah C# .....	1
1.2 Lingkungan Microsoft Visual Studio.....	2
A. Mengenal Tampilan Microsoft Visual Studio.....	3
B. Membuat Projek pada visual studio .....	6
<b>BAB II DASAR TEORI C#.....</b>	7
2.1 Tipe Data dan Variabel.....	7
A. Tipe Data.....	7
B. Variable .....	8
C. Jenis – Jenis Operator .....	8
<b>BAB III DASAR – DASAR PEMOGRAMAN C#.....</b>	10
3.1 Struktur Program C# .....	10
3.2 Lebih Jauh Mengenal Pemograman C# .....	11
A. Printah untuk mencetak karakter .....	11
B. Perintah Masukan.....	12
C. Macam – Macam Perulangan.....	13
D. Statement Pemilihan(Kondisi) .....	18
E. Array Pada C #.....	22
F. Pengantar OOP (Objek Oriented Programming) pada C#.....	24
<b>BAB IV PENGENALAN WINDOWS FORM.....</b>	38
4.1 Menggunakan Komponen Toolbox pada visual C#.....	38
4.2 Membuat Aplikasi Sederhana pada visual C#.....	52
<b>BAB V DATABASE DI C#.....</b>	81
5.1 Pengenalan MYSQL .....	81
5.2 Perintah Masukan ke Database MYSQL.....	83

5.3 Perintah untuk menampilkan data dari database MYSQL.....	87
5.4 Perintah Update ke Database MYSQL.....	90
5.5 Perintah Delete Database MYSQL.....	93
5.6 Pengenalan MS.Access.....	95
5.7 Insert,Update,Delete dan View dengan C# dan Ms.Access.....	95
<b>BAB VI REPORT DAN DATABASE LANJUTAN.....</b>	<b>101</b>
6.1 Membuat Report menggunakan ReportViewer.....	101
6.2 Membuat Report menggunakan PrintDocument dan PrintPreviewDialog.....	106
6.3 Cara Mudah Koneksi Database .....	110
<b>BAB VII BEREKSPLORASI DENGAN DATAGRIDVIEW.....</b>	<b>114</b>
7.1 Menambahkan Komponen Button Pada Data GridView.....	114
7.2 Menambahkan Komponen Checkbox pada dataGridview.....	116
7.3 Menambahkan komponen ComboBox ke datagridview di C#.....	118
7.4 Menambahkan Image pada DataGridview.....	119
<b>BAB VIII MEMBUAT FILE SETUP DENGAN VISUAL STUDIO</b>	

**Daftar Pustaka****Lampiran-Lampiran**

## BAB I PENGENALAN C# DAN MICROSOFT VISUAL STUDIO

### 1.1 SEJARAH C#

Pada akhir dekade 1990-an, Microsoft membuat program Microsoft Visual J++ sebagai sebuah langkah percobaan untuk menggunakan Java di dalam sistem operasi Windows untuk meningkatkan antarmuka dari Microsoft Component Object Model (COM). Akan tetapi, akibat masalah dengan pemegang hak cipta bahasa pemrograman Java, Sun Microsystems, Microsoft pun menghentikan pengembangan J++, dan beralih untuk membuat penganti J++, kompilernya dan mesin virtualnya sendiri dengan menggunakan sebuah bahasa pemrograman yang bersifat *general-purpose*. Untuk menangani proyek ini, Microsoft merekrut Anders Helsberg, yang merupakan mantan karyawan Borland yang membuat bahasa Turbo Pascal, dan Borland Delphi, yang juga mendesain Windows Foundation Classes (WFC) yang digunakan di dalam J++. Sebagai hasil dari usaha tersebut, C# pun pertama kali diperkenalkan pada bulan Juli 2000 sebagai sebuah bahasa pemrograman modern berorientasi objek yang menjadi sebuah bahasa pemrograman utama di dalam pengembangan di dalam *platform* Microsoft .NET Framework.

Pengalaman Helsberg sebelumnya dalam pendesain bahasa pemrograman seperti Visual J++, Delphi, Turbo Pascal) dengan mudah dilihat dalam sintaksis bahasa C#, begitu pula halnya pada inti *Common Language Runtime* (CLR). Dari kutipan atas *interview* dan makalah-makalah teknisnya ia menyebutkan kelemahan-kelemahan yang terdapat pada bahasa pemrograman yang umum digunakan saat ini, misalnya C++, Java, Delphi, ataupun Smalltalk. Kelemahan-kelemahan yang dikemukakannya itu yang menjadi basis CLR sebagai bentukan baru yang menutupi kelemahan-kelemahan tersebut, dan pada akhirnya memengaruhi desain pada bahasa C# itu sendiri. Ada kritik yang menyatakan C# sebagai bahasa yang berbagi akar dari bahasa-bahasa pemrograman lain.<sup>[1]</sup> Fitur-fitur yang diambilnya dari bahasa C++ dan Java adalah desain berorientasi objek, seperti *garbage collection*, *reflection*, akar kelas (*root class*), dan juga penyederhanaan terhadap pewarisan jamak (*multiple inheritance*). Fitur-fitur tersebut di dalam C# kini telah diaplikasikan terhadap iterasi, properti, kejadian (*event*), *metadata*, dan konversi antara tipe-tipe sederhana dan juga objek.

C# didisain untuk memenuhi kebutuhan akan sintaksis C++ yang lebih ringkas dan *Rapid Application Development* yang 'tanpa batas' (dibandingkan dengan RAD yang 'terbatas' seperti yang terdapat pada Delphi dan Visual Basic).

Agar mampu mempromosikan penggunaan besar-besaran dari bahasa C#, Microsoft, dengan dukungan dari Intel Corporation dan Hewlett-Packard, mencoba mengajukan standardisasi terhadap bahasa C#. Akhirnya, pada bulan Desember 2001, standar pertama pun diterima oleh European Computer Manufacturers Association atau *Ecma International* (ECMA), dengan nomor standar ECMA-334. Pada Desember 2002, standar kedua pun diadopsi oleh ECMA, dan tiga bulan kemudian diterima oleh *International Organization for Standardization* (ISO), dengan nomor standar ISO/IEC 23270:2006.

## 1.2 Lingkungan Microsoft Visual Studio

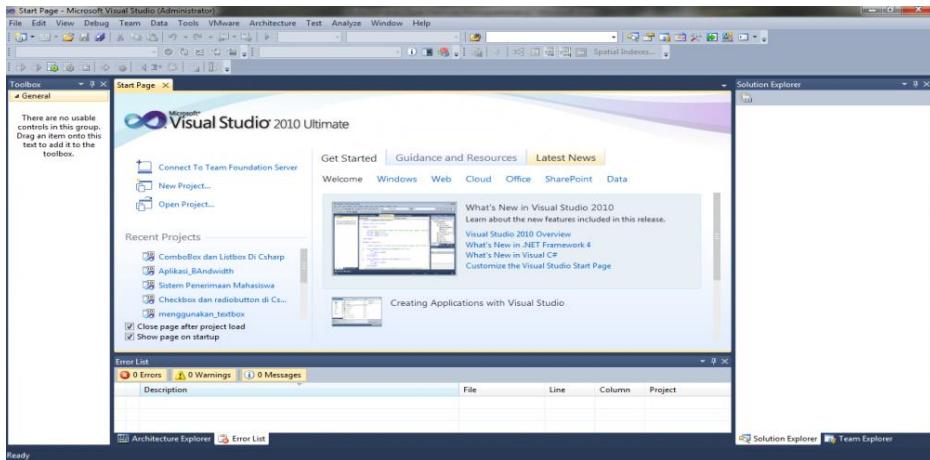
**Microsoft Visual Studio** adalah sebuah lingkungan pengembangan terpadu (IDE) dari Microsoft . Hal ini digunakan untuk mengembangkan program komputer untuk sistem operasi Microsoft Windows superfamili, serta situs web , aplikasi web dan layanan web . Visual Studio menggunakan Microsoft platform pengembangan perangkat lunak seperti API Windows , Windows Forms Windows Presentation Foundation, Windows Stor dan Microsoft Silverlight . Hal ini dapat menghasilkan baik kode asli dan kode yang dikelola .

Visual Studio mencakup kode editor pendukung IntelliSense serta refactoring kode . Terintegrasi debugger bekerja baik sebagai source-level debugger dan debugger mesin-tingkat. Built-in tools termasuk desainer bentuk untuk membangun GUI aplikasi, web designer , kelas desainer, dan skema database desainer. Ia menerima plug-in yang meningkatkan fungsionalitas pada hampir setiap tingkat-termasuk menambahkan dukungan untuk sumber-kontrol sistem (seperti Subversion dan Visual SourceSafe ) dan menambahkan toolsets baru seperti editor dan desainer visual untuk bahasa domain-spesifik atau toolsets untuk aspek-aspek lain dari siklus pengembangan perangkat lunak (seperti Team Foundation Server klien: Tim Explorer).

Visual Studio mendukung berbagai bahasa pemrograman dan memungkinkan kode editor dan debugger untuk mendukung hampir semua bahasa pemrograman, memberikan layanan bahasa spesifik ada. Built-in bahasa termasuk C , C + + dan C + + / CLI (melalui Visual C + + ), VB.NET (melalui Visual Basic. NET ), C # (via Visual C # ), dan F # (seperti Visual Studio 2010 ). Dukungan untuk bahasa lain seperti M , Python , dan Ruby antara lain tersedia melalui layanan bahasa diinstal secara terpisah. Ini juga mendukung XML / XSLT , HTML / XHTML , JavaScript dan CSS . Individu versi bahasa-spesifik Visual Studio juga ada yang menyediakan layanan bahasa yang lebih terbatas bagi pengguna: Microsoft Visual Basic, Visual J #, Visual C #, dan Visual C + +. [2]

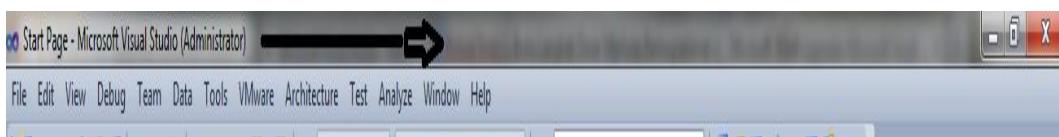
## A. Mengenal Tampilan Microsoft Visual Studio.

### 1. Tampilan Visual Studio Secara keseluruhan



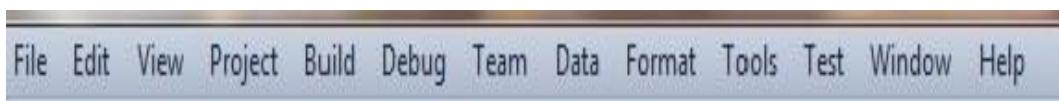
### 2. Title bar

Title Bar adalah informasi nama project yang sedang dibuat.



### 3. Menu bar

Menu bar yang terdapat pada program– program aplikasi di Windows. Menu Bar digunakan untuk melakukan proses atau perintah- perintah tertentu. Menu bar dibagi menjadi beberapa pilihan sesuai dengan kegunaan nya, seperti menu bar File digunakan untuk memproses atau menjalankan perintah-perintah yang berhubungan dengan file, seperti membuka file baru,menyimpan file, selain itu juga terdapat Menu Bar lain seperti : Edit, View, Project, Build, Debug, Data, Format, Tools, Window, dan Help.



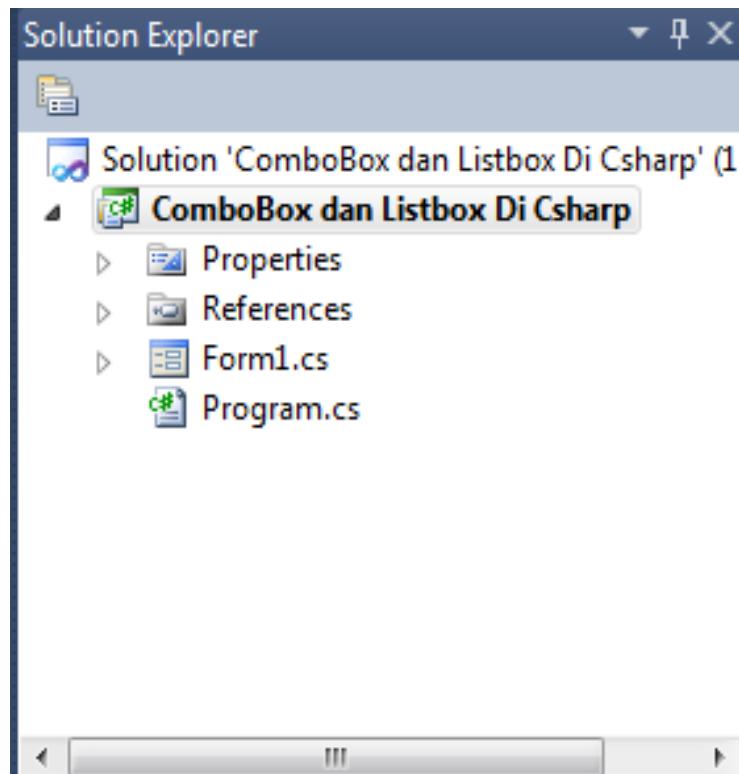
### 4. Toolbars

Toolbars pada aplikasi windows lainnya yang berisi tombol-tombol yang mewakili suatu perintah tertentu yang sering digunakan untuk keperluan dalam pemrograman dan lain-lain, toolbars dapat kita lihat dalam bentuk icon.



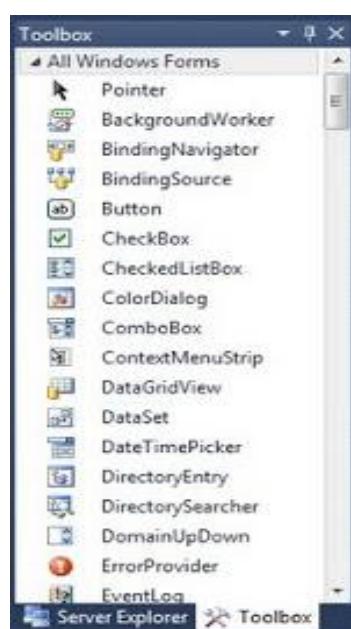
## 5. Solution Explorer

Solution Explorer adalah jendela yang menyimpan Informasi mengenai Solution, Project – project, beserta file-file, form – form ataupun resource yang digunakan pada program aplikasi.



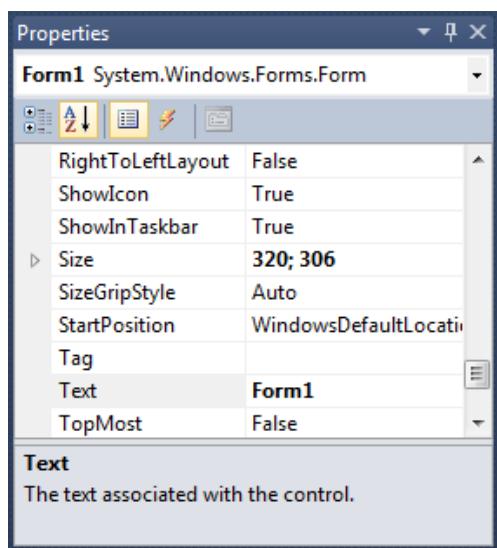
## 6. Toolbox

Toolbox adalah tempat penyimpanan kontrol-kontrol atau komponen standar yang nantinya akan kita letakkan sebagai komponen program didalam Form saat merancang sebuah aplikasi.



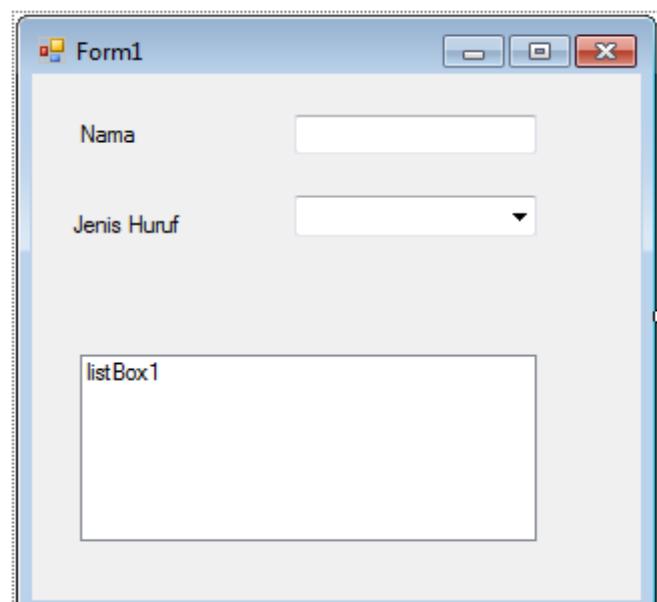
## 7. Properties

Jendela Properties berfungsi untuk memberikan informasi mengenai objek yang sedang aktif, nama objek yang sedang aktif dapat dilihat pada bagian atas jendela Properties. Properties juga digunakan untuk merubah nilai property atau karakteristik dari objek yang aktif.



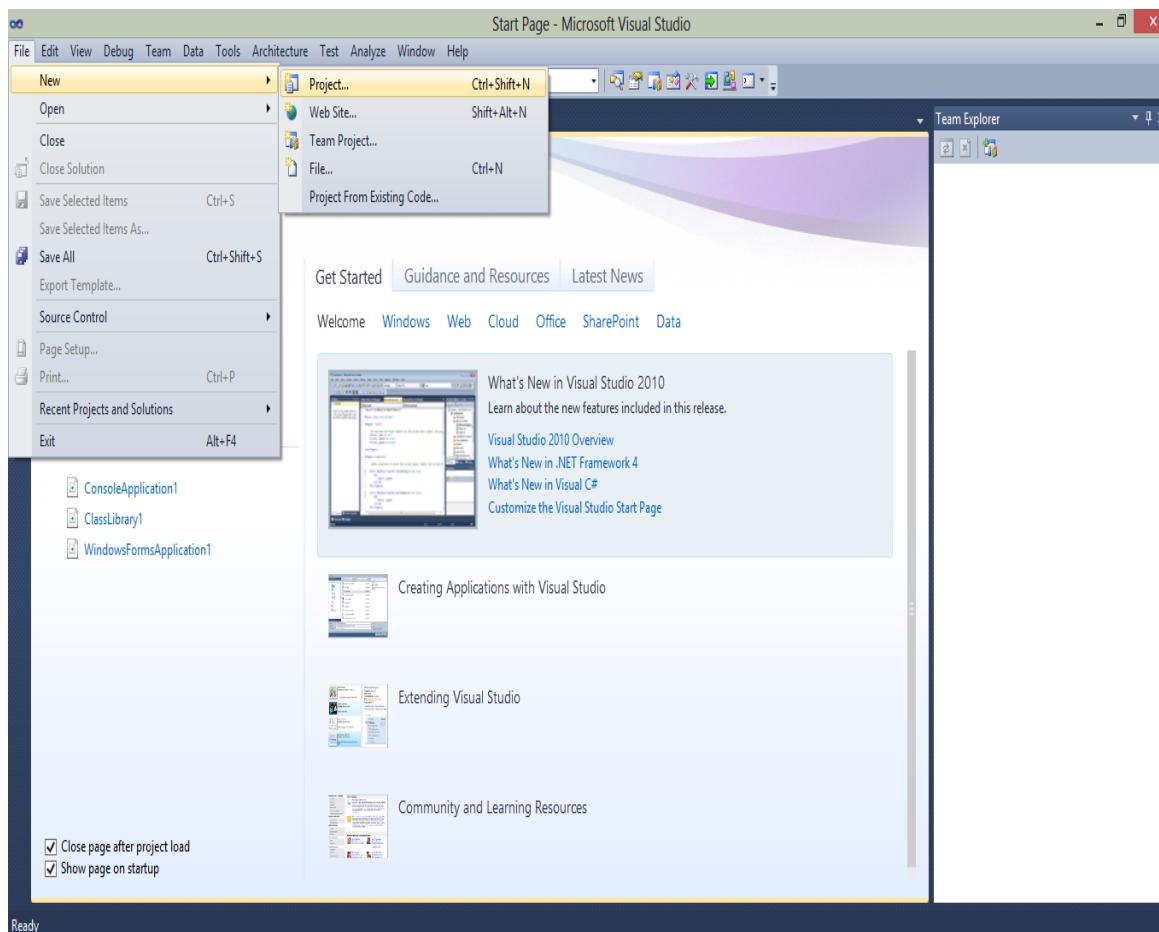
## 8. Form

Form merupakan suatu objek yang digunakan untuk merancang tampilan program.



## B. Membuat Projek pada visual studio

**Project** merupakan kumpulan beberapa file yang kita buat pada lingkungan Microsoft visual studio yang dapat di-compile untuk dapat menghasilkan program atau aplikasi. Berikut ini adalah cara membuat project pada Microsoft Visual Studio.



Pilih Menu File – New - Project

Pilih Visual C#, Console Application, Windows Form atau dll , Beri nama aplikasi (Menentukan Nama Class) , dan pilih tempat penyimpanan project, ok.

## BAB II DASAR TEORI C#

### 2.1 Tipe Data dan Variabel

#### A. Tipe Data

Tipe data merupakan jenis data yang digunakan pada program untuk dioperasikan .

Pada C# terdapat dua jenis tipe data, yaitu:

- Tipe data dasar misalkan integer, float, byte, dan sebagainya.
- Tipe data reference misalkan object dan delegates

Tipe data dasar yang terdapat pada C# adalah:

<b>Tipe Integral</b>	
<b>Byte</b>	8-bit unsigned integer
<b>Sbyte</b>	8-bit signed integer
<b>Short</b>	Short integer
<b>ushort</b>	An Unsigned Short Integer
<b>Int</b>	Integer
<b>Uint</b>	An Unsigned Integer
<b>Long</b>	Long Integer
<b>ulong</b>	An Unsigned Long Integer
<b>Tipe Floating Point</b>	
<b>Float</b>	Single-precision Float Point
<b>double</b>	Double-precision Float Point
<b>Tipe data lain</b>	
<b>char</b>	Character
<b>bool</b>	Memberikan nilai True/False
<b>decimal</b>	Bilangan bulat
<b>String</b>	Berisi karakter alfanumerik atau gabungan dari beberapa karakter

Pada Saat kita mendeklarasikan suatu variable dengan tipe data integer atau int, data tersebut tidak akan bisa diisi dengan huruf atau karakter, tetapi data tersebut dapat diisi dengan angka.

## B. Variable

Adalah suatu tempat menampung data dimemori yang mempunyai nilai atau data yang dapat berubah-ubah selama proses program. Dalam pemberian nama variabel, mempunyai ketentuan-ketentuan antara lain :

1. Tidak boleh ada spasi ( contoh : nilai uas ) dan dapat menggunakan tanda garis bawah ( \_ ) sebagai penghubung (contoh : gaji\_bersih).
2. Tidak boleh diawali oleh angka dan menggunakan operator aritmatika.

Variabel, dibagi menjadi dua jenis kelompok, yaitu :

- Variabel Numerik
- Variabel Teks

## C. Jenis – Jenis Operator

### 1. Operator Aritmatika

Operator aritmatika digunakan untuk perhitungan matematis seperti proses penambahan, pengurangan, perkalian, dan pembagian

Simbol	Keterangan
+	operator penjumlahan
-	operator pengurangan
*	operator perkalian
/	operator pembagian
%	operator sisa bagi

### 2. Operator Relasi

Operator relasi digunakan untuk membandingkan dua buah nilai yaitu (true, false)

Simbol	Keterangan
<	kurang dari
<=	kurang dari sama dengan
>	lebih dari
>=	lebih dari sama dengan
==	sama dengan
!=	tidak sama dengan

### 3. Operator logika

Operator Logika digunakan untuk menghubungkan dua buah operasi relasi menjadi sebuah ungkapan kondisi. Hasil dari operator logika ini menghasilkan nilai numerik 1 (True) atau 0 (False).

Tabel Simbol Operator Logika

Simbol	Keterangan
!	NOT
&&	short-circuit AND
	short-circuit OR

Operator logika diantaranya terdiri dari beberapa macam seperti logika NOT, AND, dan OR. Berikut adalah beberapa table kebenaran operator logika.  
Ket: B = Benar dan S = Salah

Tabel kebenaran operator logika AND

Operan 1	Operan 2	Hasil
B	B	B
B	S	S
S	B	S
S	S	S

Tabel kebenaran operator logika OR

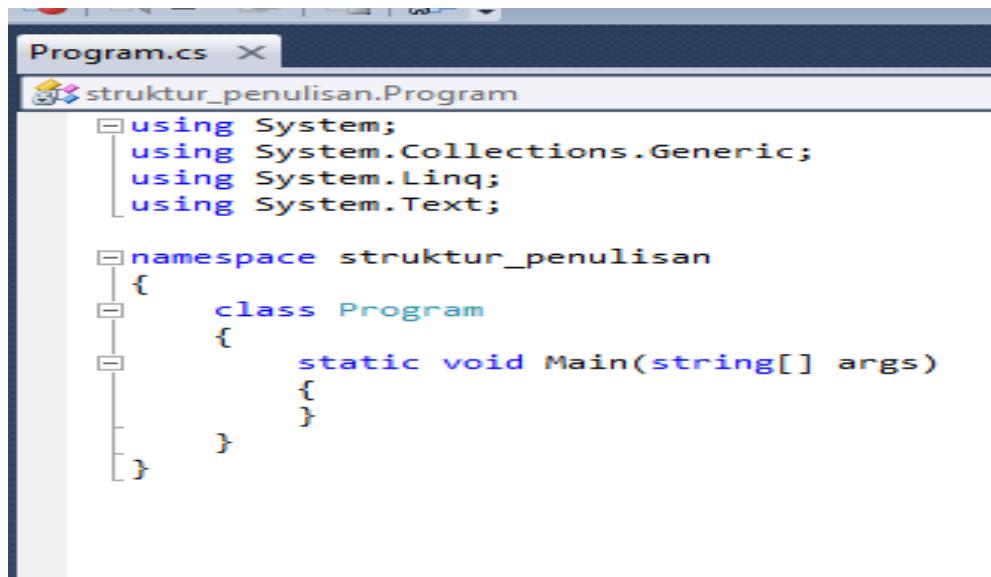
Operan 1	Operan 2	Hasil
B	B	B
B	S	B
S	B	B
S	S	S

Tabel kebenaran operator logika Negasi (NOT)

Operan	Hasil
B	S
B	S
S	B
S	B

## BAB III DASAR – DASAR PEMOGRAMAN C#

### 3.1 Struktur Program C#



```
Program.cs
struktur_penulisan.Program
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace struktur_penulisan
{
    class Program
    {
        static void Main(string[] args)
        {
        }
    }
}
```

Pembahasan :

Keyword using digunakan untuk memberitahu compiler class- class yang sedang digunakan pada aplikasi .

Keyword namespace digunakan untuk mendeklarasikan ruang lingkup dari class yang dibuat.

Static artinya menunjukan suatu tipe method

Void Merupakan suatu method atau objeck tidak di kembalikan.

String[] args merupakan tipe argumen yang akan diterima sebagai parameter, sementara args merupakan array argumen.

#### \*) Komentar Pada C#

Komentar merupakan baris program yang tidak di eksekusi, biasanya komentar digunakan untuk mengingat suatu baris program atau code pada sebuah projek. Berikut adalah contoh komentar :

//Ini komentar jadi tidak mempengaruhi apapun pada program

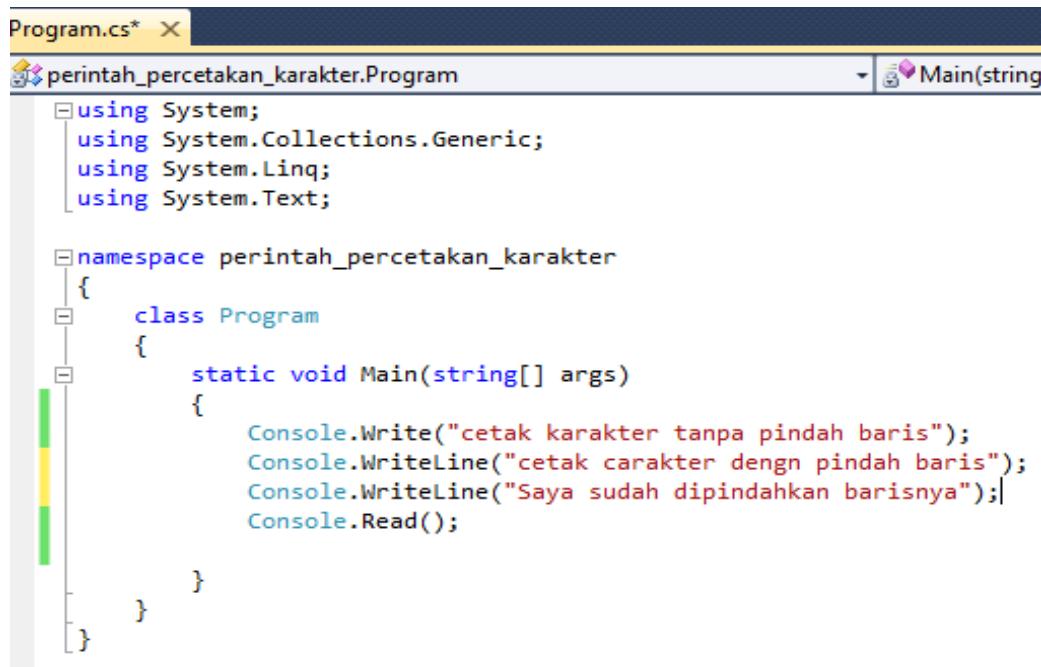
Anda juga bisa membuat komentar pada dua atau lebih baris dengan mengapit dengan tanda /\* dan \*/.

Contoh penggunaan:

```
/*ini juga merupakan komentar , walaupun syntax komentar ini berbeda dengan
yang sebelumnya, namun komentar ini juga tidak akan mempengaruhi program
anda*/
```

### 3.2 Lebih Jauh Mengenal Pemogramman C#

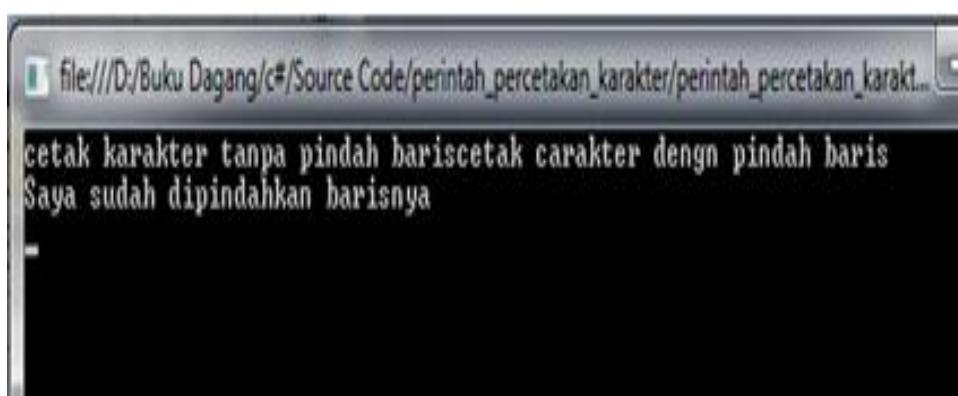
#### A. Printah untuk mencetak karakter :



```
Program.cs* X
perintah_percetakan_karakter.Program Main(string)
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace perintah_percetakan_karakter
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("cetak karakter tanpa pindah baris");
            Console.WriteLine("cetak karakter dengan pindah baris");
            Console.WriteLine("Saya sudah dipindahkan barisnya");
            Console.Read();
        }
    }
}
```

Printah diatas apabila dijalankan maka akan menghasilkan tulisan seperti gambar berikut :



```
cetak karakter tanpa pindah bariscetak karakter dengan pindah baris
Saya sudah dipindahkan barisnya
```

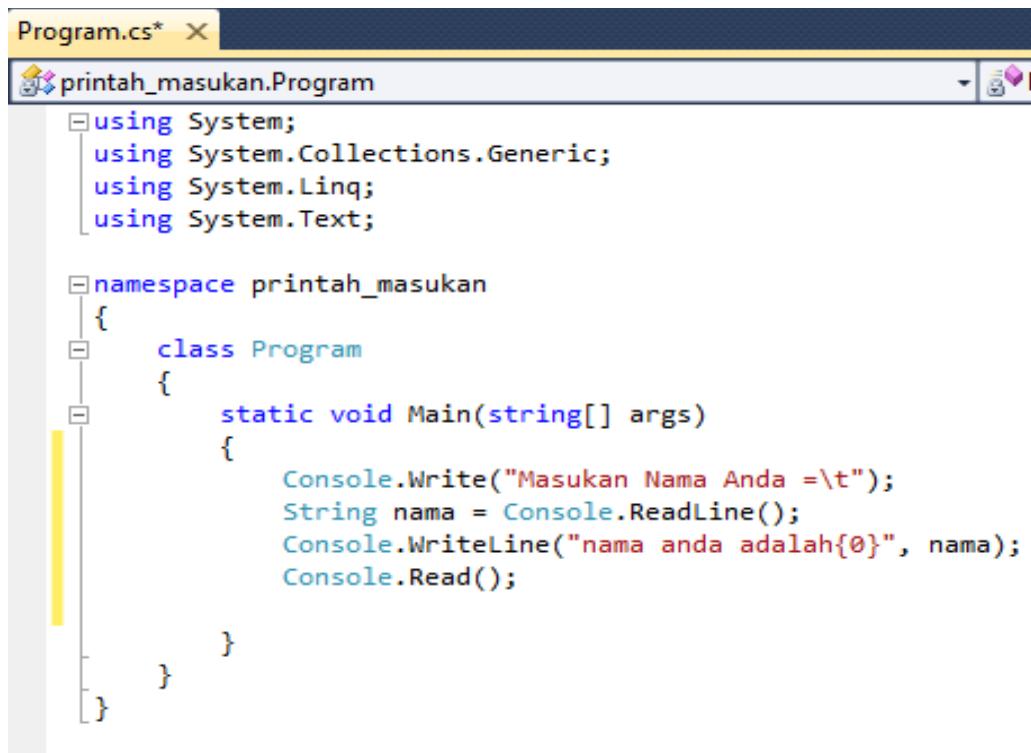
Pembahasan :

Console.Write = cetak karakter tanpa pindah baris .

Console.WriteLine = Cetak karakter dengan pindah baris berikutnya.

Console.Read = untuk membaca atau menangkap karakter

## B. Perintah Masukan



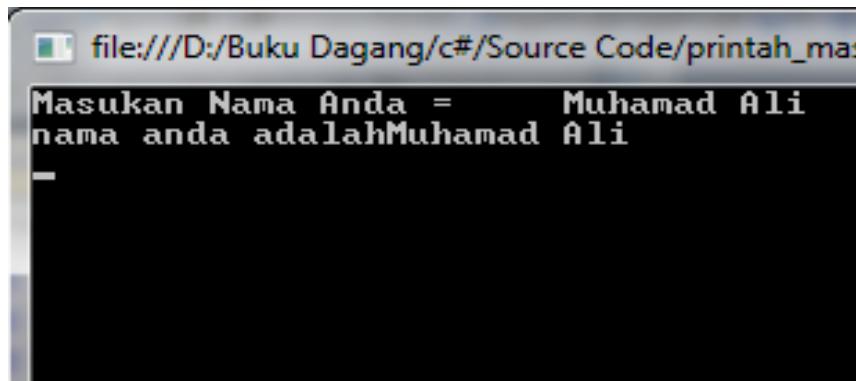
```

Program.cs* ×
printah_masukan.Program
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace printah_masukan
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Masukan Nama Anda =\t");
            String nama = Console.ReadLine();
            Console.WriteLine("nama anda adalah{0}", nama);
            Console.Read();
        }
    }
}

```

Printah diatas apabila dijalankan maka akan menghasilkan tulisan seperti gambar berikut :



```

file:///D:/Buku Dagang/c#/Source Code/printah_ma
Masukan Nama Anda = Muhamad Ali
nama anda adalahMuhamad Ali

```

Pembahasan :

Console.Write = Untuk mencetak karakter

Console.WriteLine = Untuk mencetak karakter disertai pindah baris

\t = Memberikan tab.

String nama = Tipe data string pana variable nama.

Console.ReadLine(); = Perintah masukan dari user.

{0} = Merupakan array.

### C. Macam – Macam Perulangan

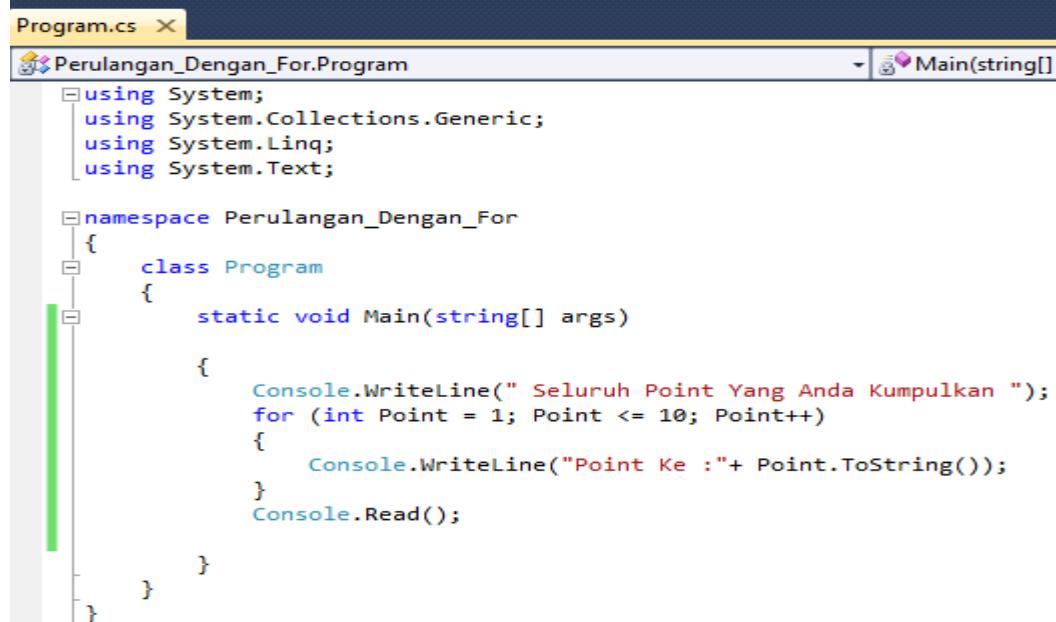
- Perulangan Dengan For

Syntak Struktur Perulangan :

**For (Nilai awal; kondisi; Increment/Decrement)**

```
{
    Statement yang akan diulang;
}
```

Contoh :



The screenshot shows a Microsoft Visual Studio interface with the following code in the 'Program.cs' file:

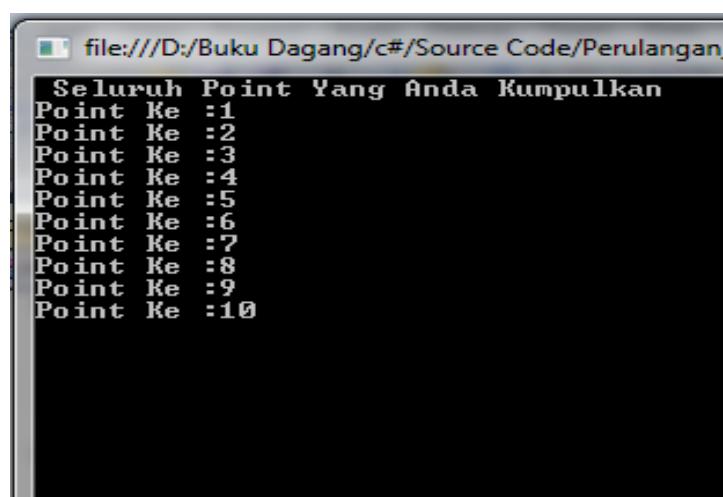
```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Perulangan_Dengan_For.Program
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine(" Seluruh Point Yang Anda Kumpulkan ");
            for (int Point = 1; Point <= 10; Point++)
            {
                Console.WriteLine("Point Ke :" + Point.ToString());
            }
            Console.Read();
        }
    }
}

```

Printah diatas apabila dijalankan maka akan menghasilkan tulisan seperti gambar berikut :



The screenshot shows a terminal window displaying the following output:

```

Seluruh Point Yang Anda Kumpulkan
Point Ke :1
Point Ke :2
Point Ke :3
Point Ke :4
Point Ke :5
Point Ke :6
Point Ke :7
Point Ke :8
Point Ke :9
Point Ke :10

```

Pembahasan :

`Console.WriteLine` = digunakan untuk mencetak karakter disertai dengan pindah baris.

`for (int Point = 1; Point <= 10; Point++)` = merupakan sintak penulisan perulangan for, dimana nilai awal point diberikan 1 dengan tipe data integer, kemudian diberikan kondisi jika kurang dari atau sama dengan sepuluh, maka point bertambah 1 (++) .

---

{

`Console.WriteLine`

`("Point Ke :" + Point.ToString());` = Statement yang akan diulang.

}

---

`+ Point.ToString()` = digunakan untuk merubah tipe data int pada point kedalam string.

### - Perulangan Dengan While

Syntak struktur Perulangan :

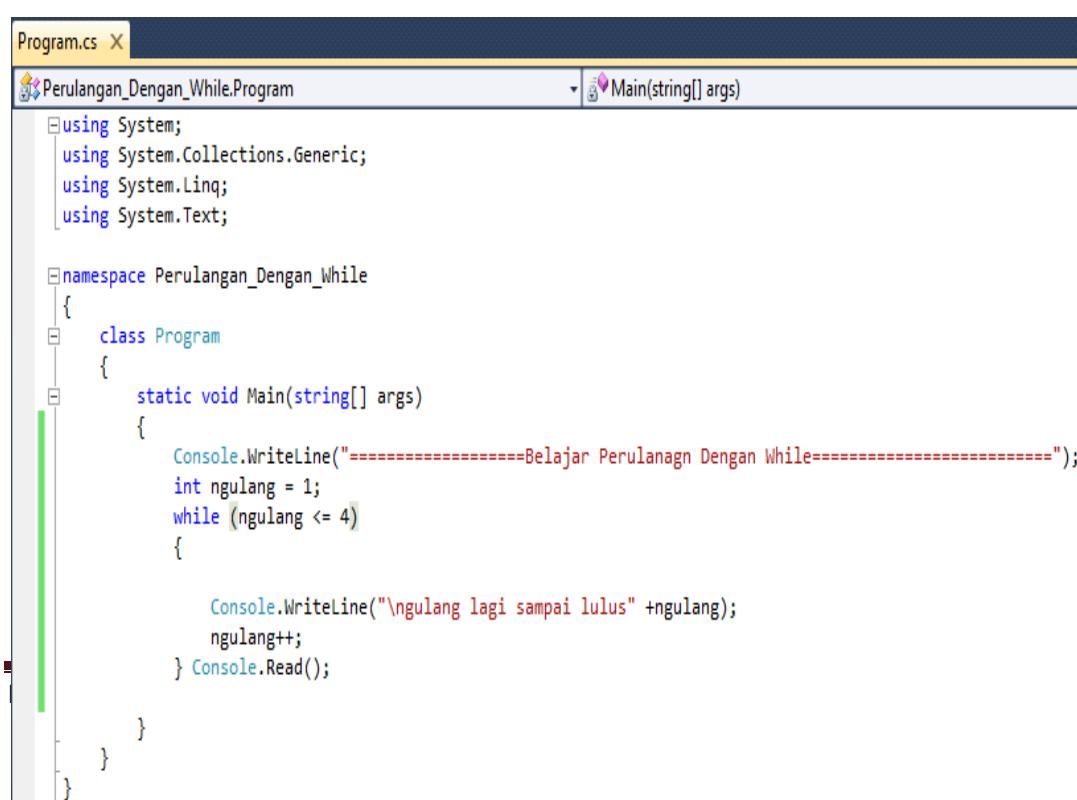
**While (Ekspresi Boolean.....)**

{

**Pernyataan perulangan yang dieksekusi;**

}

Contoh :



```

Program.cs X
Perulangan_Dengan_While.Program Main(string[] args)

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Perulangan_Dengan_While
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("=====Belajar Perulangan Dengan While=====");
            int ngulang = 1;
            while (ngulang <= 4)
            {

                Console.WriteLine("\nNgulang lagi sampai lulus" + ngulang);
                ngulang++;
            } Console.Read();
        }
    }
}

```

Printah diatas apabila dijalankan maka akan menghasilkan tulisan seperti gambar berikut :

```
file:///D:/Buku Dagang/c#/Source Code/Perulangan_Dengan_While/Perulangan_Dengan_While/bi...
=====
gulang lagi sampai lulus1
gulang lagi sampai lulus2
gulang lagi sampai lulus3
gulang lagi sampai lulus4
-
```

Tabel Pembahasan :

<b>int ngulang = 1;</b>	= mendefinisikan tipe data integer pada kata ngulang dengan diberikan nilai 1.
<b>while (ngulang &lt;= 4){</b> <b>Console.WriteLine("\ngulanglagi</b>	= ilustrasinya sintak diatas menjelaskan seandainya nilai pada ngulang kurang dari/sama dengan 4 maka akan mencetak tulisan ngulang lagi sampai lulus sebanyak 4 kali.
<b>sampai lulus" +ngulang);</b> <b>ngulang++;</b> <b>}</b>	= ngulang++ ( ini merupakan nilai ngulang yang akan bertambah terus menerus sampai kondisi/ syarat pada while terpenuhi

### - Perulangan dengan do-while

Syntak Struktur Perulangan :

**Do**

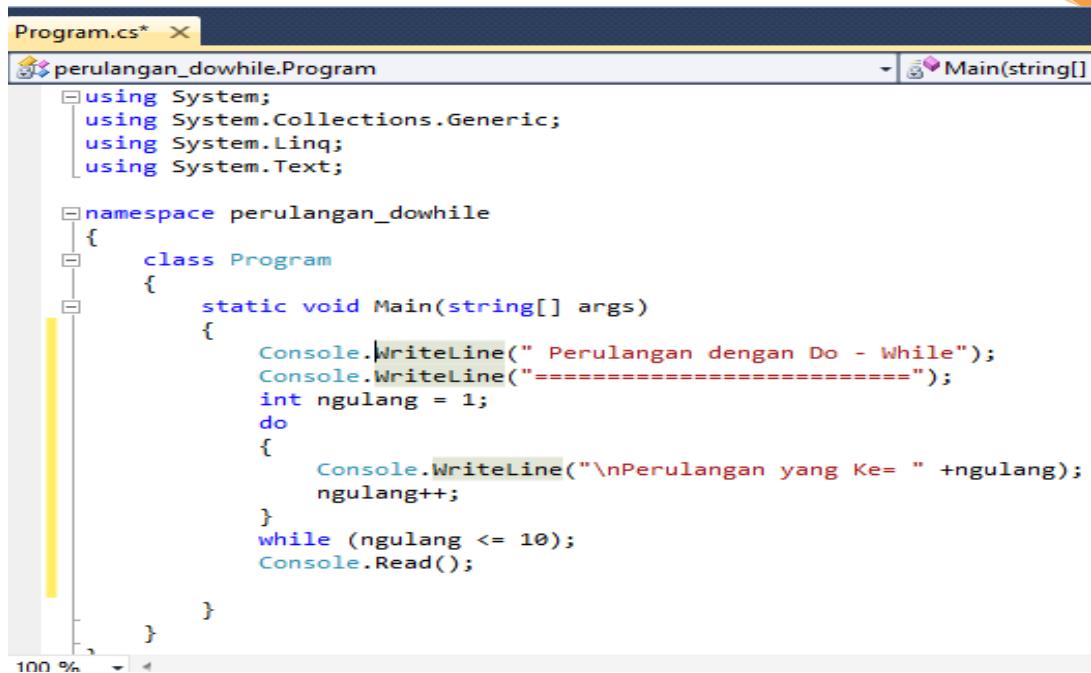
{

**Pernyataan perulangan yang dieksekusi;**

}

**While(Ekspresi Boolean.....)**

Contoh:



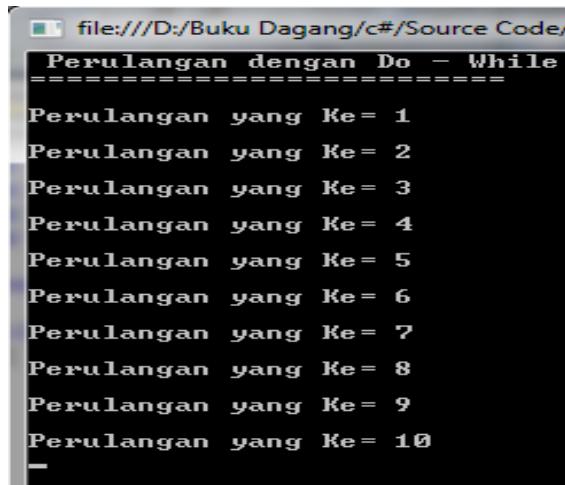
```

Program.cs* ×
perulangan_dowhile.Program
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace perulangan_dowhile
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine(" Perulangan dengan Do - While");
            Console.WriteLine("=====");
            int ngulang = 1;
            do
            {
                Console.WriteLine("\nPerulangan yang Ke= " +ngulang);
                ngulang++;
            }
            while (ngulang <= 10);
            Console.Read();
        }
    }
}

```

Printah diatas apabila dijalankan maka akan menghasilkan tulisan seperti gambar berikut :



```

file:///D:/Buku Dagang/c#/Source Code/
Perulangan dengan Do - While
=====
Perulangan yang Ke= 1
Perulangan yang Ke= 2
Perulangan yang Ke= 3
Perulangan yang Ke= 4
Perulangan yang Ke= 5
Perulangan yang Ke= 6
Perulangan yang Ke= 7
Perulangan yang Ke= 8
Perulangan yang Ke= 9
Perulangan yang Ke= 10
-

```

Tebel Pembahasan :

<code>int ngulang = 1;</code>	= mendefinisikan tipe data integer pada kata ngulang dengan diberikan nilai 1
<code>do</code> <code>{</code> <code>Console.WriteLine("\nPerulangan</code> <code>yang Ke="</code> <code>+ngulang);</code>	= Printah ini berartidijalankan terlebih dahulu sebelum kondisi batas exesuki <code>while (ngulang &lt;= 10);</code> diberikan.
<code>ngulang++;</code> <code>}</code>	= Printah ini berartidijalankan terlebih dahulu sebelum kondisi batas exesuki <code>while (ngulang &lt;= 10);</code> diberikan.

## - Perulangan dengan Foreach

Konsep foreach memiliki kesamaan dengan for. Biasanya digunakan untuk mengambil seluruh data pada list yang ada(array). Bedanya foreach tidak counter, karena proses iterasi dilakukan secara internal array.

Syntak Struktur Perulangan :

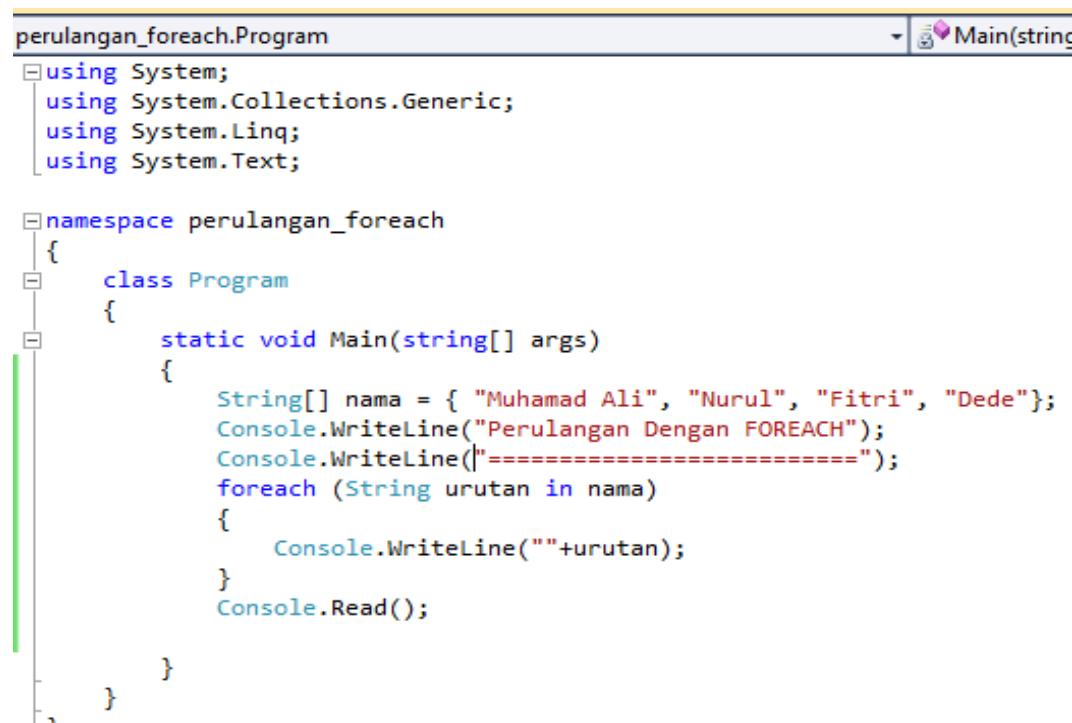
**Foreach (tipe\_data nama\_var in nama\_array)**

{

**Pernyataan ;**

}

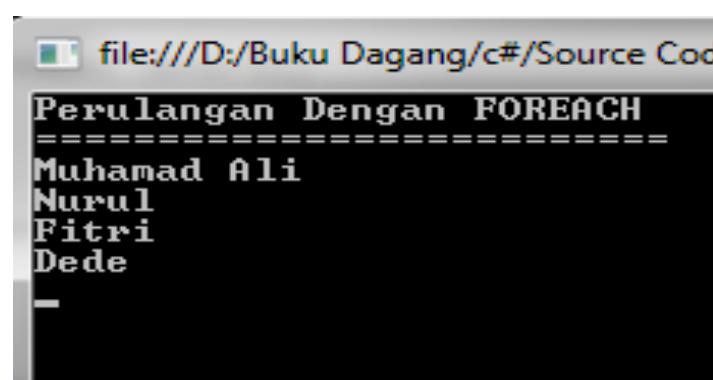
Contoh :



```
perulangan_FOREACH.Program
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace perulangan_FOREACH
{
    class Program
    {
        static void Main(string[] args)
        {
            String[] nama = { "Muhamad Ali", "Nurul", "Fitri", "Dede" };
            Console.WriteLine("Perulangan Dengan FOREACH");
            Console.WriteLine("=====");
            foreach (String urutan in nama)
            {
                Console.WriteLine("'" + urutan);
            }
            Console.Read();
        }
    }
}
```

Printah diatas apabila dijalankan maka akan menghasilkan tulisan seperti gambar berikut :



```
file:///D:/Buku Dagang/c#/Source Cod
Perulangan Dengan FOREACH
=====
Muhamad Ali
Nurul
Fitri
Dede
```

Pembahasan :

`String[] nama = { "Muhamad Ali", "Nurul", "Fitri", "Dede" };`

memberikan tipe data strin pada nama yang diisi dengan baris array ("Muhamad Ali", "Nurul", "Fitri", "Dede").

#### D. Statement Pemilihan(Kondisi)

##### 1. Statement Dengan If-Else

Syntak Struktur Program :

**IF (Kondisi/Syarat)**

{

**Keputusan;**

}

**Else**

{

**Keputusan; }**

Contoh :

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

```
namespace Kondisi_Dengan_If_Else
```

```
{
```

```
class Program
```

```
{
```

```
static void Main(string[] args)
```

```
{
```

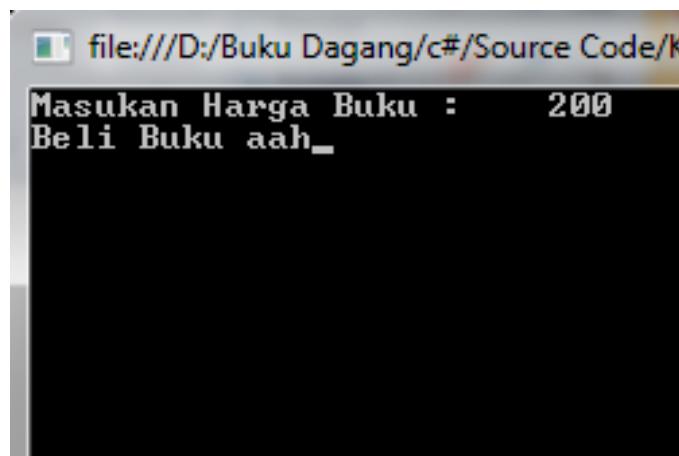
```
string harga;
```

```
Console.Write("Masukan Harga Buku :\t");
```

```
harga = Console.ReadLine();
```

```
int hargabuku = Convert.ToInt32(harga);
if (hargabuku < 25000)
{
    Console.WriteLine("Beli Buku aah");
}
else
{
    Console.WriteLine(" Mahal bangetsih gak jadi belideh");
}
Console.Read();
}
```

Hasil Program Setelah Di Running:



Pembahasan :

**string** harga = memberikan tipe data string pada harga.

harga = **Console.ReadLine()**; = Perintah inputan atau masukan ini hanya bias dilakukan oleh tipe data yang berupa string.

**int** hargabuku = **Convert.ToInt32**(harga); = untuk mengkonversi tipe data string pada harga ke tipe data integer pada hargabuku ( hal ini dilakukan karena pada kondisi if-else yang akan di bandingkan berupa tipe data integer).

Tabel pembahasan

<pre><b>if</b> (hargabuku &lt; 25000) {     <b>Console.WriteLine</b>("Beli Buku aah"); } <b>else</b> {     <b>Console.WriteLine</b>(" Mahal banetsih gak jadi belideh"); }  <b>Console.ReadLine</b>;</pre>	=	<p>Kondisi ini menjelaskan jika harga buku kurang dari 25000 maka akan mencetak tulisan “ Beli Buku aah ”, selain dari pada syarat itu maka akan dictak tulisan " Mahal banetsih gak jadi belideh".</p>
--	---	---

## 2. Statement dengan Switch Case

Syntak Struktur Program :

**Switch (Sarat)**

{

**Case kondisi :**

**Keputusan ;**

**Break;**

**Default:**

**Keputusan n;**

**Break;**

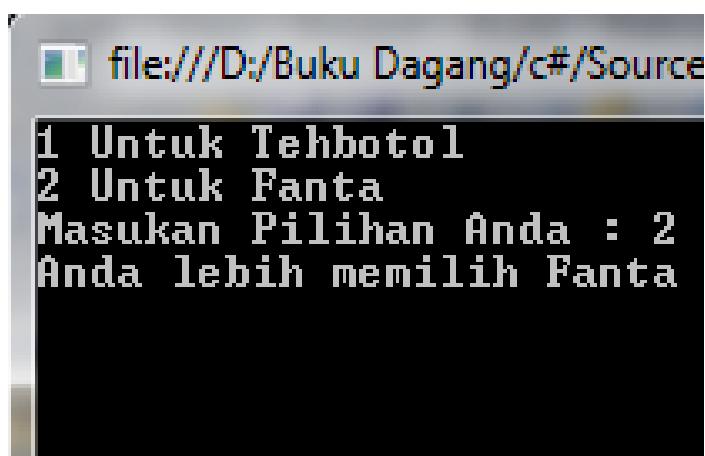
}

Contoh :

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace kondisi_dengan_switch_case
{
    class Program
```

```
{  
    static void Main(string[] args)  
    {  
        string pilihan;  
        Console.WriteLine("1 Untuk Tehbotol \n2 Untuk Fanta");  
        Console.Write("Masukan Pilihan Anda : ");  
        pilihan = Console.ReadLine();  
        switch (pilihan)  
        {  
            case "1":  
                Console.Write("Anda memilih Tehbotol");  
                break;  
            default :  
                Console.Write("Anda lebih memilih Fanta");  
                break;  
        }  
        Console.Read();  
    }  
}
```

Hasil Program Setelah Di Running:



```
file:///D:/Buku Dagang/c#/Source  
1 Untuk Tehbotol  
2 Untuk Fanta  
Masukan Pilihan Anda : 2  
Anda lebih memilih Fanta
```

Tabel Pembahasan :

<pre><b>switch</b> (pilihan) {     <b>case</b> "1":         Console.WriteLine("Anda      memilih Tehbotol");         break;     <b>default</b> :         Console.WriteLine("Anda lebih memilih Fanta");         break; }</pre>	<p>Ilustrasi dari listing disamping yaitu sebagai berikut :</p> <p>= *) <b>switch</b> (pilihan) = Syarat yang dimasukan yaitu pilihan.</p> <p>*) Case = Kondisi dari keputusan yang akan di masukan dalam listing tersebut yaitu 1.</p> <p>*) <b>default</b> : kondisi dari keputusan lain yang akan dieksekusi.</p>
--	--

## E. Array Pada C #

Array adalah sebuah struktur data yang terdiri atas banyak variabel dengan tipe data sama, dimana masing-masing elemen variabel mempunyai nilai indeks.

Contoh :

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace Array_pada_c_sharp
{
    class Program
    {
        static void Main(string[] args)
        {
            //Ini adalah Contoh Array 1 Dimensi
            int[] array = new int[] { 1, 2, 3};//Ini adalah Contoh Array
            Console.WriteLine("\t\t Belajar Bikin Array");//Ini akan dicetak
            Console.WriteLine("\n\n=====      Array      1      Dimensi      Menggunakan
C#=====");
```

```
for (int i = 0; i < array.Length; i++)//membuat perulangan
{
    Console.WriteLine("Array Satu Dimensi" + (i + 1) + " : " + array[i]);
}

//Ini adalah Array 2Dimensi

Console.WriteLine("\n\n\n=====Array      2      Dimensi      Menggunakan
C#=====");

int baris_array = 2;
int klm_array = 2;
int[,] matrik_array = new int[baris_array, klm_array];
int x = 1;

for (int s = 0; s < baris_array; s++)
{
    for (int j = 0; j < klm_array; j++)
    {
        matrik_array[s, j] = x;
        x++;
    }
}

for (int s = 0; s < baris_array; s++)
{
    for (int j = 0; j < klm_array; j++)
    {
        Console.Write("Array 2 Dimensi"+matrik_array[s, j] + "\t");//array yang akan
dicetak
    }
    Console.WriteLine();
}

Console.ReadLine();
}
```

}

Hasil Program Setelah di running :

```
file:///D:/Buku Dagang/c#/Source Code/Array pada c sharp/Array pada  
Belajar Bikin Array  
  
===== Array 1 Dimensi Menggunakan C#=====  
Array Satu Dimensi1 : 1  
Array Satu Dimensi2 : 2  
Array Satu Dimensi3 : 3  
  
=====Array 2 Dimensi Menggunakan C#=====  
Array 2 Dimensi1      Array 2 Dimensi2  
Array 2 Dimensi3      Array 2 Dimensi4  
-
```

## F. Pengantar OOP (Objek Oriented Programming) pada C#.

Objek Oriented Programming adalah sebuah teknik pemrograman modern yang memanggap segala sesuatu adalah objek. Dalam sebuah pemograman berorientasi objek, segala sesuatu ditulis dalam kelas dan kelas – kelas inilah yang akan membentuk sebuah objek(Wahana Komputer:2012).Object-oriented programming (OOP) merupakan paradigma pemrograman yang menggunakan obyek dan interaksinya untuk merancang aplikasi dan program komputer. Ada beberapa konsep pemrograman dasar dalam OOP yang dapat digunakan diantaranya yaitu :

- Class dan objek
- Encapsulation (Enkapsulasi menyembunyikan rincian implementasi dari sebuah kelas dari objek lain. didalamnya terdapat akses identifier yang menentukan siapa saja yang dapat diakses)
- Pewarisan Sifat /Inheritance (Warisan adalah cara untuk membentuk kelas baru menggunakan kelas yang telah ditetapkan)
- polimorfisme (Polimorfisme adalah proses menggunakan operator atau fungsi dalam cara yang berbeda untuk input data yang berbeda.)

Beberapa hal yang merupakan keuntungan dari konsep pemrograman berbasis object (PBO/OOP) adalah:

- Objek-objek yang dibuat bersifat reusable, sehingga dapat digunakan untuk program-program lain.
- Struktur program lebih jelas, trackable (kesalahan mudah dilacak), dan mudah untuk dikembangkan.

## 1. Class

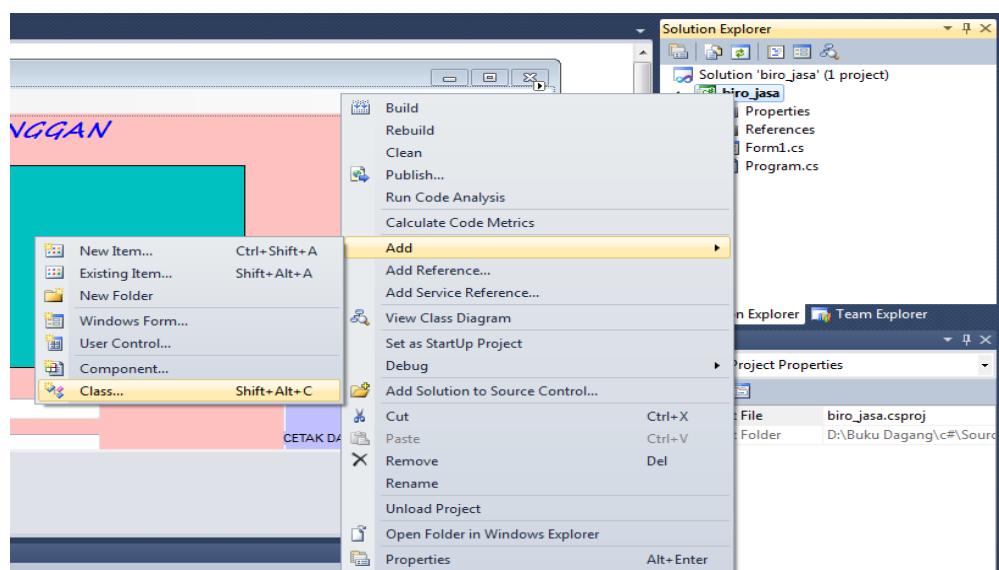
Secara umum Class merupakan rancangan/sketsa atau wadah dari sebuah objek. Jika dianalogikan sebagai sebuah lukisan maka klas merupakan sketsa sebelum lukisan itu diwarnai .Class memiliki dua macam anggota yaitu field dan method. Field dapat diartikan sebagai atribut dari object sedangkan method dapat diartikan sebagai aksi/tindakan yang dapat dilakukan oleh sebuah object. Untuk memahami contoh sebuah script yang menerangkan sebuah class

```
class Program //ini merupakan main class main class.
```

```
{
    static void Main(string[] args) // dan ini adalah main method
    {
    }
}
```

Berikut adaah gambar yang menjelaskan cara membuat kelas menggunakan MS.Visual studio C#.

Gambar Membuat Class



## 2. Objek

Jika kelas merupakan sebuah krangkan dari sebuah objek maka objek didalam adalah konsep software yang dibundel bersama-sama.Terdri dari data dan fungsi. Sebuu Objek ini memiliki sebuah aksi dan tindakan,misalnya objek motor yang mempunyai tindakan bergerak, objek kapal yang mempunyai tindakan terbang dan lain sebagainya.

Untuk memahami sebuah Objek dan kelas pada sebuah program silakan buatlah projek baru dan beri nama **Class\_Pada\_OOP** .

- Setelah itu buatlah sebuah class baru dengan nama **class\_pesawat.cs**, lalu ketikan script berikut.

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

namespace Class_Pada_OOP
{
    class class_pesawat
    {

        //ini merupakan atribut / variabel.

        public string nama;
        public string ketinggian;

        //ini merupakan baris Method.
        public void terbang()
        {
            Console.WriteLine("Pesawat dengan nama {0} , sedang take off", this.nama);
        }
        public void sudafterbang()
        {
            Console.WriteLine("Pesawat sekarang berada pada ketinggian {0}", this.ketinggian);
        }

    }
}
```

- Kemudian ketikan juga listing berikut pada **Program.cs**

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

```

using System.Text;

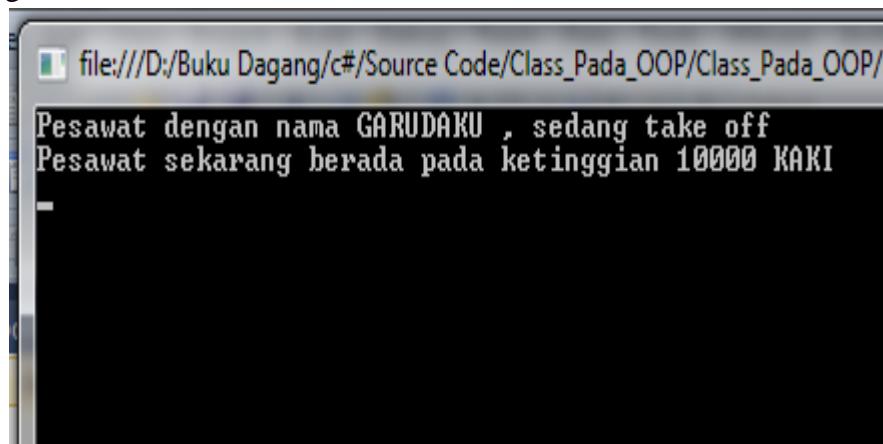
namespace Class_Pada_OOP
{
    class Program
    {
        static void Main(string[] args)
        {
            class_pesawat pesawat = new
class_pesawat();/*instantiasi objek (objek yang akan dibuat harus di
instantiasi terlebih dahulu*)

            pesawat.nama = "GARUDAKU"; //mengakses
atribut class
            pesawat.ketinggian = "10000 KAKI";

            pesawat.terbang();//mengakses method pada
kelas
            pesawat.sudafterbang();
Console.Read();
        }
    }
}

```

- Jika Sudah Jalankan program anda jika berhasil , maka akan seperti gambar berikut .



### 3. Encapsulation

Enkapsulasi sebenarnya merupakan sebuah metode untuk menyembunyikan elemen tertentu dari sebuah class sehingga tidak dapat dikses penuh atau full control oleh kelas lain, ini memungkinkan pengguna lain hanya sebatas dapat membacanya saja atau *read* saja. Sebelum memberikan contoh mengenai Encapsulation ada baiknya kita mengenal yang namanya access identifier karena dalam bahasa pemograman tidaklah lepas dari namanya access identifier yang

memungkinkan atau tidaknya sebuah pengguna dapat mengakses properti yang akan digunakan.

Ada beberapa access identifier yang sering digunakan diantaranya adalah :

a. Public

Public ini berarti bahwa anggota class tersebut (attribut/method/property) dapat diakses oleh siapa saja atau class yang lain.

b. Private

Private berarti bahwa anggota class tersebut hanya boleh diakses oleh class itu sendiri

c. Protected

Protected berarti bahwa anggota class tersebut hanya boleh diakses oleh dirinya sendiri dan Class – Class turunannya.

Tabel Hubungan antara Class ,Atribut dan Method

CLASS	pesawat
ATRIBUT	+ Nama : String -ketinggian : String -JumlahPenumpang : int -JumlahRoda : int
METHOD	+Pesawattakeoff() +pesawatketinggian() +pesawatkapasitas() +rodapesawat()

Untuk membungkus suatu objek pada C# kita dapat menggunakan metode getter dan seter , untuk lebih memahaminya mari langsung saja kita praktikan.

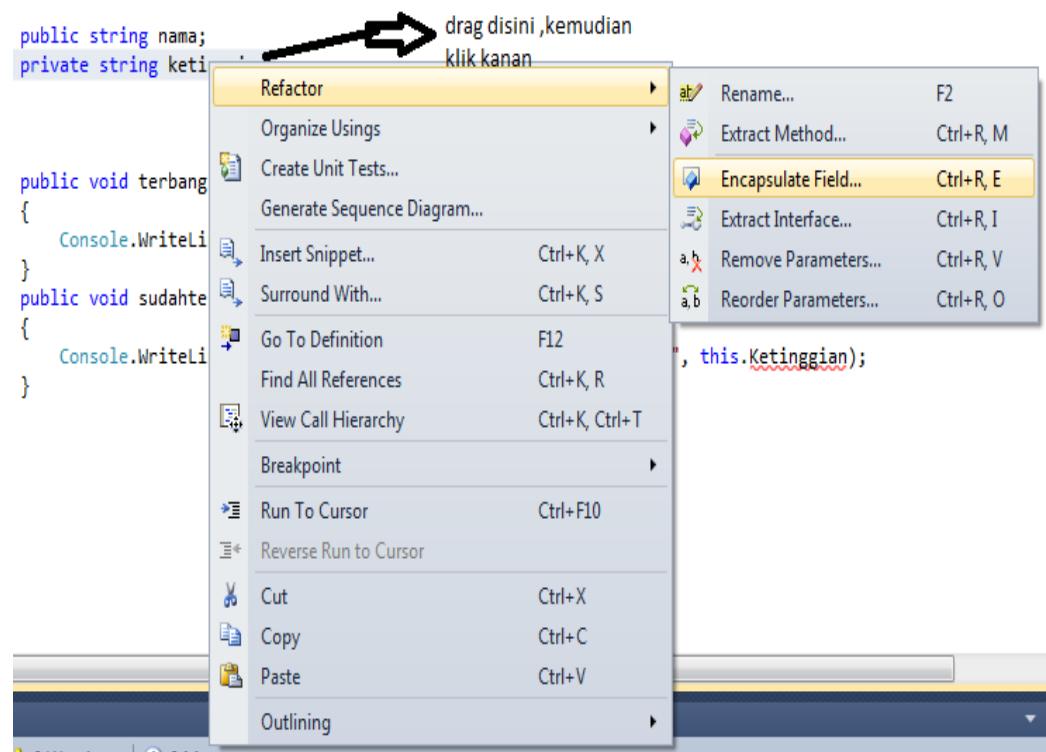
- Silakan buka kembali projek yang anda buat dengan nama **Class\_Pada\_OOP** kemudian buka class **class\_pesawat.cs** dan rubah properti atributnya menjadi seperti dibawah ini.

```
public string nama;  
  
private string ketinggian;
```

- Pada kenyataanya variable ketinggian tidak dapat diakses dari luar kelas karena menggunakan atribut private, pemecahanya kita dapat membungkusnya menggunakan metode getter dan setter agar dapat diakses dari luar kelas perhatikan listing berikut.

```
public string Ketinggian  
{  
    get { return ketinggian; }  
    set { ketinggian = value; }  
}
```

- Anda tidak perlu menulis code tersebut bersusah payah karena dengan visual studio bisa dilakukan dengan cepat dengan namanya property, perhatikan ilustrasi gambar dibawah.



- untuk lebih memahami tentang materi ini silakan perhatikan program keseluruhan berikut.

### Class\_pesawat.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace Class_Pada_OOP
{
    class class_pesawat
    {
        public string nama;
        private string ketinggian;

        public string Ketinggian
        {
            get { return ketinggian; }
            set { ketinggian = value; }
        }
        public void terbang()
    }
  
```

```

    {
        Console.WriteLine("Pesawat dengan nama {0} ,
sedang take off", this.nama);
    }
    public void sudahterbang()
    {
        Console.WriteLine("Pesawat sekarang berada pada
ketinggian {0}", this.Ketinggian);
    }

}
}

```

**Programm.cs**

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Class_Pada_OOP
{
    class Program
    {
        static void Main(string[] args)
        {
            class_pesawat pesawat = new class_pesawat();
            pesawat.nama = "GARUDAKU";
            pesawat.Ketinggian = "10000 KAKI";

            pesawat.terbang();
            pesawat.sudahterbang();
            Console.Read();

        }
    }
}

```

**4. Pewarisan Sifat / Inheritance**

Adalah cara untuk membentuk kelas baru menggunakan kelas yang telah ditetapkan. Ilustrasinya yaitu jika sebuah **class pesawat** mempunya atribut (+ Nama : String, -ketinggian : String, -JumlahPenumpang : int-, JumlahRoda : int), maka atribut dari class pesawat tersebut dapat diwariskan ke kelas lainnya contoh Class Pesawat\_tempur, dan atribut pada class pesawat juga bisa digunakan pada method di class pesawat\_tempur. Contoh :

- Buatlah projek baru dengan nama **Inheritance**, setalah itu tambahan juga class baru dengan nama **pesawat.cs** lalu ketikan listing program dibawah.

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Inheritance
{
    class pesawat
    {
        public string Nama;
        public int JumlahRoda;
        private string ketinggian;
        private string jumlahPenumpang;

        public string Ketinggian
        {
            get { return ketinggian; }
            set { ketinggian = value; }
        }
        public string JumlahPenumpang
        {
            get { return jumlahPenumpang; }
            set { jumlahPenumpang = value; }
        }
        public void terbang()
        {
            Console.WriteLine("Pesawat dengan nama {0}, mempunyai jumlah
roda {1}, sedang berada pada ketinggian{2}dengan
membawa, jumlah penumpang
sebanyak{3}", this.Nama, this.JumlahRoda, this.Ketinggian, this.
JumlahPenumpang);
        }
    }
}

```

- Setelah itu silakan buatlah sebuah Class baru dengan nama **pesawat\_tempur.cs** , kelas ini nantinya akan mewarisi atribut dari **Class pesawat**. Kemudian ketikan listing program berikut.

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Inheritance
{
    class pesawat_tempur:pesawat //ini merupakan inheritance
    dari class pesawat
    {

```

```

        public void terbang()
    {
Console.WriteLine("Pesawat Tempur dengan nama {0}, yang
mempunyai jumlah roda {1}, sedang berada pada ketinggian{2}
dengan membawa jumlah penumpang sebanyak{3} Akan meledakan
senjata", this.Nama, this.JumlahRoda, this.Ketinggian,
this.JumlahPenumpang);
    }
}
}

```

- Terakhir ketikan program dibawah pada **Program.cs**

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Inheritance
{
    class Program
    {
        static void Main(string[] args)
        {
            pesawat pesawat = new pesawat();// instantiasi objek
            pesawat_tempur pswttempur = new pesawat_tempur();// instantiasi objek

            // seting atribut/variabel
            pesawat.Nama = "Helly";
            pesawat.JumlahRoda = 3;
            pesawat.Ketinggian = "1000 kaki";
            pesawat.JumlahPenumpang = "3 penumpang";

            pswttempur.Nama = "AZ500TPU";
            pswttempur.JumlahRoda = 5;
            pswttempur.Ketinggian = "200 kaki";
            pswttempur.JumlahPenumpang = "2 Penumpang ples
kopilot dan asistance";
            //memanggil method dari suatu kelas
            pesawat.terbang();
            pswttempur.terbangtinggi();

            Console.Read();
        }
    }
}

```

## 5. Polimorfisme

Polimorfisme merupakan kemampuan suatu objek untuk mengungkapkan banyak hal melalui satu cara yang sama. Polimorfisme mengizinkan kelas induk untuk mendefinisikan sebuah method general (bersifat umum) untuk semua kelas turunannya, dan selanjutnya kelas-kelas turunan dapat memperbarui implementasi dari method tersebut secara lebih spesifik sesuai dengan karakteristiknya masing-masing. Polimorfisme adalah proses menggunakan operator atau fungsi dalam cara yang berbeda untuk input data yang berbeda. Untuk mengimplementasikan polimorfisme dapat dilakukan menggunakan dua cara yaitu:

### A. Methode overloading

Method overloading adalah membuat dua atau lebih method yang bernama sama, tetapi dengan jumlah atau jenis parameter yang berbeda.

Contoh :

- Buatlah sebuah projek baru ,kemudian buat juga class dengan nama **data**.  
Masukan dua script berikut.
- Script untuk **Data.cs**

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

namespace overloadingmethod
{
    class data
    {
        public string nama;
        public double nomor_telepon;

        public void print(string nama)
        {
            this.nama = nama;
            Console.WriteLine("Namanya adalah : {0}",
this.nama);
        }

        public void print(double nomor_telepon)
        {
            this.nomor_telepon = nomor_telepon;
            Console.WriteLine("Nomor HP : {0}",
this.nomor_telepon);
        }
    }
}
```

- Script untuk Program.cs

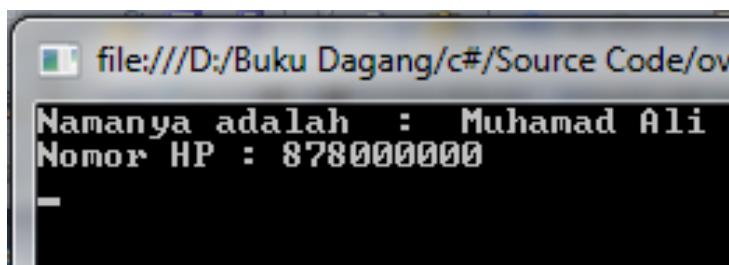
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace overloadingmethod
{
    class Program
    {
        static void Main(string[] args)
        {
            data datanew = new data();

            datanew.print("Muhamad Ali");
            datanew.print(0878000000);

            Console.Read();
        }
    }
}
```

- Hasil jika program dijalankan akan seperti gambar berikut



## B. Methode overriding dengan pewarisan

Overriding digunakan oleh Subclass yang berusaha memodifikasi tingkah laku yang diwarisi dari superclass. Tujuannya adalah supaya subclass memiliki tingkah laku yang lebih spesifik. Ini dilakukan dengan cara mendeklarasikan kembali method milik parent class di subclass.

- Contoh Buatlah sebuah projek dengan nama **overriding**, dan ketikan script berikut.

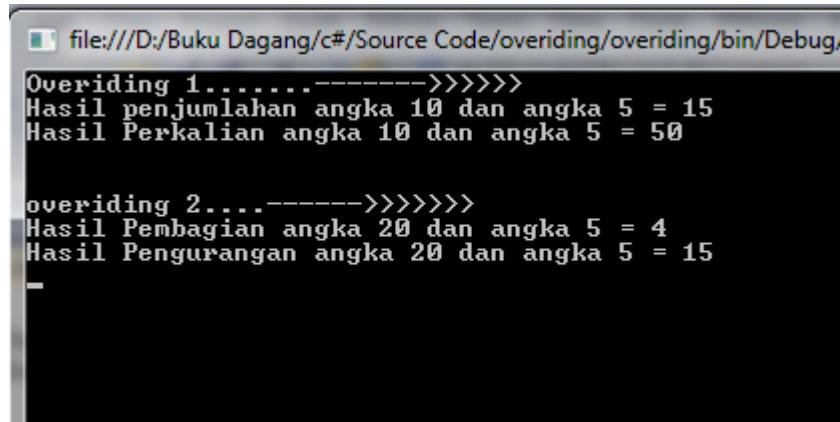
```
using System;
using System.Collections.Generic;
using System.Linq;
```

```
using System.Text;

namespace overriding
{
    class Program
    {
        public int bilangan1, bilangan2;
        public void bilangan(int angka1, int angka2)
        {
            bilangan1 = angka1;
            bilangan2 = angka2;
        }
        public virtual void penjumlahanperkaliandll()
        {
Console.WriteLine("Overiding 1.....----->>>>");
Console.WriteLine("Hasil penjumlahan angka {0} dan angka {1}
= {2}",bilangan1,bilangan2,bilangan1+bilangan2);
Console.WriteLine("Hasil Perkalian angka {0} dan angka {1} =
{2} \n\n", bilangan1, bilangan2, bilangan1 * bilangan2);
        }
    }
    class overide:Program
    {
        public override void penjumlahanperkaliandll()
        {
Console.WriteLine("overiding 2.....----->>>>>");  
Console.WriteLine("Hasil Pembagian angka {0} dan angka {1} =
{2}",bilangan1,bilangan2,bilangan1/bilangan2);
Console.WriteLine("Hasil Pengurangan angka {0} dan angka {1} =
{2}", bilangan1, bilangan2, bilangan1 - bilangan2);
        }
        static void Main(string[] args)
        {
            Program overriding = new Program();
            overriding.bilangan(10, 5);
            overriding.penjumlahanperkaliandll();
            overriding = new overide();
            overriding.bilangan(20, 5);
            overriding.penjumlahanperkaliandll();

            Console.Read();
        }
    }
}
```

- Jalankan program jika berhasil maka hasilnya akan seperti gambar berikut .



```
file:///D:/Buku Dagang/c#/Source Code/overiding/overiding/bin/Debug/Overiding.exe
Overiding 1.....----->>>>
Hasil penjumlahan angka 10 dan angka 5 = 15
Hasil Perkalian angka 10 dan angka 5 = 50

overiding 2.....----->>>>>
Hasil Pembagian angka 20 dan angka 5 = 4
Hasil Pengurangan angka 20 dan angka 5 = 15
-
```

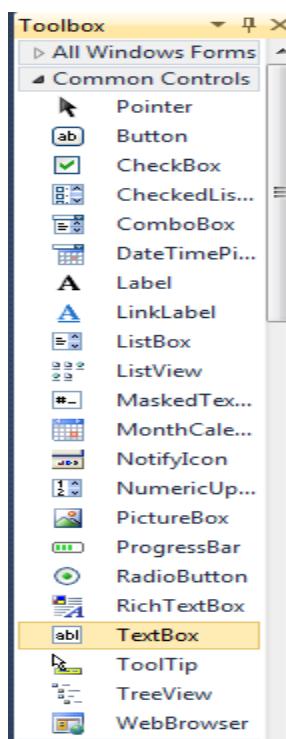
## BAB IV PENGENALAN WINDOWS FORM

### 4.1 Menggunakan Komponen Toolbox pada visual C#

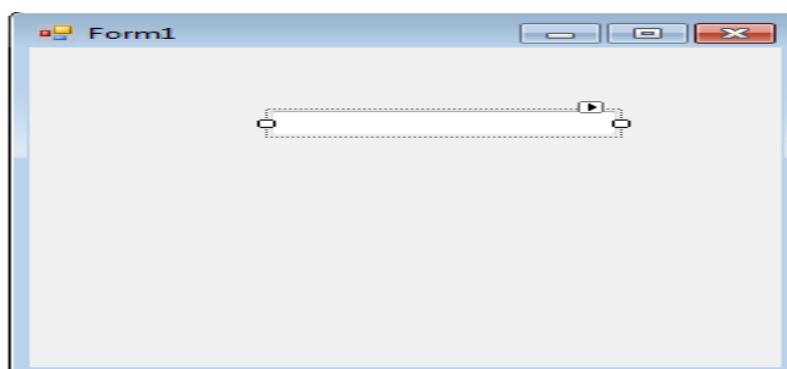
Pada bagian ini kita akan mempelajari bagaimana menggunakan komponen – komponen Toolbox yang sering di gunakan dalam membuat aplikasi menggunakan Microsoft visual studio dengan bahasa C#.

#### A. Mengenal Label Dan TextBox Pada C#

- Silakan Buat projek baru pada visual studio dengan nama : menggunakan\_textbox.
- Setelah itu klik komponen textbox pada toolbox seperti gambar dibawah.



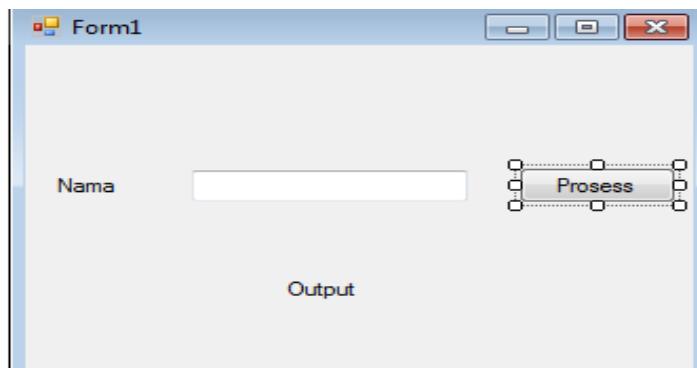
- Kemudian letakan textbox pada desainer Form1.cs[Design] dengan cara klik pada bagian formnya, seperti gambar dibawah.



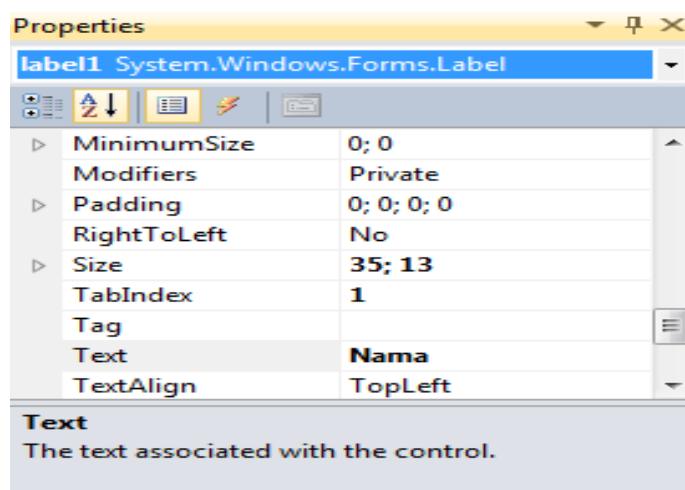
- ambil komponen label sebanyak 2 dan 1 button pada toolbox letakan seperti gambar berikut.



- Kemudian Seting Text Properties label1,label2,dan button sehingga menjadi seperti dibawah ini.



Catatan : properties terletak di pojok kanan bawah pada visual studio. Untuk mensettingnya anda bias melihat gambar dibawah ini:



- Selanjutnya silakan anda klik 2 kali pada button1/button prosess lalu masukan script dibawah ini.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
```

```
namespace menggunakan_textbox
```

```
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
    }
```

```
private void button1_Click(object sender, EventArgs e)
{
    label2.Text = textBox1.Text; /*printah ini akan mencetak inputan dari textbox1 ke
label2*/
}
```

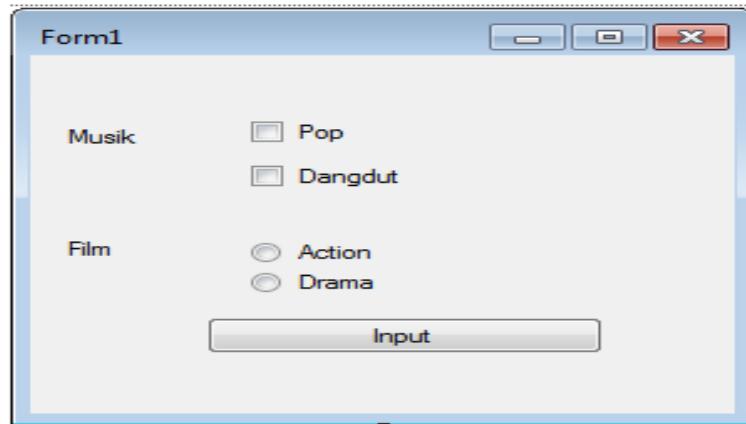
- kemudian tekan f5 pada key board untuk meruning atau menjalankan uplikasi yang sudah dibuat. Sampai sini aplikasi yang kita buat sudh bisa digunakan.

- Jika berhasil maka hasil aplikasi akan terlihat seperti gambar dibawh.



### B. Checkbox dan radiobutton di C#

Silakan Buat Projek Baru berinama checkbox\_dan\_radiobutton, lalu siapkan komponen yang diperlukan seperti 2 label, 2 checbox,2 radio button,dan 1 buah label kemudian atur seperti gambar dibawah.



- Selanjutnya silakan anda klik 2 kali pada button1/button input lalu masukan script dibawah ini.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
```

```
namespace Checkbox_dan_radiobutton_di_Csharp
```

```
{  
public partial class Form1 : Form  
{  
string musik,film;  
public Form1()  
{  
InitializeComponent();  
}  
private void button1_Click(object sender, EventArgs e)  
{  
if (checkBox1.Checked)  
{  
musik = "pop";  
}  
else if (checkBox2.Checked)  
{  
musik = "dangdut";  
}  
else  
{  
musik = "pop, dangdut";  
}  
if (radioButton1.Checked)  
{  
film = "Action";  
}  
else  
{  
film = "Drama";  
}  
}
```

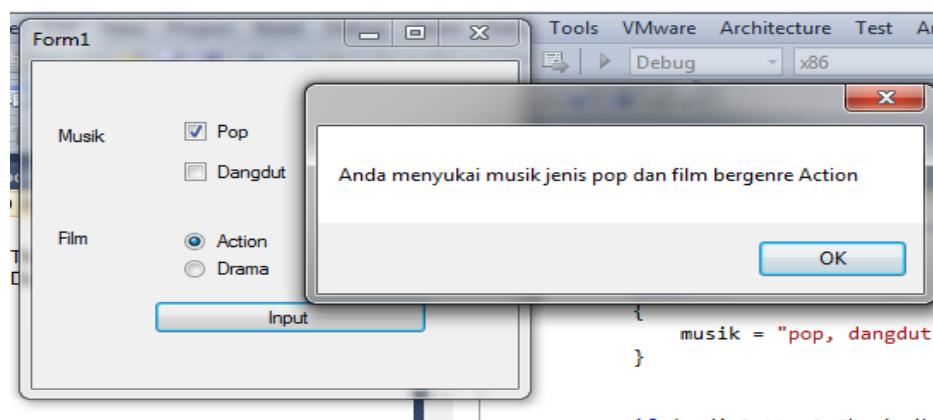
```

    MessageBox.Show("Anda menyukai musik jenis "+musik+" dan film bergenre
"+film+"");
}

}

```

- kemudian tekan f5 pada key board untuk merunning atau menjalankan uplikasi yang sudah dibuat. Jika berhasil maka akan deperti gambar dibawah ini.

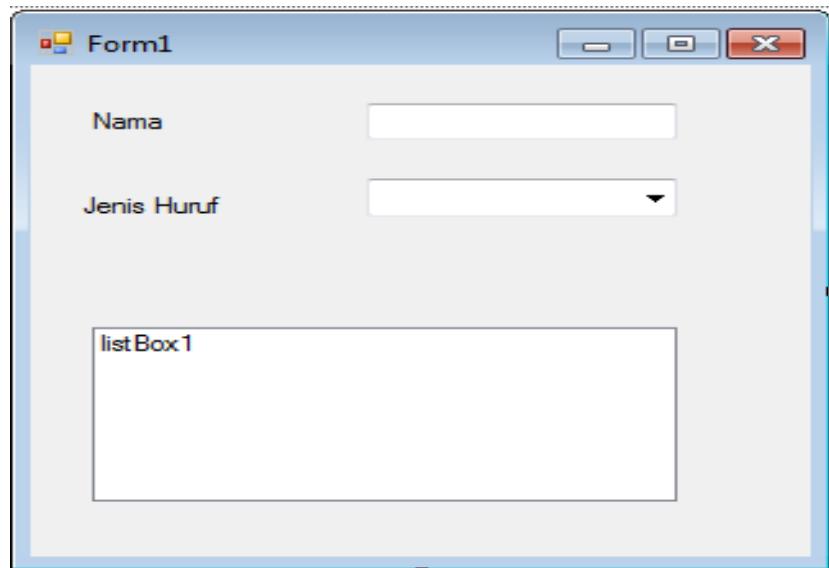


Tabel Pembahasan :

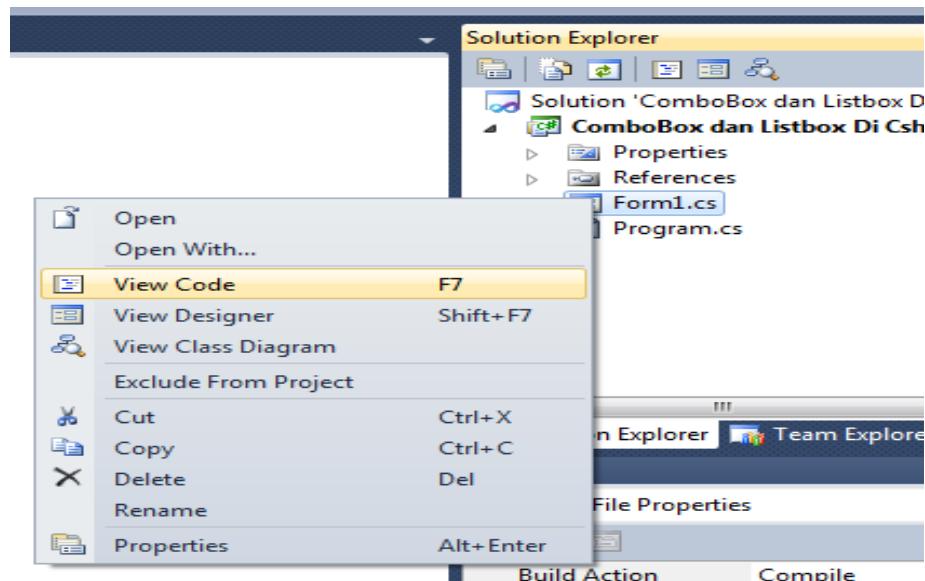
<code>string musik,film;</code>	= Memberikan tipe data string pada variable music dan film
<code>private void button1_Click(object sender, EventArgs e)</code>	= Ini dipanggil jika button1 di klik oleh user,untuk membuat method ini cukup double klik pada buton11 di windows form
<code>if (checkBox1.Checked) {     musik = "pop"; } else if (checkBox2.Checked) {     musik = "dangdut"; } else {     musik = "pop, dangdut"; }</code>	= Ilustrasi dari listing diatas yaitu jika checkbox1 dalam kondisi di ceklis maka akan diberikan huruf "pop" pada variable music, dan begitu seterusnya, sampai semua kondisi terpenuhi dan pada kondisi tterakhir akan diberikan "pop,dangdut".
<code>if (radioButton1.Checked) {     film = &gt;Action"; } else {     film = &gt;Drama"; }</code>	= Pembahsan sam seperti diatas , namun yang diberikan kondisi disini yaitu radiobutton
<code>MessageBox.Show("Anda menyukai musik jenis "+musik+" dan film bergenre "+film+"");</code>	= Perintah disamping digunakan untuk menampilkan sebuah pesan kotak dialog.

### C. ComboBox dan Listbox Di C#

- Silakan Buat Projek Baru berinama **ComboBox\_dan\_Listbox\_Di\_Csharp**, lalu siapkan komponen yang diperlukan seperti 2 label, 1 textbox, 1 combobox, dan 1 buah listview . kemudian atur design seperti gambar dibawah.



- Selanjutnya klik kanan pada Form1.cs di solution explorer – pilih view code seperti gambar dibawah.



- Langkah berikutnya masukan code berikut.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
```

```
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace ComboBox_dan_Listbox_Di_Csharp
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

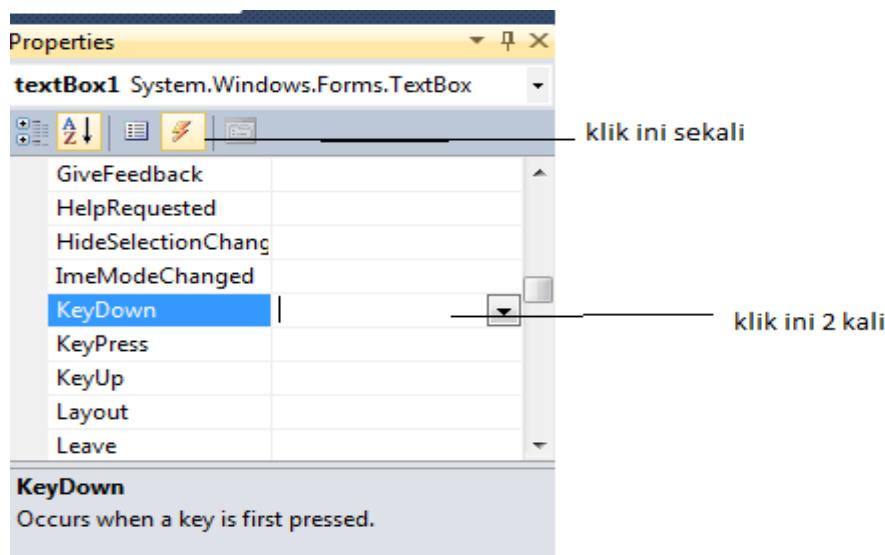
        public void isikombo()
        {
            comboBox1.Items.Add("TEBAL");
            comboBox1.Items.Add("MIRING");
        }

        public void koreksi_huruf()
        {
            if (comboBox1.Text == "TEBAL")
            {
                listBox1.Font = new Font(listBox1.Font, FontStyle.Bold);
            }
            else
            {
                listBox1.Font = new Font(listBox1.Font, FontStyle.Italic);
            }
        }
    }
}
```

- Berikutnya yaitu kembali kedesinernya lalu klik 2 kali pada formnya dan copykan code berikut .

```
private void Form1_Load(object sender, EventArgs e)
{
    isikombo();
}
```

- Atur properties untuk event key pada textbox1 dengan cara klik textbox1 satu kali saja kemudian klik gambar petir pada properties lalu pilih keydown dengan cara klik dua kali seperti gambar dibawah.



- Setelah semua sudah dilakukan langkah berikutnya isikan listing berikut .

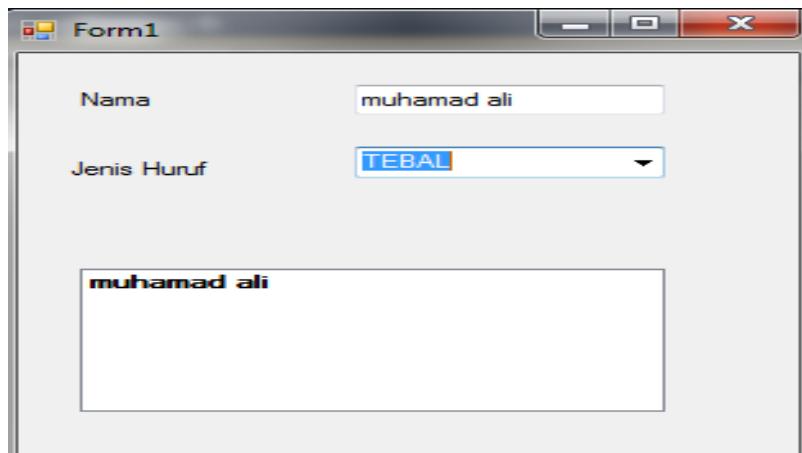
```
private void textBox1_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
    {
        listBox1.Items.Add(textBox1.Text);
    }
}
```

- langkah terakhir yaitu silakan kembali lagi pada desainer anda lalu klik dua kali pada comboBox1 dan ketikan listing berikut.

```
private void comboBox1_SelectedIndexChanged(object sender, EventArgs e)
{
    koreksi_huruf();
```

```
}
```

Sampai disini aplikasi sudah selesai dibuat silakan jalankan projek anda untuk melihat hasilnya jangan lupa isikan pada textput dan tekan enter setelah selesai mengisi. Jika berhasil maka akan seperti gambar berikut .



Tabel Pembahasan

<code>listBox1.Font = new Font(listBox1.Font, FontStyle.Bold);</code>	=	Menseting huruf menjadi huruf tebal/BOLD pada listBox1
<code>private void comboBox1_SelectedIndexChanged(object sender, EventArgs e)</code>	=	Perintah ini akan dieksekusi jika combobox1 dalam keadaan terpilih/ diklik.
<code>private void textBox1_KeyDown(object sender, KeyEventArgs e) {     if (e.KeyCode ==         Keys.Enter)</code>	=	Perintah ini mengilustrasikan penekanan tombol tertentu pada keyboard, di listing tersebut penekanan tombol keyboard yang akan dibaca yaitu tombol enter.

#### D. Menggunakan Timer Pada C#

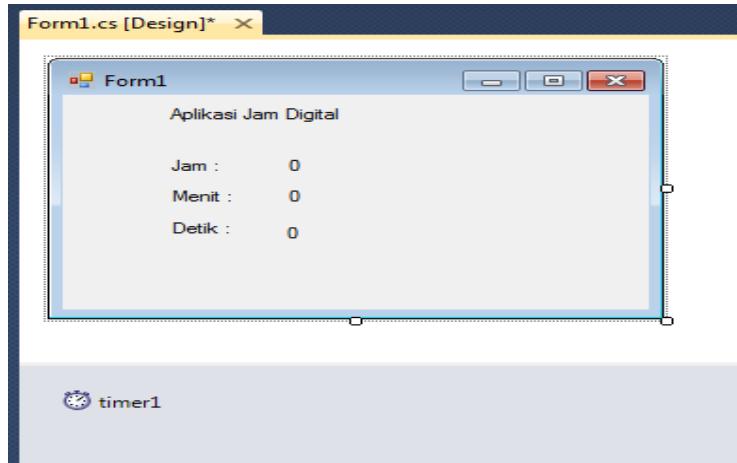
- Silakan Buat Projek Baru berinama **Timer\_Pada\_Csharp**, lalu siapkan komponen yang diperlukan seperti 7 label, dan 1buah Timer. kemudian atur properties dan design seperti gambar dibawah.

Tabel Properties.

ToolBox	Properties	
Label1	Text	Aplikasi Jam Digital
Label2	Text	Jam :
Label3	Text	Menit :
Label4	Text	Detik :
Label5	Text	0 (nol)
Label6	Text	0
Label7	Text	0
timer1	Text	timer1

- Untuk lebih jelasnya silakan lihat gambar Desain berikut :

Gambar Design Form



- Setelah itu silakan klik dua kali pada form kemudian kerikan listing berikut :

```
private void Form1_Load(object sender, EventArgs e)
{
    timer1.Enabled = true;
}
```

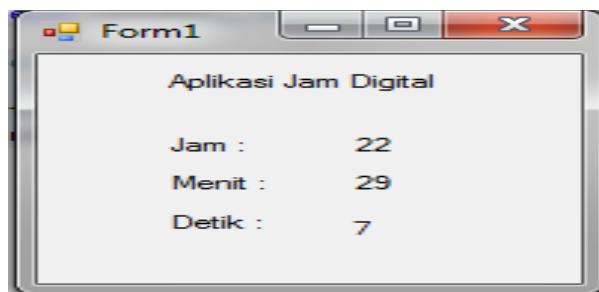
- Langkah selanjutnya yaitu silakan kembali ke form designer lalu double klik pada timer1 dan ketikan lagi listing dibawah ini. Sampai disini aplikasi sudah selesai dibuat dan sudah siap dijalankan.



```
private void timer1_Tick(object sender, EventArgs e)
```

```
{
    DateTime jamku = DateTime.Now;
    label5.Text = jamku.Hour.ToString();
    label6.Text = jamku.Minute.ToString();
    label7.Text = jamku.Second.ToString();
}
```

Jika berhasil maka aplikasi akan tampak seperti gambar dibawah.

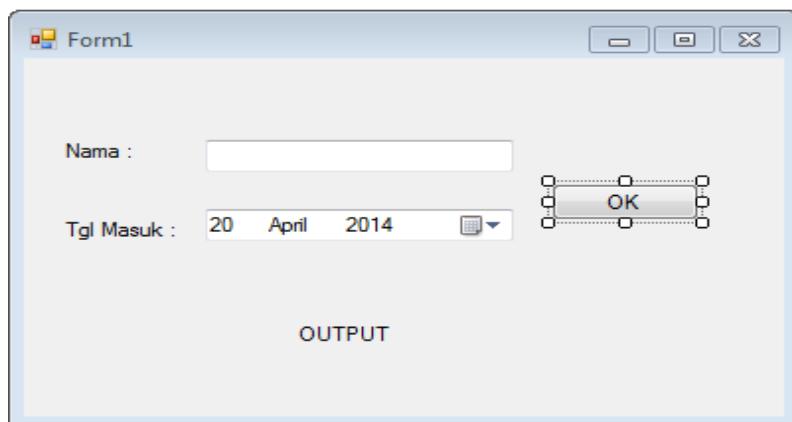


Tabel Pembahasan

<code>timer1.Enabled = true;</code>	=	Perintah ini diberikan untuk mengaktifkan fungsi timer
<code>DateTime jamku = DateTime.Now;</code>	=	Digunakan untuk memanggil fungsi waktu yang kemudian ditampung ke variable jamku
<code>label5.Text = jamku.Hour.ToString(); label6.Text = jamku.Minute.ToString(); label7.Text = jamku.Second.ToString();</code>	=	Mengambil fungsi jam,menit dan detik yang akan di tampilkan ke label

### E. Menggunakan DateTimepicker pada C#

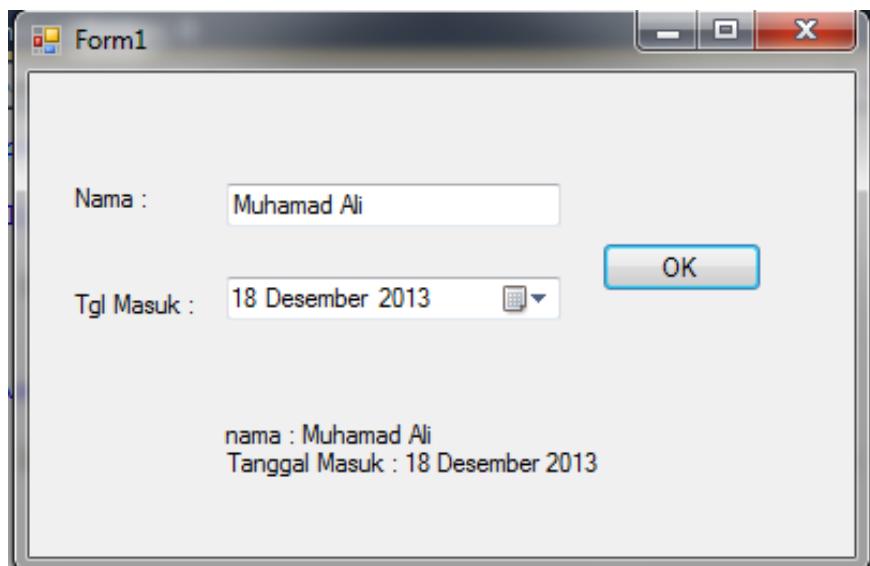
- Silakan Buat Projek Baru berinama **DateTimepicker\_pada\_Csharp**, lalu siapkan komponen yang diperlukan seperti 3 label,1 textbox dan 1buah Datetimepicker. kemudian untuk properties pada label3 beri Text “OUTPUT” dan button1 beri text “OK”, sementara untuk yang lainnya biarkan default saja dan design tampilan seperti gambar dibawah .



- Kemudian Silakan double klik pada button1/button ok ,kemudian ketikan listing berikut.

```
private void button1_Click(object sender, EventArgs e)
{
    label3.Text = "nama : "+textBox1.Text+"\nTanggal Masuk :
"+dateTimePicker1.Text;
}
```

- Jika Sudah Jalankan Program jika berhasil maka akan tampil output seperti gambar berikut.

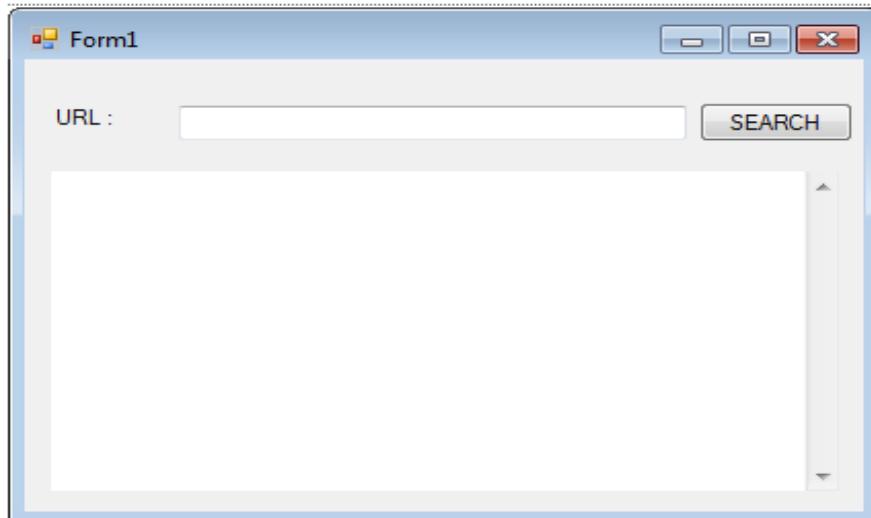


Tabel Pembahasan

label3.Text = "nama : "+textBox1.Text+"\nTanggal Masuk : "+dateTimePicker1.Text;	=	Mencetak kalimat nama: kemudian memanggil isi dari textBox1 yang akan dicetak ,dan tanggal masuk : kemudian memanggil isi dari dateTimePicker1 yang akan dicetak.
---	---	---

## F. Menggunakan Komponen WebBrowser pada C#

- Silakan Buat Projek Baru berinama **WebBrowser\_sederhana\_pada\_Csharp**, lalu siapkan komponen yang diperlukan seperti 1 webbrowser, 1 label,1 textbox dan 1buah button. kemudian untuk properties name biarkan default saja,sementara untuk text properties bisa dilihat pada bagian design tampilan form seperti gambar dibawah .



- Kemudian double klik pada button1/button search lalu ketikan listing dibawah ini.

```
private void button1_Click(object sender, EventArgs e)
{
    webBrowser1.Navigate(textBox1.Text); /*saya rasa perintah ini mudah untuk dipahami*/
}
```

- Silakan jalankan program untuk melihat hasilnya,jika berhasil aplikasi akan berjalan seperti gambar dibawah ini.



Baik demikian pengenalan windows form pada C#. pada halaman berikutnya akan dibahas mengenai study kasus pembuatan aplikasi – aplikasi sederhan pada C#. Jangan lupa lebih bereksplorasi atau berexperiment sendiri untuk meningkatkan pemahaman dan skill anda pada bahasa pemogramman C#.

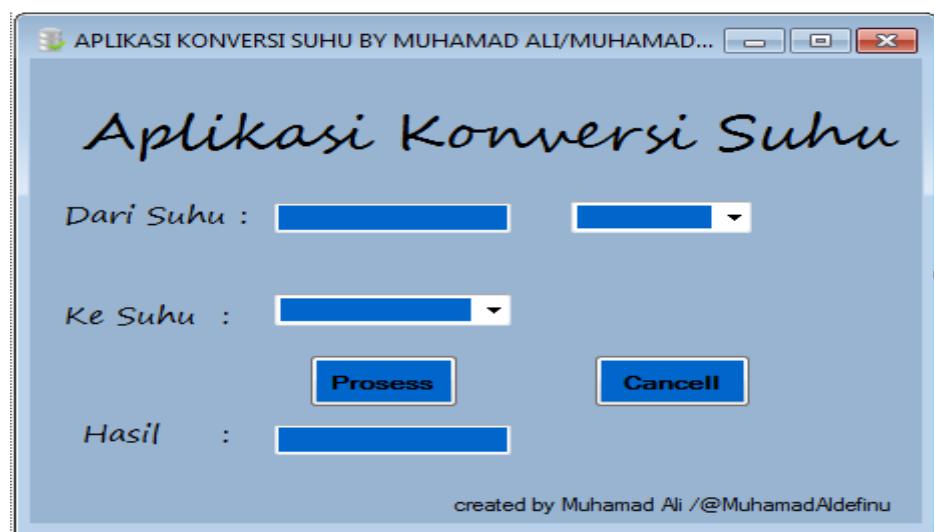
## 4.2 Membuat Aplikasi Sederhana pada visual C#

Untuk lebih memahami mengenai pembelajaran pada bab ini maka Pada halaman ini akan dibahas membahas mengenai studi kasus membuat aplikasi sederhana menggunakan C# diantaranya membuat aplikasi konversi suhu, aplikasi kalkulator sederhana dan aplikasi – aplikasi lainya.

### 1. Aplikasi Konversi Suhu

- Langkah pertama silakan design tampilan dan atur properties seperti gambar berikut .

**Gambar design Form**



**Tabel Properties**

Toolbox	Properties	keterangan
<b>Form1</b>	Text	APLIKASI KONVERSI SUHU BY MUHAMAD ALI/MUHAMAD ALI ALDEFINU
	BackColor	ActiveCaption
	Icon	Ganti sesuai keinginan anda
<b>label1</b>	text	Aplikasi Konversi Suhu
	name	Label1
	Font	Segoe Script; 24pt
<b>label2</b>	text	Dari Suhu :
	name	label2
	Font	Segoe Print; 12pt
<b>label3</b>	text	Ke Suhu :
	name	label3
	Font	Segoe Print; 12pt
<b>label4</b>	text	Hasil :

	name	label3
	Font	Segoe Print; 12pt
<b>label4</b>	text	Created by.....
<b>textBox1</b>	name	textBox1
<b>comboBox1</b>	name	comboBox1
	items	celcius reamur fahrenheit kelvin
<b>comboBox2</b>	name	comboBox2
	items	celcius reamur fahrenheit kelvin
<b>button1</b>	name	button1
	text	Prosess
<b>button1</b>	name	button2
	text	Cancel
<b>textBox2</b>	name	textBox2

- Setelah semua sudah di atur sulakan klik kana pada form kemudian pilih view code dan tuliskan script dibawah :

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace aplikasi_konversi_suhu
{
    public partial class Form1 : Form
    {
        string a, b, c;
        double g,hitung;
    }
}

```

```
public Form1()
{
    InitializeComponent();
}
```

- Selanjutnya adalah klik dua kali pada button1 / button proeses lalu masukan script berikut.

```
private void button1_Click(object sender, EventArgs e)
{
    a = textBox1.Text;
    b = comboBox1.Text;
    c = comboBox2.Text;
    g = Convert.ToDouble(a);
    if ((b.Equals("celcius")) && (c.Equals("reamur")))
    {
        hitung = g /5 * 4;
        textBox2.Text = hitung.ToString();
    }
    else if ((b.Equals("celcius")) && (c.Equals("fahrenheit")))
    {
        hitung = g / 5 * 9 + 32;
        textBox2.Text = hitung.ToString();
    }
    else if ((b.Equals("celsius")) && (c.Equals("kelvin")))
    {
        hitung = g / + 273.15;
        textBox2.Text = hitung.ToString();
    }
    else if ((b.Equals("reamur")) && (c.Equals("celcius")))
}
```

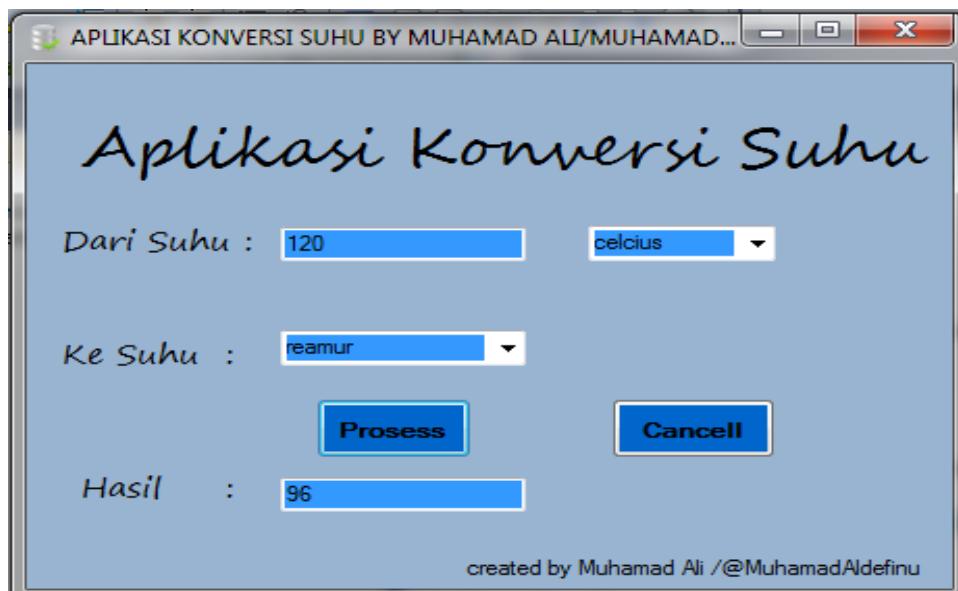
```
{  
    hitung = g / 4 * 5;  
    textBox2.Text = hitung.ToString();  
}  
  
else if ((b.Equals("reamur")) && (c.Equals("fahrenheit")))  
{  
    hitung = g / 4 * 9 + 32;  
    textBox2.Text = hitung.ToString();  
}  
  
else if ((b.Equals("reamur")) && (c.Equals("kelvin")))  
{  
    hitung = g / 4 * 9 + 32;  
    textBox2.Text = hitung.ToString();  
}  
  
else if ((b.Equals("fahrenheit")) && (c.Equals("celcius")))  
{  
    hitung = (g - 32) / 9 * 5;  
    textBox2.Text = hitung.ToString();  
}  
  
else if ((b.Equals("fahrenheit")) && (c.Equals("reamur")))  
{  
    hitung = (g - 32) / 9 * 4;  
    textBox2.Text = hitung.ToString();  
}  
  
else if ((b.Equals("fahrenheit")) && (c.Equals("kelvin")))  
{  
    hitung = (g - 32) / 9 * 5 + 273.15;  
    textBox2.Text = hitung.ToString();  
}  
  
else if ((b.Equals("kelvin")) && (c.Equals("celcius")))
```

```
{  
    hitung = g - 273.15;  
    textBox2.Text = hitung.ToString();  
}  
  
else if ((b.Equals("kelvin")) && (c.Equals("fahrenheit")))  
{  
    hitung = (g - 273.15) / 5 * 9 + 32;  
    textBox2.Text = hitung.ToString();  
}  
  
else  
{  
    hitung = (g - 273.15) / 5 * 4;  
    textBox2.Text = hitung.ToString();  
}  
}
```

- Langkah terakhir yaitu silakan double klik pada button2 dan masukan script berikut .

```
private void button2_Click(object sender, EventArgs e)  
{  
    textBox1.Text = "";  
    textBox2.Text = "";  
}
```

Sampai disini aplikasi sudah selesai dibuat silakan jalankan untuk melihat hasilnya. Jika Berhasil maka aplikasi akan terlihat seperti gambar berikut .



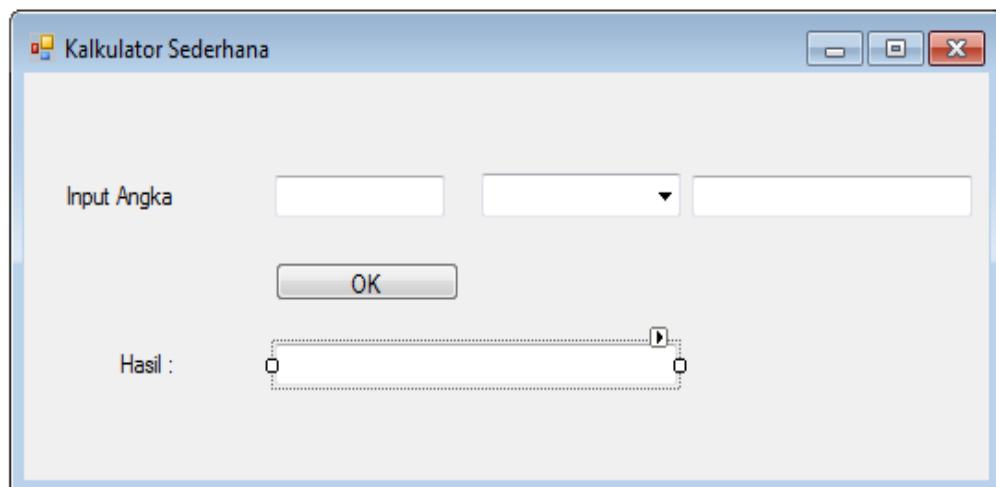
Tabel Pembahasan

<code>string a, b, c;</code>	=	Memberikan tipe data string pada "a,b,c" dan tipe data double pada "g" dan "hitung"
<code>a = textBox1.Text;</code> <code>b = comboBox1.Text;</code> <code>c = comboBox2.Text;</code>	=	Menampung isi text dari textbox kedalam variable a,b,c
<code>g = Convert.ToDouble(a);</code>	=	Mengkonversi tipe data string pada variable "a" kedalam tipe data double yang akan ditampung pada variable "g"
<code>helpcs help = new helpcs();</code> <code>help.Show();</code>	=	Instantiasi objek dan menampilkan form help.cs

## 2. Aplikasi Kalkulator sederhana

- buatlah projek dengan nama aplikasi\_kalkulator\_sederhana, atur desain dan propertiesnya seperti gambar dan table dibawah ini.

Gambar Design Form



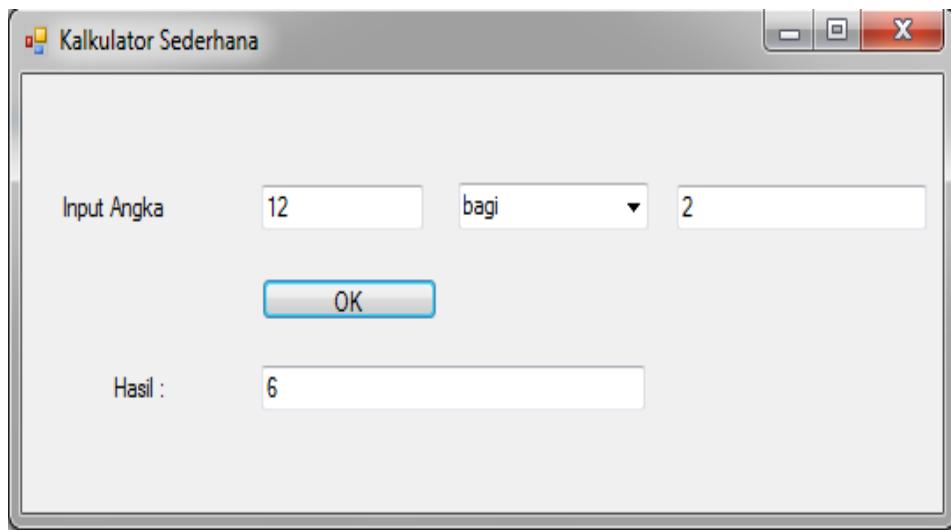
**Tabel Properties**

ToolBox	Properties	Keterangan
<b>Form1</b>	text	Kalkulator Sederhana
<b>label1</b>	text	Input Angka :
<b>label2</b>	text	Hasil :
<b>textbox1</b>	name	textbox1
<b>textbox2</b>	name	textbox2
<b>button1</b>	text	OK
<b>Combobox1</b>	items	kali tambah kurang bagi

- Langkah terakhir yaitu double klik pada button1 dan ketikn listing berikut .

```
private void button1_Click(object sender, EventArgs e)
{
    double a, b;
    a = Convert.ToDouble(textBox1.Text);
    b = Convert.ToDouble(textBox2.Text);
    if (comboBox1.Text == "kali")
    {
        textBox3.Text = (a * b).ToString();
    }
    else if (comboBox1.Text == "tambah")
    {
        textBox3.Text = (a + b).ToString();
    }
    else if (comboBox1.Text == "kurang")
    {
        textBox3.Text = (a - b).ToString();
    }
    else
    {
        textBox3.Text = (a / b).ToString();
    }
}
```

Sampai disini aplikasi sudah bisa di jalankan, jika berhasil maka hasilnya akan terlihat seperti gambar dibawah ini.



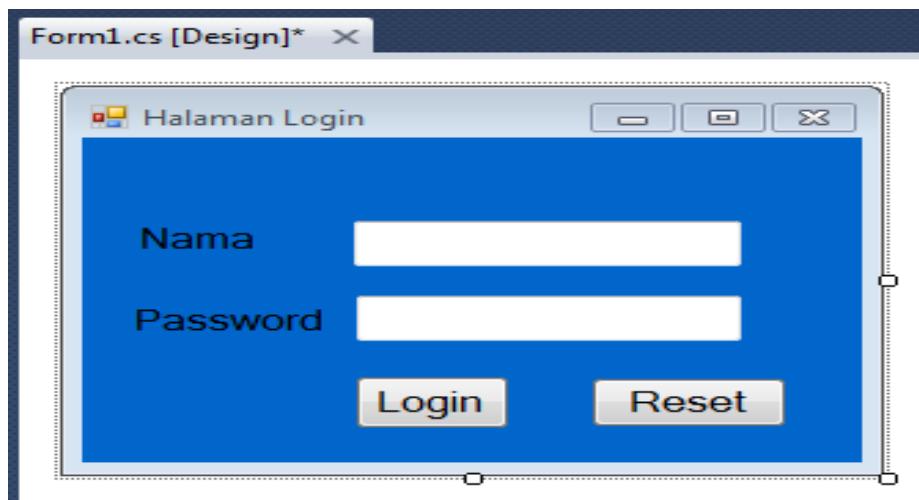
Tabel Pembahasan

<code>a = Convert.ToDouble(textBox1.Text);</code>	= Printah yang digunakan untuk mengkonversi tipe data string pada textbox kedalam tipe data double
<code>textBox3.Text = (a * b).ToString();</code>	= Mengalikan variable "a" dan "b" kemudian mengubahnya ke tipe data string.

### 3. Membuat Aplikasi Hotel Reception Sederhana.

- Silakan buat aplikasi dengan nama aplikasi\_hotel, kemudian siapkan beberapa komponen yang diperlukan seperti 2 label,2 textbox dan dua buah button:
- Kemudian atur design dan propertiesnya seperti gambar dibawah untuk form 1.

Gambar design form1



Tabel Properties

Toolbox	Properties	Keterangan
<b>label1</b>	text	Nama
	font	Microsoft Sans Serif; 12pt
<b>label2</b>	text	Password
	font	Microsoft Sans Serif; 12pt
<b>textbox1</b>	name	nama
<b>textbox2</b>	name	pss
	PasswordChar	*
<b>button1</b>	text	Login
<b>button2</b>	text	Reset
<b>Form1</b>	text	Halaman login
	BackColor	HotTrack

- Setelah itu double klik pada button1 dan Silakan ketikan listing berikut.

```
private void button1_Click(object sender, EventArgs e)
{
    string a = pss.Text; /* untuk menampung nilai yg di text pss/textbox2 yg
propertiesnya di rubah mjd pss */

    if (a.Equals("hotel"))
    {
        MessageBox.Show("Selamat login success");

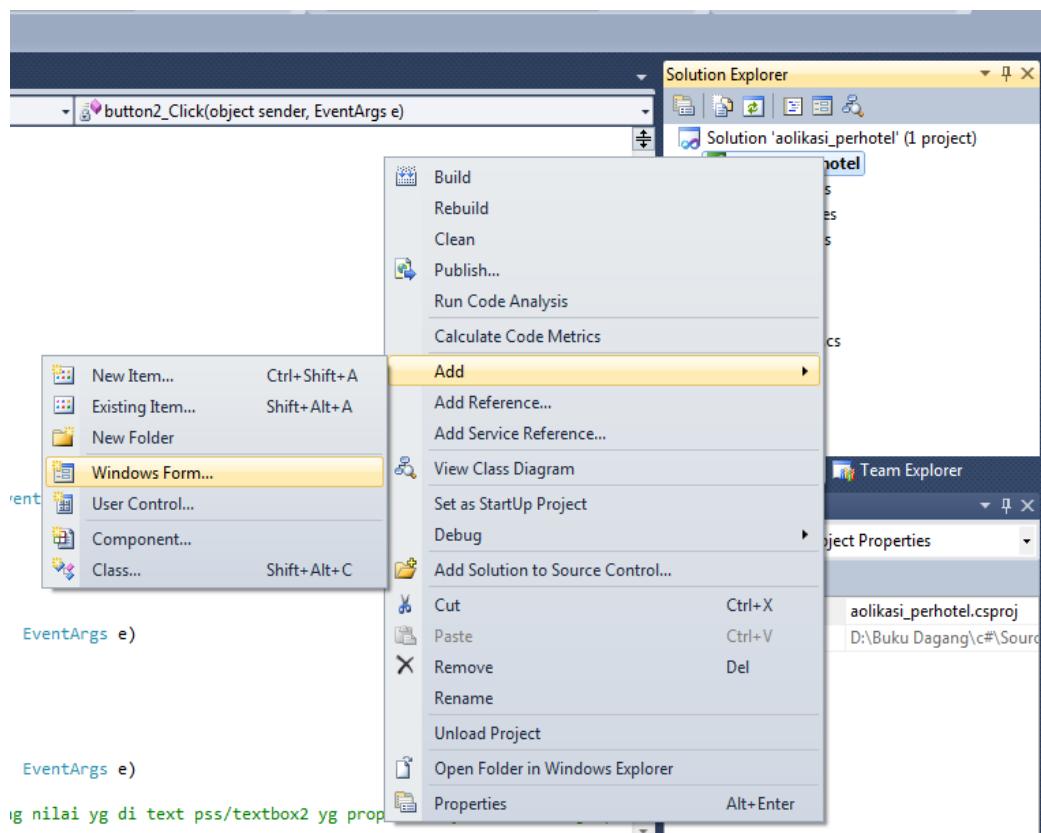
        hotel h = new hotel(); /*instatiasi hotel atau mendefenisikan sebuah
kelas baru*/

        h.Show(); /*perintah unruk menampilkan hotel.cs*/
        this.Hide(); // untuk menyembunyikan form1
    }
    else
    {
        MessageBox.Show("maaf password salah");
    }
}
```

- Kemudian double klik juga pada button2 dan ketikan listing berikut.

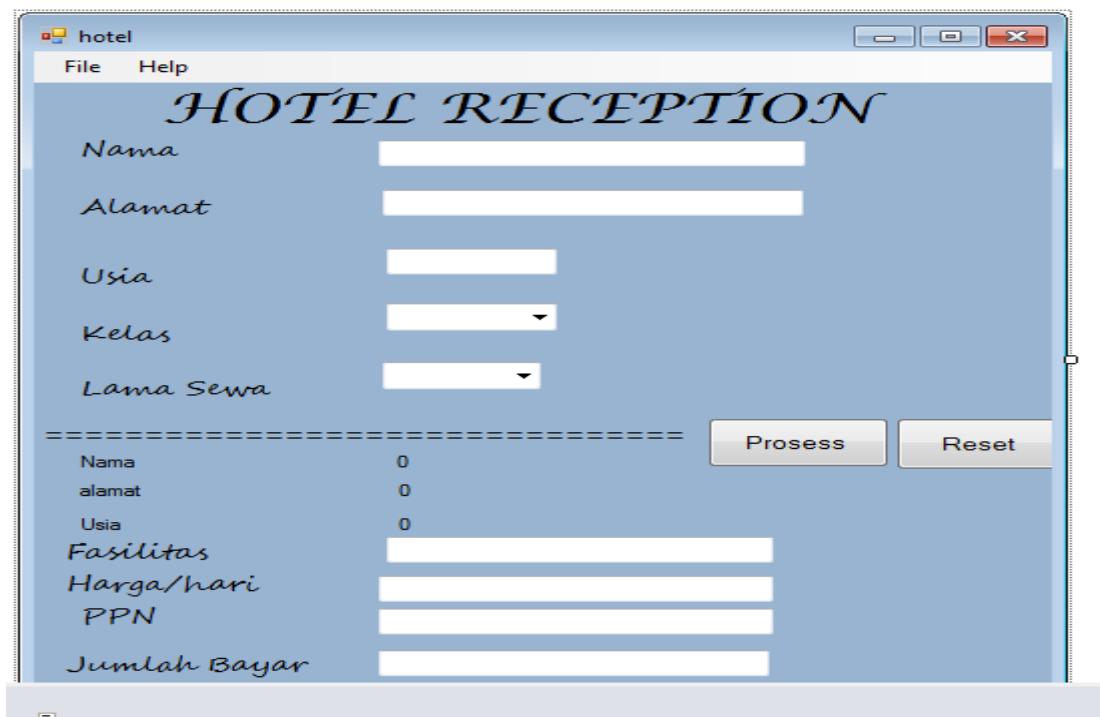
```
private void button2_Click(object sender, EventArgs e)
{
    nama.Text = "";
    pss.Text = "";
}
```

- Langkah berikutnya yaitu buat form baru dengan cara klik kanan pada nama projek di solution explore dan pilih add -> window form



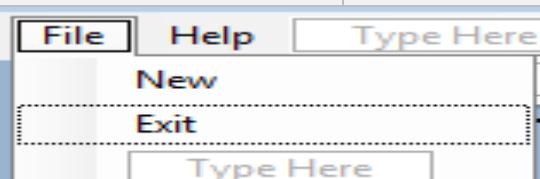
- berikutnya yaitu atur properties dan designnya untuk label silakan lihat saja properties textnya seperti gambar form dibawah sementara untuk yang lainya bisa dilihat ditabel properties.

Gambar Design



Tabel Properties.

Toolbox	Properties	Keterangan
textBox1	name	textBox1
textBox2	name	textBox2
textBox3	name	textBox3
comboBox1	name	comboBox1
	Items	VIP STANDART HEMAT
comboBox2	name	comboBox2
	items	1 2 3 4 5 6 7 8 9 10 11 12 13

		14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
<b>button1</b>	text	Prosess
<b>button2</b>	text	Reset
<b>textBox4</b>	name	textBox4
<b>textBox5</b>	name	textBox5
<b>textBox6</b>	name	textBox6
<b>textBox7</b>	name	textBox7
<b>menuStrip1</b>		

- Berikutnya yaitu double klik pada button1/button prosess dan ketikan listing berikut.

```
private void button1_Click(object sender, EventArgs e)
{
    string a, b, c, combo1, combo2;
    int harga, ppn, total, lama;
    a = textBox1.Text;
    b = textBox2.Text;
    c = textBox3.Text;
    combo1 = comboBox1.Text;
    combo2 = comboBox2.Text;
    lama = Convert.ToInt16(combo2);
```

```
if (combo1.Equals("VIP")) {  
    textBox4.Text = "Kamar Mewah,TV dan Hotspot";  
    harga = 700000;  
    ppn = 20000;  
    total = (harga * lama) + ppn;  
    label12.Text = a;  
    label13.Text = b;  
    label14.Text = c;  
    textBox5.Text = harga.ToString();  
    textBox6.Text = ppn.ToString();  
    textBox7.Text = total.ToString();  
}  
  
else if (combo1.Equals("STANDART"))  
{  
    textBox4.Text = "Kamar Mewah dan TV";  
    harga = 500000;  
    ppn = 20000;  
    total = (harga * lama) + ppn;  
    label12.Text = a;  
    label13.Text = b;  
    label14.Text = c;  
    textBox5.Text = harga.ToString();  
    textBox6.Text = ppn.ToString();  
    textBox7.Text = total.ToString();  
}  
  
else if (combo1.Equals("HEMAT"))  
{  
    textBox4.Text = "Kamar Mewah ";  
    harga = 300000;  
    ppn = 20000;
```

```

total = (harga * lama) + ppn;

label12.Text = a;
label13.Text = b;
label14.Text = c;
textBox5.Text = harga.ToString(); ;
textBox6.Text = ppn.ToString();
textBox7.Text = total.ToString();

}
}

```

- Selanjutnya silakan duble klik pada button resert dan ketikan listing berikut.

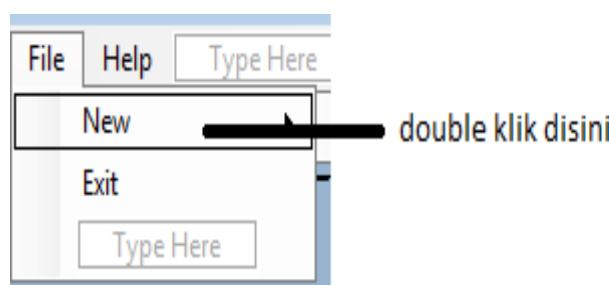
```

private void button2_Click(object sender, EventArgs e)

{
    textBox1.Text = "";
    textBox2.Text = "";
    textBox3.Text = "";
    textBox4.Text = "";
    textBox5.Text = "";
    textBox6.Text = "";
    textBox7.Text = "";
    label12.Text = "";
    label13.Text = "";
    label14.Text = "";
}

```

- Double klik juga pada file-new di menustrip1 seperti gambar dibawah dan silakan ketikan listing berikut.



```
private void newToolStripMenuItem_Click(object sender, EventArgs e)
```

```
{  
    textBox1.Text = "";  
    textBox2.Text = "";  
    textBox3.Text = "";  
    textBox4.Text = "";  
    textBox5.Text = "";  
    textBox6.Text = "";  
    textBox7.Text = "";  
    label12.Text = "";  
    label13.Text = "";  
    label14.Text = "";  
}
```

- berikutnya double klik juga pada file -> exit di menustrip1 dan masukan code berikut.

```
private void exitToolStripMenuItem_Click(object sender, EventArgs e)
```

```
{  
    Form1 f = new Form1();  
    f.Close();/*untuk mengklos form1*/  
    this.Close();/*untuk mengklos form hotelcs*/  
}
```

- double klik juga pada menu help di menustrip1 lalu katikan listing berikut.

```
private void helpToolStripMenuItem_Click(object sender, EventArgs e)
```

```
{  
    Help tolong = new Help();  
    tolong.Show();  
}
```

- langkah berikutnya buat form baru dengan nama Help.cs, atur desain dan propertiesnya seperti gambar dibawah.

Gambar Form

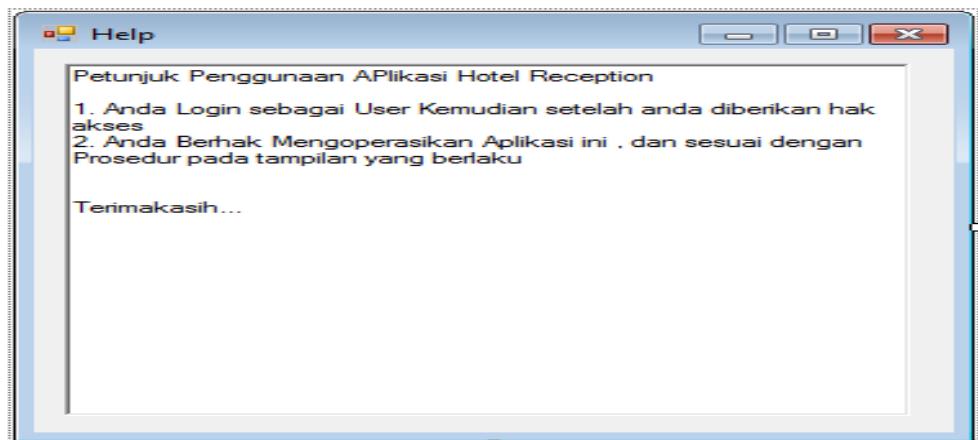


Table properties.

Toolbox	Properties	keterangan
richTextBox1	text	<p>Petunjuk Penggunaan Aplikasi Hotel Reception</p> <p>1. Anda Login sebagai User Kemudian setelah anda diberikan hak akses</p> <p>2. Anda Berhak Mengoperasikan Aplikasi ini , dan sesuai dengan Prosedur pada tampilan yang berlaku</p> <p>Terimakasih...</p>

Nah sampai disini aplikasi sudah selesai dibuat ,silakan jalankan untuk melihat hasilnya. Jika berhasil aplikasi akan berjalan seperti gambar dibawah.

Nama	muhamad
Alamat	jakarta
Usia	20
Kelas	STANDART
Lama Sewa	7
-----	-----
Nama	muhamad
alamat	jakarta
Usia	20
Fasilitas	Kamar Mewah dan TV
Harga/hari	500000
PPN	20000
Jumlah Bayar	3520000

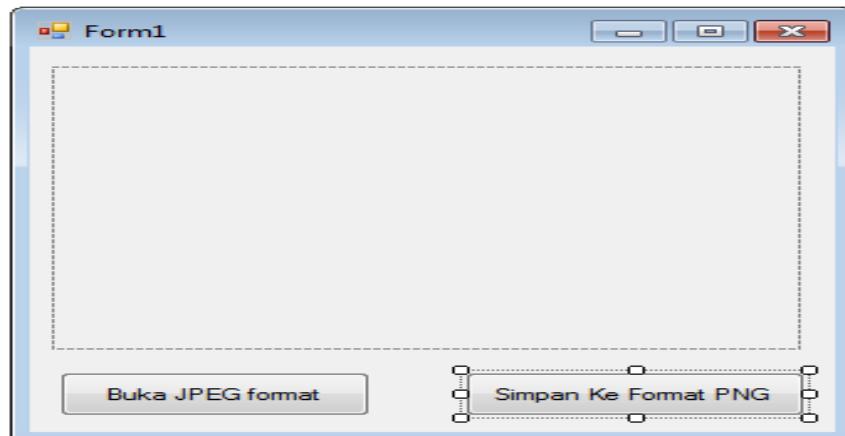
- Untuk pembahasan yang belum pernah dibahas pada halaman – halaman sebelumnya dalam projek tersebut sudah penulis sertakan dalam bentuk komentar.

#### 4. Aplikasi Load dan Save Gambar

Aplikasi ini untuk meload gambar dengan format JPEG dan menyimpanya hanya dengan format PNG.

- Silakan anda buat projek baru dengan nama buka\_simpan\_gambar buat gambar dan yang perlu diatur di propertiesnya hanya text pada button saja sesuaikan seperti gambar form dibawah.
- Adapun komponen toolbox yang digunakan yaitu hanya 1 picturebox dan 2 buah button.

Gambar Form



- Jika Sudah Silakan tambahkan library `using System.Drawing.Imaging;`
- kemudian double klik pada button1/button buka, lalu ketikan listing dibawah.

```
private void button1_Click(object sender, EventArgs e)
{
    OpenFileDialog open = new OpenFileDialog();
    open.Filter = "Jpeg Format|*.Jpg";
    if (open.ShowDialog() == DialogResult.OK)
    {
        pictureBox1.SizeMode = PictureBoxSizeMode.StretchImage;
        pictureBox1.ImageLocation = open.FileName;
    }
}
```

- setelah itu double klik juga pada button2 dan ketikan listing berikut.

```
private void button2_Click(object sender, EventArgs e)
{
    SaveFileDialog simpan = new SaveFileDialog();
    simpan.RestoreDirectory = true;
    if (simpan.ShowDialog() == DialogResult.OK)
    {
        pictureBox1.Image.Save(simpan.FileName, ImageFormat.Png);
    }
}
```

Sampai Disini Aplikasi sudah selesai dibuat silakan jalankan aplikasi ,jika berhasil maka akan seperti gambar dibawah.



Tabel Pembahasan

<code>OpenFileDialog open = new OpenFileDialog(); open.Filter = "Jpeg Format *.Jpg";</code>	=	Digunakan untuk membuka gambar dengan format Jpg
<code>pictureBox1.SizeMode = PictureBoxSizeMode.StretchImage;</code>	=	Untuk menyesuaikan ukuran gambar sesuai picturebox(gambar distretch memenuhi ukuran picturebox)
<code>pictureBox1.ImageLocation = open.FileName;</code>	=	Untuk memasukan gambar ke pictureBox
<code>SaveFileDialog simpan = new SaveFileDialog(); simpan.RestoreDirectory = true;</code>	=	SaveFileDialog digunakan untuk menyimpan gambar
<code>pictureBox1.Image.Save(simpan.FileName, ImageFormat.Png);</code>	=	Gambar disimpan dengan format Png

## 5. Membuat Aplikasi Billing Sederhana

Pada halaman ini kita akan coba membuat aplikasi billing sederhana, yang hanya mengilustrasikan pilihan paket dan timming saja . Untuk membaatnya silakan ikuti langkah – langkah berikut :

- Buat lagi Form 2 Kasih Nama Form dengan nama Billing.cs
- Klik Kanan Pada Form2(Billing.Cs) ➔ ➔ VIEW CODE . Lalu Masukan Source Code Sebagai Berikut :
- Tambahkan `using System.Runtime.InteropServices;` diatas program.

```
public partial class Billing : Form
```

```
{
    private string text = "";
    string p, q, r, h ;
    int s, t, u, i;
    public string Text
    {
        get { return this.text; }
        set { this.text = value; }
    }
    public Billing()
    {
```

```

InitializeComponent();

}

}

```

Klik Form1.cs[desain] di bagian Form 1 Sebagai Berikut :

Toolbox	Properties	Keterangan		
panel1	-	-	radioButton3	Name r3 Text 6000
groupBox1	Text	Paket Anda Rp.	radioButton4	Name r4 Text 8000
groupbox2	Text	Fasilitas	radioButton5	Name r5 Text 11000
pictureBox1	Image	Terserah Anda	radioButton6	Name r6 Text 13000
label1	Text	Pilihan Paketmu	TextBox1	Name minuman
label2	Text	30 Menit	button1	
label3	Text	1 Jam		
label4	Text	2 Jam		
label5	Text	3 Jam		
label6	Text	4 Jam		
label7	Text	5 Jam		
label8	Text	Masukan Nama Minuman		
radioButton1	Name r1 Text 1500			
radioButton2	Name r2 Text 3000			



Klik Kanan Pada Form1(Form1Cs) ➔ ➔ VIEW CODE . Lalu Masukan Source Code Sebagai Berikut :

```

public partial class Form1 : Form
{
    public Billing f2;
    string a, b;
    public Form1()
    {
        InitializeComponent();
    }

```

- Kembali ke Form1.cs double klik pada Button1 dan masukan listing berikut.

```
private void button1_Click(object sender, EventArgs e)
{
    if (r1.Checked)
    {
        a = r1.Text;
        b = minuman.Text;
        MessageBox.Show("anda telah memilih paket \r"+a+"\r dengan
Fasilitas \r" +b+ "");
        Billing f2 = new Billing();
        f2.Text = a;
        f2.Show();
        this.Hide();
    }

    else if (r2.Checked)
    {

        a = r2.Text;
        b = minuman.Text;
        MessageBox.Show("anda telah memilih paket \r" + a +
"\r dengan Fasilitas \r" + b + "");
        Billing f2 = new Billing();
        f2.Text = a;
        f2.Show();
        this.Hide();
    }

    else if (r3.Checked)
    {

        a = r3.Text;
        b = minuman.Text;
        MessageBox.Show("anda telah memilih paket \r" + a +
"\r dengan Fasilitas \r" + b + "");
        Billing f2 = new Billing();
        f2.Text = a;
        f2.Show();
        this.Hide();
    }

    else if (r4.Checked)
    {

        a = r4.Text;
```

```
        b = minuman.Text;
MessageBox.Show("anda telah memilih paket \r" + a +
"\rdengan Fasilitas \r" + b + "");
Billing f2 = new Billing();
f2.Text = a;
f2.Show();
this.Hide();
}

else if (r5.Checked)
{

    a = r5.Text;
    b = minuman.Text;
MessageBox.Show("anda telah memilih paket \r" + a +
"\rdengan Fasilitas \r" + b + "");
Billing f2 = new Billing();
f2.Text = a;
f2.Show();
this.Hide();
}

else if (r6.Checked)
{

    a = r6.Text;
    b = minuman.Text;
MessageBox.Show("anda telah memilih paket \r" + a +
"\rdengan Fasilitas \r" + b + "");
Billing f2 = new Billing();
f2.Text = a;
f2.Show();
this.Hide();

}

}
```

Silakan Disain Form2nya Dan Atur Propertinya Sesuai Tabel Di bawah

Tools	Properties	Keterangan
label1	Text	Waktu Anda :
label2	Text	Harga :
label3	Text	S
label4	Text	M
label5	Text	H
label6	Name Text	detik 0
label7	Name Text	menit 0
label8	Name Text	jam 0
label9	Name Text	harga 0
button1	Text	Selesai
timer1	-	-



- Kembali ke Form2(Billing.Cs[Design] kemudian double klik pada form dan masukan listing berikut.

```
private void Billing_Load(object sender, EventArgs e)
{
    timer1.Enabled = true;
    harga.Text = text;
}
```

- Berikutnya Double klik juga Pada Timer1 dan Ketikan listing berikut.

```
private void timer1_Tick(object sender, EventArgs e)
{
    DateTime jamku = DateTime.Now;
    h = harga.Text;
    i = Convert.ToInt16(h);

    detik.Text = jamku.Second.ToString();
    menit.Text = jamku.Minute.ToString();
    jam.Text = jamku.Hour.ToString();
    p= detik.Text ;
```

```
q = menit.Text;
r = jam.Text;

if ((i) == (1500))
{
    s = Convert.ToInt16(p);
    t = Convert.ToInt16(q);
    u = Convert.ToInt16(r);
    while (t == 30)
    {
        s = 0;
        DialogResult di =
MessageBox.Show("paket anda sudah habis,Ingin Tambah Paket
Lagi???", "Important", MessageBoxButtons.YesNo,
MessageBoxIcon.Question, MessageBoxDefaultButton.Button1);
        if (di == DialogResult.Yes)
        {
            Form1 form = new Form1();
            form.Focus();
            form.Visible = true;
            this.Hide();
        }
        else
        {
            Process.Start("shutdown", "/s /t 0");
        }
    }
}

else if ((i) == (3000))
{
    s = Convert.ToInt16(p);
    t = Convert.ToInt16(q);
    u = Convert.ToInt16(r);

    while (u == 1)
    {
        t = 0;
        s = 0;
    }
    DialogResult di = MessageBox.Show("paket anda sudah
habis,Ingin Tambah Paket Lagi???", "Important",
MessageBoxButtons.YesNo, MessageBoxIcon.Question,
MessageBoxDefaultButton.Button1);

    if (di == DialogResult.Yes)
    {
        Form1 form = new Form1();
```

```
        form.Focus();
        form.Visible = true;
        this.Hide();
    }
    else
    {

        Process.Start("shutdown", "/s /t 0");
    }

}

else if ((i) == (6000))
{

    s = Convert.ToInt16(p);
    t = Convert.ToInt16(q);
    u = Convert.ToInt16(r);

    while (u == 2)
    {
        t = 0;
        s = 0;
    DialogResult di = MessageBox.Show("paket anda sudah
habis,Ingin Tambah Paket Lagi???", "Important",
MessageBoxButtons.YesNo, MessageBoxIcon.Question,
MessageBoxDefaultButton.Button1);

        if (di == DialogResult.Yes)
        {
            Form1 form = new Form1();
            form.Focus();
            form.Visible = true;
            this.Hide();
        }
        else
        {

            Process.Start("shutdown", "/s /t 0");
        }

    }

    else if ((i) == (8000))
{
```

```
s = Convert.ToInt16(p);
t = Convert.ToInt16(q);
u = Convert.ToInt16(r);

while (u == 3)
{
    t = 0;
    s = 0;
    DialogResult di = MessageBox.Show("paket anda sudah
    habis,Ingin Tambah Paket Lagi???", "Important",
    MessageBoxButtons.YesNo, MessageBoxIcon.Question,
    MessageBoxDefaultButton.Button1);

    if (di == DialogResult.No)
    {
        Form1 form = new Form1();
        form.Focus();
        form.Visible = true;
        this.Hide();
    }
    else
    {

        Process.Start("shutdown", "/s /t 0");
    }
}

else if ((i) == (11000))
{
    s = Convert.ToInt16(p);
    t = Convert.ToInt16(q);
    u = Convert.ToInt16(r);

    while (u == 4)
    {
        t = 0;
        s = 0;
        DialogResult di = MessageBox.Show("paket anda sudah
        habis,Ingin Tambah Paket Lagi???", "Important",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question,
        MessageBoxDefaultButton.Button1);

        if (di == DialogResult.Yes)
    {
```

```
        Form1 form = new Form1();
        form.Focus();
        form.Visible = true;
        this.Hide();
    }
    else
    {

        Process.Start("shutdown", "/s /t 0");
    }

}

else if ((i) == (13000))
{

    s = Convert.ToInt16(p);
    t = Convert.ToInt16(q);
    u = Convert.ToInt16(r);

    while (u == 5)
    {
        t = 0;
        s = 0;
    }

    DialogResult di = MessageBox.Show("paket anda sudah
    habis,Ingin Tambah Paket Lagi???", "Important",
    MessageBoxButtons.YesNo, MessageBoxIcon.Question,
    MessageBoxDefaultButton.Button1);

    if (di == DialogResult.Yes)
    {
        Form1 form = new Form1();
        form.Focus();
        form.Visible = true;
        this.Hide();
    }
    else
    {

        Process.Start("shutdown", "/s /t 0");
    }

}
}
```

- Double klik pada button selesai lalu ketikan listing berikut.

```

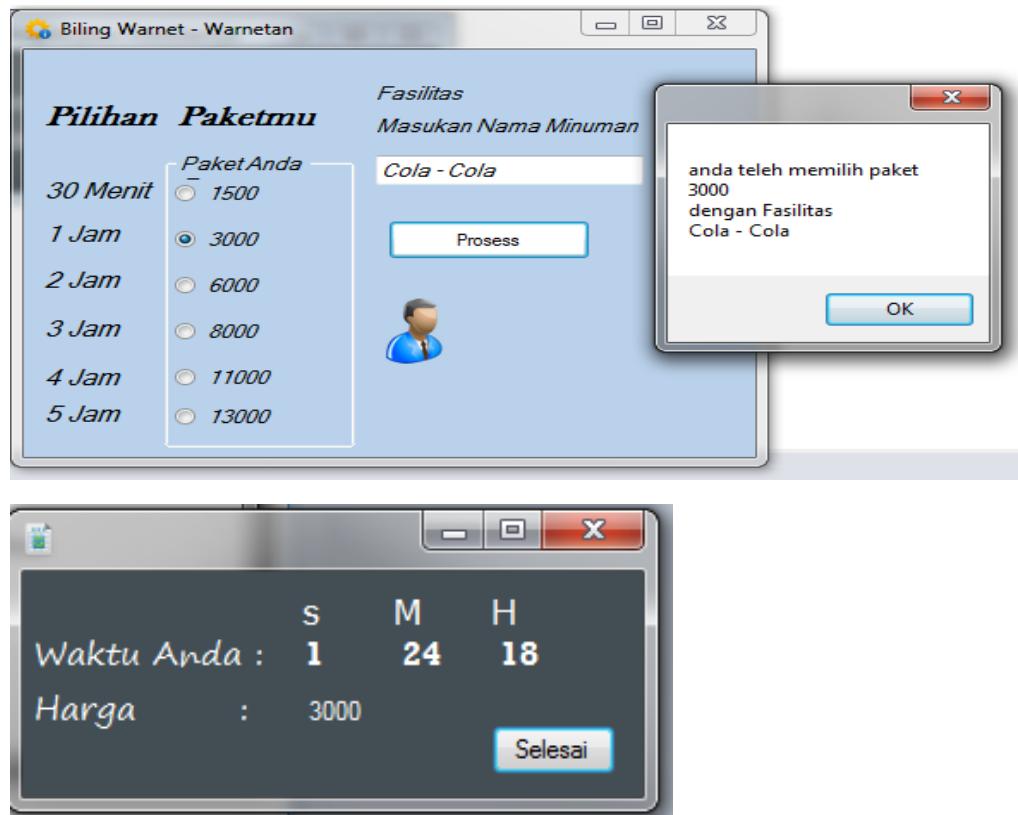
private void button1_Click(object sender, EventArgs e)
{
    DialogResult di = MessageBox.Show("Yakin mau keluar???", 
    "Important", MessageBoxButtons.YesNo,
    MessageBoxIcon.Question,
    MessageBoxDefaultButton.Button1);

    if (di == DialogResult.No)
    {
        Form1 form = new Form1();
        form.Focus();
        form.Visible= true;
        this.Hide();
    }
    else
    {

        Process.Start("shutdown", "/s /t 0");
    }
}

```

Sampai disini aplikasi sudah selesai dibuat silakan jalankan program untuk melihat hasilnya. Jika berhasil maka hasilnya akan terlihat seperti gambar berikut.



Tabel Pembahasan

<pre>public string Text {     get { return this.text; }     set { this.text = value; } }</pre>	=	Sudah dibahas pada bagian OOP
<pre>DialogResult di = MessageBox.Show("paket anda sudah habis,Ingin Tambah Paket Lagi???", "Important", MessageBoxButtons.YesNo, MessageBoxIcon.Question, MessageBoxDefaultButton.Button1);</pre>	=	Suatu DialogResult untuk menampung MessageBox yang mempunyai button YesNO dengan tipe Question kemudian button yang deseting untuk dijadikan default yaitu Button1 atau Button Yes, ini semua diampung pada variable “di”.
<pre>if (di == DialogResult.Yes) {     form = new Form1();     form.Focus();     form.Visible = true;     this.Hide(); }</pre>	=	Jika Dialogresult pada Messagebox yang ditekan adalah tombol Yes, maka Focus pada Form1 dan form1 diperlihatkan,lalu form billing.cs disembunyikan kembali
<pre>Process.Start("shutdown", "/s /t 0");</pre>	=	Perintah ini digunakan untuk menshutdown windows
<pre>Billing f2 = new Billing(); f2.Text = a; f2.Show(); this.Hide();</pre>	=	Instaniasi objek Billing.Cs dan akan mentransfer nilai pada variable “a” ke f2(Billing.Cs) yang disimpan divariabel “Text”, kemudian form1 disembunyikan.

## BAB V DATABASE DI C#

### 5.1 Pengenalan MYSQL

#### - MySQL is a database management system.

Database adalah kumpulan data terstruktur . Untuk menambah , mengakses, dan memproses data yang tersimpan dalam database komputer , Anda memerlukan sistem manajemen database seperti MySQL Server,xampp dll . Selama komputer masih sangat baik dalam menangani sejumlah besar data , sistem manajemen database memainkan peran sentral dalam komputasi , sebagai utilitas standalone , atau sebagai bagian dari aplikasi lain .

#### - MySQL databases are relational.

Sebuah relasional database yang menyimpan data dalam tabel terpisah kemudian menempatkan semua data dalam satu gudang besar . Struktur database disusun dalam file fisik dioptimalkan untuk kecepatan . Model logis , dengan benda-benda seperti *databases*, *tables*, *views*, *rows*, dan *columns*, menawarkan lingkungan pemrograman yang fleksibel . Anda dapat membuat aturan yang mengatur hubungan antara bidang data yang berbeda , seperti *one-to-one*, *one-to-many*, *unique*, *required or optional*, dan “*pointers*” antara tabel yang berbeda . Database memberlakukan aturan-aturan ini , sehingga dengan database yang dirancang dengan baik , aplikasi Anda tidak pernah terlihat tidak konsisten , menduplikasi , yatim piatu , out-of -date , atau data hilang .

SQL bagian dari " MySQL " singkatan dari " Structured Query Language " . SQL adalah bahasa standar yang paling umum digunakan untuk mengakses database . Tergantung pada lingkungan pemrograman Anda , Anda mungkin memasukkan SQL secara langsung ( misalnya , untuk menghasilkan laporan ) , pernyataan SQL menanamkan ke dalam kode yang ditulis dalam bahasa lain , atau menggunakan API bahasa tertentu yang menyembunyikan sintaks SQL .

SQL didefinisikan oleh ANSI / ISO SQL Standard . Standar SQL telah berkembang sejak tahun 1986 dan beberapa versi ada. Dalam manual ini , " SQL - 92 " mengacu pada standar yang dirilis pada tahun 1992 , " SQL : 1999" mengacu pada standar dirilis pada tahun 1999 , dan " SQL : 2003" mengacu pada versi terbaru dari standar . Kami menggunakan frasa " SQL standar " berarti versi terbaru dari Standard SQL setiap saat .

#### - MySQL software is Open Source.

Open Source berarti bahwa adalah memungkinkan bagi siapa saja untuk menggunakan dan memodifikasi perangkat lunak. Siapa saja dapat men-download software MySQL dari internet dan menggunakan tanpa harus membayar apapun . Jika Anda ingin, Anda dapat mempelajari source code dan mengubahnya sesuai dengan kebutuhan anda .

- **The MySQL Database Server is very fast, reliable, scalable, and easy to use.**

MySQL Server dapat digunakan dengan baik pada computer desktop atau laptop di samping aplikasi lainnya , web server , dan sebagainya .MySQL Server pada awalnya dikembangkan untuk menangani sebagian besar database yang jauh lebih cepat daripada solusi yang ada dan telah berhasil digunakan dalam lingkungan produksi selama beberapa tahun . Meskipun dalam pengembangan yang konstan, MySQL Server saat ini menawarkan satu set yang mempunyai banyak fungsi . Konektivitas , kecepatan , dan keamanan membuat MySQL Server sangat cocok untuk mengakses database di Internet .

- **MySQL Server works in client/server or embedded systems.**

MySQL Perangkat Lunak Basis Data adalah sistem client / server yang terdiri dari SQL server multi-threaded yang mendukung backends berbeda , beberapa program klien yang berbeda dan perpustakaan , perangkat administrasi , dan berbagai antarmuka pemrograman aplikasi ( API ) .

Mysql juga menyediakan MySQL Server sebagai embaded perpustakaan multi-threaded yang dapat Anda link ke dalam aplikasi Anda untuk mendapatkan yang lebih kecil , lebih cepat , lebih mudah - untuk-mengelola produk standalone .

- **A large amount of contributed MySQL software is available.**

MySQL Server memiliki seperangkat fitur praktis yang dapat dikembangkan oleh pengguna . Hal ini sangat mendukung aplikasi anda. [5]

Salah satu Software DMBS untuk mengolah database mysql yang paling sering digunakan yaitu xampp.

- **APA ITU XAMPP**

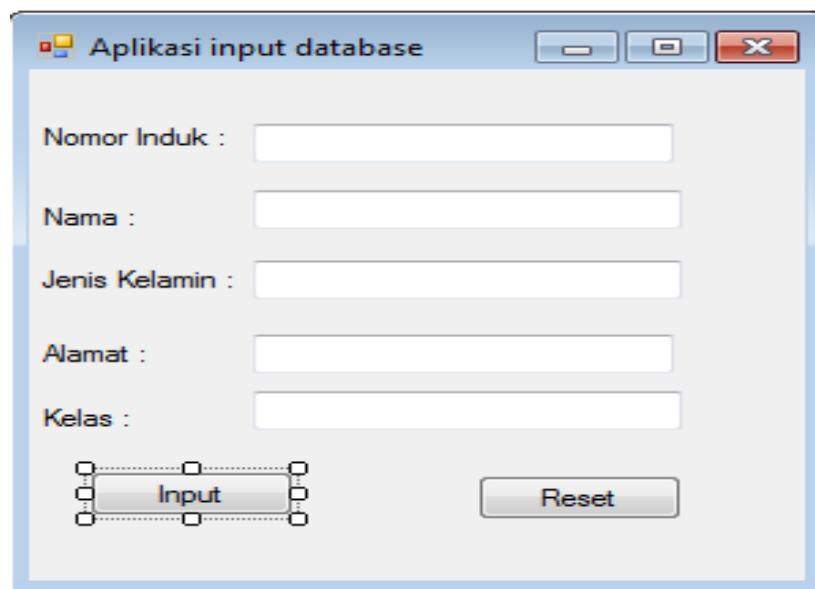
XAMPP adalah paket open-source web server yang bekerja pada berbagai platform . Hal ini sebenarnya merupakan singkatan dengan X yang berarti berarti " CROSS " platform , A untuk server Apache HTTP , M untuk MySQL , P untuk PHP , dan P untuk Perl . XAMPP dirancang untuk membantu pengembang halaman Web , programer , dan desainer memeriksa dan meninjau pekerjaan mereka menggunakan komputer mereka bahkan tanpa koneksi ke web atau internet . Jadi , pada dasarnya XAMPP dapat digunakan untuk berdiri sendiri /stand alone tanpa koneksi. Hal ini juga dapat digunakan untuk membuat dan mengkonfigurasi database MySQL yang ditulis dalam dan / atau SQLite . Dan karena XAMPP dirancang sebagai paket server cross-platform , tersedia untuk berbagai sistem operasi dan platform seperti Microsoft Windows, Mac OS X , Linux , dan Solaris.

## 5.2 Perintah Masukan ke Database MYSQL

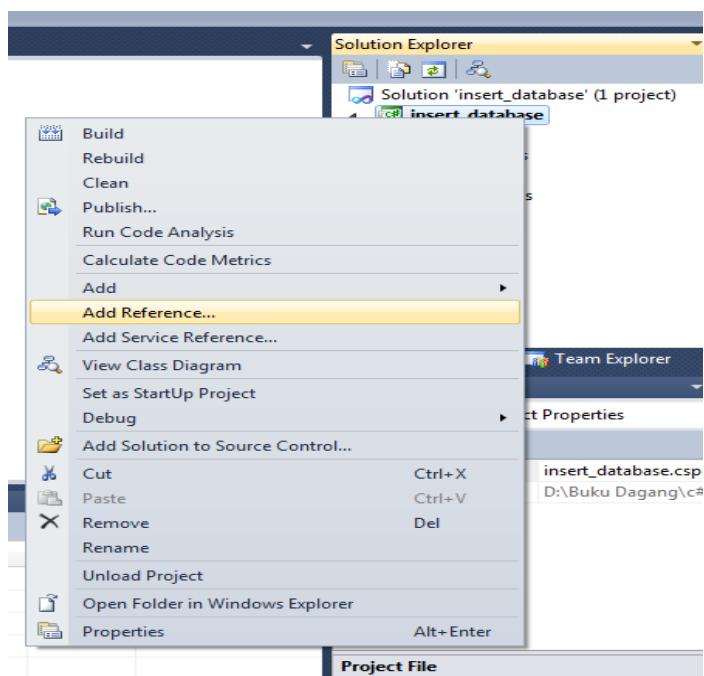
- Sebelumnya silakan anda download terlebih mysqlconnector.net -nya, untuk koneksi mysql dengan Net di <http://cdn.mysql.com/Downloads/Connector-Net/mysql-connector-net-6.1.3.zip> kemudian silakan anda install seperti biasa ,lalu buatlah terlebih dahulu databasenya dengan xampp dengan nama sekolah ,oh iyah xampp yang digunakan oleh penulis yaitu menggunakan xampp dengan versi yang sudah cukup lama yaitu versi 1.5.3 kemudian buat juga tabelnya dengan nama siswa seperti gambar dibawah.

	Field	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	<u>nomor induk</u>	int(12)			No		auto_increment	
<input type="checkbox"/>	nama	varchar(20)	latin1_general_ci		No			
<input type="checkbox"/>	jenkel	varchar(10)	latin1_general_ci		No			
<input type="checkbox"/>	alamat	varchar(100)	latin1_general_ci		No			
<input type="checkbox"/>	kelas	varchar(10)	latin1_general_ci		No			

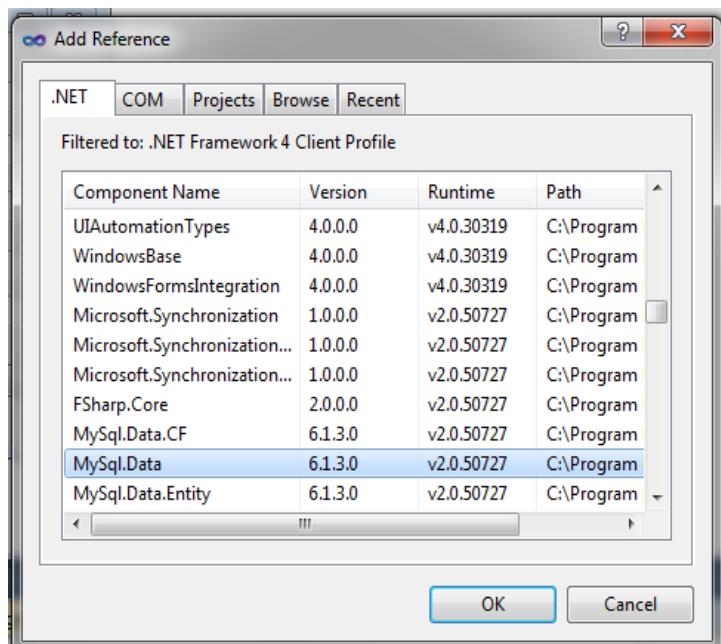
- kemudian Silahkan buatlah projek baru dengan nama insert-database, setelah itu buatlah design form seperti gambar dibawah.



- setelah itu tambahkan reference library mysql.Data; pada form dengan cara seperti gambr dibawah.



Pilih MySql.Data -> OK.



- Setelah itu silakan ketikan listing berikut.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
```

```
using System.Windows.Forms;
using MySql.Data.MySqlClient;

namespace insert_database
{
    public partial class Form1 : Form
    {
        string database = ("server = localhost; uid = root;
database=sekolah; pwd='');");
        public MySqlConnection koneksi;
        public MySqlCommand cmd;
        public Form1()
        {
            InitializeComponent();
        }

        public void Query(string query)
        {
            koneksi = new MySqlConnection(database);
            try
            {
                koneksi.Open();
                cmd = new MySqlCommand(query, koneksi);
                cmd.ExecuteNonQuery();
            }
            catch (Exception ali)
            {
                MessageBox.Show(ali.Message);

            }
            finally
            {
                koneksi.Close();
            }
        }
        private void label2_Click(object sender, EventArgs e)
        {

        }
        private void button1_Click(object sender, EventArgs e)
        {
Query("insert into siswa values('" +this.textBox1.Text +
"', '" +this.textBox2.Text + "','" +this.textBox3.Text +
"', '" +this.textBox4.Text + "','" +this.textBox5.Text +
 "')");
        }
    }
}
```

```

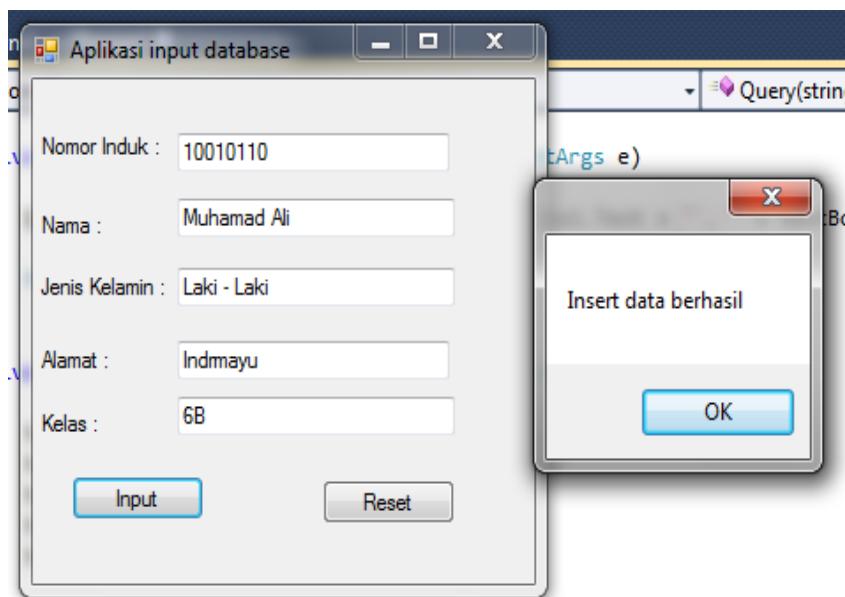
        MessageBox.Show("Insert data berhasil");
    }

    private void button2_Click(object sender, EventArgs e)
    {
        textBox1.Text = "";
        textBox2.Text = "";
        textBox3.Text = "";
        textBox4.Text = "";
        textBox5.Text = "";
    }
}
}

```

- terakhir jalankan program jika berhasil maka akan terlihat seperti gambar dibawah ,tidak lupa silakan cek database anda apakah sudah ada datanya atau belum.

Gambar tampilan aplikasi



Gambar data database yang sudah masuk

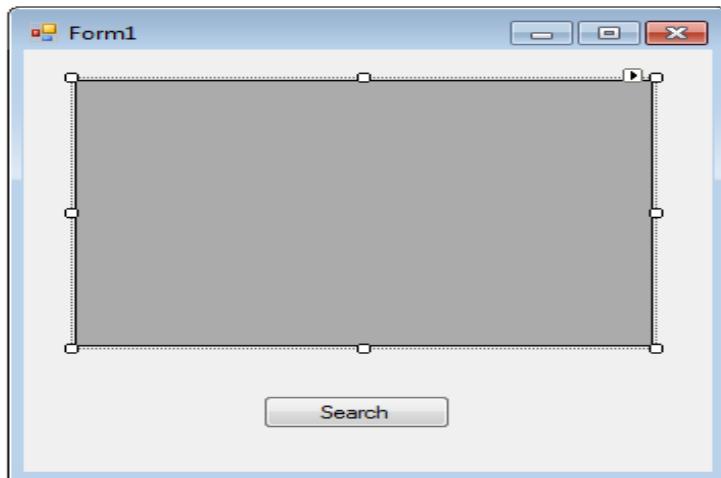
		<u>nomor induk</u>	<u>nama</u>	<u>jenkel</u>	<u>alamat</u>	<u>kelas</u>
		X 2147483647	Muhamad Ali	laki - laki	Indramayu	5B
		X 10010110	Muhamad Ali	Laki - Lak	Indramayu	6B

Tabel Pembahasan

<code>using MySql.Data.MySqlClient;</code>	=	Library tambahan untuk database Mysql
<code>string database = ("server = localhost; uid = root; database=sekolah; pwd='';");</code>	=	String yang suwaktu – waktu dapat dipanggil untuk perintah koneksi database
<code>public MySqlConnection koneksi;</code> <code>public MySqlCommand cmd;</code>	=	Koneksi dan cmd merupakan variable dari MySqlConnection dan MySqlCommand yang digunakan untuk perintah koneksi dan query di database .
<code>koneksi = new MySqlConnection(database);</code>	=	Membuat koneksi ke database.
<code>koneksi.Open();</code>	=	Membuka koneksi ke Mysql
<code>try</code> { } <code>catch (Exception ali)</code> { <code>MessageBox.Show(ali.Message);</code> }	=	Dapat digunakan untuk menangkap kesalahan, jika perintah pada try gagal di exsekusi maka akan menampilkan pesan kesalahan pada MessageBox
<code>finally {</code> <code>koneksi.Close();</code> }	=	Perintah Finally akan selalu dijalankan , koneksi.Close untuk menutup koneksi dari database.
<code>Query("insert into siswa values('"+this.textBox1.Text + "','" +this.textBox2.Text + "','" +this.textBox3.Text + "','" +this.textBox4.Text + "','" +this.textBox5.Text + "')");</code>	=	Digunakan untuk memasukan data ke database Mysql.
<code>textBox1.Text = "";</code> <code>textBox2.Text = "";</code> <code>textBox3.Text = "";</code> <code>textBox4.Text = "";</code> <code>textBox5.Text = "";</code>	=	Digunakan untuk membersihkan Textbox

### 5.3 Perintah untuk menampilkan data dari database MYSQL

- Kita masih menggunakan database yang sebelumnya untuk kasus ini, jadi jangan tutup dulu dbms xmpp anda.
- Kemudian silakan buat projek baru dengan nama **view\_database**, lalu design tampilan seperti gambar dibawah dan atur properties dan toolbox yang digunakan seperti button dan 1 buah **datagridview** . Jangan lupa pula tambahkan library untuk **Msqli.Data** –nya .



- Setelah itu ketikan listing program dibawah ini .

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using MySql.Data.MySqlClient;

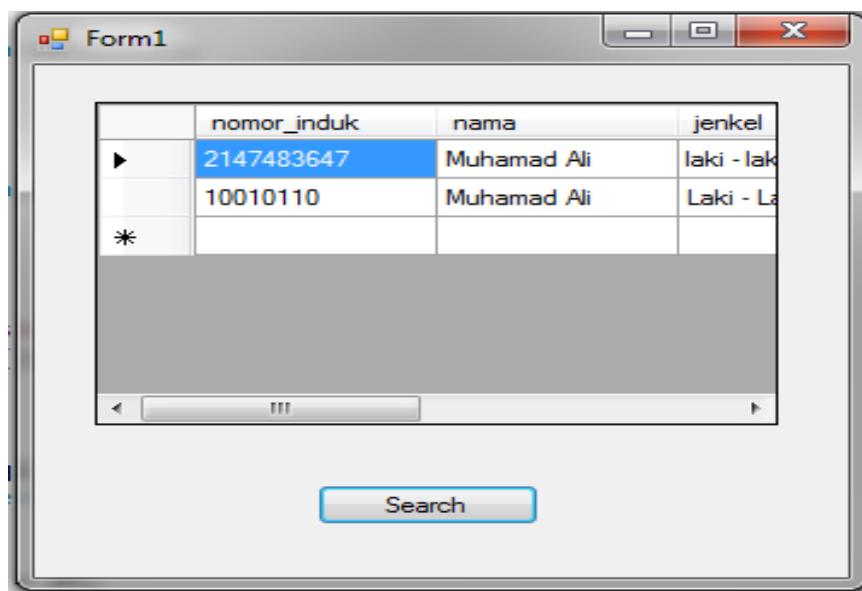
namespace view_database
{
    public partial class Form1 : Form
    {
        string database = "server = localhost;
database=sekolah; uid=root; pwd=''";
        public MySqlConnection koneksi;
        public MySqlCommand cmd;
        public MySqlDataAdapter adp;
        public Form1()
        {
            InitializeComponent();
        }

        public void koneksi()
        {
            koneksi = new MySqlConnection(database);
            koneksi.Open();
        }
        public void disconek()
        {
            koneksi = new MySqlConnection(database);
```

```
koneksi.Close();
}
public DataTable baca()
{
    string sql = "select * from siswa";
    DataTable dt = new DataTable();
    try
    {
        koneksi();
        cmd = new MySqlCommand(sql,koneksi);
        adp = new MySqlDataAdapter(cmd);
        cmd.ExecuteNonQuery();
        adp.Fill(dt);
        dataGridView1.DataSource = dt;
    }
    catch(Exception ali)
    {
        MessageBox.Show(ali.Message);
    }
    disconeck();
    return dt;
}

}
private void button1_Click(object sender, EventArgs e)
{
    baca();
}
}
```

- Sampai disini aplikasi sudah selesai dibuat , silakan jalankan program untuk melihat hasilnya. Jika berhasil maka akan terlihat seperti gambar dibawah ini.

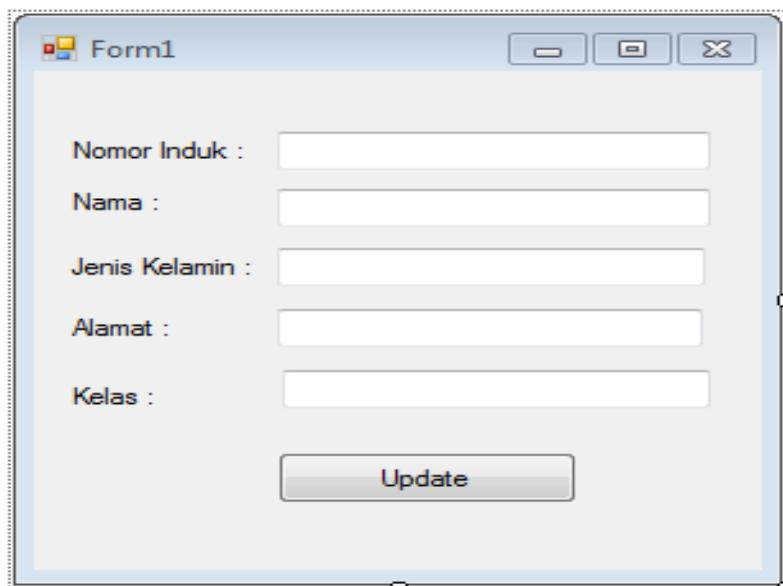


Tabel Pembahasan

<pre>public void koneksi() { koneksi = new MySqlConnection(database); koneksi.Open(); } public void disconek() { koneksi = new MySqlConnection(database); koneksi.Close(); }</pre>	=	Membuat Method koneksi dan diskoneksi
<pre>DataTable dt = new DataTable();</pre>	=	Variabel dt digunakan untuk mewakili DataTable.
<pre>adp.Fill(dt); dataGridView1.DataSource = dt;</pre>	=	Memasukan nilai ke datatable dari dataadapter yang diwakili variable "adp" dan "dt". Kemudian datasource dari dataGridView1 diperoleh dari variable "dt"

#### 5.4 Perintah Update ke Database MYSQL

- Kita masih menggunakan database yang sebelumnya untuk kasus ini, jadi jangan tutup dulu dbms xmpp anda.
- langahnya masih sama seperti diatas, silakan buat projek baru dengan nama **Update\_database**, lalu atur designya seperti gambar dibawah. Jangan Lupa pula dengan tambahkan **add reference** nyah.



- Setelah itu silakan isikan listing program berikut.

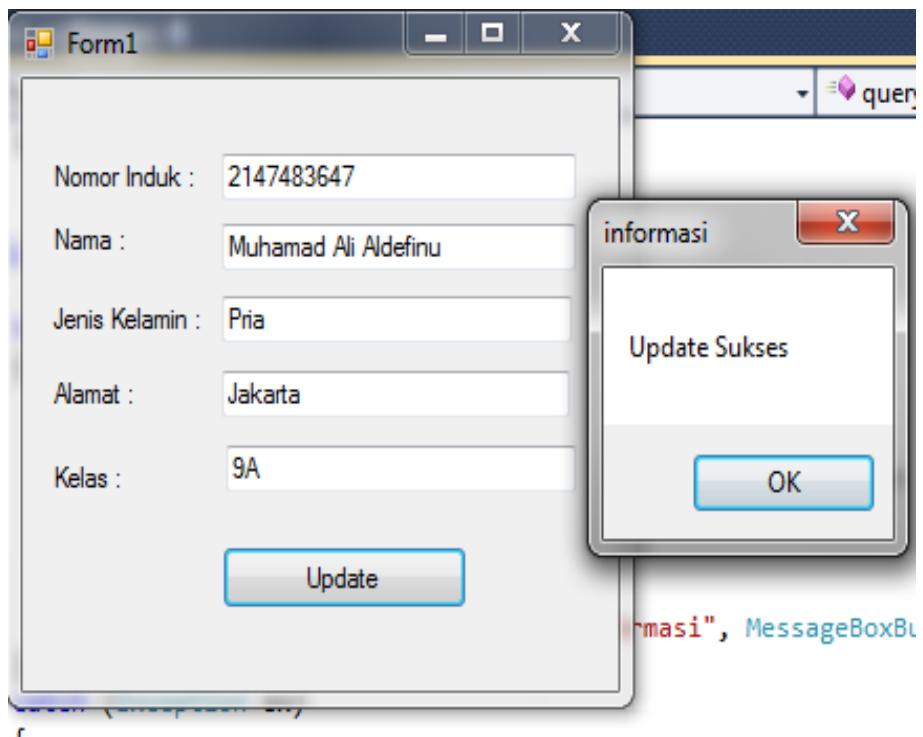
```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using MySql.Data.MySqlClient;

namespace Update_database
{
    public partial class Form1 : Form
    {
        string database = "server = localhost; database=sekolah;
uid=root; pwd=''";
        public MySqlConnection koneksi;
        public MySqlCommand command;
        public Form1()
        {
            InitializeComponent();
        }

        public void query()
        {
            try
            {
                koneksi = new MySqlConnection(database);
                koneksi.Open();
                string update = "UPDATE siswa SET nama='" + textBox2.Text +
                "',jenkel ='" + textBox3.Text + "',alamat='"
                + textBox4.Text + "',kelas='"
                + textBox5.Text + "' WHERE nomor_induk='"
                + textBox1.Text + "'";
                command = new MySqlCommand(update, koneksi);
                command.ExecuteNonQuery();
                koneksi.Close();
                MessageBox.Show("Update Sukses", "informasi",
                MessageBoxButtons.OK);
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.Message);
            }
        }
        private void button1_Click(object sender, EventArgs e)
        {
            query();
        }
    }
}
```

}

- Sampai disini aplikasi sudah selesai dibuat, silakan jalankan program untuk melihat hasilnya, oh iya jangan lupa masukan Nomor induk yang ingin di database. Jika berhasil aplikasi akan berjalan seperti gambar dibawah.

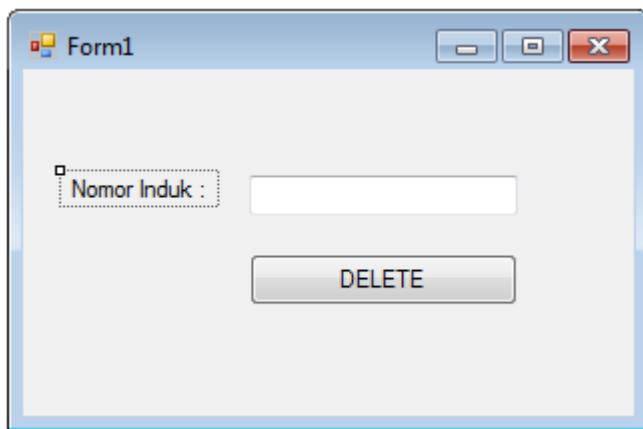


## Tabel Pembahasan

<pre>string update = "UPDATE siswa SET nama=' " + textBox2.Text + ',jenkel =' " + textBox3.Text + "',alamat=' " + textBox4.Text + ',kelas=' " + textBox5.Text + "'WHERE nomor_induk=' " + textBox1.Text + "';</pre>	=	<p>String yang suwaktu – waktu dapat di panggil untuk perintah Update Database.</p>
<pre>MessageBox.Show("Update Sukses", "informasi", MessageBoxButtons.OK);</pre>	=	<p>Menampilkan sebuah MessageBox dengan diseratai Tombol Button OK</p>

## 5.5 Printah Delete Database MySql

- Kita juga masih akan menggunakan database yang sebelumnya untuk kasus ini, jadi jangan tutup dulu dbms xmpp anda.
- langahnya masih sama seperti diatas, silakan buat projek baru dengan nama **deletemysql**, lalu atur designya seperti gambar dibawah. Jangan Lupa pula dengan tambahkan add **reference** nyah.



- Berikut adalah listing program secara keseluruhan, silakan ketikan pada projek anda.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using MySql.Data.MySqlClient;

namespace deletemysql
{
    public partial class Form1 : Form
    {
        string database = "server = localhost; database=sekolah;
uid=root; pwd=''";
        public MySqlConnection koneksi;
        public MySqlCommand command;
        public Form1()
        {
            InitializeComponent();
        }
        public void query()
```

```

    {
        try
        {
            koneksi = new MySqlConnection(database);
            koneksi.Open();
            string del = "delete from siswa WHERE
nomor_induk='"+textBox1.Text+"'";
            command = new MySqlCommand(del, koneksi);
            command.ExecuteNonQuery();
            koneksi.Close();
            MessageBox.Show("Delete Sukses", "informasi",
MessageBoxButtons.OK);
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.Message);
        }
    }

    private void button1_Click(object sender, EventArgs e)
    {
        query();
    }
}

```

- Silakan jalankan program untuk melihat hasilnya jika berhasil,maka akan terlihat seperti gambar dibawah. Jangan lupa cek pula di database anda apakah sudah berhasil di delete atau belum.



Gambar data database dengan nomor id yg di delete sudah hilang.

	<u>nomor induk</u>	<u>nama</u>	<u>jenkel</u>	<u>alamat</u>	<u>kelas</u>
	10010110	Muhamad Ali	Laki - Lak	Indrmayu	6B
	13112098	Muhamad Arif	Pria	Jakarta	13.B

Tabel Pembahasan

<code>string del = "delete from siswa WHERE nomor_induk='"+textBox1.Text+"'";</code>	=	String yang suwaktu – waktu dapat di panggil untuk perintah Delete Database.
<code>command = new MySqlCommand(del, koneksi); command.ExecuteNonQuery();</code>	=	Untuk memasukan query ke database

## 5.6 Pengenalan MS.Access

Microsoft Access , juga dikenal sebagai Microsoft Office Access adalah sebuah sistem manajemen database dari Microsoft yang menggabungkan relasional Microsoft Jet Database Engine dengan antarmuka pengguna grafis dan software-development. Ini merupakan anggota dari Microsoft Office suite aplikasi , termasuk dalam edisi profesional dan lebih tinggi atau dijual secara terpisah .

Microsoft Access menyimpan data dalam format sendiri berdasarkan Akses Jet Database Engine . Hal ini juga dapat mengimpor atau link langsung ke data yang tersimpan dalam aplikasi dan database lain . Pengembang perangkat lunak dan arsitek data yang dapat menggunakan Microsoft Access untuk mengembangkan perangkat lunak aplikasi , dan " power user " dapat menggunakannya untuk membangun aplikasi perangkat lunak . Seperti aplikasi Office lainnya , Akses didukung oleh Visual Basic untuk aplikasi , bahasa pemrograman berorientasi objek yang yang dapat mereferensikan berbagai tools termasuk DAO (Data Access Objects ) , ActiveX Data Objects , dan banyak komponen ActiveX lainnya . Objek visual digunakan dalam bentuk dan laporan mengekspos metode dan properti di lingkungan pemrograman VBA mereka , dan kode VBA modul dapat menyatakan dan memanggil Windows Fungsi sistem operasi .

- Apa Manfaat Microsoft Acess ?

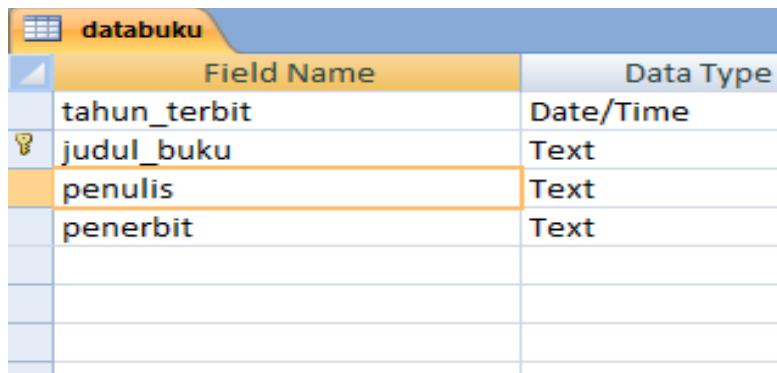
Sangat sederhana, Microsoft Access adalah alat manajemen informasi yang membantu Anda menyimpan informasi untuk referensi, pelaporan, dan analisis. Microsoft Access membantu Anda menganalisis sejumlah besar informasi, dan mengelola data terkait lebih efisien daripada Microsoft Excel atau aplikasi spreadsheet lainnya. Artikel ini menunjukkan Anda kapan harus menggunakan Access, dan bagaimana hal itu dapat membantu membuat Anda lebih efektif.

## 5.7 Insert,Update,Delete dan View dengan C# dan Ms.Access

Pada intinya printah akses database Ms. Acceess dengan C# hampir mirip dengan mysql mungkin hanya akan ada beberapa perbedaan saja,namun jika anda sudah memahami konsep manipulasi database mysql dengan C# anda tentu tidak akan kesulitan sekalipun anda menggunakan MS.Access. Akan tetapi pada halaman ini

kita akan mengakses database dengan konsep yang berbeda yaitu kita akan menggunakan **Class** untuk mengakses databasenya.

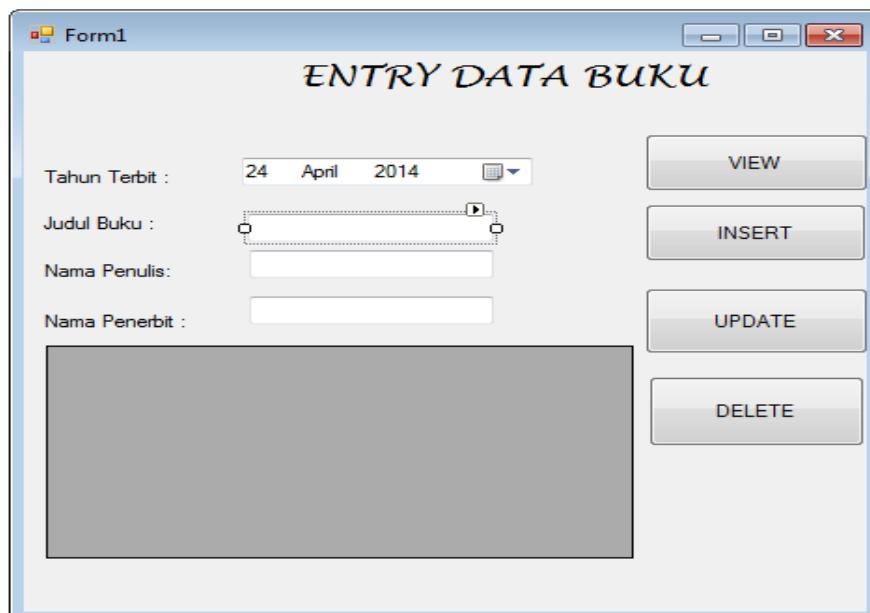
- Sebelum membuat projek baru, apastikan terlebih dahulu membuat database dengan Ms.Access dengan nama “**buku**”, dan buatlah struktur tablenya seperti gambar dibawah dengan nama **databuku**. Jangan lupa untuk sementara simpan file databasenya di direktori **\bin\Debug** projek anda.



A screenshot of the Microsoft Access application showing the table structure for 'databuku'. The table has four fields: 'tahun\_terbit' (Date/Time), 'judul\_buku' (Text), 'penulis' (Text), and 'penerbit' (Text). The 'judul\_buku' field is currently selected.

	Field Name	Data Type
	tahun_terbit	Date/Time
!	judul_buku	Text
!	penulis	Text
	penerbit	Text

- Silakan buatlah projek baru dan atur designnya seperti gambar dibawah. Adapun beberapa komponen yang dibutuhkan yaitu , datetimepicker, button, label, textbox dan datagridview.



- Setelah Itu silakan buatlah sebuah kelas baru dengan nama **koneksidatabase.cs** . Sudah tahu kan cara membuat klasnya seperti yang dibahs di bab OOP. Setelah itu silakan ketikan listing berikut.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Windows.Forms;
```

```
using System.Data.OleDb;
using System.Data;

namespace insert_update_delete_view_dengan_access
{

    class koneksidatabase
    {
        string database = "provider=Microsoft.ACE.OLEDB.12.0;Data
Source= buku.accdb";
        string sql = " SELECT * FROM databuku";
        public OleDbConnection koneksi;
        public OleDbCommand perintah;
        public OleDbDataAdapter adp;

        public void connect()
        {
            koneksi = new OleDbConnection(database);
            koneksi.Open();
        }
        public void disconected()
        {
            koneksi = new OleDbConnection(database);
            koneksi.Close();
        }

        public DataTable tampil()
        {
            DataTable dt = new DataTable();
            try
            {
                connect();
                perintah = new OleDbCommand(sql, koneksi);
                adp = new OleDbDataAdapter(perintah);
                perintah.ExecuteNonQuery();
                adp.Fill(dt);
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString());
            }
            return dt;
        }

        public void QUERY (string query)
    {

```

```
try
{
    connect();
    perintah = new OleDbCommand(query, koneksi);
    perintah.ExecuteNonQuery();

}
catch (Exception ali)
{
    MessageBox.Show(ali.Message);

}
finally
{
    disconected();
}
}
```

- Dan berikut adalah listing untuk **Form1.cs** -nya. Silakan diketik di projek anda.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace insert_update_delete_view_dengan_access
{

    public partial class Form1 : Form
    {

        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            koneksidatabase koneksi = new koneksidatabase();
            DataTable dt = new DataTable();
            dt = koneksi.tampil();
            dataGridView1.DataSource = dt;
        }
    }
}
```

```

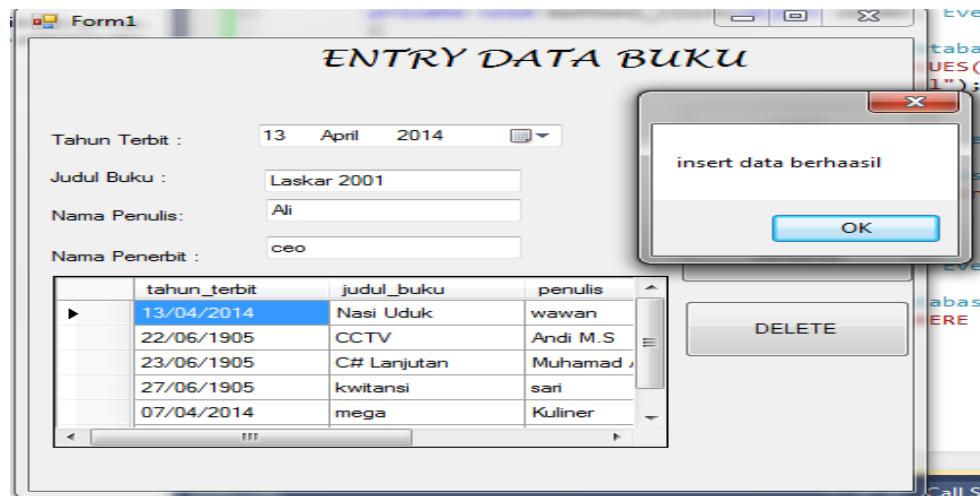
private void Form1_Load(object sender, EventArgs e)
{
}

private void button2_Click(object sender, EventArgs e)
{
    koneksi database kont = new koneksi database();
    kont.QUERY("INSERT INTO databuku VALUES('" +
    this.dateTimePicker1.Text + "','" + this.textBox1.Text +
    "','" + this.textBox2.Text + "','" + this.textBox3.Text +
    "')");
    MessageBox.Show("insert data berhaasil");
}
private void button3_Click(object sender, EventArgs e)
{
    koneksi database kont = new koneksi database();
    kont.QUERY("UPDATE databuku set tahun_terbit =
    '" + this.dateTimePicker1.Text + "',penulis = '" +
    this.textBox2.Text + "',penerbit='" + this.textBox3.Text +
    "' WHERE judul_buku='" + this.textBox1.Text + "'");
    MessageBox.Show("Update success");
}

private void button4_Click(object sender, EventArgs e)
{
    koneksi database kont = new koneksi database();
    kont.QUERY("DELETE * from databuku WHERE
    judul_buku= '" + textBox1.Text + "'");
    MessageBox.Show("Delete success");
}
}

```

- Sampai disini aplikasi sudah selesai dibuat , silakan running untuk melihat aplikasinya.



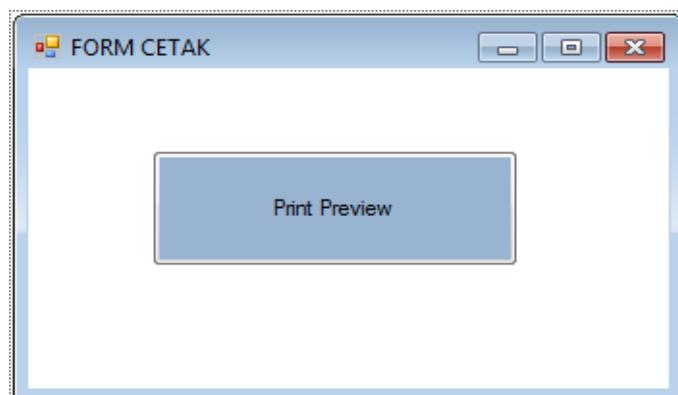
- Untuk pembahasan diatas saya rasa sangat mudah dipahami karena hamper mirip dengan pembahasan pada halaman – halaman sebelumnya, silakan anda coba untuk memahaminya sendiri.

## BAB VI REPORT DAN DATABASE LANJUTAN

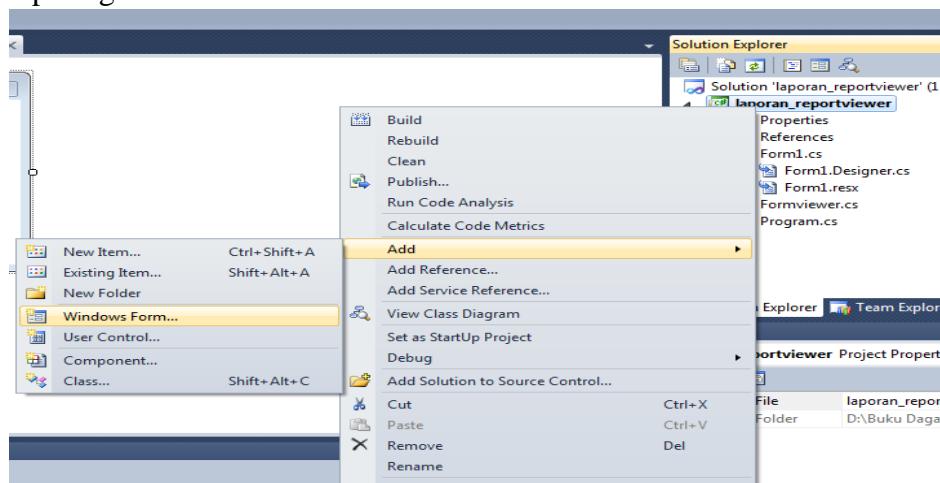
### 6.1 Membuat Report menggunakan ReportViewer

Report atau laporan sangatlah diperlukan dalam suatu aplikasi apalagi bila aplikasi yang kita buat adalah aplikasi berbasis bisnis, pasti tidaklah lepas dari yang namanya laporan seperti laporan data barang, laporan stok gudang dll. Kita masih menggunakan database buku yang dibuat dengan Ms.Access pada bab sebelumnya, untuk dijadikan report pada aplikasi kali ini. Jangan lupa Copy dulu file databasenya dan taro di tempat yang menurut anda lebih mudah dicari.

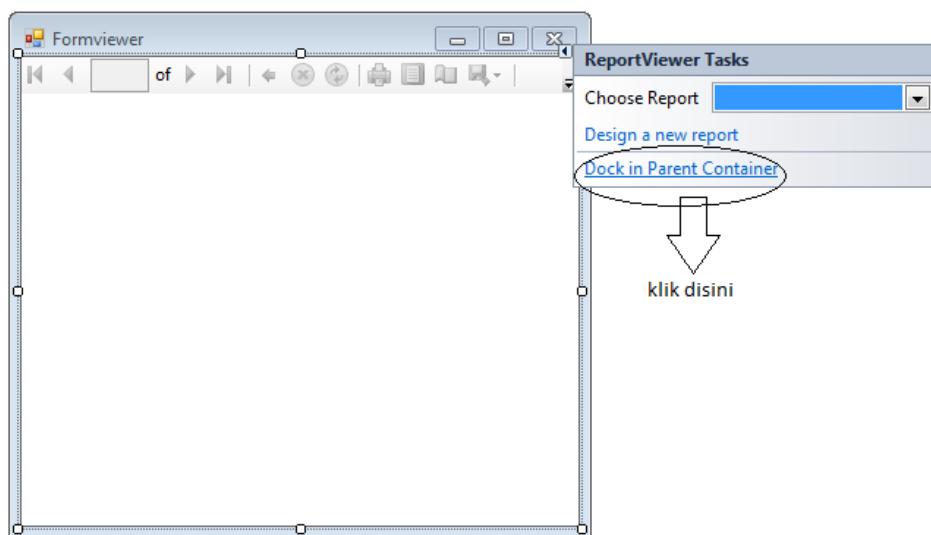
- Pertama – tama silakan buatlah terlebih dahulu projek dengan nama laporan\_reportviewer pada visual c# anda, kemudian atur desain gambar dibawah ini.



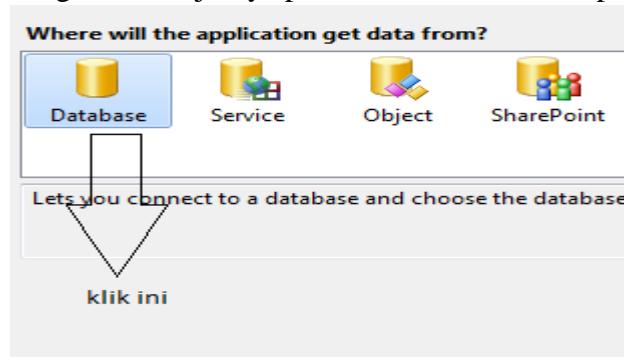
- Kemudian buat lagi form2 atau tambahkan form dengan cara dengan cara seperti gambar dibawah ini.



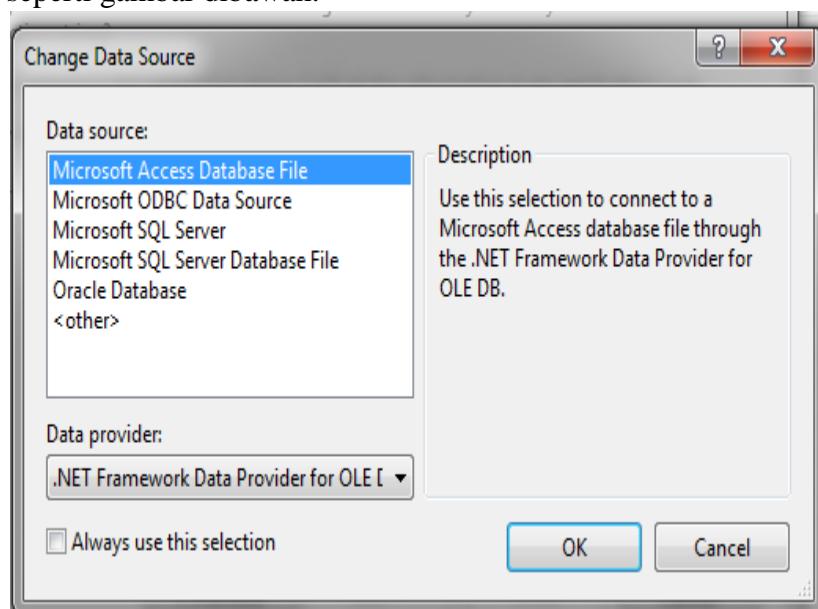
- Jangan lupa kasih nama formnya berikan saja nama **Formviewer.cs**
- Kemudian tambahkan komponen report viewer dan atur tampilannya seperti gambar form berikut.



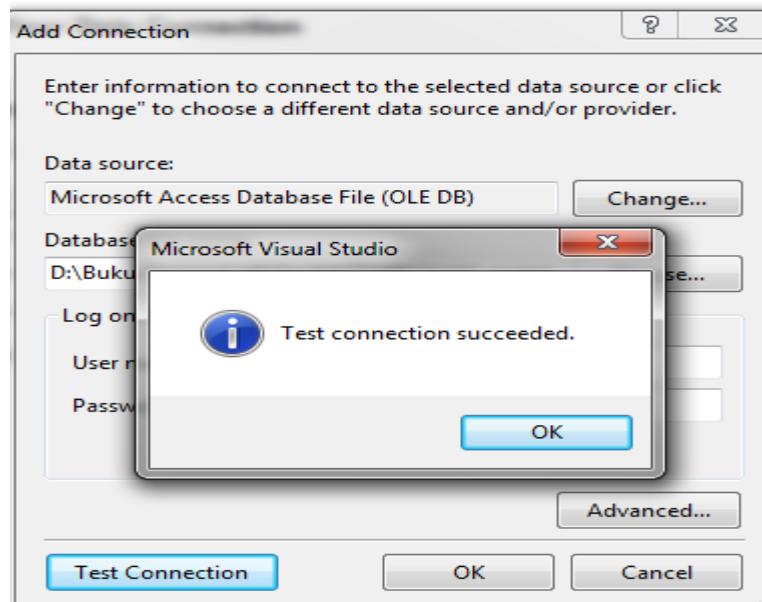
- Langkah berikutnya silahkan klik juga **design a new report** seperti diatasnya Dock in parent Container
- Langkah Selanjutnya pilih database -> next seperti gambar dibawah.



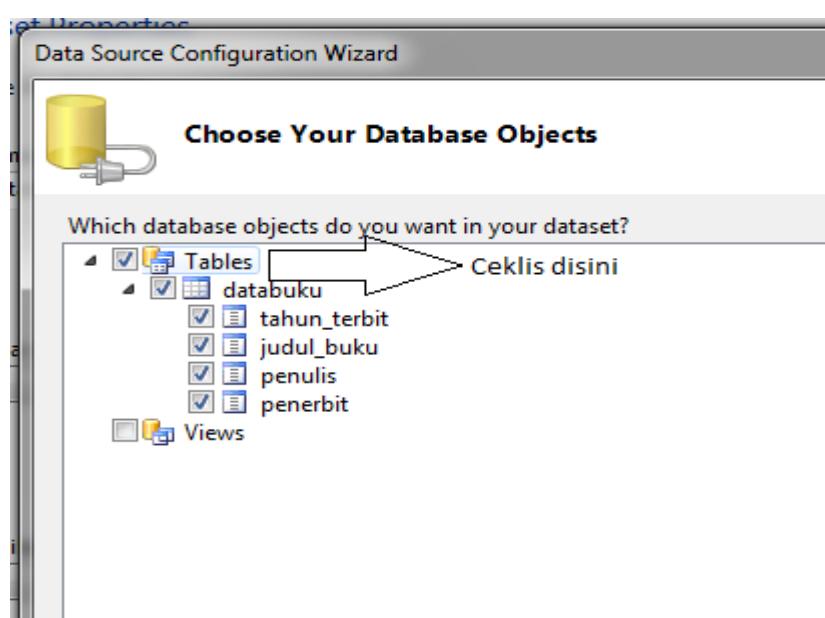
- Klik dataset-.next kemudian klik juga new connection
- Klik cange connection dan pilih Microsoft Access database file -> ok, seperti gambar dibawah.



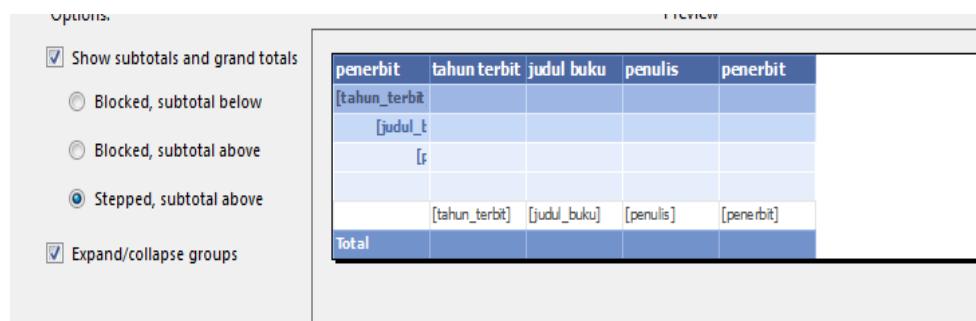
- Kemudian pada pilan database file name klik browse,dan cari file database access anda.
- Kemudian pastikan cek dulu koneksi databasenya dengan cara klik test connection. Jika berhasil maka akan mendapat pesan seperti gambar dibawah.



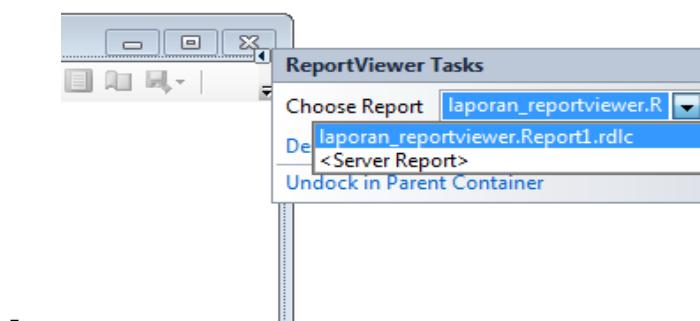
- Username isikan saja Admin kemudian password kosongkan(hal ini supaya password tidak lupa) untuk belajar sebaiknya kosongkan saja.
- Kemudian klik next-yes-next saja . oh iya connection string itu , variable yang bias dipanggil kalo sewaktu – waktu kita gunakan database itu lagi.
- Nah jika berhasil akan seperti gambar dibawah ini, Jangan ceklis pada Tables lalu klik finis lalu next.



- klik dan tahan geser dari available fields ke row grup Value. Atau anda juga bisa klik tahan geser ke pilihan lainnya.
- Selanjutnya next , dan atur design sesuai keinginan anda , perhatikan pula gambar berikut .



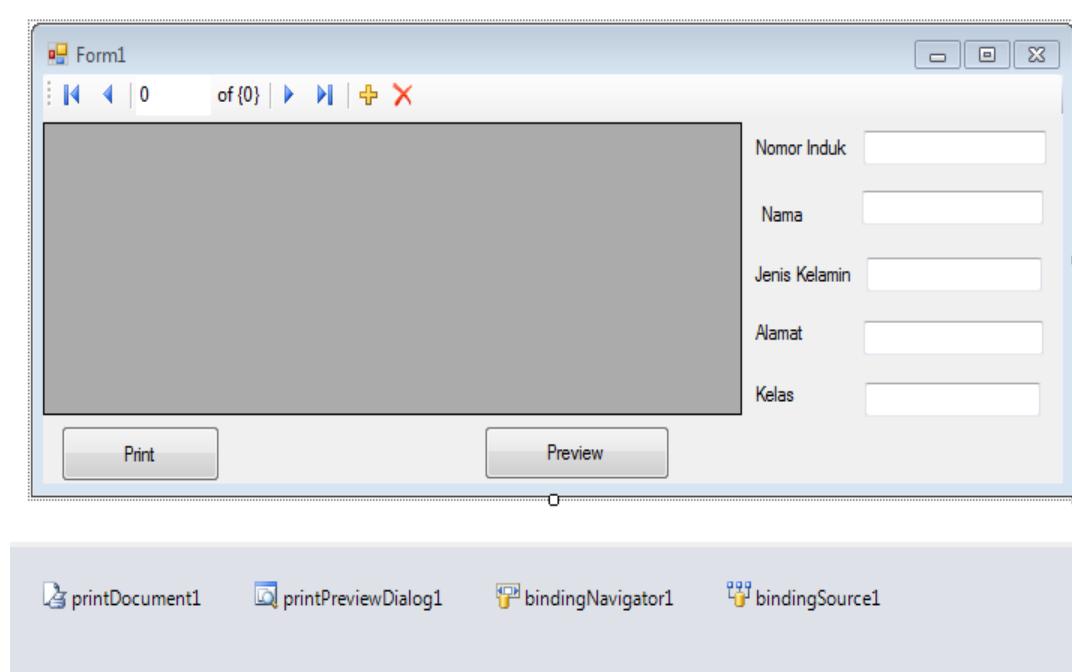
- Klik next saja - > Finish, kemudian akan dibawa kehalaman **Report1.rdlc** . atur saja seperti gambar dibawah.
- Pada **formviewer.cs** atur seperti gambar dibawah.



- Kembali ke Form1.cs[design], kemudian double klik pada button dan masukan listing berikut.
- ```
private void button1_Click(object sender, EventArgs e)
{
    Formviewer fw = new Formviewer();
    fw.Show();
}
```
- Program sudah selesai dibuat silahkan jalankan untuk melihat hasilnya. Jika berhasil ketika tombol Print preview di klik,maka akan tampil laporan preview.

## 6.2 Membuat Report menggunakan PrintDocument dan PrintPreviewDialog

Pertama – tama silahkan anda buat terlebih dahulu projek baru anda dengan nama **laporanprintdialog**, kemudian atur deaignya seperti gambar dibawah ini.



- Untuk komponen – komponen yang dibutuhkan yaitu lihat table propertie dibawah. Sementara untuk komponen textbox biartkan default saja.

| Toolbox             | Properties | Keterangan          |
|---------------------|------------|---------------------|
| label1              | text       | Nomor Induk         |
| label2              | text       | Nama                |
| label3              | text       | Jenis Kelamin       |
| label4              | text       | Alamat              |
| label5              | text       | Kelas               |
| button1             | text       | print               |
| button2             | text       | Preview             |
| dataGridView1       | name       | dataGridView1       |
| bindingNavigator1   | name       | bindingNavigator1   |
| bindingSource1      | name       | bindingSource1      |
| printDocument1      | name       | printDocument1      |
| printPreviewDialog1 | name       | printPreviewDialog1 |

- Kemudian silakan ketikan listing dibawah ini.
- `using System;`  
`using System.Collections.Generic;`  
`using System.ComponentModel;`  
`using System.Data;`

```
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using MySql.Data.MySqlClient;
using System.Data;

namespace laporanprintdialog
{
    public partial class Form1 : Form
    {
        public string databases = "server=localhost;
database=sekolah; uid=root; pwd=''";
        public MySqlConnection koneksi;
        public MySqlCommand command;
        public MySqlDataAdapter adp;
        public MySqlDataReader rdr;

        public Form1()
        {
            InitializeComponent();
            this.MaximizeBox = false;
            this.MinimizeBox = false;
            this.FormBorderStyle =
FormBorderStyle.FixedSingle;

        }
        public DataTable tampil()
        {
            string sql = "select * from siswa";
            DataTable dt = new DataTable();
            try
            {
                koneksi = new MySqlConnection(databases);
                koneksi.Open();
                command = new MySqlCommand(sql, koneksi);
                adp = new MySqlDataAdapter(command);
                adp.Fill(dt);
                command.ExecuteNonQuery();
                bindingSource1.DataSource = dt;
                dataGridView1.DataSource = bindingSource1;
                bindingNavigator1.BindingSource = bindingSource1;
            }
            catch (Exception ali)
            {
                MessageBox.Show(ali.Message);
            }
            koneksi.Close();
        }
    }
}
```

```
        return dt;
    }

    public void pindahketextbox()
    {
textBox1.Text =
dataGridView1.CurrentRow.Cells["nomor_induk"].Value.ToString();
textBox2.Text = dataGridView1.CurrentRow.Cells["nama"].Value.ToString();
textBox3.Text = dataGridView1.CurrentRow.Cells["jenkel"].Value.ToString();
textBox4.Text = dataGridView1.CurrentRow.Cells["alamat"].Value.ToString();
textBox5.Text = dataGridView1.CurrentRow.Cells["kelas"].Value.ToString();

}
private void Form1_Load(object sender, EventArgs e)
{
    tampil();

}

private void dataGridView1_SelectionChanged(object sender, EventArgs e)
{
    pindahketextbox();

}

private void button2_Click(object sender, EventArgs e)
{
    printPreviewDialog1.Document = printDocument1;
    printPreviewDialog1.PrintPreviewControl.Zoom = 1;
    printPreviewDialog1.ShowDialog();
}

private void button1_Click(object sender, EventArgs e)
{

printDocument1.PrinterSettings.DefaultPageSettings.Landscape = false;
printDocument1.Print();
}

private void printDocument1_PrintPage(object sender, System.Drawing.Printing.PrintEventArgs e)
{
var font = new Font("Khmer UI", 12);
e.Graphics.DrawString("\t\t\t Data Siswa - Siswi
\n\n\n", font, Brushes.Black, 25, 10);
```

```

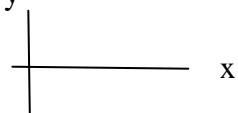
e.Graphics.DrawString("NIM           \t:" +
textBox1.Text + "\n\n", font, Brushes.Black, 20, 30);
e.Graphics.DrawString("NAMA          \t:" +
textBox2.Text + "\n\n", font, Brushes.Black, 20, 50);
e.Graphics.DrawString("Jenis Kelamin \t:" +
textBox3.Text + "\n\n", font, Brushes.Black, 20, 80);
e.Graphics.DrawString("Alamat        \t:" +
textBox4.Text + "\n\n", font, Brushes.Black, 20, 110);
e.Graphics.DrawString("Kelas         \t:" +
textBox5.Text + "\n\n", font, Brushes.Black, 20, 130);

}
}

}

```

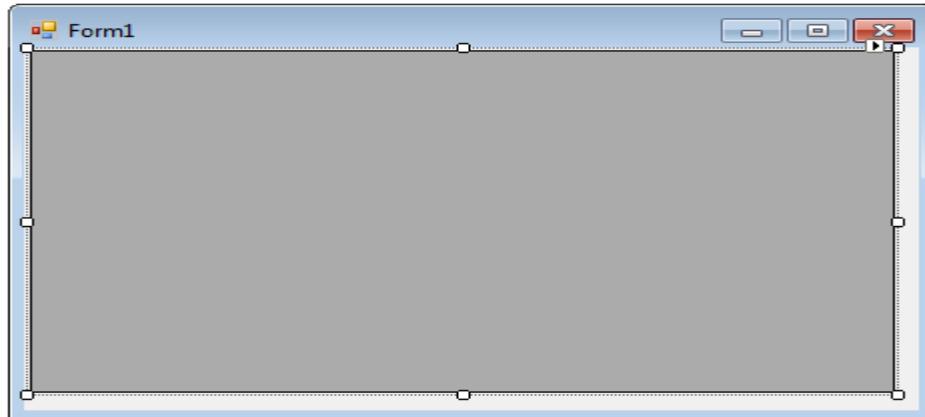
Tabel Pembahasan

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>private void<br/>dataGridView1_SelectionChanged(o<br/>bject sender, EventArgs e)</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | = | Printah ini dibangkitkan bila datagridview1 dalam keadaan terseleksi                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| <code>printDocument1.Print();</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | = | Digunakan untuk mengeprint document                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| <code>private void<br/>printDocument1_PrintPage(object<br/>sender,<br/>System.Drawing.Printing.PrintPageE<br/>ventArgs e)<br/>{<br/>var font = new Font("Khmer UI", 12);<br/>e.Graphics.DrawString("\t\t\t Data<br/>Siswa - Siswi \n\n\n", font,<br/>Brushes.Black, 25, 10);<br/>e.Graphics.DrawString("NIM<br/>\t:" + textBox1.Text + "\n\n", font,<br/>Brushes.Black, 20, 30);<br/>e.Graphics.DrawString("NAMA<br/>\t:" + textBox2.Text + "\n\n", font,<br/>Brushes.Black, 20, 50);<br/>e.Graphics.DrawString("Jenis<br/>Kelamin \t:" + textBox3.Text +<br/>"\n\n", font, Brushes.Black, 20, 80);<br/>e.Graphics.DrawString("Alamat<br/>\t:" + textBox4.Text + "\n\n", font,<br/>Brushes.Black, 20, 110);<br/>e.Graphics.DrawString("Kelas<br/>\t:" + textBox5.Text + "\n\n", font,</code> | = | <p>Printah ini digunakan untuk menseting printdocument,dan memasukan listing ini dengan cara klik dua kali pada printdocument1,</p> <p>Sementara</p> <p>e.Graphics.DrawString("\t\t\t Data<br/>Siswa - Siswi<br/>\n\n\n", font,<br/>Brushes.Black, 25, 10); ini artinya untuk menseting huruf <b>Brushes.Black</b> artinya dengan warna hitam pada document dengan menseting penempatanya pada kordinat x dan y .</p>  |

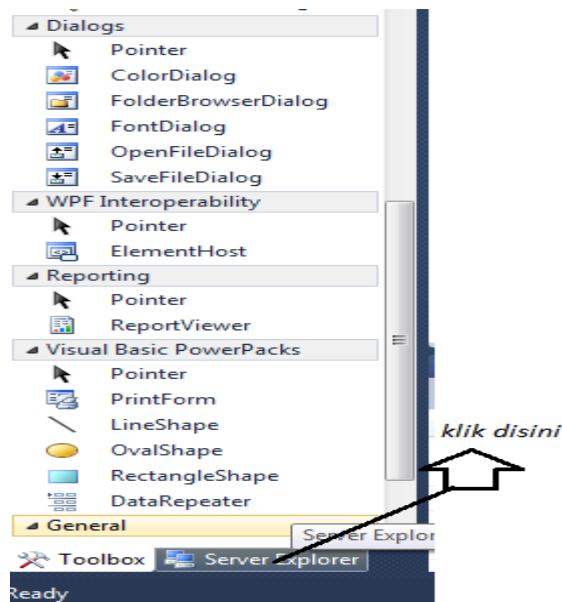
|                                                                                                                                                     |                                                |
|-----------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------|
| Brushes.Black, 20, 130);<br>} <td>Brus</td>                                                                                                         | Brus                                           |
| printPreviewDialog1.Document =<br>printDocument1;<br><br>printPreviewDialog1.PrintPreviewControl.Z<br>oom = 1;<br>printPreviewDialog1.ShowDialog(); | = Seting Properti pada<br>printpreviewdialog1. |

#### 6.4 Cara Mudah Koneksi Database

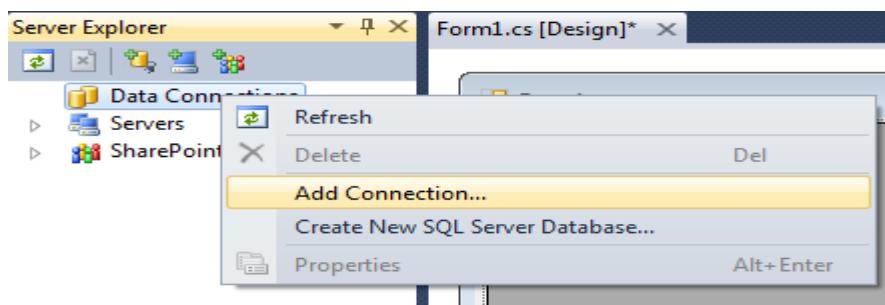
- Pertama – tama silakan anda buat terlebih dahulu projeknya kasih nama saja dengan nama **koneksi\_cepat**, kemudian buat desain dan ikuti langkah – langkahnya seperti gambar berikut. Sementara tool yang digunakan yaitu hanya satu buah Gridview.



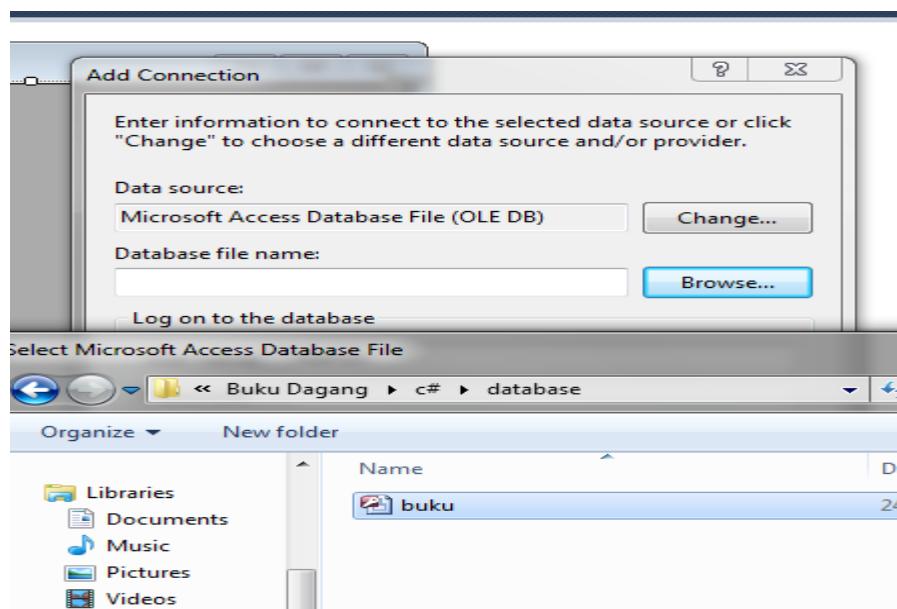
- Kemudian ikutilah langkah – langkahnya seperti gambar – gambar berikut.



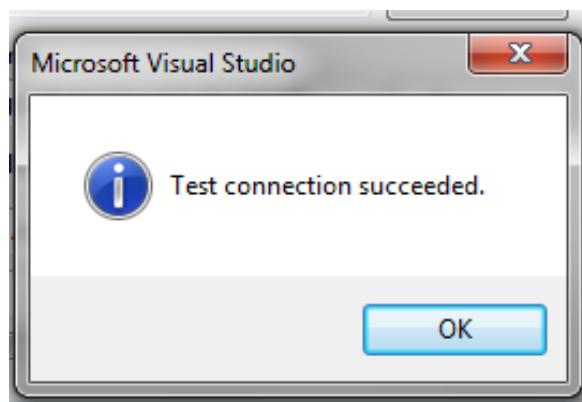
- Kemudian klik kanan pada data connection dan pilih add Connection



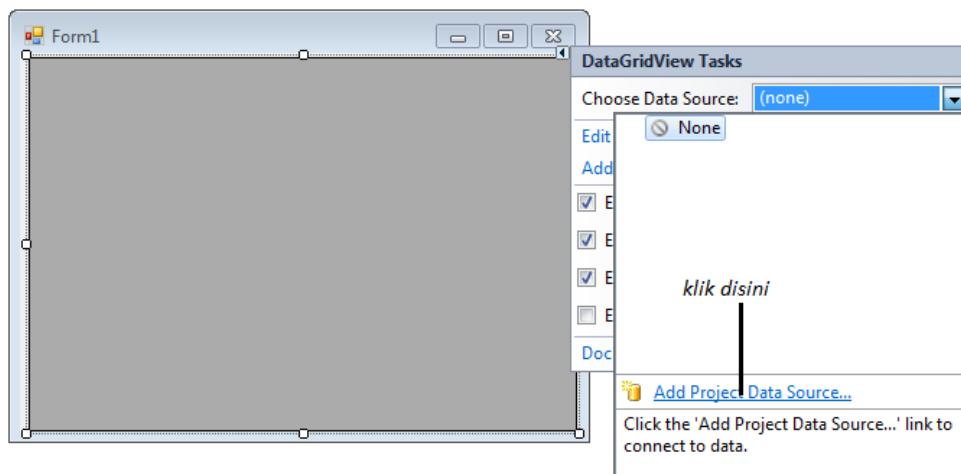
- Kemudian klik browse dan pilih lokasi file penyimpanan database anda, dalam contoh ini kita masih menggunakan database buku, yang sudah pernah dibuat sebelumnya.



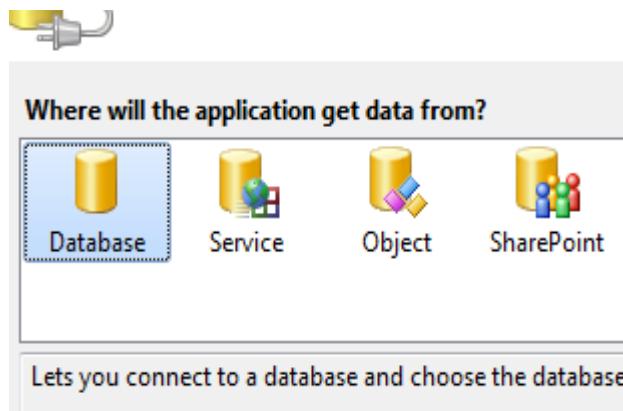
- Kemudian silahkan klik test connection untuk memastikan koneksi berhasil



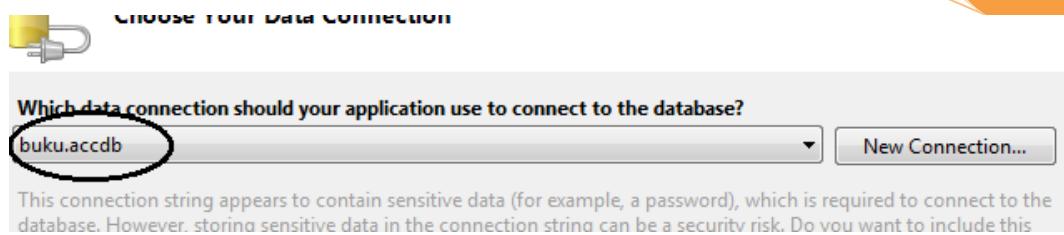
- Berikutnya klik – OK
- Berikutnya klik pada pojok kanan atas komponen datagridview1, perhatikan gambar dibawah.



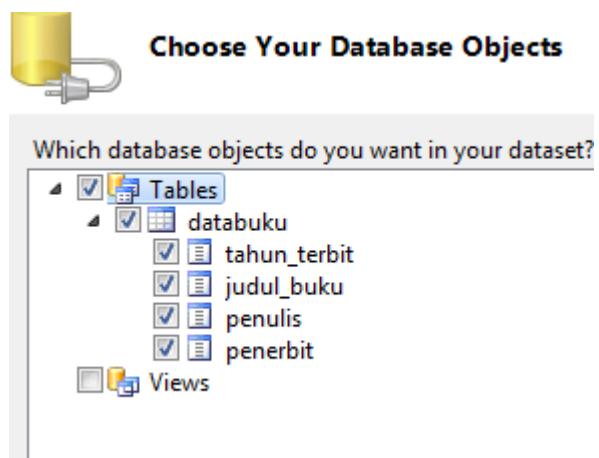
- Pilih Database dan klik next-dataset-next



- Perhatikan gambar dibawah, terdapat sebuah koneksi yang sudah kita buat.



- Berikutnya klik saja next – yes – nex .. kemudian ceklis tables seperti gambar dibawah.



- Finish, dan silakan jalankan aplikasi anda ,jika berhasil akan seperti gambar berikut.

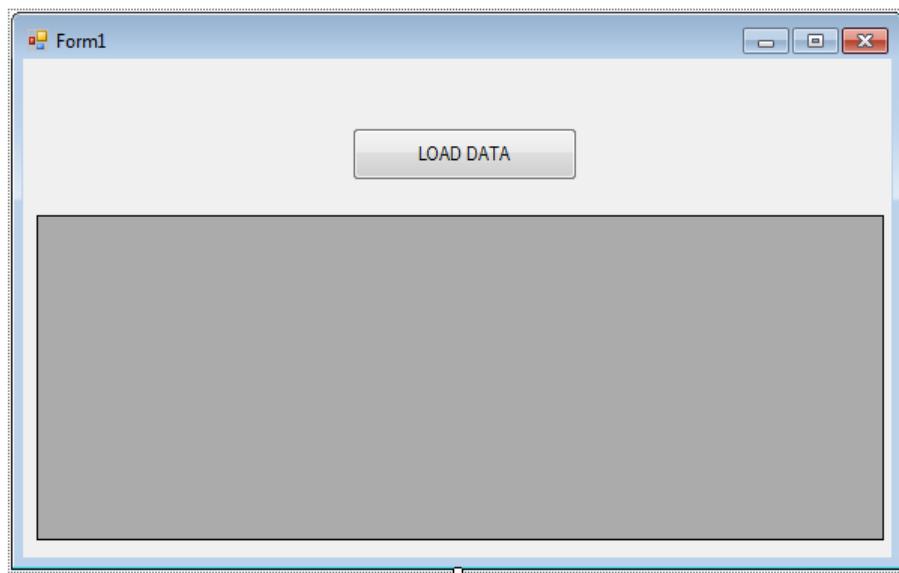
|   | tahun_terbit | judul_buku  | penulis     | penerbit |
|---|--------------|-------------|-------------|----------|
|   | 13/04/2014   | Nasi Uduk   | wawan       | ceo      |
|   | 13/04/2014   | Laskar 2001 | Ali         | ceo      |
|   | 22/06/1905   | CCTV        | Andi M.S    | idie     |
|   | 23/06/1905   | C# Lanjutan | Muhamad Ali | Indie    |
|   | 27/06/1905   | kwitansi    | sari        | -        |
|   | 07/04/2014   | mega        | Kuliner     | indie    |
| * |              |             |             |          |

## BAB VII BEREKSPLORASI DENGAN DATAGRIDVIEW

Pada DataGridView kita tidak hanya bisa menampilkan data dari database, akan tetapi kita juga dapat mengeksplorasi lebih jauh seperti menambahkan komponen button,checkbox dan lain sebagainya. Pada bab ini kita akan belajar bagaimana kita dapat lebih jauh lagi mengeksplorasi datagridview tersebut. Untuk meningkatkan skill kita tentang apa yang akan dibahas mari kita praktikan dengan mengikuti pembahasan demi pembahasan dalam bab ini.

### 7.1 Menambahkan Komponen Button Pada Data GridView

- Silahkan buat projek baru dan beri nama dengan **addbutton\_togrid**, kemudian desain tampilan seperti gambar form berikut.



- Kemudian Double klik pada button1 dan masukan script berikut.

```
private void button1_Click(object sender, EventArgs e)
{
    dataGridView1.ColumnCount = 4;
    dataGridView1.Columns[0].Name = "Nim";
    dataGridView1.Columns[1].Name = "Nama";
    dataGridView1.Columns[2].Name = "Kelas";
    dataGridView1.Columns[3].Name = "IPK";

    string[] baris = new string[] { "1211109",
"Muhamad Ali", "13.6B.11", "3.81" };
    dataGridView1.Rows.Add(baris);
    baris = new string[] { "13000011", "Aldefinu",
"12.6AA.3", "3.45" };
    dataGridView1.Rows.Add(baris);
```

```

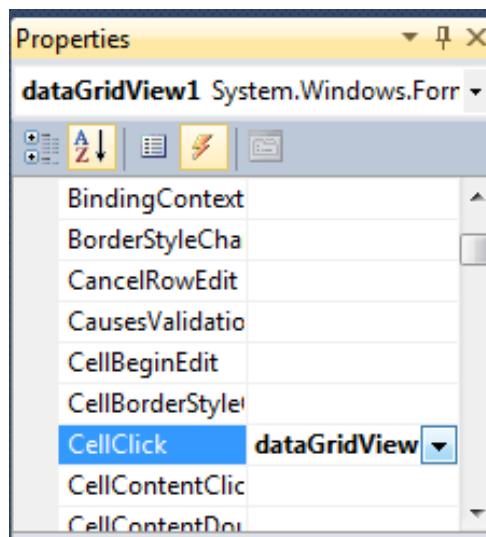
        baris = new string[] { "1234533", "Dede",
"12.34.BB", "2.98" };
        dataGridView1.Rows.Add(baris);
        baris = new string[] { "34567323", "Wawan",
"12.6b.33", "3.00" };
        dataGridView1.Rows.Add(baris);

        DataGridViewButtonColumn button = new
DataGridViewButtonColumn();
        dataGridView1.Columns.Add(button);
        button.HeaderText = "Klik Pesan";
        button.Text = "Coba klik";
        button.Name = "btn";
        button.UseColumnTextForButtonValue = true;

    }

```

- Berikutnya berikan event cell klik pada gridview seperti gambar berikut.



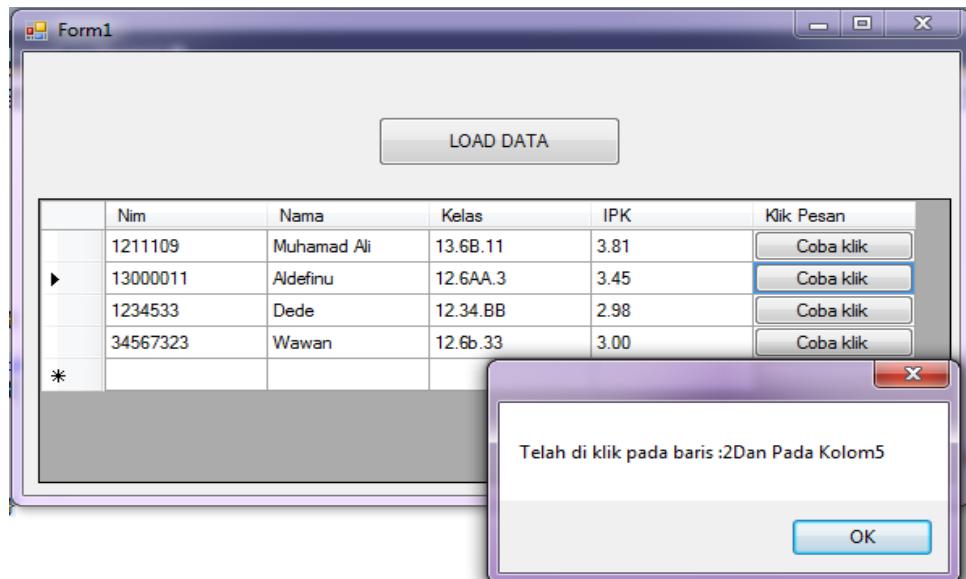
- Langkah berikutnya silakan masukan atau ketikan listing program berikut.

```

private void dataGridView1_CellClick(object sender,
DataGridViewCellEventArgs e)
{
    if (e.ColumnIndex == 4)
    {
        MessageBox.Show("Telah di klik pada baris :"
+ (e.RowIndex + 1) + "Dan Pada Kolom" + (e.ColumnIndex + 1)
+ "");
    }
}

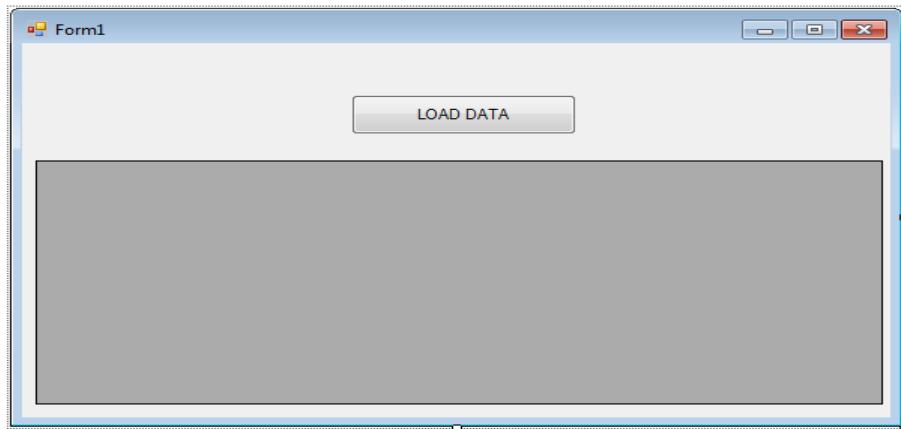
```

- Sampai disini aplikasi sudah berhasil dibuat silakan jalankan program, jika berhasil maka aplikasi akan terlihat seperti gambar di bawah ini.



## 7.2 Menambahkan Komponen Checkbox pada DataGridView

- Silahkan buatlah projek baru dan beri nama dengan **addcheckbox\_togrid**, kemudia desain tampilan sama seperti desain tampilan pada projek **addbutton\_togrid**.



- Jika tampilan sudah dibuat sama dengan projek sebelumnya ,silahkan double klik pada button1 dan ketikan listing berikut.

```
private void button1_Click(object sender, EventArgs e)
{
    dataGridView1.ColumnCount = 4;
    dataGridView1.Columns[0].Name = "Nim";
    dataGridView1.Columns[1].Name = "Nama";
    dataGridView1.Columns[2].Name = "Kelas";
    dataGridView1.Columns[3].Name = "IPK";
```

```

        string[] baris = new string[]{"112345", "Muhamad
Ali", "13.6b.11", "4.00"};
        dataGridView1.Rows.Add(baris);
        baris = new string[] { "3635453", "Aldefinu",
"13.6B.11", "1.60" };
        dataGridView1.Rows.Add(baris);
        baris = new string[] { "3635453", "Alfian",
"13.6B.11", "1.60" };
        dataGridView1.Rows.Add(baris);

        DataGridViewCheckBoxColumn cek = new
DataGridViewCheckBoxColumn();
        dataGridView1.Columns.Add(cek);
        cek.HeaderText = "Sava IPK";
        cek.Name="cek";
        dataGridView1.Rows[0].Cells[4].Value = false;

    }

```

- Kemudian berikan juga event cellclick seperti pada projek sebelumnya , dan masukan script berikut.

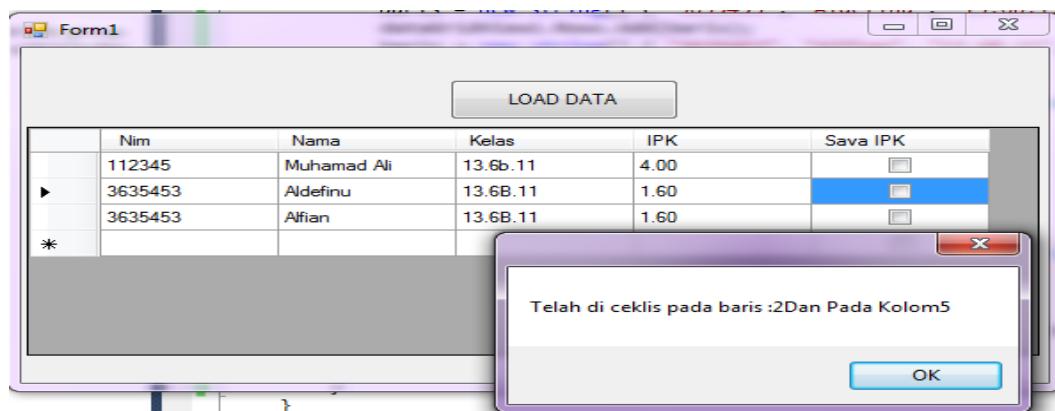
```

private void dataGridView1_CellClick(object sender,
DataGridViewCellEventArgs e)
{
    if (e.ColumnIndex == 4)
    {
        MessageBox.Show("Telah di ceklis pada
baris :" + (e.RowIndex + 1) + "Dan Pada Kolom" +
(e.ColumnIndex + 1) + "");
    }

}

```

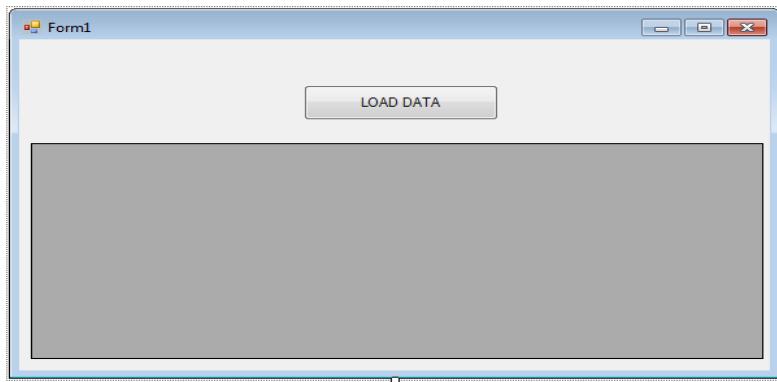
- Silakan Jalankan program untuk melihat hasilnya,jika berhasil maka akan terlihat seperti gambar dibawah.



## 7.5 Menambahkan komponen ComboBox ke datagridview di C#

Listing yang akan digunakan pada pembahasan ini kurang lebih sama seperti pada projek sebelumnya ,akan tetapi hanya ada sedikit perbedaan namun ini tidak terlalu sulit untuk dipahami jika anda mengikuti langkah – langkah sebagai berikut:

- Buatlah projek baru dan berikan nama dengan **addcombo\_togrid**. Desain tampilan sama seperti projek sebelumnya :



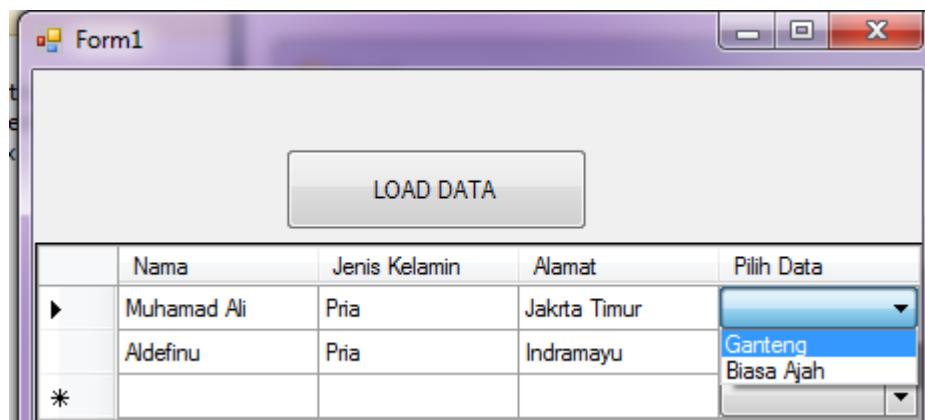
- ,kemudian double klik pada button dan ketikan listing berikut.

```
private void button1_Click(object sender, EventArgs e)
{
    dataGridView1.ColumnCount = 3;
    dataGridView1.Columns[0].Name = "Nama";
    dataGridView1.Columns[1].Name = "Jenis Kelamin";
    dataGridView1.Columns[2].Name = "Alamat";

    string[] baris = new string[] { "Muhamad Ali",
    "Pria", "Jakarta Timur" };
    dataGridView1.Rows.Add(baris);
    baris = new string[] { "Aldefinu", "Pria",
    "Indramayu" };
    dataGridView1.Rows.Add(baris);

    DataGridViewComboBoxColumn cmbo = new
    DataGridViewComboBoxColumn();
    cmbo.HeaderText = "Pilih Data";
    cmbo.Name = "cmb";
    cmbo.MaxDropDownItems = 2;
    cmbo.Items.Add("Ganteng");
    cmbo.Items.Add("Biasa Ajah");
    dataGridView1.Columns.Add(cmbo);
}
```

- Berikutnya jalankan program untuk melihat hasilnya.



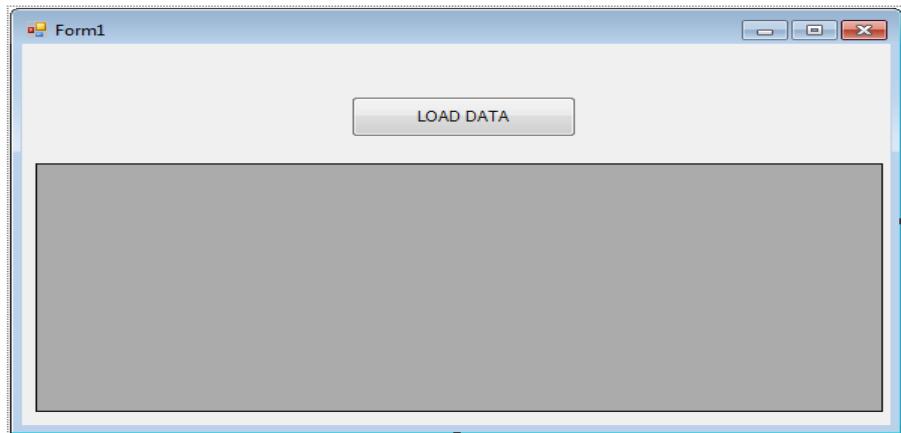
## 7.6 Menambahkan Image pada DataGridview

Sebenarnya untuk menambahkan image pada datagridview cukup menambahkan listing seperti berikut saja :

```
DataGridViewImageColumn foto = new
DataGridViewImageColumn();
Image image = Image.FromFile("Lokasi foto
anda");
foto.Image = image;
dataGridView1.Columns.Add(foto);
foto.HeaderText = "Image";
foto.Name = "Nama Foto";
```

Tetapi agar lebih memahami lagi mari ikuti latihan berikut :

- Buat Projek baru dan beri nama **addimage\_togrid** dan desain tampilan masih sama seperti pada pembahasan sebelumnya. Jangan Lupa taruh imgame dengan nama “contoh” di lokasi direktori bin/debug projek anda.



- Kemudian double klik pada button1 dan ketikan listing berikut.

```
private void button1_Click(object sender, EventArgs e)
{
    dataGridView1.ColumnCount = 3;
    dataGridView1.Columns[0].Name = "Nama";
    dataGridView1.Columns[1].Name = "Jenis Kelamin";
    dataGridView1.Columns[2].Name = "Alamat";

    string[] baris = new string[] { "Muhamad Ali",
    "Pria", "Indramayu" };
    dataGridView1.Rows.Add(baris);
    baris = new string[] { "Ali", "Pria",
    "Indramayu" };
    dataGridView1.Rows.Add(baris);
    baris = new string[] { "Aldefinu", "Pria",
    "Singapur" };
    dataGridView1.Rows.Add(baris);

    DataGridViewImageColumn foto = new
    DataGridViewImageColumn();
    Image image = Image.FromFile("contoh.png");
    foto.Image = image;
    dataGridView1.Columns.Add(foto);
    foto.HeaderText = "Foto";
    foto.Name = "identitas";

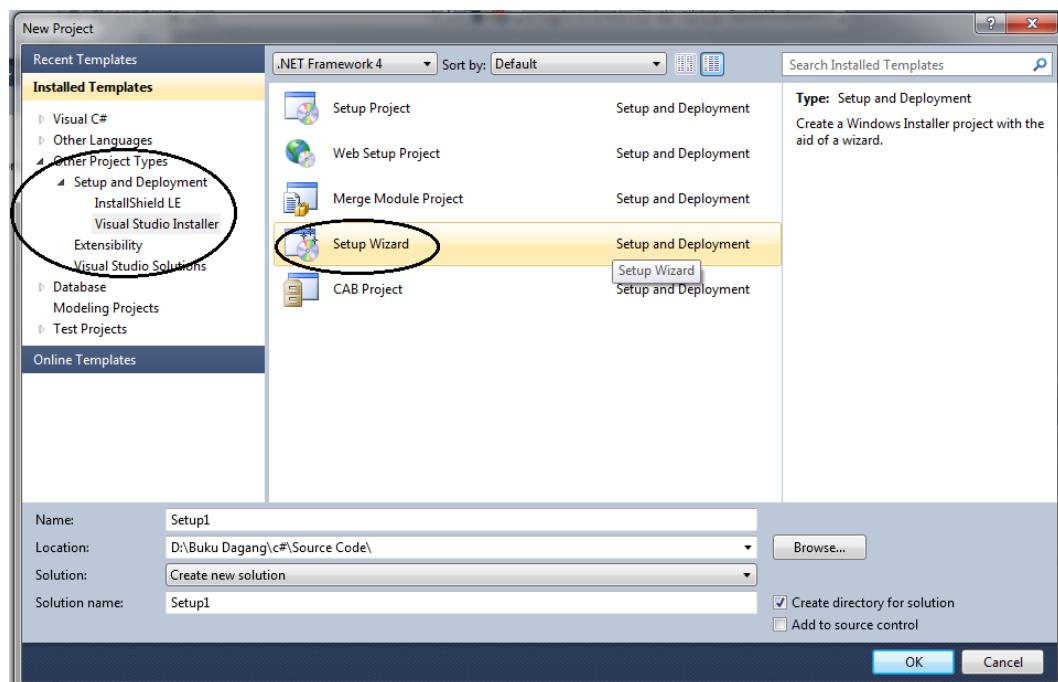
}
```

- Silakan jalankan program untuk melihat hasilnya.

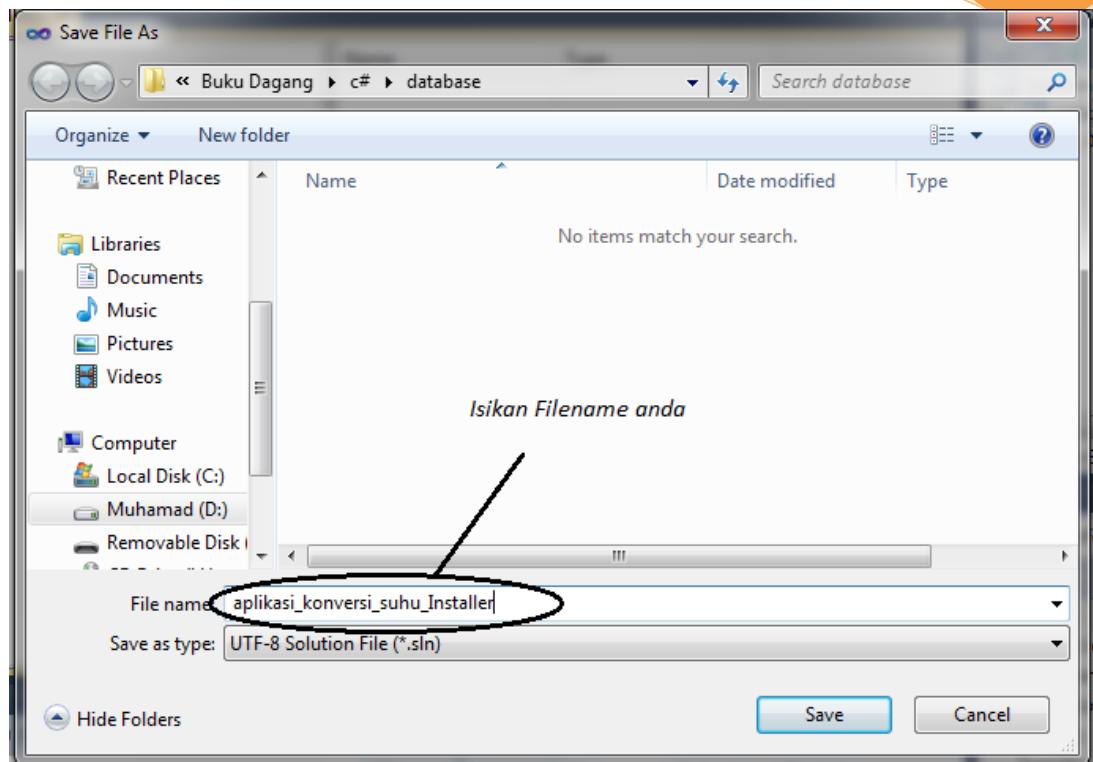
## BAB VIII MEMBUAT FILE SETUP DENGAN VISUAL STUDIO

Dalam sebuah aplikasi biasanya tidaklah lepas dari sebuah Install – menginstall aplikasi yang biasa disebut installer atau Setup. Ini diperlukan agar dapat mempermudah pengguna dalam menggunakan aplikasi yang sudah kita buat sehingga pengguna tidak perlu bersusah payah mencari tahu bagaimana menjalankan aplikasi C# yang sudah kita buat atau mencari file yang bisa dijalankan langsung yang terletak di folder bin/debug. Untuk dapat membuatnya tidaklah susah jika anda menggunakan Microsoft Visual Studio karena hanya dengan beberapa langkah anda dapat membuatnya , sehingga kini aplikasi anda sudah bisa di install dikomputer mana saja. Berikut beberapa langkah membuat aplikasi **Setup Wizard**.

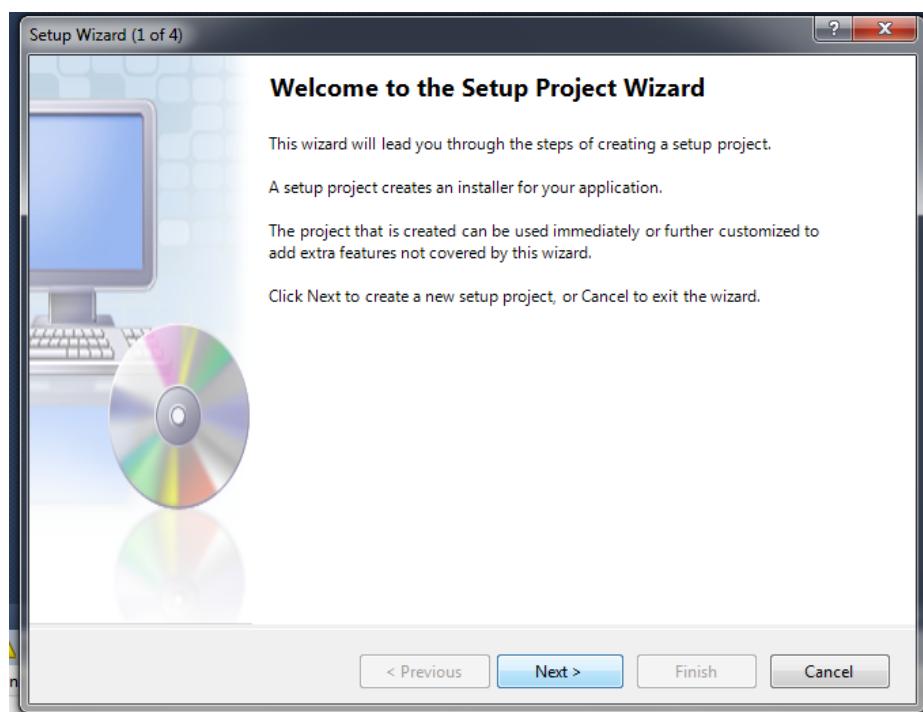
- Pertama – tama silakan buat projek baru ,kemudian pilih **Other Project Types** -> **Setup and Deployment** -> **Visual studio Installer** -> **Setup Wizard**. Kemudian berinama aplikasi anda sesuka hati anda. Karena disini penulis menggunakan **aplikasi\_konversi\_suhu** yang akan dibuatkan installer.
- maka menulis memberikan nama **aplikasi\_konversi\_suhu\_Installer** pada projek ini.

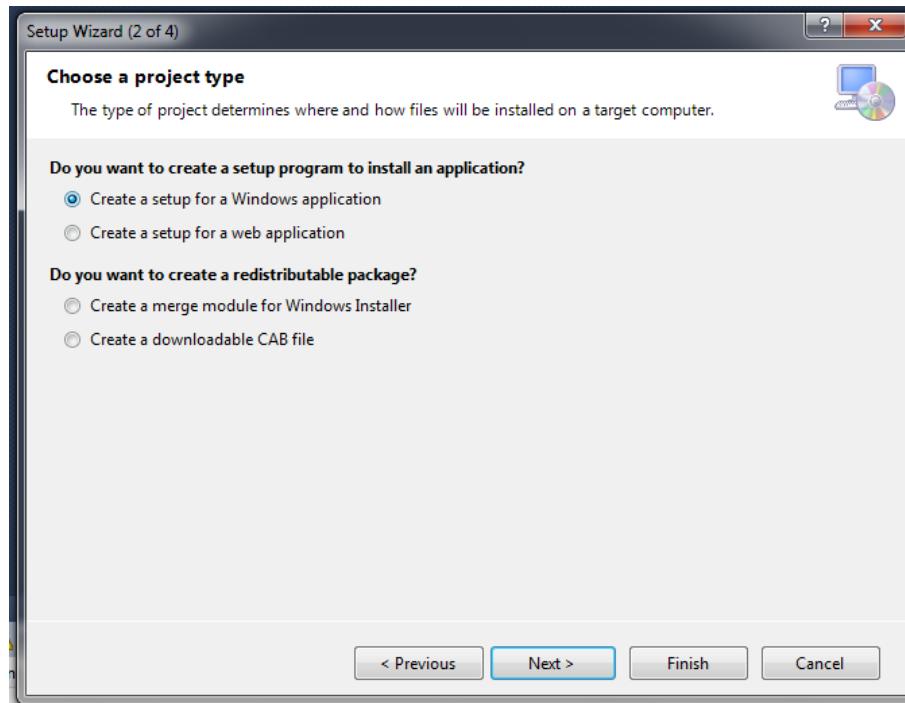


- Jika sudah klik saja OK. Langkah berikutnya ikuti saja gambar dibawah.

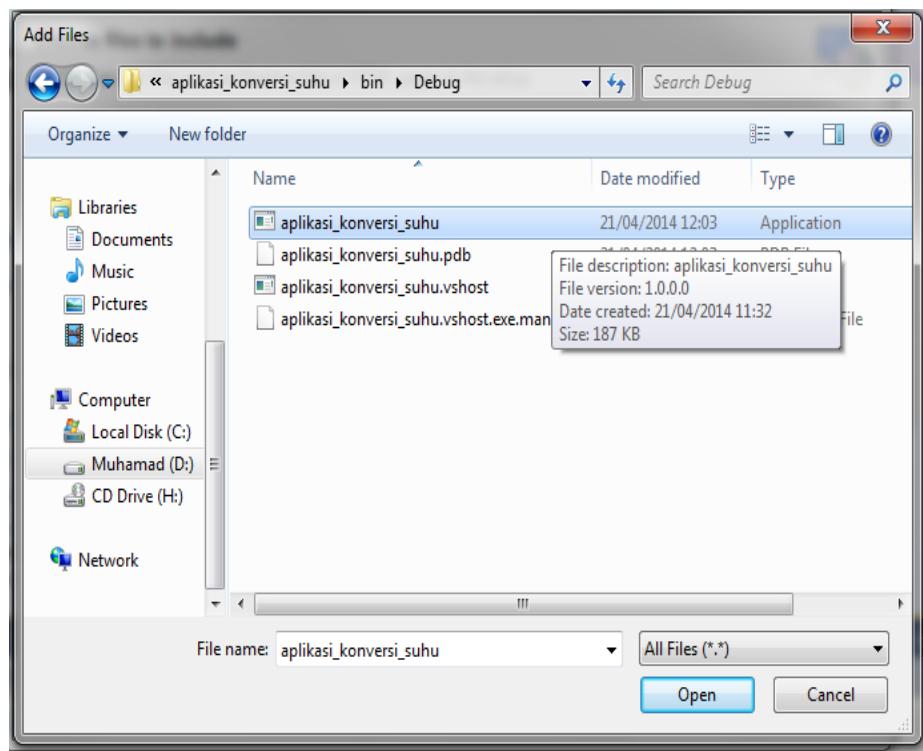


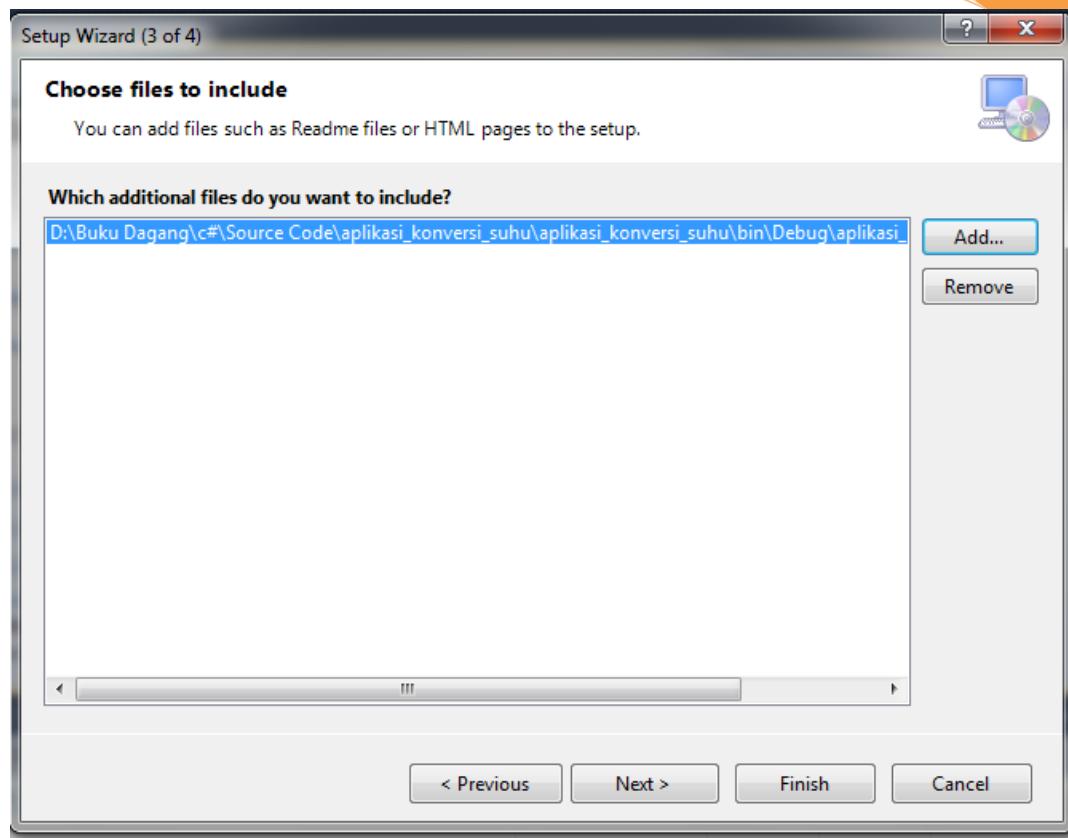
- Klik next -> next lagi



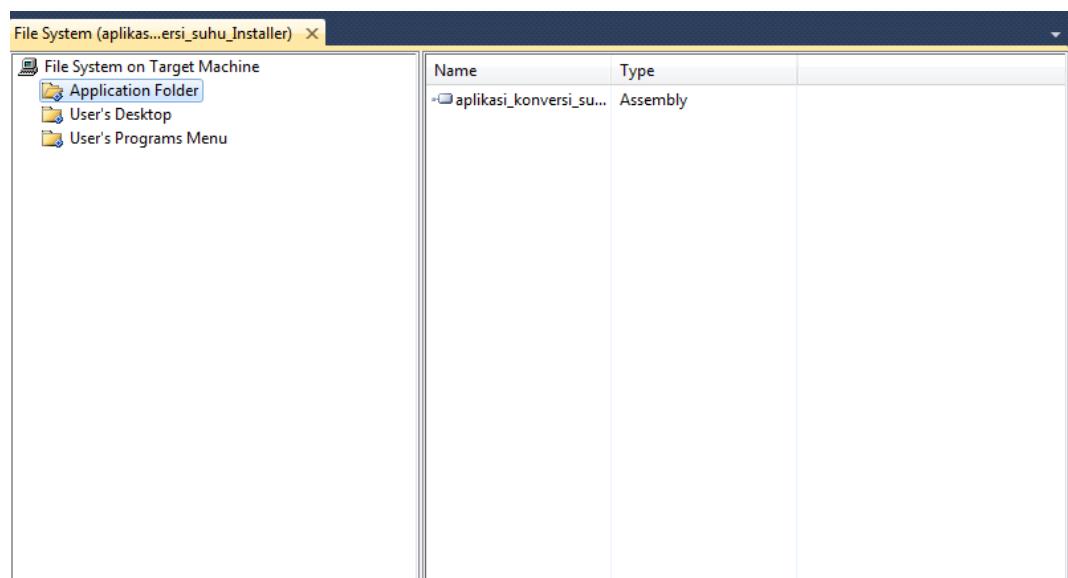


- Klik Add, lalu masukan file – file aplikasi anda

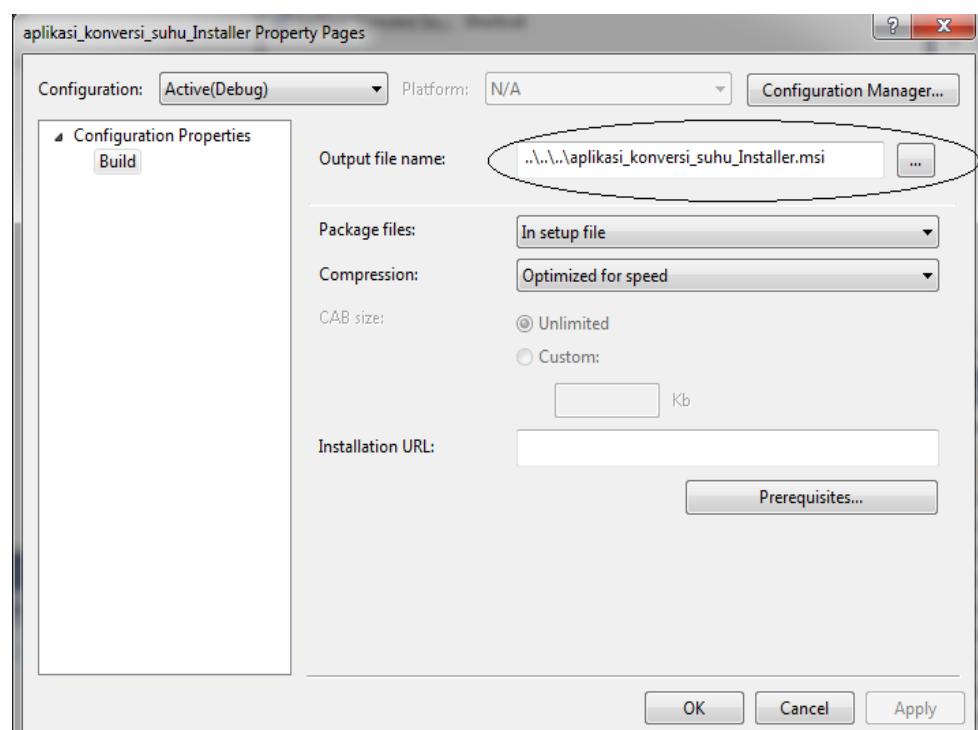
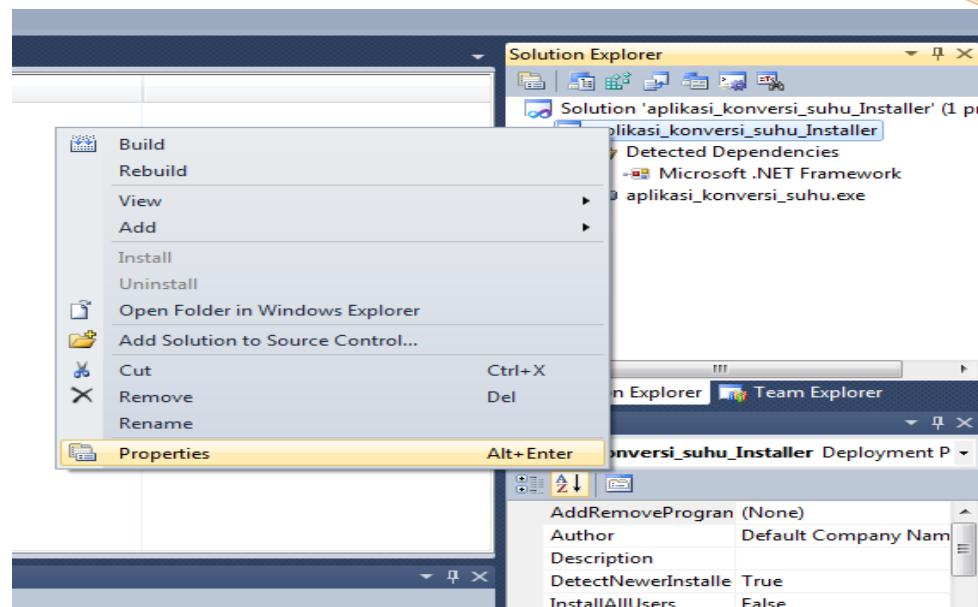




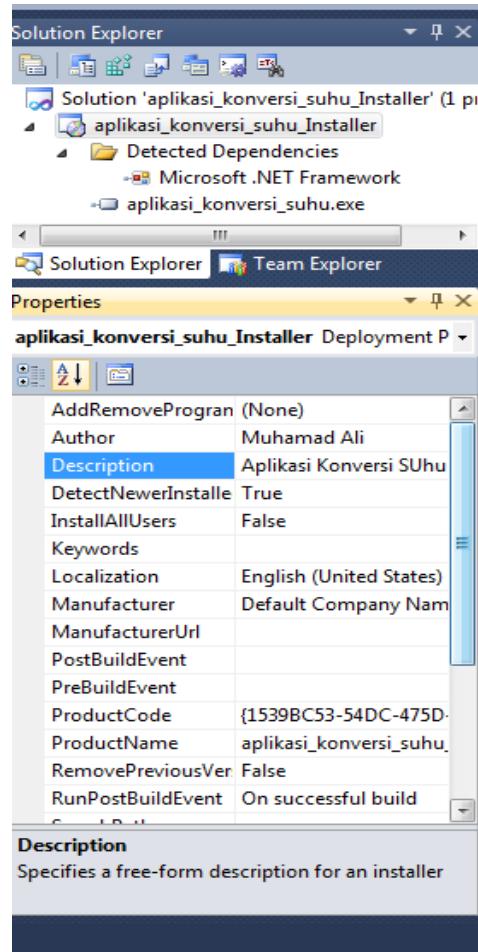
- Klik Next -> Finish
- File yang tadi kita import ada di application Folder



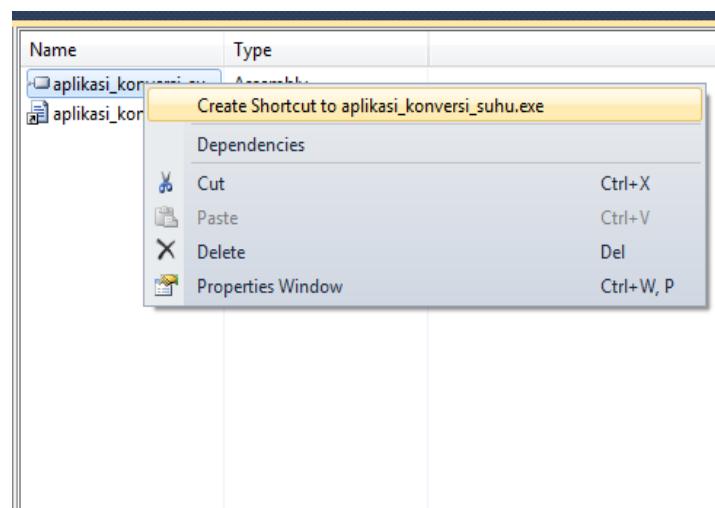
- Kemudian atur lokasi fle output setelah di build untuk file setup kita, lihat gambar dibawah

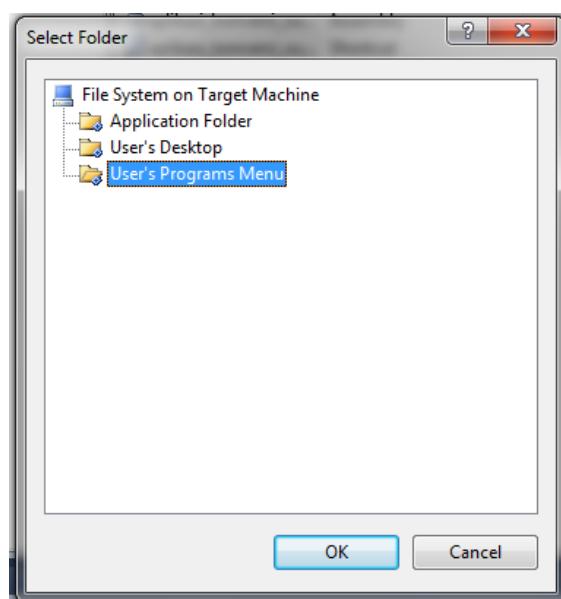
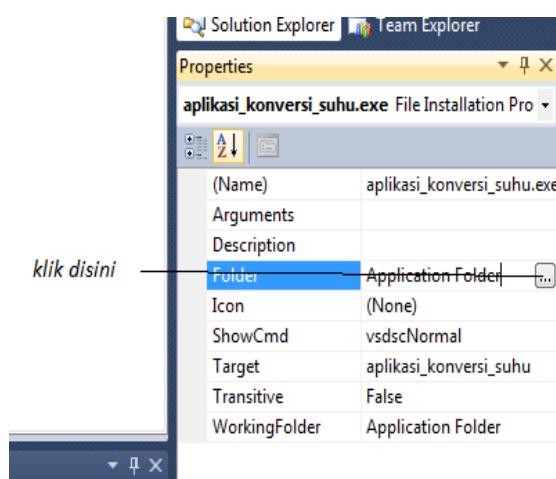
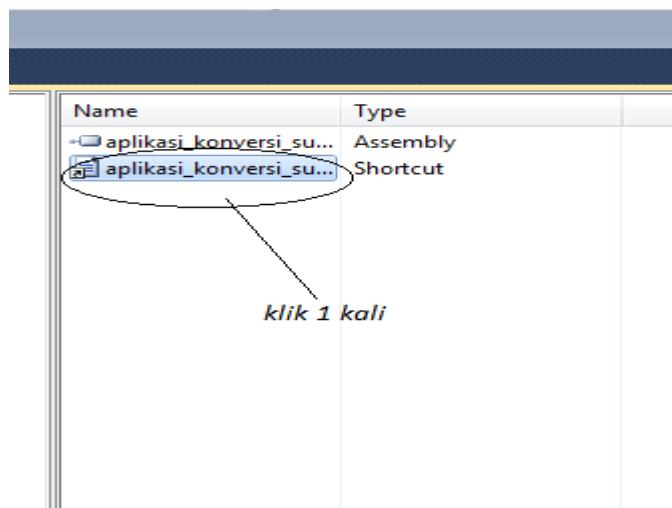


- Anda Juga bisa mengatur untuk properties autor dan lain sebagainya. Lihat gambar.



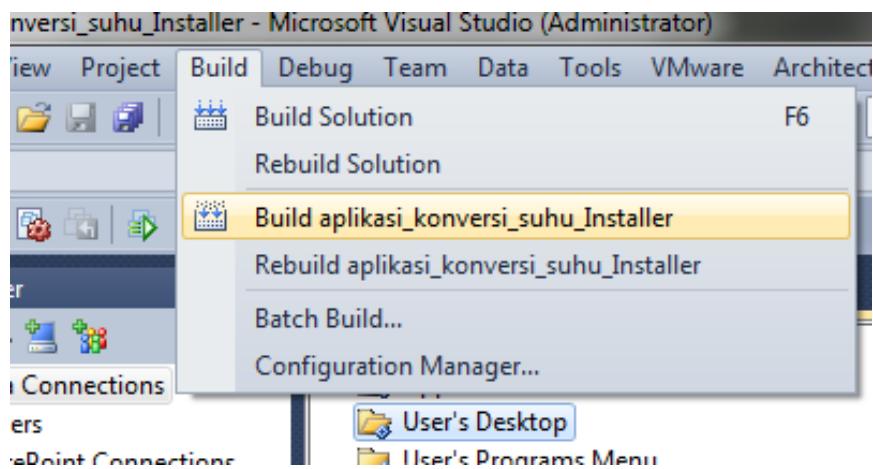
- Untuk membuat shortcut di startmenu



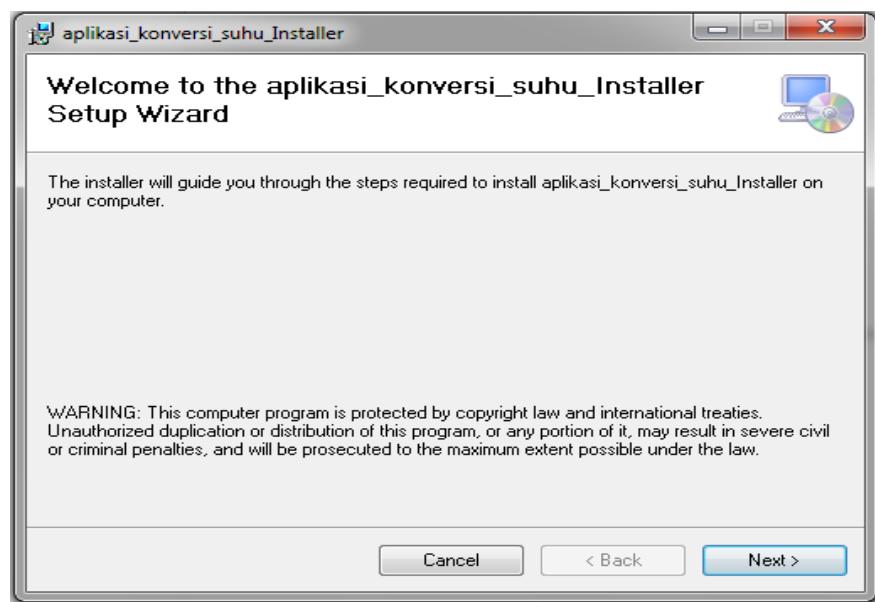
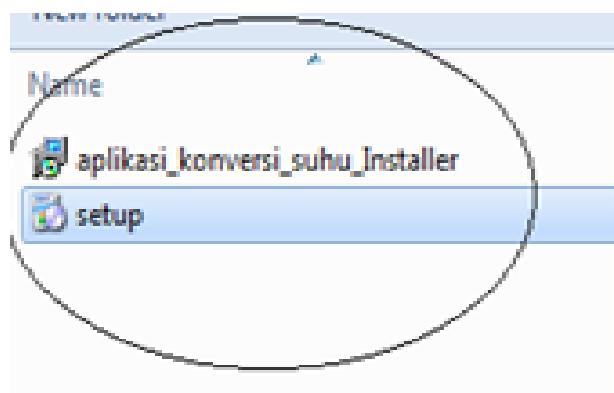


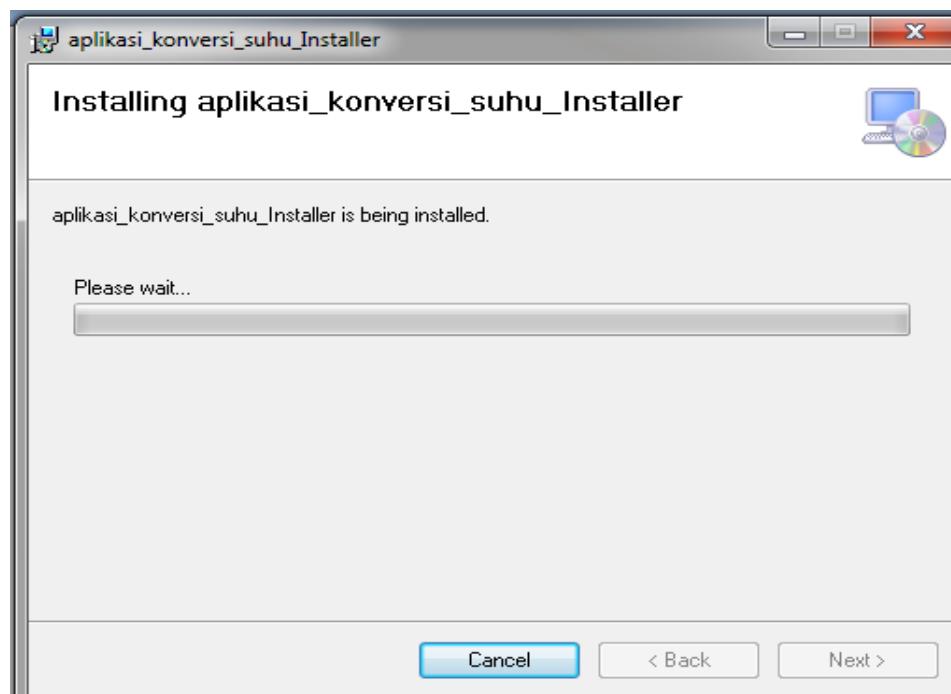
- Sementara Untuk membuat shortcut di destop silahkan lakukan langkah yang sama.

- Kemudian built program seperti gambar dibawah.



- Sampai disini aplikasi installer sudah berhasil dibuat. Silahkan jalankan Programm untuk melihat hasilnya. Silahkan double klik untuk menginstall aplikasi.





## DAFTAR PUSTAKA

[http://id.wikipedia.org/wiki/C\\_sharp](http://id.wikipedia.org/wiki/C_sharp)

[http://en.wikipedia.org/wiki/Microsoft\\_Visual\\_Studio](http://en.wikipedia.org/wiki/Microsoft_Visual_Studio)

<http://eirenekw.blogspot.com/2011/04/pengenalan-microsoft-visual-studio-2010.html>

<http://zetcode.com/lang/csharp/oopi/>

<http://spacenotalone.wordpress.com/artikel/oop-polimorfismeoverloadingoverriding/>

<http://anak-banyuwangi.blogspot.com/2012/04/polimorfisme.html>

<http://doubledoortikom.wordpress.com/2012/05/11/oop2-bab-iii-polymorphisms-part-1-overriding/>

<http://csharp.net-informations.com/gui/cs-textbox.htm>

<http://dev.mysql.com/doc/refman/4.1/en/what-is-mysql.html>

<http://www.qwhatis.com/what-is-xampp/>

[http://en.wikipedia.org/wiki/Microsoft\\_Access](http://en.wikipedia.org/wiki/Microsoft_Access)

<http://www.opengatesw.net/ms-access-tutorials/What-Is-Microsoft-Access-Used-For.htm>

<http://csharp.net-informations.com/datagridview/csharp-datagridview-button.htm>

<http://csharp.net-informations.com/datagridview/csharp-datagridview-checkbox.htm>

<http://csharp.net-informations.com/datagridview/csharp-datagridview-combobox.htm>

<http://csharp.net-informations.com/datagridview/csharp-datagridview-image.htm>

Wahaana Komputer ,2012, VISUAL BASIC 2010 Programming,: C.V ANDI OFFSET(Penerbit Andi).Yogyakarta.

## LAMPIRAN

## Tentang Penulis



Muhamad Ali , merupakan mahasiswa tingkat akhir jurusan Tehnik Komputer di AMIK BINA SARANA INFORMATIKA serta merangkap Jurusan Tehnik Informatika di kampus STMIK NUSA MANDIRI. Penulis juga pernah menjadi pengajar private freelance pada bidang pemograman php selain kegiatanya sehari – hari sebagai mahasiswa tingkat akhir , penulis juga aktif menulis tutorial – tutorial computer baik melalui situs internet dalam bentuk ebook ataupun tutorial dalam bentuk artikel di website. Selain itu juga akhir – akhir ini penulis mulai merambah ke dunia tulis menulis dalam bentuk buku. Penulis dapat dihubungi melalui email [ali.muhamad20@gmail.com](mailto:ali.muhamad20@gmail.com) atau melalui twiter @muhamadaldefinu.