# **BAB I**

# **MENGENAL .NET FRAMEWORK**

## Penjelasan Tentang .Net Framework

Net Framework adalah sebuah [perangkat lunak](https://id.wikipedia.org/wiki/Perangkat_lunak) [kerangka kerja](https://id.wikipedia.org/wiki/Kerangka_kerja) yang berjalan utamanya pada [sistem operasi](https://id.wikipedia.org/wiki/Sistem_operasi) [Microsoft Windows](https://id.wikipedia.org/wiki/Microsoft_Windows), Kerangka kerja ini menyediakan sejumlah besar pustaka pemrograman komputer dan mendukung beberapa bahasa pemrograman sehingga memungkinkan bahasa-bahasa tersebut berfungsi satu dengan lain dalam pengembangan sistem serta mampu meminimalkan proses dalam pembangunan suatu aplikasi atau suatu sistem informasi dalam linkungan terdistribusi internet.

Selain itu tujuan dari .NET Framework adalah:

* Menyediakan lingkungan pemograman berorientasi objek atau Object Oriented Programming (OOP) yang konsisten meskipun kode objek disimpan dan dijalankan secara lokal tetapi dapat disebarkan melalui internet dan di jalankan secara remote ( di jalankan dari suatu tempat).
* Menyediakan lingkungan untuk menjalankan suatu kode yang dapat mengeliminasi masalah performa dari lingkungan scripted (halaman).
* Menyediakan lingkungan untuk menjalankan suatu kode yang menjamin keamanan saat kode di jalankan, termasuk kode yang di buat oleh pihak yang tidak di ketahui/ pihak ketiga yang setengah dipercaya.

.NET Framework terdiri dari dua buah bagian utama, yaitu Common Language Runtime (CLR) dan Base Class Library (BCL).

1. **Common Language Runtime (CLR)** adalah pondasi utama (bahasa umum) dari .NET Framework ,yang menjalankan aplikasi .NET Framework menyediakan sejumlah layanan pada berbagai hal seperti :

* Pengaturan memory,
* Mengelola kode (melakukan eksekusi kode)
* Melakukan verifikasi terhadap keamanan kode
* Menentukan hak akses dari kode, melakukan kompilasi kode, dan berbagai layanan system lainnya.

1. **Base Case Library (BCL)** bersifat berorientasi terhadap objek yang akan menyediakan jenis dari fungsi-fungsi pengaturan kode.

Seperti telah disebutkan sebelumnya bahwa .Net Framework memilki beberapa bahasa pemograman di dalamnya, termasuk VB.Net, maka struktur .Net Framewok paling utama dibangun oleh bahasa pemograman .Net. Dapat rekan-rekan lihat di bawah ini untuk struktur dasar .Net Framework ;



Common Language SPesification (CLS)

Data ADO.Net & XML

Base Class Library (BCL)

Common Language Runtime (CLR)

Service Web Xml

VB.Net

ASP.Net

C#.Net

Form-form Web

J#.Net

C++.Net

Forms Windows

## .Net Enterprise Server

Sebelum kita memulai menjalankan Visual Basic , alangkah baiknya kita mengenal terlebih dahulu secara singkat Microsoft visual studio yang akan kita gunakan nanti.

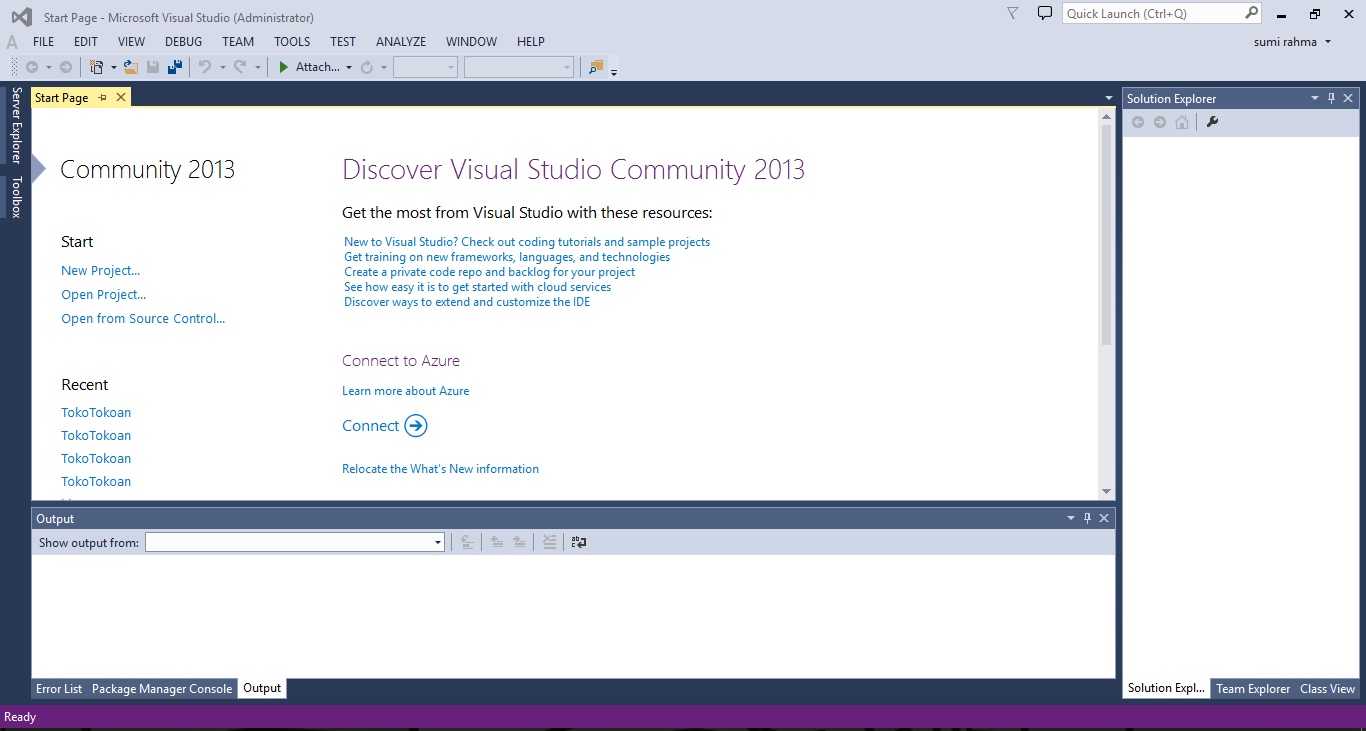
Microsoft Visual Studio  merupakan sebuah Perangkat Lunak lengkap yang dapat digunakan untuk melakukan pengembangan aplikasi, baik itu aplikasi bisnis, aplikasi personal, ataupun komponen aplikasinya, dalam bentuk aplikasi console, aplikasi Windows, ataupun aplikasi Web.

Visual Studio mencakup Kompiler (penyusun), SDK, Integrated Development Environment (IDE), dan dokumentasi (umumnya berupa MSDN Library). Kompiler yang dimasukkan ke dalam paket Visual Studio antara lain Visual C++, Visual C#, Visual Basic, Visual Basic .NET, Visual InterDev,Visual J++, Visual J#, Visual FoxPro, dan Visual SourceSafe.

Microsoft Visual Studio dapat digunakan untuk mengembangkan aplikasi dalam *native code* (dalam bentuk bahasa mesin yang berjalan di atas Windows) ataupun *managed code* (dalam bentuk Microsoft Intermediate Language di atas .NET Framework). Selain itu, Visual Studio juga dapat digunakan untuk mengembangkan aplikasi Silverlight, aplikasi Windows Mobile (yang berjalan di atas .NET Compact Framework).

1. Langkah pertama Buka Microsoft Visual Studio.

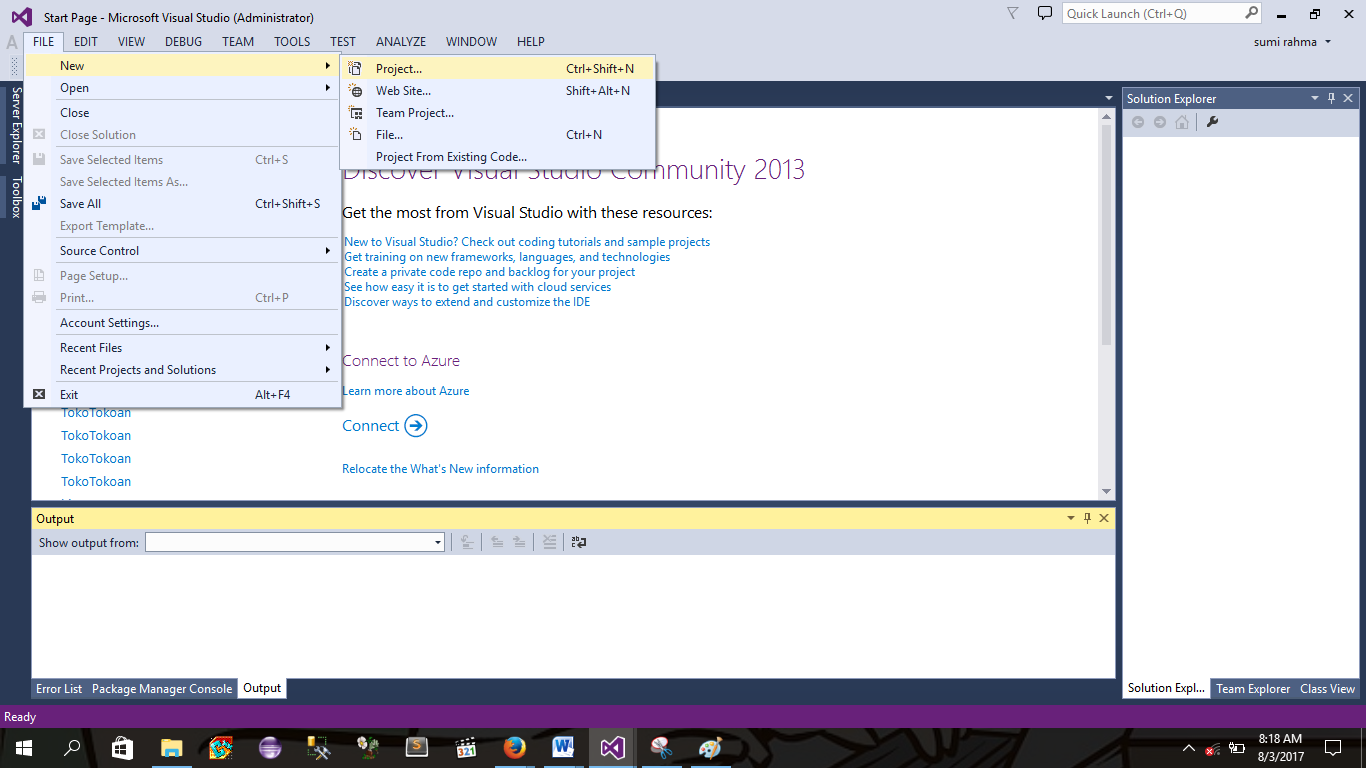
Gambar dibawah merupakan tampilan awal saat kita pertama membuka Microsoft visual studio 2010.



Gambar 1.1

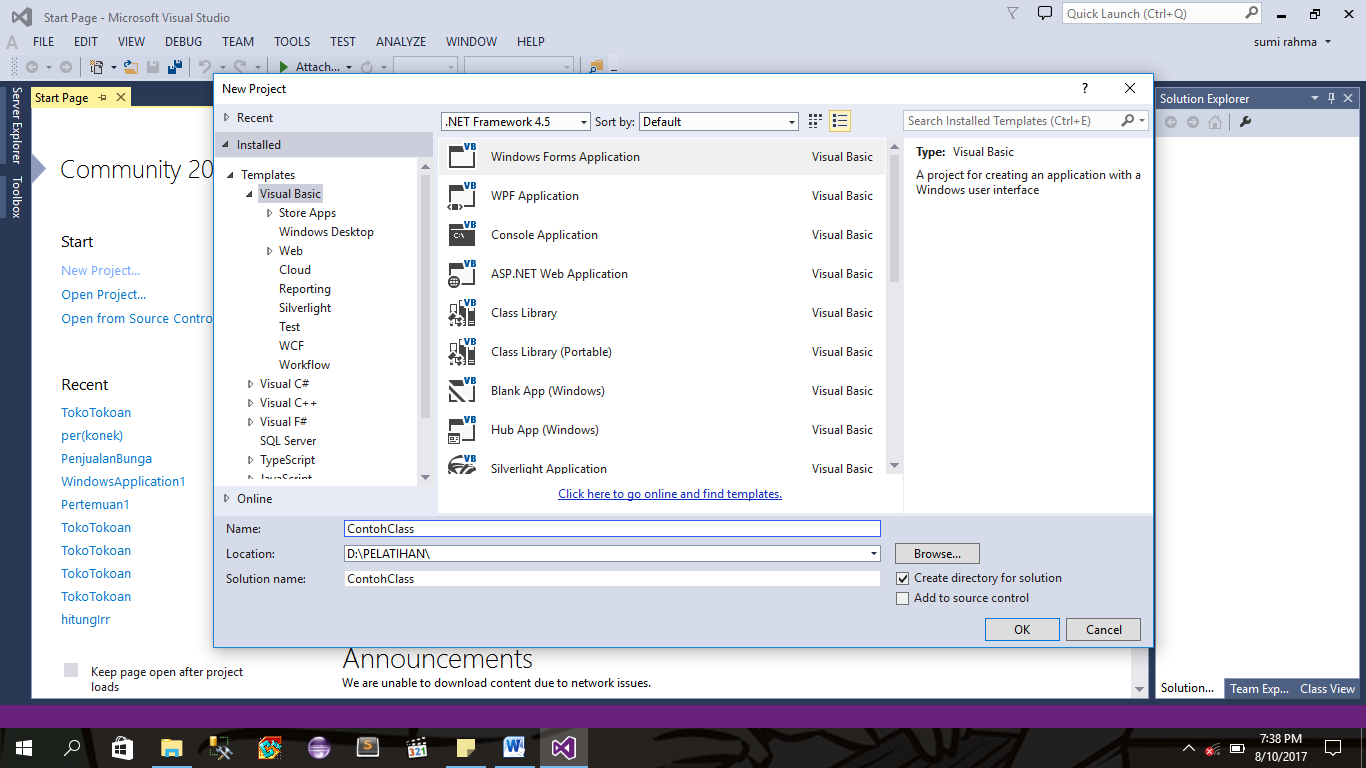
Untuk melihat apa yang terdapat pada Microsoft Visual Studio 2010. Maka kita perlu membuat sebuah solution. Terdapat 2 cara dalam membuat solution baru.

* kita hanya mengklik *new projek* yang berada pada tampilan awal atau kita dapat mengklik **file 🡪 new 🡪 projek**. (dapat kita lihat pada gambar 1.2).



Gambar 1.2

* Selanjutnya akan menampilkan (gambar 04) pilih **Visual Basic 🡪 Windows 🡪 windows Forms Application**. Kemudian ketik nama solution yang akan kita buat pada kolom *name.* dan kolom *location* untuk tempat penyimpanan solution yang akan kita buat. Pada modul ini kita beri nama “ContohClass”.



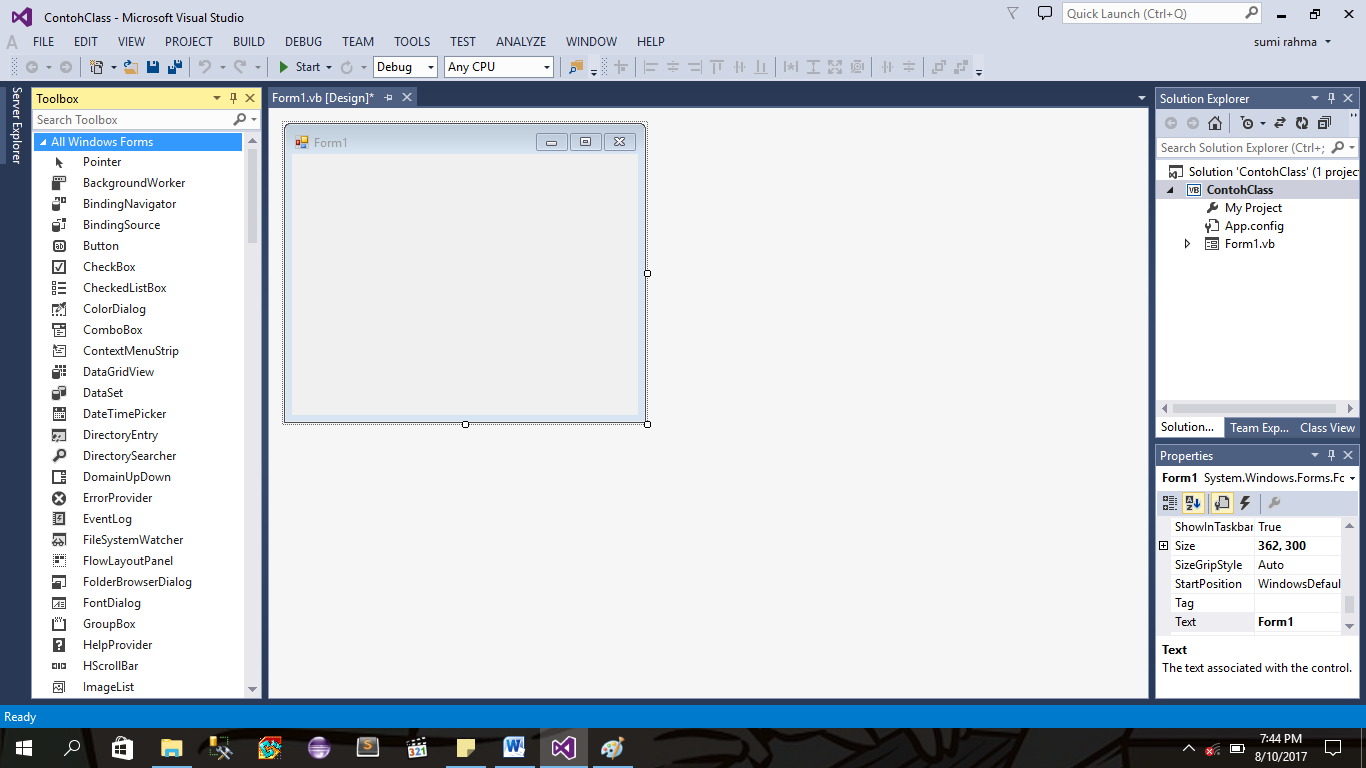
Gambar 1.3

* Lalu klik **OK.**

Maka akan tampil gambar sebagai berikut:

**Toolbar**

**Menu bar**



**Solution Exploler, window property,**

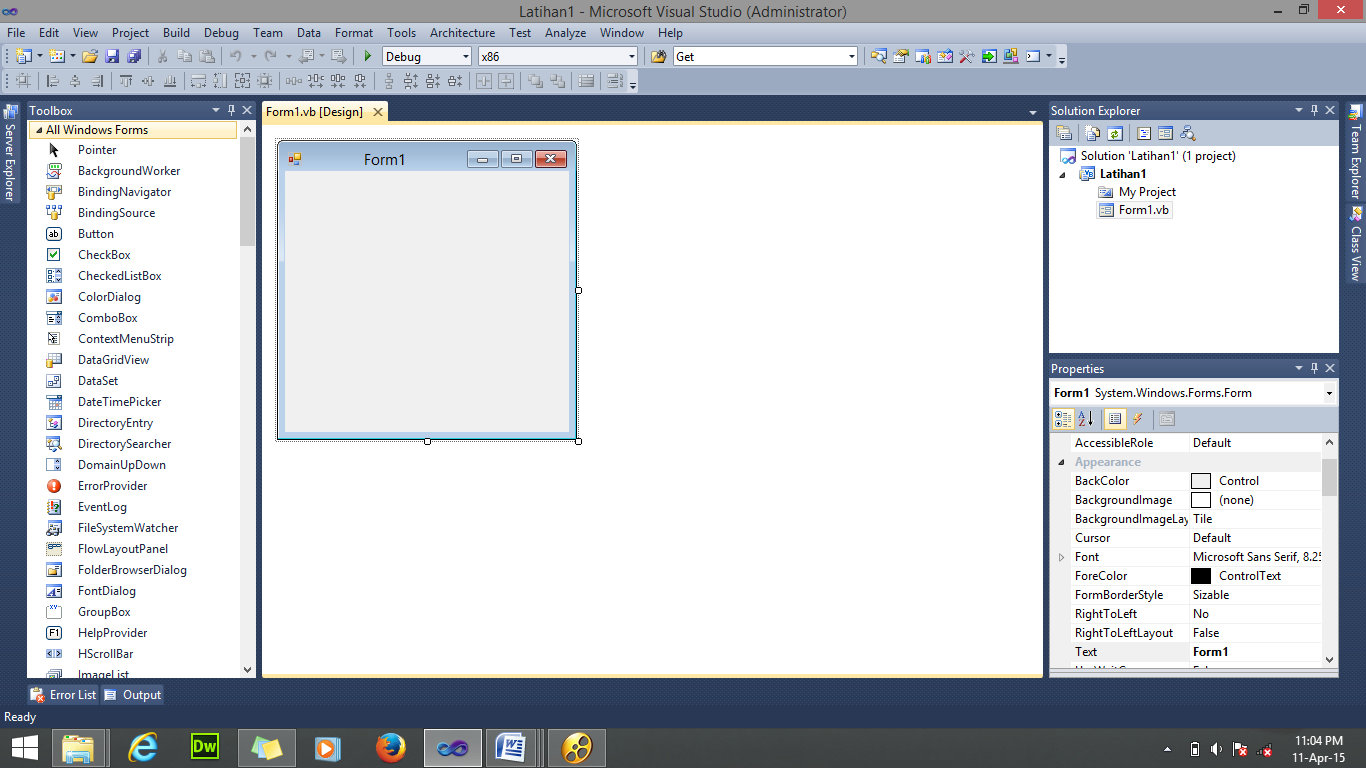
**Toolbox**

**Form Tempat Pembuatan projek**

Gambar 1.4

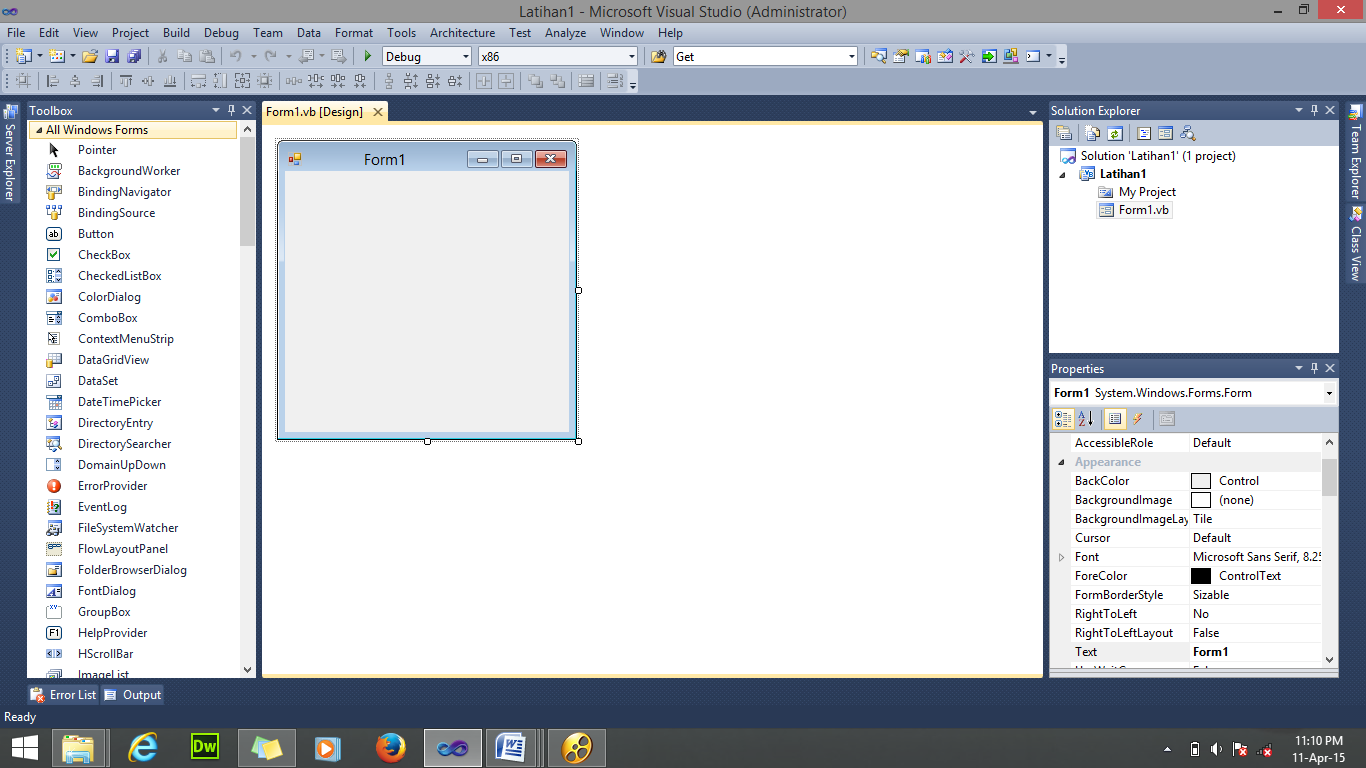
Penjelasan mengenai bagian-bagian yang terdapat pada Gambar 1.4 untuk mempermudah kita dalam pembuatan program aplikasi.

* **MenuBar**

****

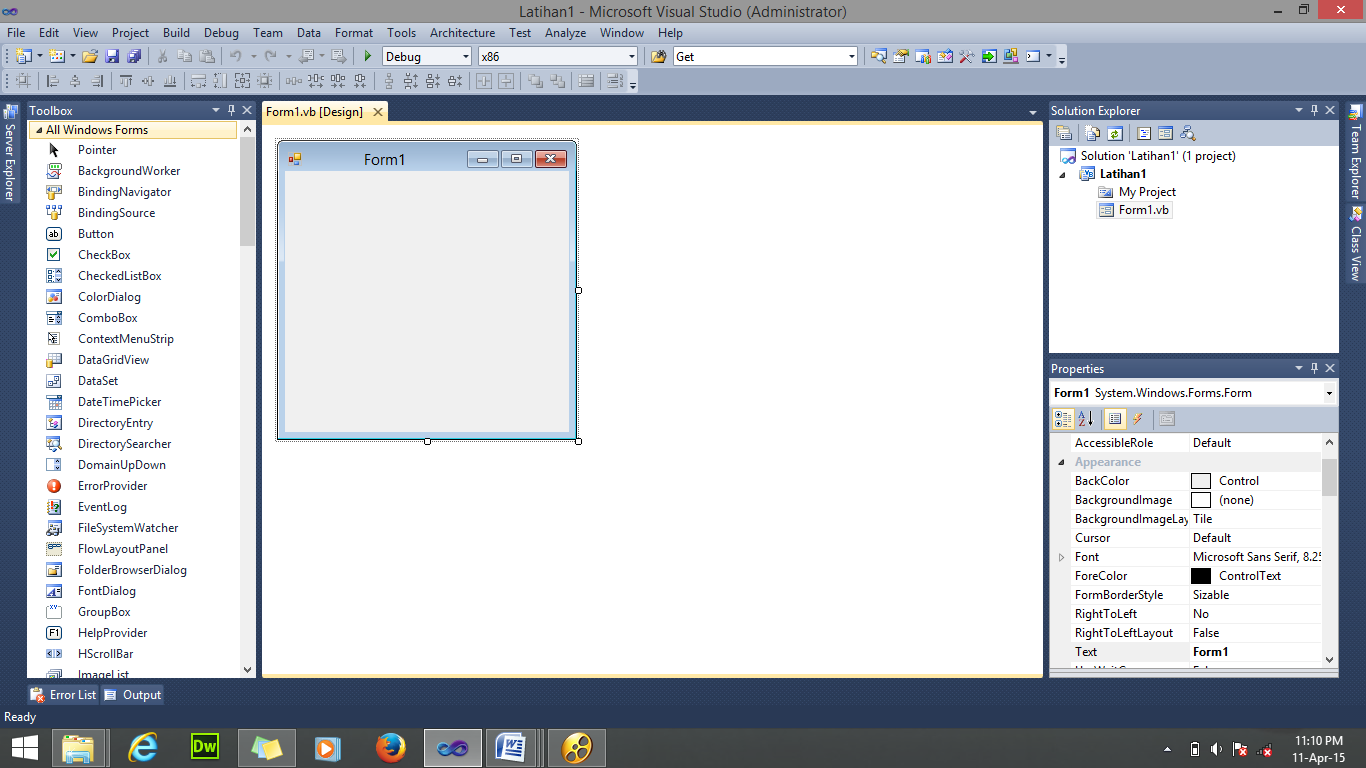
Menubar merupakan kumpulan dari perintah-perintah dalam menjalankan aplikasi yang dikelompokkan berdasarkan kriteria operasinya masing-masing.

* **Toolbar**



Toolbar merupakan tombol/simbol yang mewakili suatu perintah yang terdapat pada didalam menubar untuk mengoperasikan file program. Rekan-rekan dapat menambah ataupun mengurangi toolbar dengan cara klik view pada menubar pilih toolbar.

* **Toolbox**

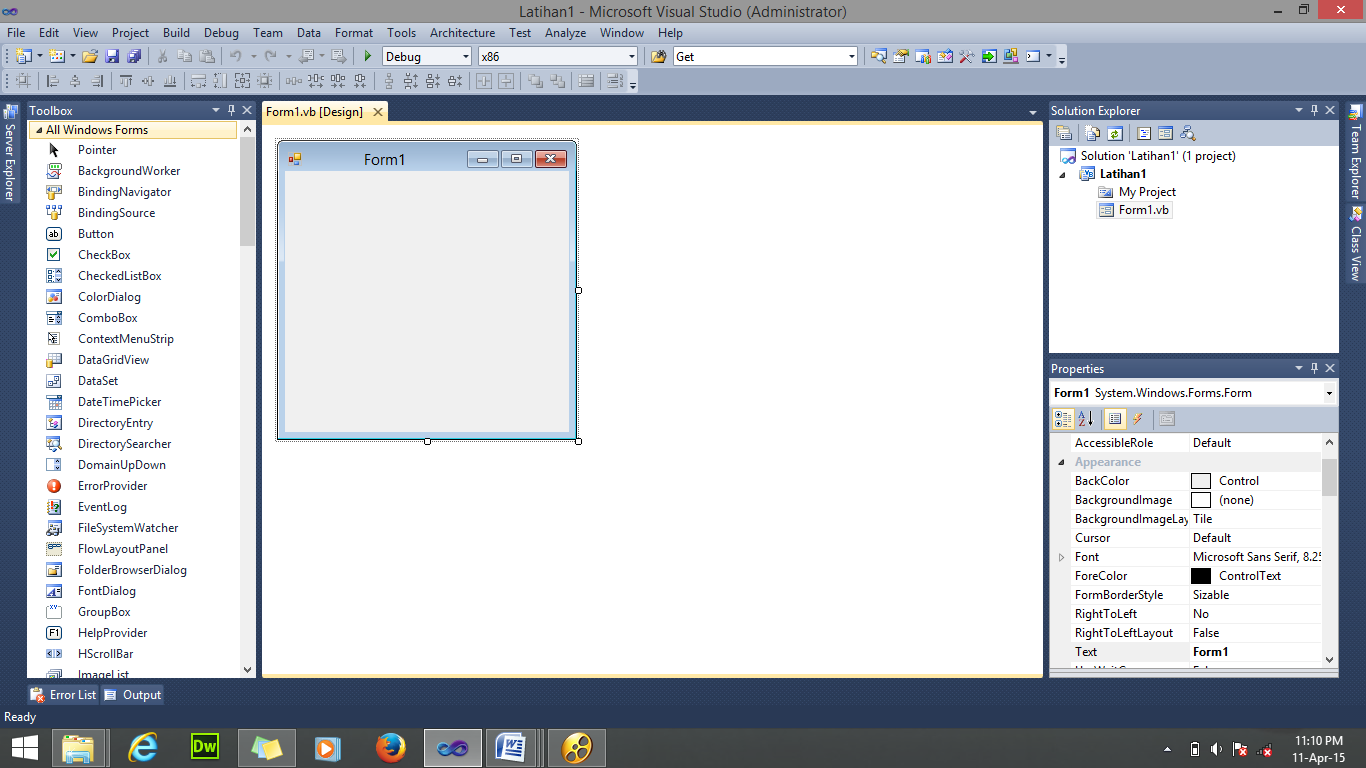


Gambar 1.5

Toolbox merupakan komponen-komponen atau tool-tool untuk membuat sebuah program aplikasi berbasis windows atau web. ToolBoxadalah tempat dimana kontrol dan komponen yang dilambangkan dengan icon. Kontrol dan komponen sangat membantu pada saat proses merancang tampilan Form dalam pembuatan program. kontrol dan komponen diletakkan pada tab-tab berdasarkan kegunaannnya.

Toolbox dapat dilihat ketika file yang aktif tersebut berhubungan dengan designer (form) atau antarmuka.

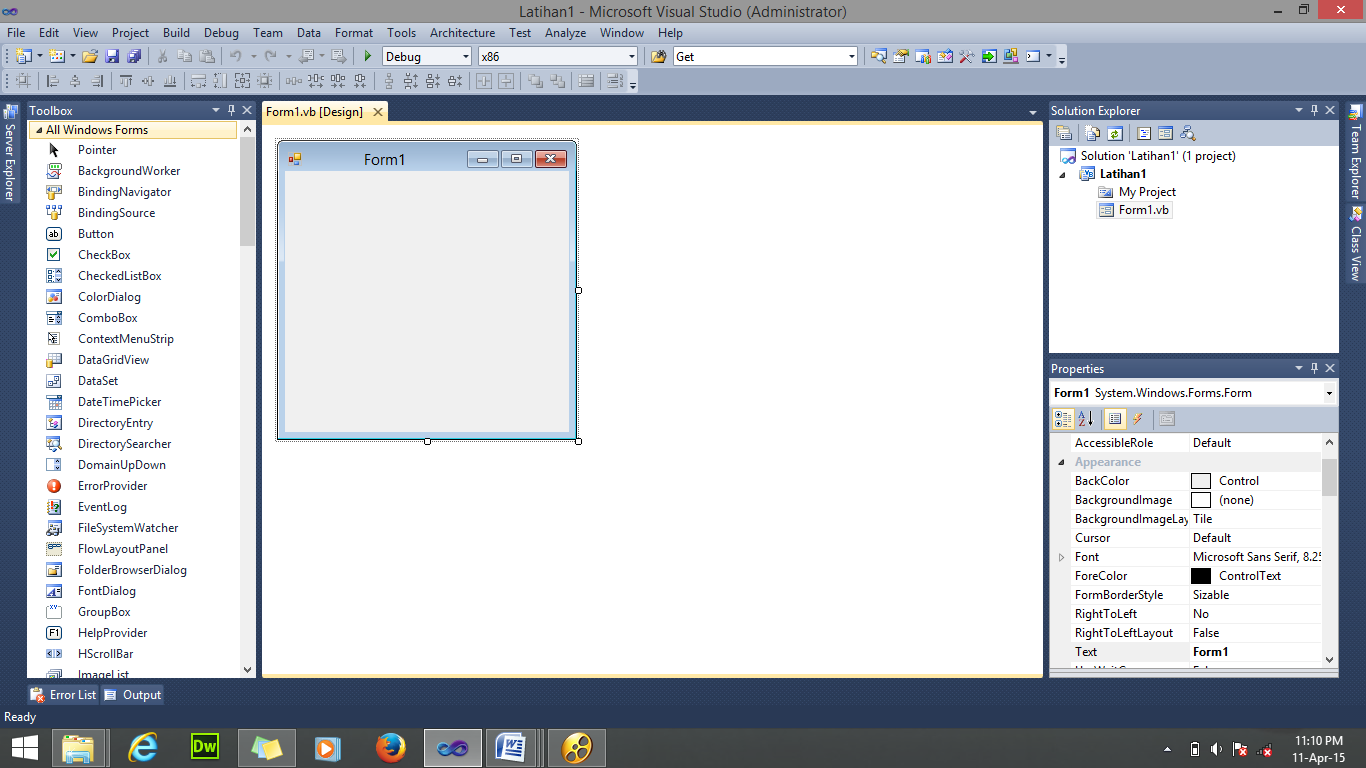
* **Window Project / Solution Exploler**



Gambar 1.6

Window Project berfungsi untuk menampilkan daftar form dan modul- modul yang kita tambahkan di project aplikasi yang sedang dikerjakan.

* **Window Properties**



Gambar 1.7

Window Properties berfungsi untuk menampilkan daftar properti dari sebuah komponen atau tool yang sedang aktif. Kita dapat mengubah properti dari sebuah komponen atau tool dengan cara mengaktifkan (mengklik/memilih) komponen atau tool tersebut, kemudian mengubah nilai propertinya di Window Properties.

# 

# **BAB II**

# **DASAR – DASAR OOP (OBJECT ORIENTED PROGRAMMING)**

## 2.1 Pengertian OOP

Pada awalnya setiap bahasa pemrograman menggunakan konsep terstruktur untuk membangun aplikasi, seiring berkembangnya ilmu pemrograman kini konsep pembangunan aplikasi berkembang menjadi konsep pemrograman yang berorientasikan pada objek atau OOP (Object Oriented Programming). OOP itu sendiri adalah suatu metode dalam pembuatan program, dengan tujuan untuk menyelesaikan kompleksnya berbagai masalah program yang terus meningkat. Object-oriented programs terdiri dari objects yang berinteraksi satu sama lainnya untuk menyelesaikan sebuah tugas. OOP merupakan cara yang paling efisien untuk menulis program komputer yang sangat mudah untuk dikombinasikan dan untuk dipergunakan kembali.

Tujuan utama dari pengembangan pendekatan berorientasi objek (OOP) adalah untuk menghilangkan beberapa kelemahan yang terdapat pada pendekatan procedural. Pemrograman procedural adalah suatu aktifitas pemrograman dengan memperhatikan urutan langkah-langkah perintah secara sistematis, logis , dan tersusun berdasarkan algoritma yang sederhana dan mudah dipahami. Adapun kelemahan pendekatan procedural diantaranya seperti :

* Pendekatan prosedural tidak cocok untuk projek aplikasi yang besar karena akan menghasilkan code yang banyak
* Apabila terjadi eror akan sulit untuk menemukan erornya karena terlalu banyaknya code
* Harus menulis ulang jika ada fungsi yang sama

## 2.2 Kelebihan dan Kekurangan konsep OOP

1. **Kelebihan**

* Efektif digunakan untuk menyelesaikan masalah besar, karena OOP terdiri dari class-class yang memisahkan setiap code program menjadi kelompok-kelompok kecil, sesuai dengan fungsinya.
* Fungsi-fungsi algoritma yang terbagi menjadi beberapa class dapat memudahkan kita dalam memahami program, begitu juga ketika ada bug pada program, kita bisa lebih mudah menemukan penyebab errornya dibanding ketika menggunakan prosedural programming.
* Object dan class dapat digunakan berkali-kali, sehingga dapat menghemat space memori.
* Cocok sekali digunakan dalam kasus pembuatan software yang rumit dan kompleks karena memberikan berbagai kemudahan kepada programmer.

1. **Kekurangan**

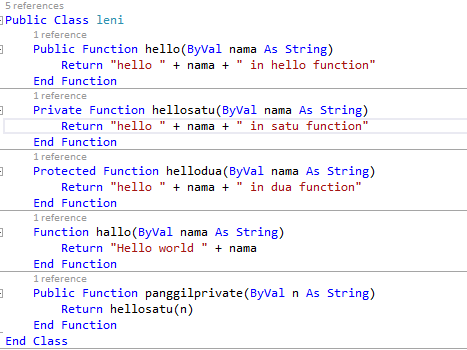
* Programmer yang telah terbiasa membuat program dengan konsep prosedural sering merasa kesulitan ketika dihadapkan dengan OOP. Biasanya "mereka" bingung dengan banyaknya istilah-istilah dalam OOP.
* Tidak semua masalah program dapat diselesaikan dengan OOP.

* 1. **Akses Modifier**

Akses Modifier yang berfungsi sebagai pengatur **hak akses** untuk mengakses suatu **Class, Attribut dan Method.** Untuk Class, Atribut dan Method akan dibahas di BAB selanjutnya.

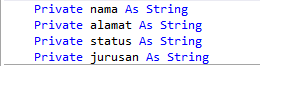
1. **Public**

Public, seperti namanya, atribut atau fungsi yang diberi akses modifier ini bebas diakses dari class manapun.



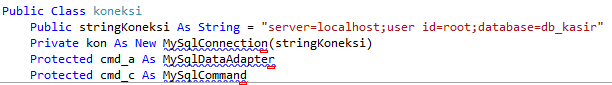
1. **Private**

Atribut atau fungsi yang diberi akses modifier private hanya bisa diakses didalam kelas itu sendiri.



1. **Protected**

Protected merupakan suatu akses modifier yang hanya bisa di akses untuk didalam kelas itu sendiri dan kelas turunannya.



# **BAB III**

# **KOMPONEN – KOMPONEN OOP**

## 3.1 Class

Class adalah sebuah deskripsi/gambaran dari sekumpulan objek yang memiliki operasi, attribute dan hubungan antar objek yang sama. Class merupakan cetak biru (Blueprint) dari objek yang mendefinisikan variabel-variabel dan method-method.

Contoh yang paling sering digunakan yaitu cetakan kue agar-agar. Cetakan agar-agar di analogikan sebagai sebuah class, sedangkan kue agar-agarnya adalah objek atau instance dari cetakan tadi. Jadi, cetakan agar-agar adalah bukan agar-agar.

Demikian juga class, class bukanlah sebuah objek. Tetapi, class adalah blue print (cetakan biru) yang membentuk sebuah objek, dan sebuah objek adalah instance dari sebuah class.

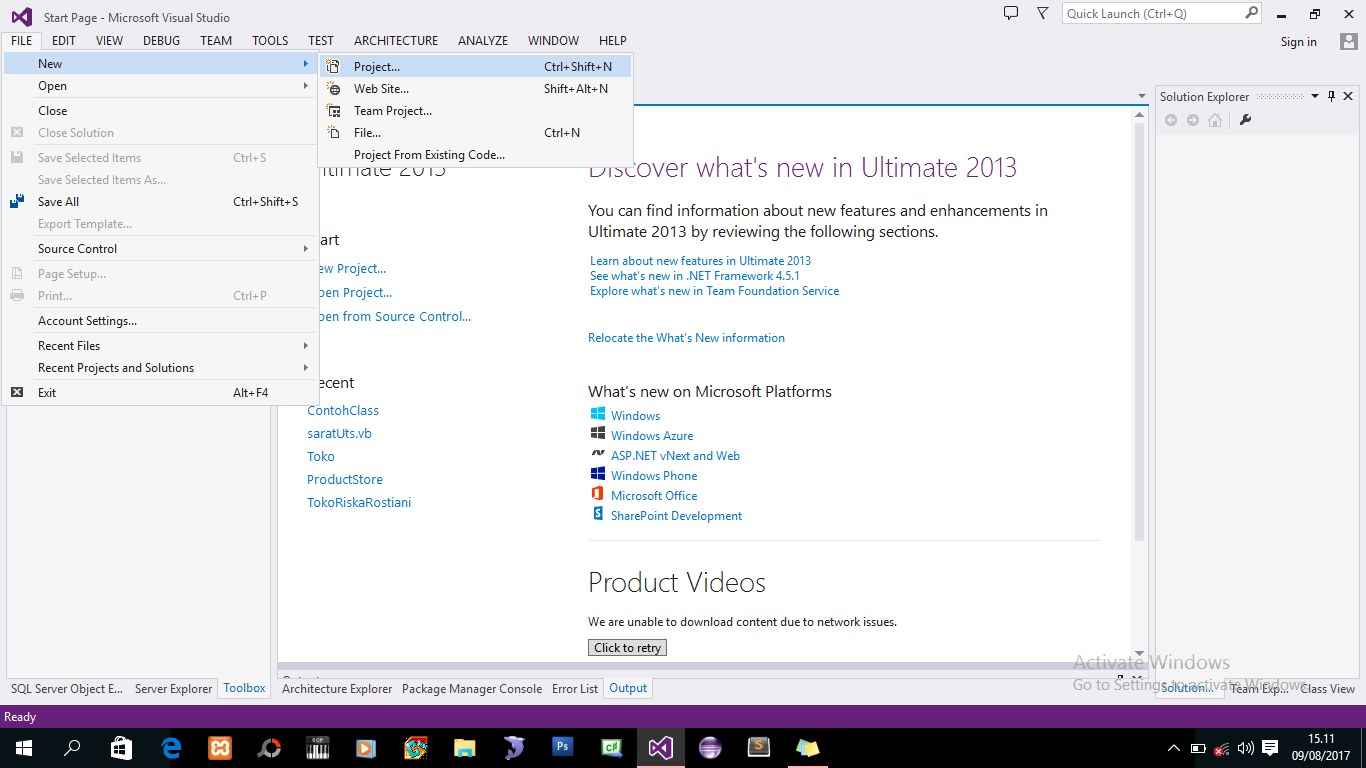
Contoh dari class : class kendaraan, class animal, class buah, class mamalia dan lain-lain.

Contoh hubungan antar objek yang sama bisa diilustrasikan sebagai berikut :

Bayangkan jika kita mempunyai 2 objek yang sama, yaitu sama-sama mamalia : objek kucing dan objek sapi. Dan memiliki satu class yaitu class mamalia.

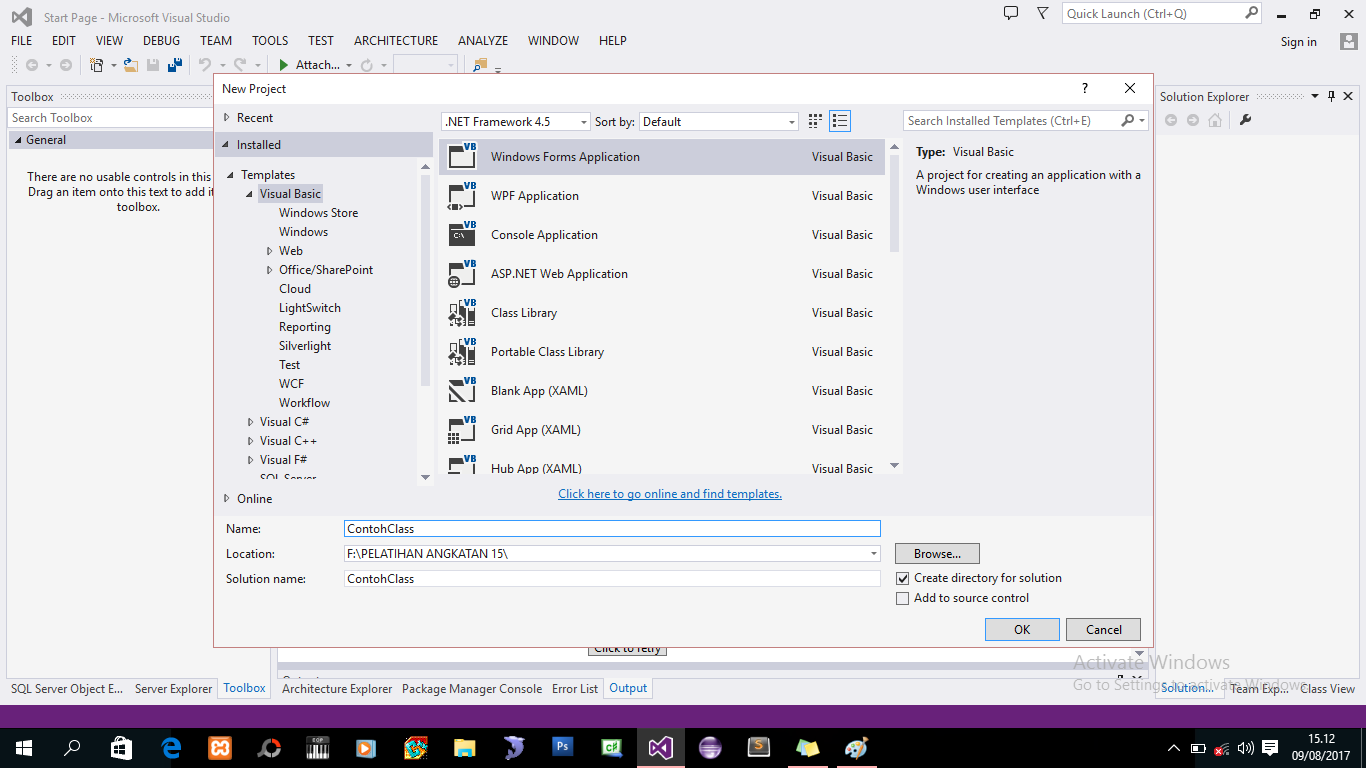
Supaya lebih jelas, lihat dan praktikan contoh berikut :

1. Buka visual studio nya dan membuat file baru, klik file -> new -> project seperti berikut :



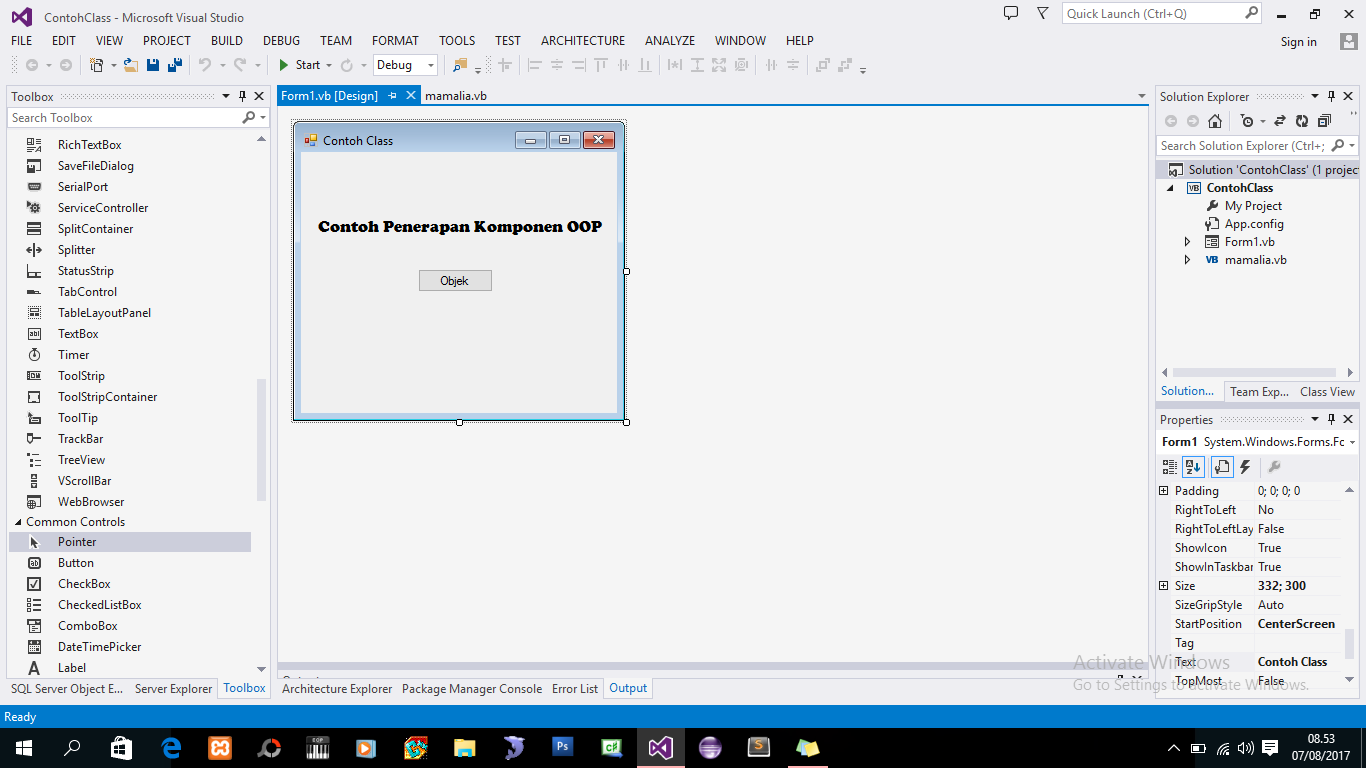
Gambar 3.1

1. Maka akan muncul tampilan seperti dibawah ini. Pilih Visual basic-> Windows form Application. Kemudian memberikan nama project ContohClass, dan klik OK seperti berikut :



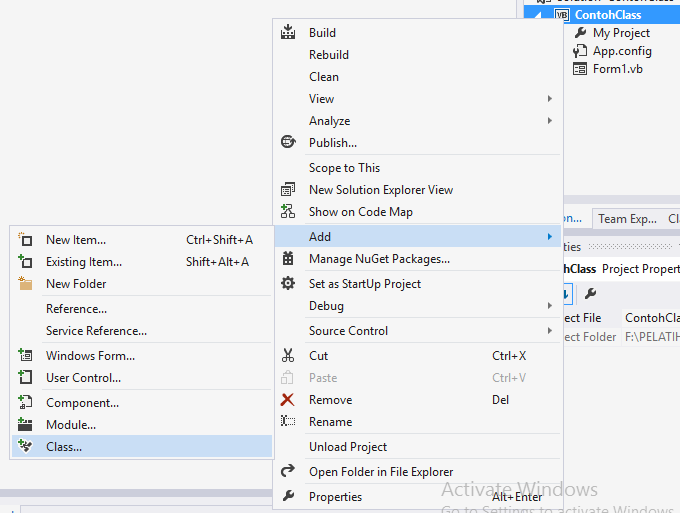
Gambar 3.2

1. Kemudian akan muncul tampilan yang hanya ada 1 form. Dalam form tambahkan 1 buah label dan ganti nama label dengan “Contoh Penerapan Komponen OOP”, kemudian 1 buah button dan beri nama Class seperti berikut:



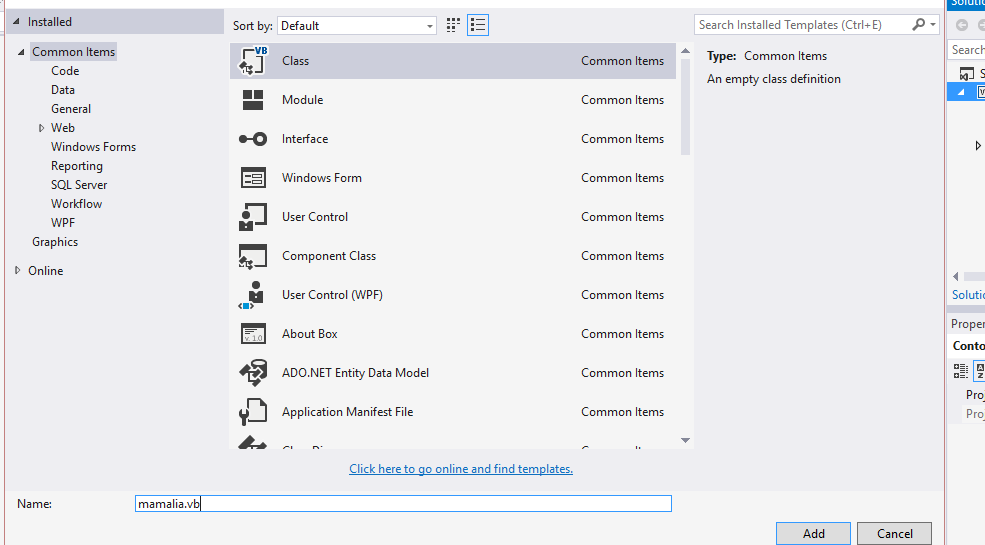
Gambar 3.3

1. Setelah itu buat class baru dengan cara klik kanan pada nama project -> add->class, seperti berikut :



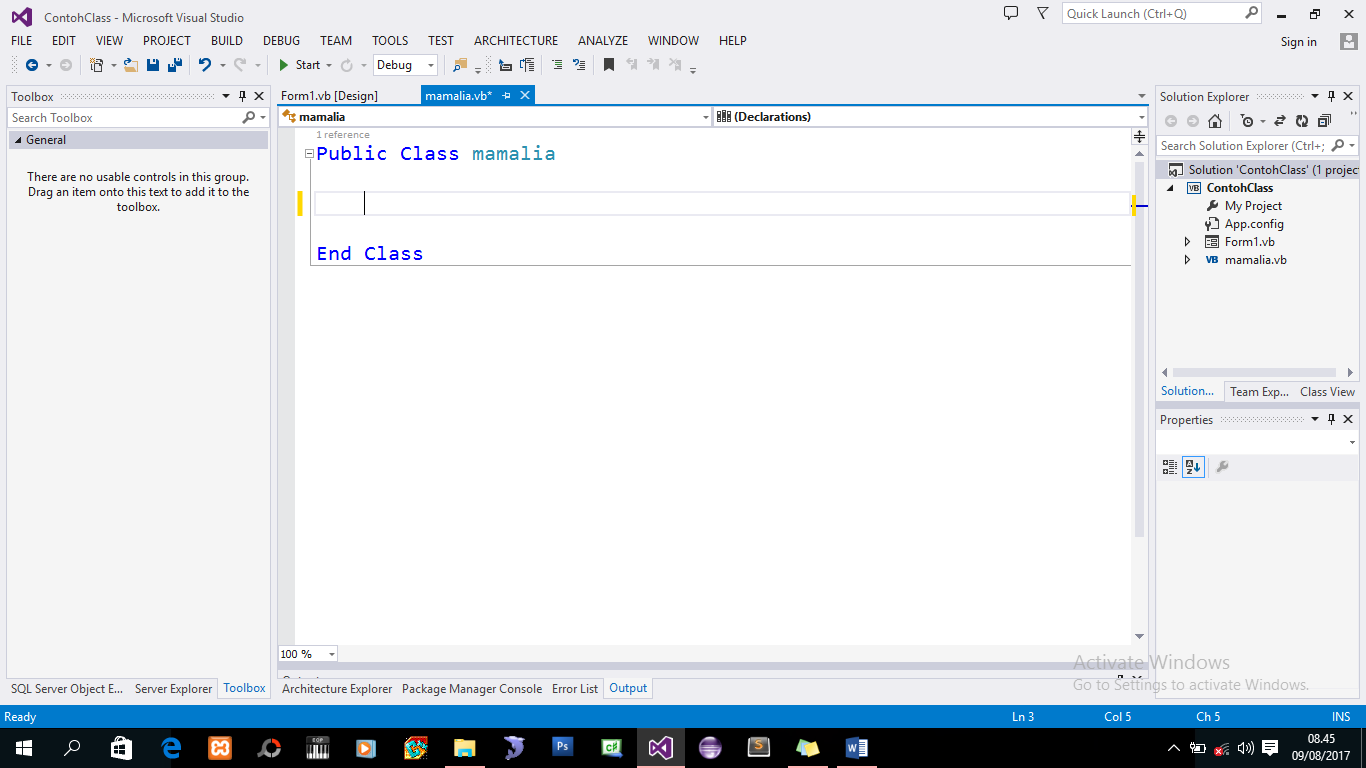
Gambar 3.4

1. Kemudian berikan nama class tersebut mamalia.vb, dan Klik Add seperti berikut :



Gambar 3.5

1. Tampilan sebuah class :



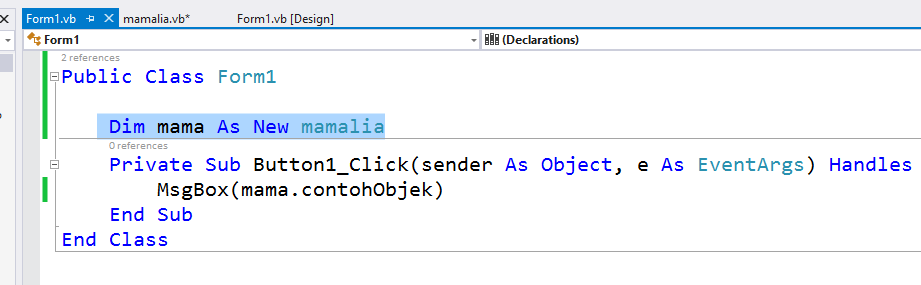
Gambar 3.6

## 3.2 Objek

Objek adalah representasi dari fakta dunia nyata. Objek sering kali disebut sebuah benda yang dapat anda kenali atau bayangkan fisik maupun konseptual.

Contoh fisik : rumah, mobil, sepeda motor, gelas, komputer, meja dan sebagainya. Contoh konseptual: sebuah proses kimia. Namun bagi programmer, objek adalah sebuah benda yang dapat anda nyatakan dalam suatu program dengan cara instansiasi/perwujudan dari sebuah class. Setiap objek akan mempunyai attributedan methodyang dimiliki oleh classnya. Setiap objek dapat berinteraksi dengan objek lainnya meskipun berasal dari class yang berbeda. Alasan mengapa kita menyatakan objek-objek tersebut dalam sebuah program adalah untuk memungkinkan kita menggunakan kembali code (code re-use) yang berada pada class yang di instansiasi tersebut.

Contoh penerapan objek, masih menggunakan class mamalia. Seperti berikut :



Gambar 3.7

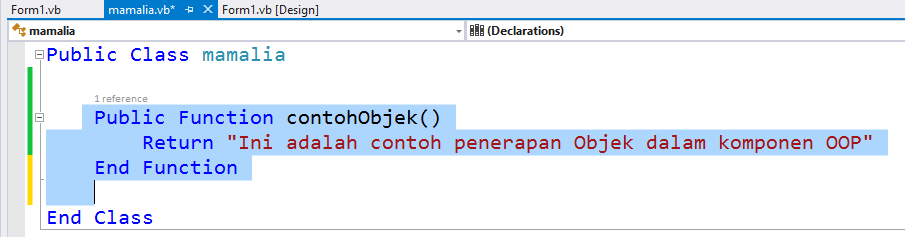
Baris kode yang diblok itulah yg disebut instansiasi/perwujudan suatu Class yaitu objek dari Claas mamalia yang diberi nama “mama”. Objek mama akan mempunyai attributedan methodyang dimiliki oleh class mamalia.

## 3.3 Method

Method adalah suatu tindakan/aksi yang di lakukan oleh objek atau serangkaian statements dalam suatu class yang menghandle suatu task tertentu atau suatu Operasi berupa fungsi-fungsi yang dapat dikerjakan oleh objek. Method dipanggil melalui objek.

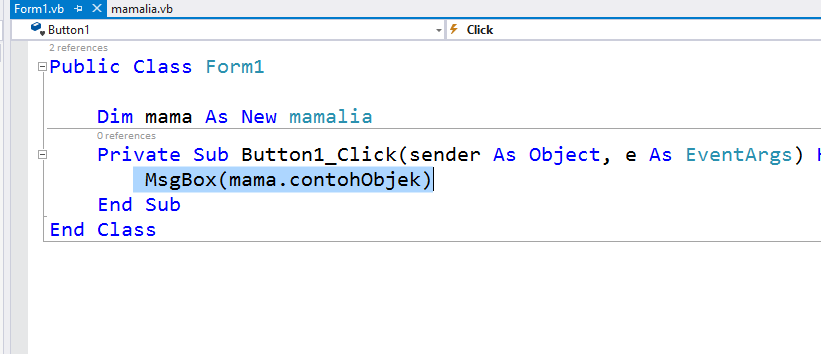
Misalkan didalam suatu class hewan dan kita membuat objek sapi. Dalam class hewan terdapat method SuaraSapi. Didalam method SuaraSapi() akan mengembalikan nilai “Suara Sapi Moo”.

Supaya lebih jelas, maka kita gunakan contoh method masih dalam class yang kita buat sebelumnya yaitu class mamalia seperti berikut :



Gambar 3.8

Dari baris kode method di atas jika kita memanggil method tersebut dengan objek mama dari class mamalia tersebut maka akan mengembalikan nilai “ini adalah contoh penerapan objek dalam komponen OOP. Pemanggilan method tersebut seperti dibawah ini:



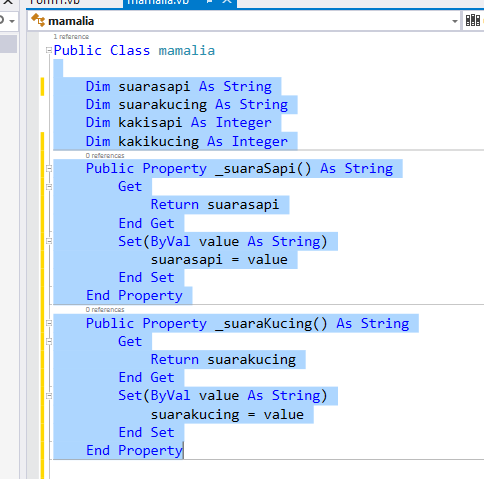
Gambar 3.9

## 3.4 Attribute

Attribute merupakan property dari sebuah class yang memiliki nama dan mendeskripsikan suatu range / nilai dari properti tersebut.

Bentuk attribute mirip dengan variabel yaitu dapat di baca dan di set secara langsung dan memiliki type data. Type yang umum untuk suatu attribute adalah integer, boolean, string dan lain-lain.

Contoh attribut masih dalam class yang kita buat sebelumnya, seperti berikut :



Gambar 3.10

Baris kode yang diblok tersebut contoh dari attribut yang dimiliki oleh Class mamalia. Itu merupakan contoh yang dibuat oleh penulis.

# **BAB IV**

# **PENANGANAN ERROR (ERROR HANDLING)**

## 4.1 Try – Catch

Penanganan error yang berkaitan erat dengan OOP (object oriented programming) yaitu Exception Handling. Sebuah *Exception* adalah masalah yang timbul selama program berjalan. *Exception* adalah respon terhadap keadaan yang biasa muncul ketika sebuah program sedang berjalan, seperti mencoba untuk membagi sebuah dengan nol.

*Exception* menyediakan cara untuk mentransfer kontrol dari satu bagian dari program yang lain. Penanganan masalah ini di VB.net dibangun dalam empat kata kunci: **Try**, **Catch**, **Finally** dan **Throw.**

* **Try** : mengidentifikasi blok kode yang berisi kode-kode tertentu yang diharapkan untuk dilaksanakan oleh program atau aplikasi. Blok Try diikuti oleh satu atau lebih blok Catch.
* **Catch** : Saat terjadi kesalahan pada blok Try, maka blok ini akan menangkap kesalahan dengan yang isinya biasanya menampilkan pesan tertentu atau menuju ke blok lain untuk mengeksekusi perintah yang diinginkan. Kata kunci Catch menunjukkan penangkapan pengecualian.
* **Finally** : blok ini digunakan untuk menjalankan satu set pernyataan, apakah eksepsi dilemparkan atau tidak. Misalnya, jika kita membuka file, itu harus ditutup apakah eksepsi dimunculkan atau tidak, atau saat operasi dengan database pada blok ini ditambahkan perintah untuk menutup koneksi yang telah dibuka pada blok Try.
* **Throw** : Suatu program melempar pengecualian ketika masalah muncul. Hal ini dilakukan dengan menggunakan kata kunci Throw.

**Try**

Kode-kode yang diharapkan dalam suatu proses disimpan disini

**Catch ex as Exception**

Kode-kode jika terjadi kesalahan dalam blok diatas disimpan disini.Messagebox.Show(ex.Message)

**Finally**

Kode-kode untuk melakukan sesuatu jika terjadi exception atau tidak. Kode-kode pada blok ini pasti dijalankan.

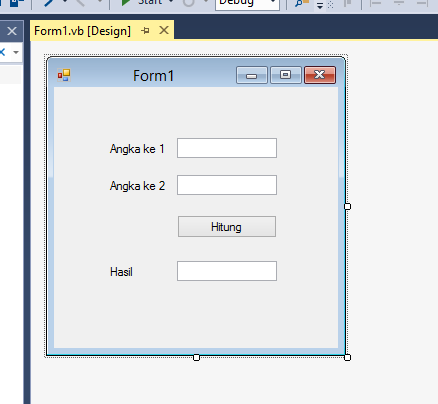
**End Try**

|  |  |
| --- | --- |
|  |  |
| Berikut adalah struktur dari try catch statement: |  |

“**Kelebihan try catch yaitu dapat menangani error tanpa membuat program itu berhenti.”**

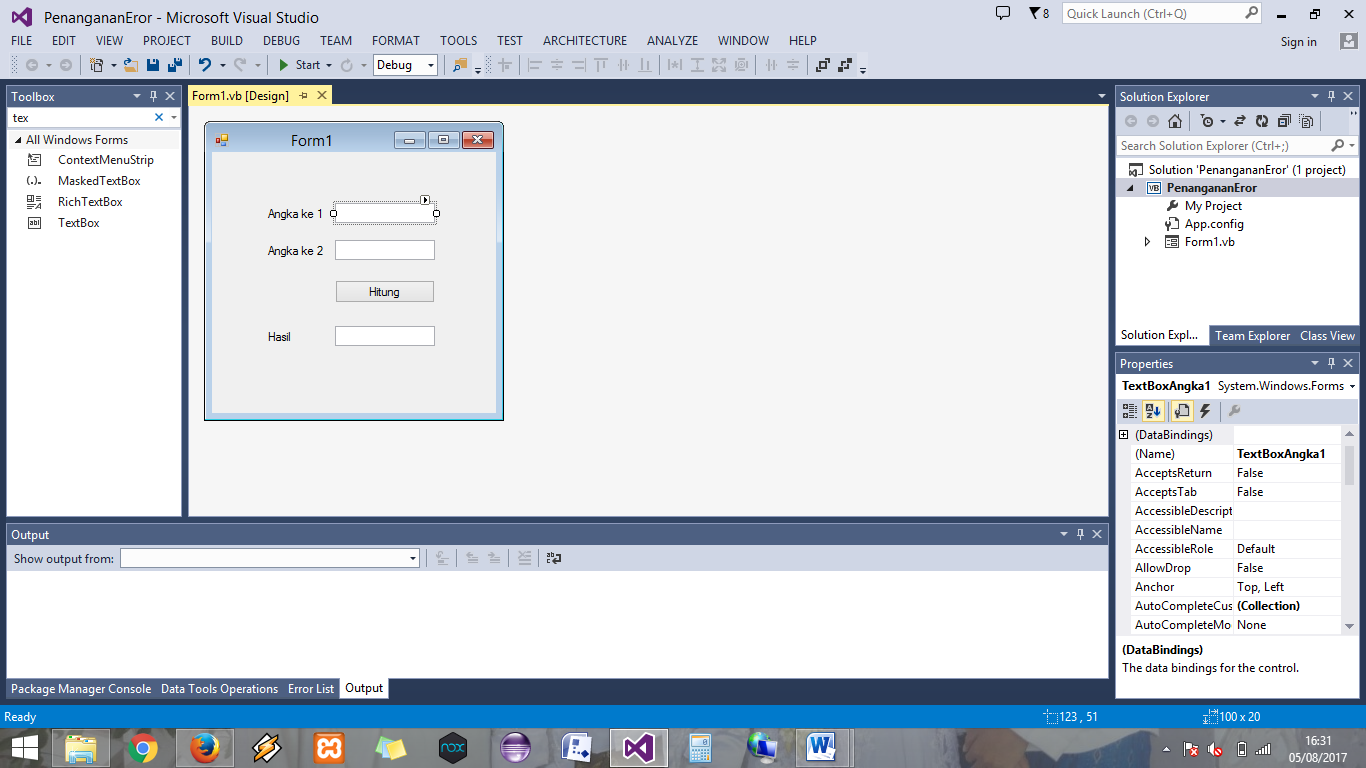
Contoh implementasi :

1. Buatlah sebuah mini project dengan langkah-langkah yang telah diterangkan di BAB I dengan nama project PenangananEror.vb dan buatlah tampilan dengan menambahkan label, textbox dan button seperti berikut ini !



Gambar 4.1

1. Selanjutnya ganti setiap nama TextBox dan button di Properties agar mudah dikenali.



Gambar 4.2

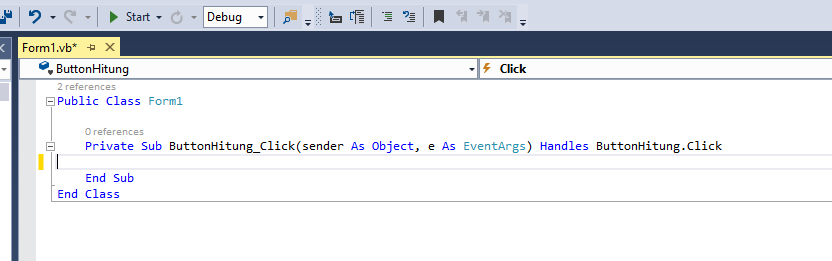
TextBox1 ubah menjadi TextBoxAngka1

TextBox2 ubah menjadi TextBoxAngka2

TextBox3 ubah menjadi TextBoxHasil

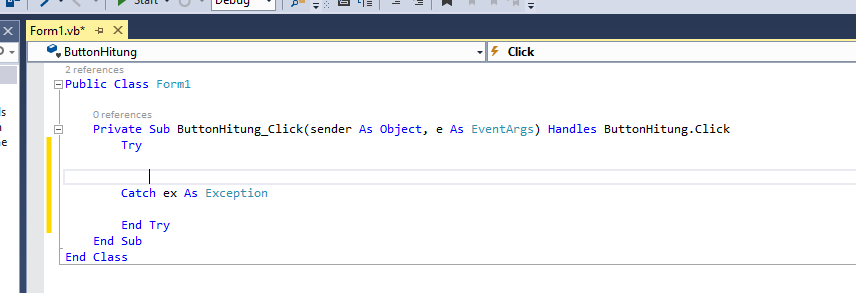
Button ubah menjadi ButtonHitung

1. Klik 2x pada tombol button sehingga akan tampil seperti berikut



Gambar 4.3

Tulis code **Try** kemudian Tab 2x sehingga seperti berikut :



Gambar 4.4

1. Tulis code seperti berikut :

Public Class Form1

Private Sub ButtonHitung\_Click(sender As Object, e As EventArgs) Handles ButtonHitung.Click

Try

Dim angka1 As Integer = TextBoxAngka1.Text

Dim angka2 As Integer = TextBoxAngka2.Text

TextBoxHasil.Text = angka1 + angka2

Catch ex As Exception

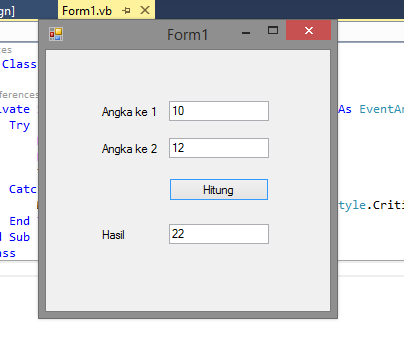
MsgBox("Tolong masukkan angka aja", MsgBoxStyle.Critical, "Perhatian")

End Try

End Sub

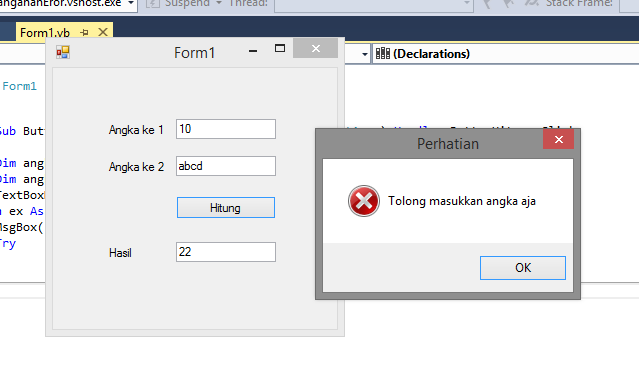
End Class

Kemudian Run ! inputkan angka pada TextBox Angka ke 1 dan TextBox Angka ke 2 -> button Hitung sehingga tampilan yang muncul kan seperti berikut :



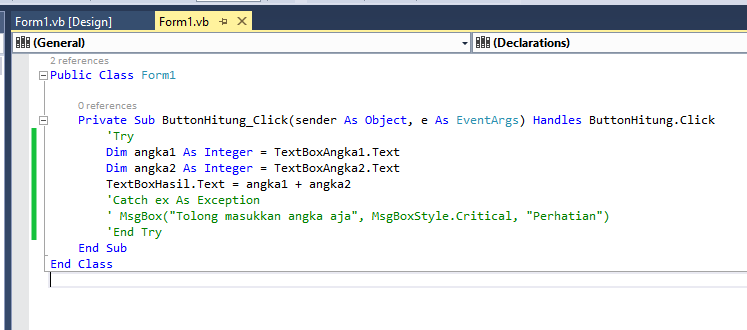
Gambar 4.5

Ini adalah tampilan jika inputan sesuai dengan ketentuan, kemudian coba inputkan disalah satu TextBox (Angka ke 1 / Angka ke 2) dengan selain angka misalnya menginputkan karakter, maka yang terjadi akan seperti berikut :



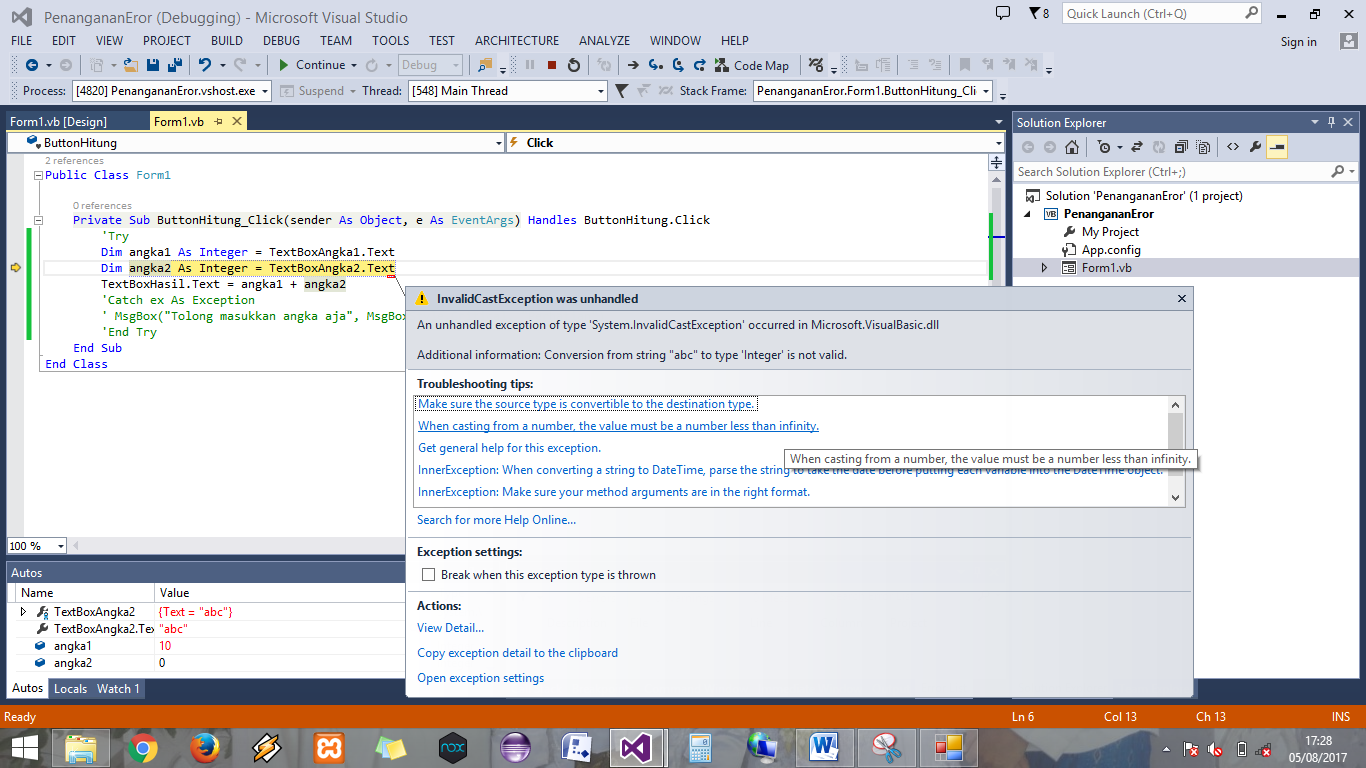
Gambar 4.6

Akan ada message bahwa inputan tidak sesuai/eror , namun dengan menggunakan Try Catch walaupu eror aplikasi akan tetap berjalan. Coba kita jalankan aplikasi dengan tidak menggunakan Try Catch dan inputan tidak sesuai dengan ketentuan. Cobalah coment code Try Catch seperti berikut :



Gambar 4.7

Kemudian Run kembali , maka yang akan muncul adalah peringatan eror dan aplikasi tidak berjalan.



Gambar 4.8

# **BAB V**

# **FITUR – FITUR DALAM OOP**

Sebelum membahas lebih dalam, perlu diketahui sebagai **ketentuan dasar** apa saja yang masuk ke dalam fitur – fitur OOP, yaitu :

1. **Inheritance**
2. **Polymorphism**
3. **Encapsulation**

Sudah tau kan apa saja yang masuk ke dalam fitur – fitur OOP, di ingat terus yaa dan jangan di lupakan !!! Oke sekarang mari kita bahas secara mendalam satu persatu dari fitur – fitur OOP

## 5.1 Inheritance

Atau sering disebut penurunan sifat/pewarisan, inheritance merupakan ciri khas dari OOP yang tidak terdapat pada pemrograman prosedural gaya lama. Mengapa disebut dengan penurunan sifat/ pewarisan ??? karena inheritance merupakan kemampuan untuk membuat class turunan, yang akan mewarisi semua property dan method yang ada pada class induknya yang sifatnya public dan protected, jadi semua property dan method yang ada pada class induk bisa digunakan kembali baik untuk dimodifikasi atau tidak pada class turunan.nya .

Lebih jelasnya misalkan :

*A sebagai Ayah memiliki Anak bernama B , A berkulit Hitam maka B akan berkulit Hitam juga.*

Ini merupakan inheritance terhadap A kepada B , dan inheritance bukan hanya dapat 1x diwariskan , tetapi dapat lebih dari 1x. seperti berikut :  
*A sebagai Ayah memiliki Anak bernama B , A berkulit Hitam maka B akan berkulit Hitam jg . dan B pun memiliki anak C maka C pun akan berkulit hitam*

****

**1. Base Class**

Atau bisa juga disebut class induk merupakan class dasar atau class yang utama yang dapat mewariskan behavior dan attribute kepada class turunannya ibaratkan Base Class ini adalah Ayah (parrent class) yang dapat mewariskan segala yang ada pada dirinya.

**2. Derived Class**

Adalah class turunannya/ class anak yang dibentuk dari proses pewarisan class induk dengan segala sifatnya. Proses pewarisan dari base class ke derived class disebut **deriving**.

Contoh Program Inherits

* Buka visual basic.Net dan buat project baru
* Tambah class baru dengan cara klik kanan pada nama project dan klik add class
* Isi nama class dengan nama ***Parent,*** maka akan muncul class baru dengan code sebagai berikut

Public Class parent

End Class

Class Parent ini adalah Base Class/ Class Utamanya

* Kemudian buat 2 class lagi dengan cara yang sama seperti membuat class Parent, beri nama Class ***Son*** dan ***Daughter.***

Pada masing – masing class akan muncul code sama seperti pada code yang ada di class Parent, namun class Son dan Daughter akan menginherits ke class Parent dengan cara sebagai berikut

Public Class Son

**Inherits parent**

End Class

Berikut Codingan lengkap untuk contoh program Inherits

Pada Class Parent

Public Class parent

Public Function ClassParent()

Return "This Method from ClassParent"

End Function

End Class

Penjelasan :

Pada class Parent sebagai Base Class mempunyai method bernama ClassParent yang akan mengembalikan nilai string “This method from ClassParent”

Pada Class Son

Public Class son

Inherits parent

Public Function ClassSon()

Return "This Method from ClassSon and inherits from " + ClassParent()

End Function

End Class

Penjelasan :

Pada class Son sebagai Derived Class ia mengInherits pada class Parent dan mempunyai method bernama ClassSon yang akan mengembalikan nilai string “This method from ClassSon” dan di tambah return nilai string yang ada pada method ClassParent pada Class Parent

Pada Class Daughter

Public Class daughter

Inherits parent

Public Function ClassDaughter()

Return "This Method from ClassDaughter and inherits from " + ClassParent()

End Function

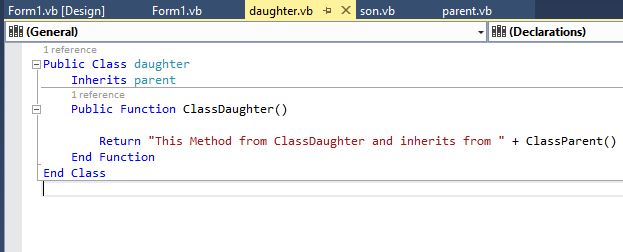
End Class

Penjelasan :

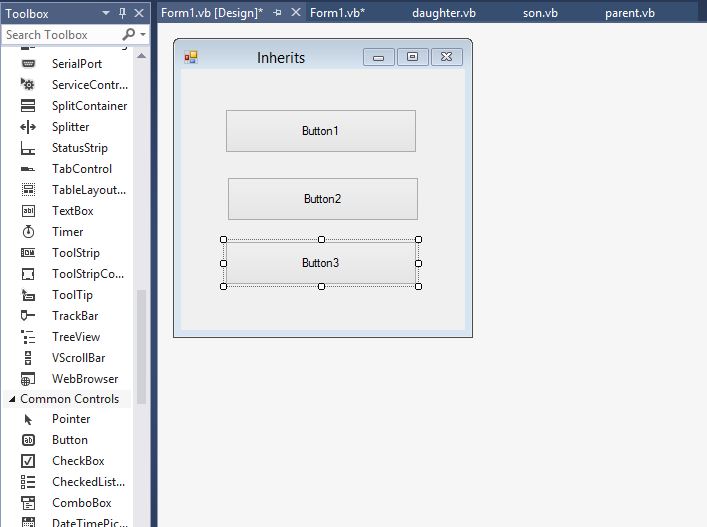
Pada class Daughter sebagai Derived Class ia mengInherits pada class Parent dan mempunyai method bernama ClassDaughter yang akan mengembalikan nilai string “This method from ClassDaughter” dan di tambah return nilai string yang ada pada method ClassParent pada Class Parent

* Kemudian pada Form Desaign tambahkan 3 Button untuk mengecek masing – masing method dengan cara klik form Design dan drag 3 button pada toolbox kemudian ganti nama button dengan cara klik button lalu ke properties dan tulis nama button pada Text

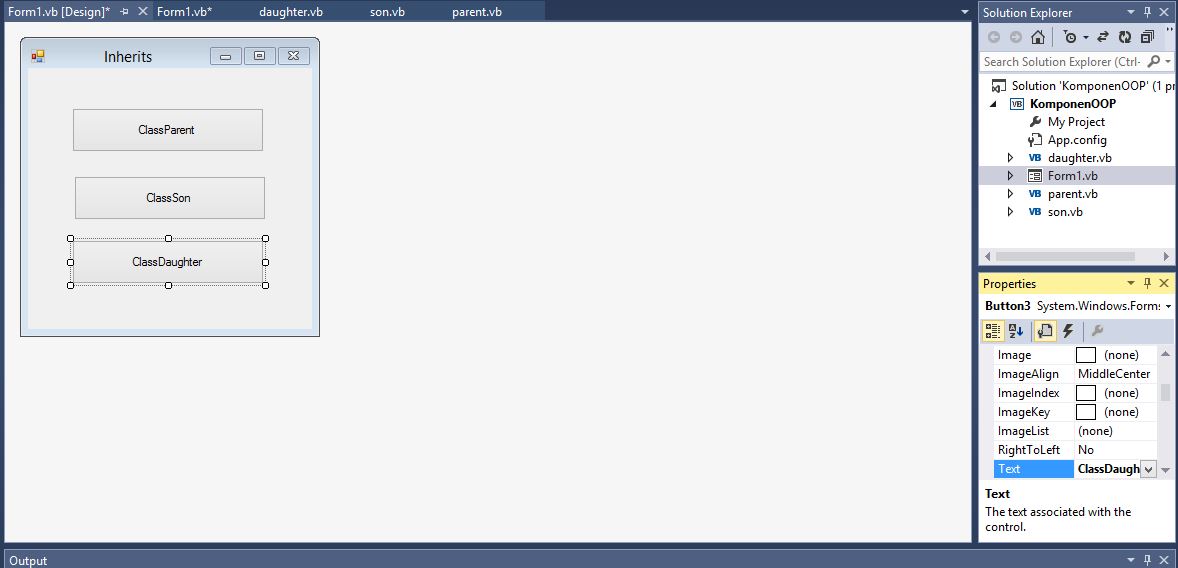
*Gambar 1 Klik form desaign*



*Gambar 2 drag button dari toolbox*

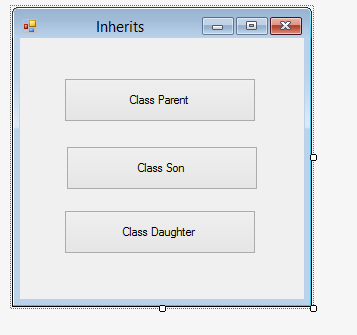
**

*Gambar 3 Ubah Nama Button*

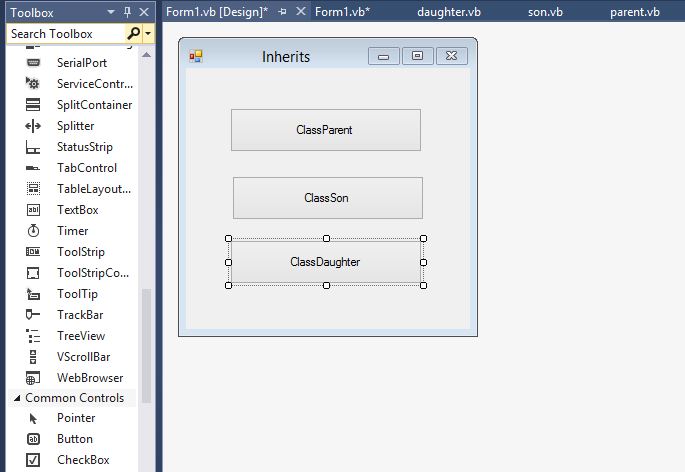
**

Maka tampilan form desaign akan seperti gambar 4

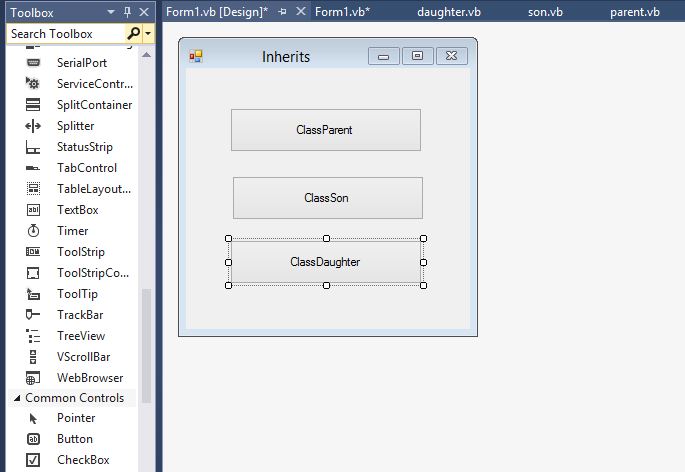
*Gambar 4 Tampilan form desaign*



*Gambar 5 klik class form*

**

*Gambar 6 klik 2x pada button*

**

* Tuliskan code berikut pada class form dengan mengklik class form seperti pada gambar 5 , dan untuk button klik 2x pada button yang ada pada form design seperti pada gambar 6 sesuai dengan nama button.nya.

Public Class Form1

Dim obj\_Son As New son

Dim obj\_Daughter As New daughter

Private Sub Button1\_Click(sender As Object, e As EventArgs) Handles Button1.Click

MsgBox(obj\_Son.ClassParent())

End Sub

Private Sub Button2\_Click(sender As Object, e As EventArgs) Handles Button2.Click

MsgBox(obj\_Son.ClassSon())

End Sub

Private Sub Button3\_Click(sender As Object, e As EventArgs) Handles Button3.Click

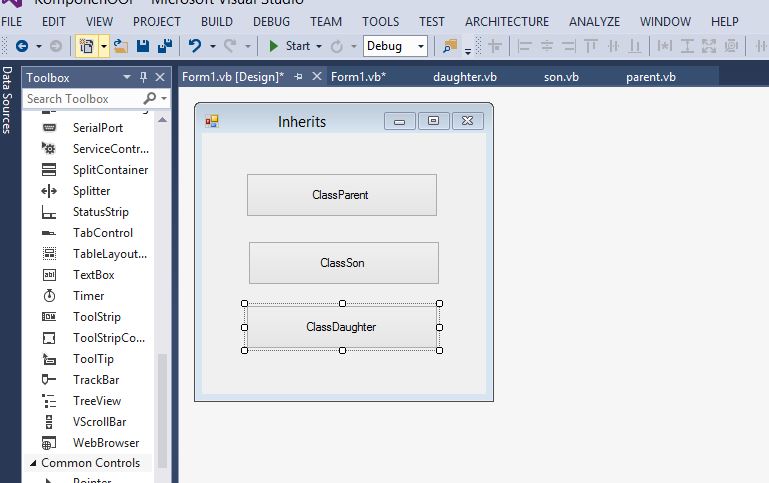
MsgBox(obj\_Daughter.ClassDaughter())

End Sub

End Class

* Kemudian running dengan cara klik toolbox runing seperti pada gambar 7 dan coba pahami apa perbedaannya.

*Gambar 7 Running*

**

## 5.2 Polymorphism

Sering disebut banyak bentuk / perubahan bentuk, polymorphism merupakan salah satu sifat class yang memungkinkan sebuah method mempunyai nama yang sama tetapi dengan aksi yang berbeda. Kalau biasanya pada metode struktural apabila terdapat nama method yang sama pasti terjadi error, nah pada OOP ini bisa diatasi dengan penggunaan Polymorphism.

Ada 2 method pada polymorphism yaitu :

1. **Method Overloading**
2. **Method Overriding**

Ingat – ingat yaa, selanjutnya mari kita bahas kedua method tersebut

1. Method Overloading

Adalah suatu konsep yang terdapat sejumlah method yang sama namun memiliki fungsi yang berbeda dalam suatu class yang sama atau class yang berbeda.

Ciri – ciri Overloading :

* Memiliki nama method sama.
* Class berbeda
* Memiliki jumlah parameter atau type data parameter yang berbeda.
* Nilai yang di return harus memiliki type yang sama.
* Menggunakan kata kunci ***overloads.***

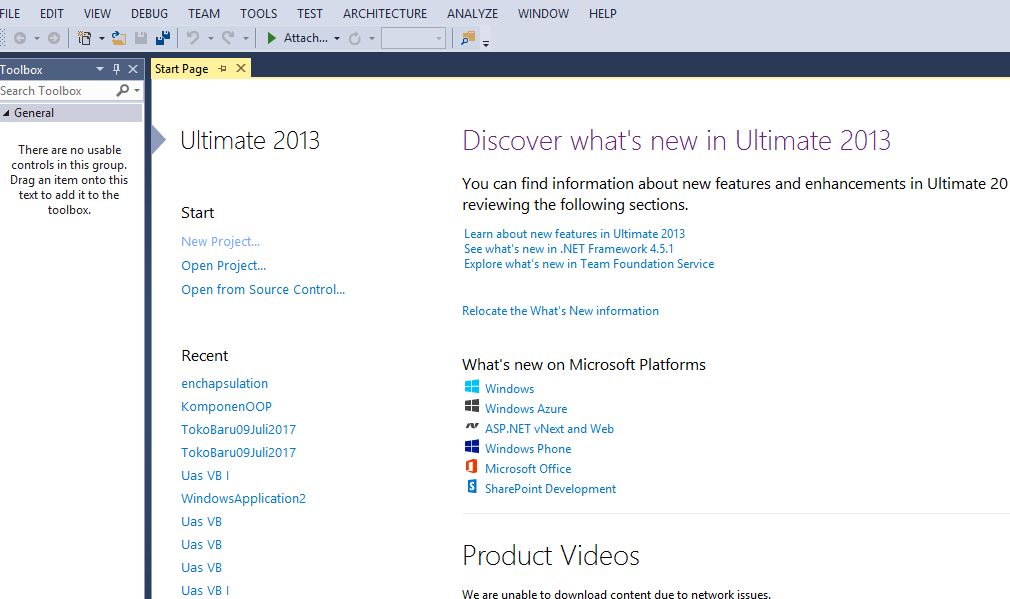
Contoh Programnya :

Berikut Langkah dan Ketentuannya :

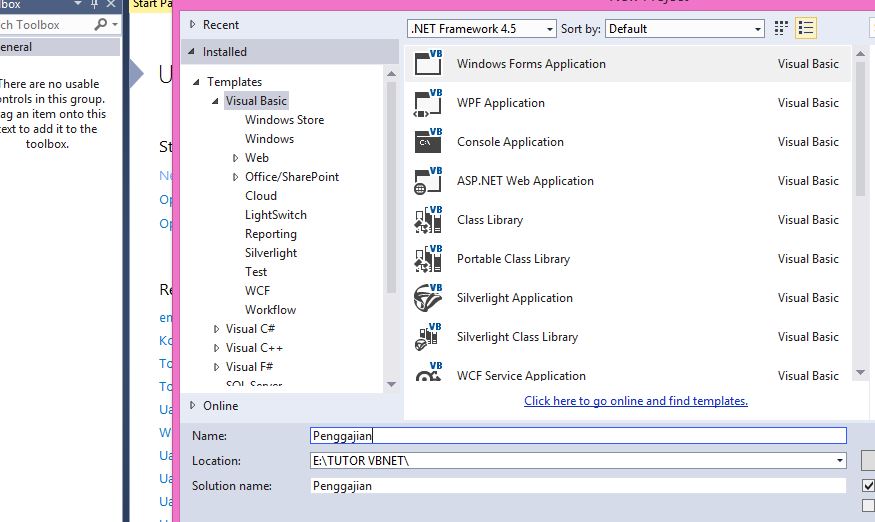
Buat suatu program penghitung gaji, dengan ketentuan berikut:

* Buatlah new project seperti gambar 8 kemudian beri nama penggajian seperti gambar 9.

*Gambar 8 membuat New Project*

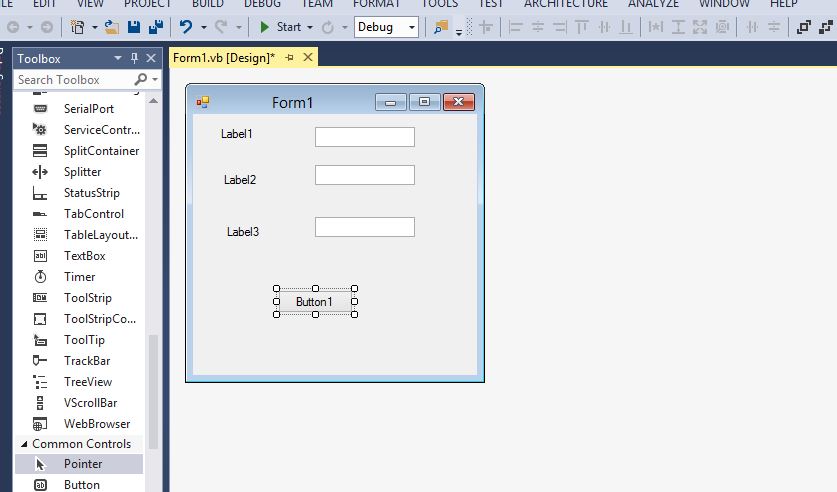
**

*Gambar 9 Beri Nama Penggajian*

**

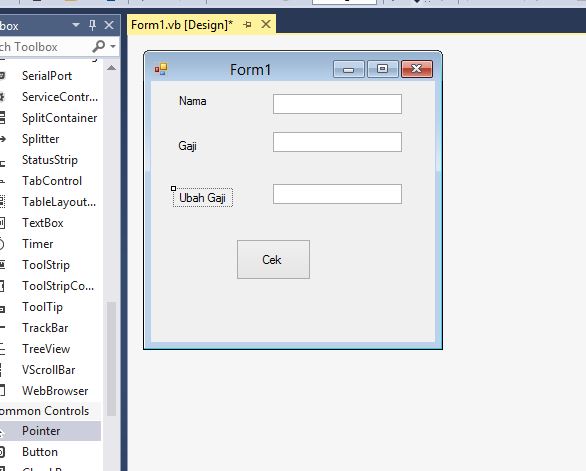
* Dan buatlah form yang berisi 3 buah label, 3 buah textbox dan 1 buah button seperti pada gambar 10.

*Gambar 10 Memberi 3 label 3 textbox dan 1 button*

**

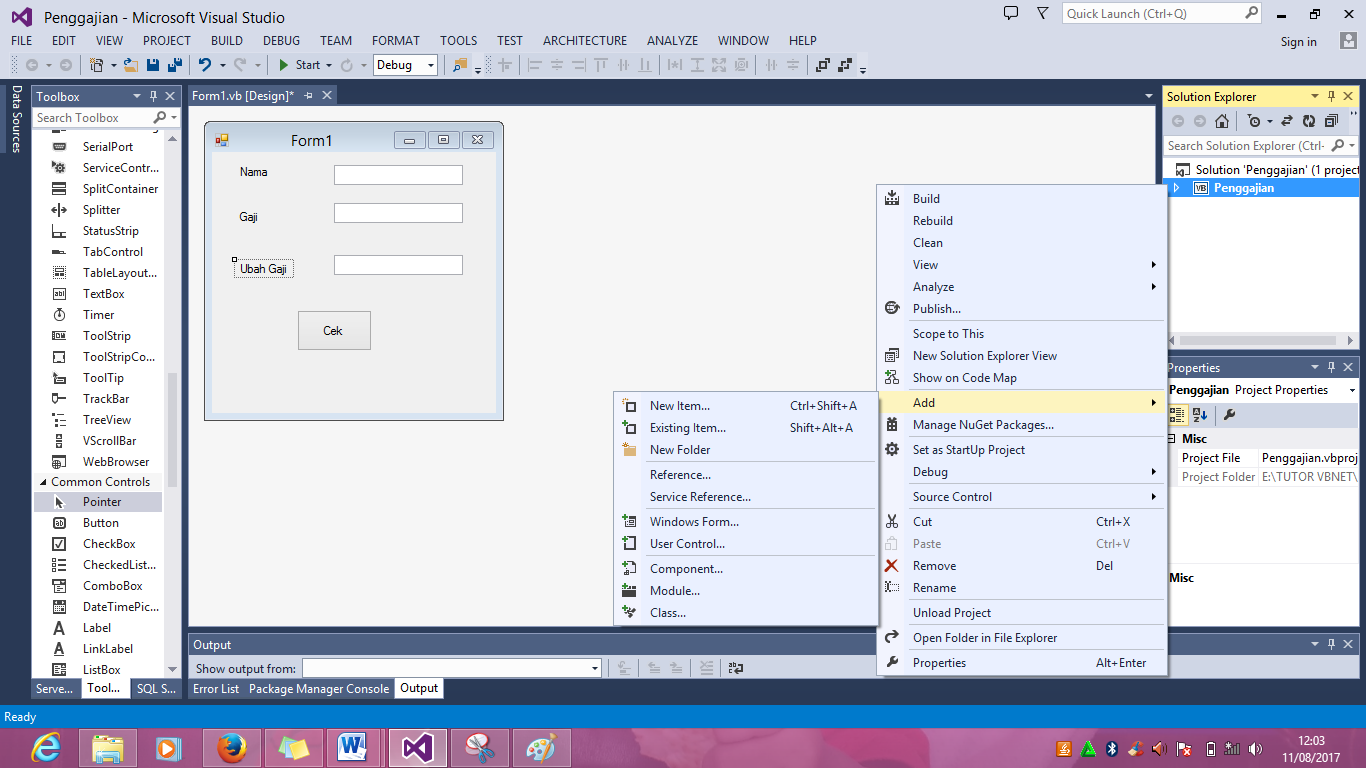
* Ganti Nama Label dan Button seperti cara pada gambar 3, kemudian hasilnya akan seperti gambar 11.

*Gambar 11 hasil Form Desaign*

**

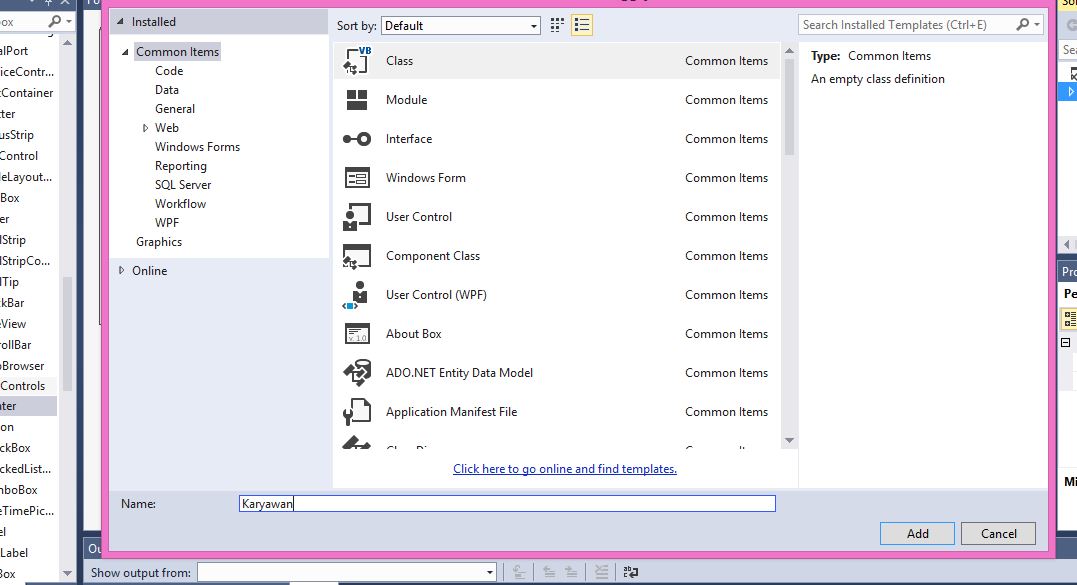
* Buat Base classnya dengan nama Karyawan dengan cara klik kanan pada nama project pilih add pilih class seperti pada gambar 12.

*Gambar 12 Membuat Base Class Karyawan*

**

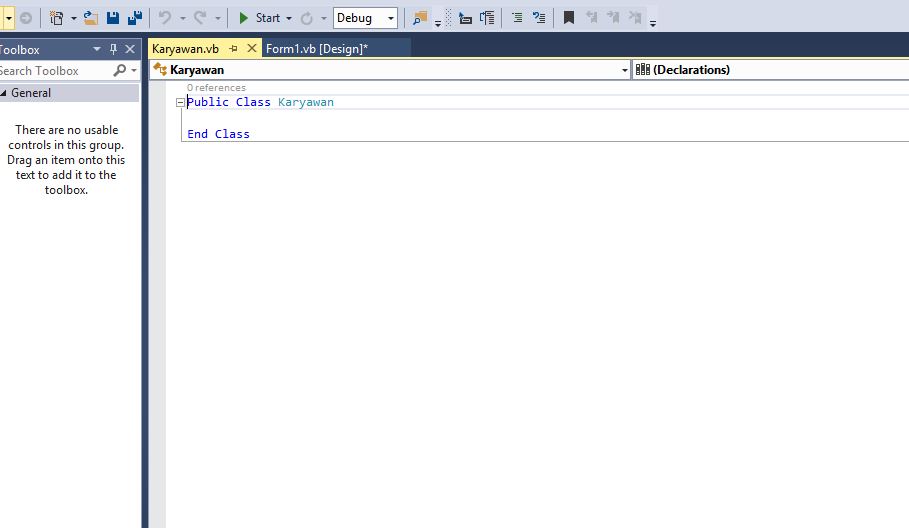
* Kemudian akan muncul tampilan nama class dan beri nama Karyawan kemudian klik add seperti gambar 13.

*Gambar 13 Beri Nama Karyawan sebagai base class*

**

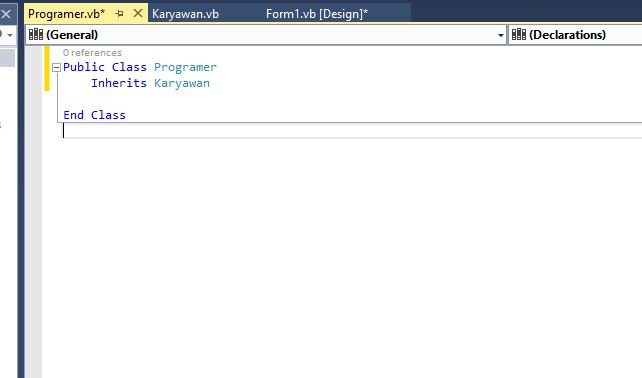
* Maka akan muncul tampilan seperti gambar 14.

*Gambar 14 Tampilan Class Karyawan*

**

* Kemudian buatlah Derived classnya dengan nama programmer, dengan cara sama seperti pada gambar 12 13 14,class karyawan menurunkan sifatnya pada class programmer dengan kata kunci ***inherits*** seperti pada gambar 15.

*Gambar 15 Class Programer menginherits ke Class Karyawan*

**

Kode programnya:

Langkah – langkah dalam menyelesaikan program Overloading, sebagai berikut:

* Buat Code seperti di bawah pada class karyawan

Public Class karyawan

Public \_Nama As String = "Yunita Niken"

Public Function getGaji(ByVal iAbsen As Integer, ByVal dGajiPokok As Double) As Double

Return iAbsen \* dGajiPokok

End Function

End Class

Penjelasan : Membuat variabel \_Nama dengan type data string dan diisi dengan “Yunita Niken” (isi bebas) Kemudian membuat fungsi bernama getGaji yang mempunyai parameter iAbsen dengan type integer dan dGajiPokok dengan type double kemudian fungsi akan mengembalikan nilai iAbsen \* dGajiokok

* Pada class programer tulislah code sebagai berikut

Public Class programmer

Inherits karyawan

Public Overloads Function getGaji(ByVal iAbasen As Integer, ByVal dGajiPokok As Double, ByVal bonusSkill As Double) As Double

Return iAbasen \* dGajiPokok + bonusSkill

EndFunction

EndClass

* Kemudian klik form desaign seperti pada gambar 1 dan tuliskan code dibawah ini pada button dengan klik button 2x seperti pada gambar 6.

Public Class Form1

Private Sub Button1\_Click(sender As Object, e As EventArgs) Handles Button1.Click

Dim obj As New Programer

TextBox1.Text = obj.\_Nama

TextBox2.Text = obj.getGaji(30, 1500000)

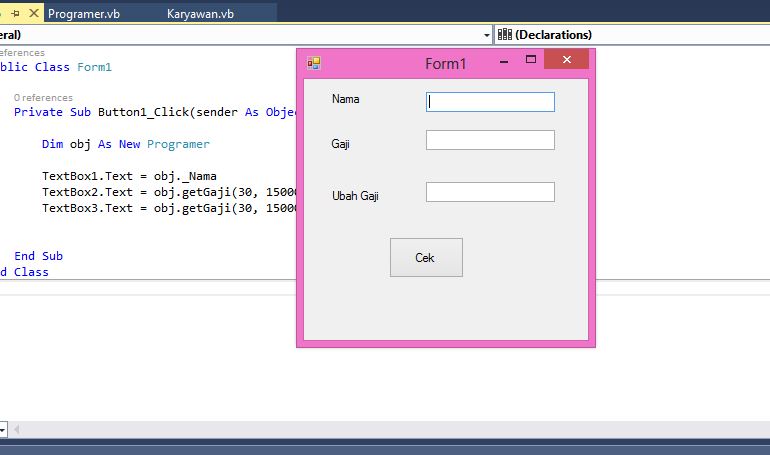
TextBox3.Text = obj.getGaji(30, 1500000, 500000)

End Sub

End Class

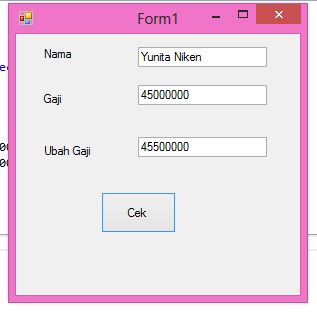
* Selanjutnya jalankan programnya dengan me-running seperti pada gambar 7 dan klik tombol Cek seperti pada gambar 16.

*Gambar 16 Klik Button Cek*

**

* Maka akan muncul tampilan hasil seperti pada gambar 17

*Gambar 17 Hasil*

**

2. Method Overriding

Overriding merupakan suatu keadaan dimana kelas anak dapat mengubah atau bisa kita bilang memodifikasi atau memperluas data dan method pada kelas induk. Keuntungan Overriding : dapat menambahkan sifat / atribut pada kelas induknya

Ciri – ciri Overriding :

* Memiliki nama method sama
* Daftar parameter harus sama
* Nilai yang di return memiliki type yang sama
* Menggunakan kata kunci **overridable** dan **overrides**

Contoh Programnya :

Berikut ini langkah-langkah dan ketentuannya :

* Buatlah new project seperti gambar 8 dengan nama Operriding seperti gambar 9.
* Dan buatlah form yang berisi 2 buah label, 2 buah textbox dan 1 buah button seperti gambar 10.
* Proses menampilkan data menggunakan button dengan nama Cek
* Base classnya adalah Parent
* Derived classnya adalah Child
* Class Parent akan di Override oleh class Child dengan cara memberi kata kunci Overridable pada method yang ada class Parent dan memberi kata kunci Overrides pada class Child yang memiliki method dengan nama yang sama dengan method pada class Parent yang di beri kata kunci Overridable.

Kode programnya:

Langkah – langkah dalam menyelesaikan program Overriding, sebagai berikut:

* Buat Base Class dengan Nama Parent dengan cara seperti pada gambar 12 dan 13. Kemudian tulis code seperti dibawah ini.

Public Class Parent

Public Overridable Function hasil()

Return "Parent"

End Function

End Class

* Kemudian buat Class Child sama seperti gambar 12 dan13 dan beri nama Child kemudian inherits ke class parent dan buat fungsi Overrides seperti pada pada code berikut

Public Class Child

Inherits Parent

Public Overrides Function hasil()

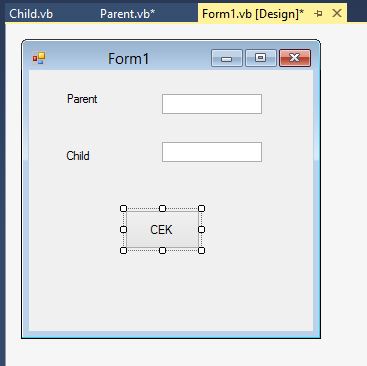
Return "Child"

End Function

End Class

* Sekarang buat form tampilan dengan cara seperti pada gambar 10 dan buat tampilan seperti pada gambar 18 dan buat event click pada tombol CEK (Button 1) dengan cara double klik pada button proses seperti pada gambar 6 serta ketikkan kode – kode program seperti berikut ini.

*Gambar 18 Tampilan Form*

**

Public Class Form1

Private Sub Button1\_Click(sender As Object, e As EventArgs) Handles Button1.Click

Dim objek1 As New Parent

Dim objek2 As New Child

TextBox1.Text = objek1.hasil

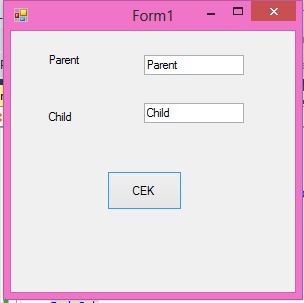
TextBox2.Text = objek2.hasil

End Sub

End Class

* Selanjutnya jalankan programnya dengan running seperti pada gambar 7 dan klik tombol Cek seperti pada gambar 16.
* Maka akan muncul hasil tampilan seperti pada gambar 19.

*Gambar 19 Tampilan Form Hasil*

**

## 5.3 Encapsulation

Adalah pembungkusan, maksudnya di sini semua data yang kita definisikan / bisa juga informasi kita sembunyikan di dalam enkapsulasi tersebut agar tidak dapat dibaca oleh kelas lain yang tidak berhak membacanya. maksudnya secara lebih jelas agar memastikan agar data hanya dapat di ubah oleh object yang mendapat ijin untuk mengubahnya , bila tidak memiliki ijin maka tidak dapat mengubahnya atau membacanya.

Konsep enchapsulasi sangat penting dilakukan untuk menjaga program tersebut.source code enhapsulation biasanya menggunakan keyword “public property”. Enchapsulasi biasa dilakukan menggunakan setter getter. Setter artinya menyetting / mengirimkan nilai ke variabel kemudian getter artinya mengembalikan/mereturn nilai variabel.

Contoh Program Enchapsulation

* Buat new project dan add class beri nama Hewan seperti pada gambar 12 dan 13
* Lalu buat variable dengan akses modifier private
* Kemudian buatlah enchapsulasi.nya dengan codingan sebagai berikut

Public Class Hewan

Private XName As String

Private XAge As Byte

Public Property Name() As String

Get

Return XName

End Get

Set(ByVal value As String)

XName = value

End Set

End Property

Public Property Age() As String

Get

Return XAge

End Get

Set(ByVal value As String)

XAge = value

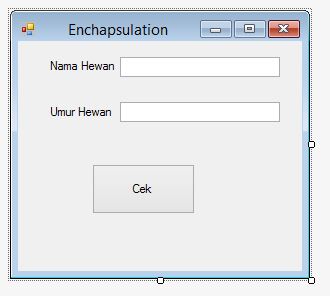
End Set

End Property

End Class

* Buatlah tampilan Desaign di Form desaign dengan cara seperti pada gambar 10 dan buatlah form seperti pada gambar 20
* Ubah nama button dan label sesuai dengan tampilan form pada gambar 20 dengan cara seperti pada gambar 3

*Gambar 20 Tampilan Form Enchapsulation*



* Dan kemudian tuliskan Code sebagai berikut di Class Form dengan cara seperti pada gambar 5

Public Class Form1

Dim obj\_animal As New Hewan

Private Sub Button1\_Click(sender As Object, e As EventArgs) Handles Button1.Click

obj\_animal.Name = TextBox1.Text

obj\_animal.Age = TextBox2.Text

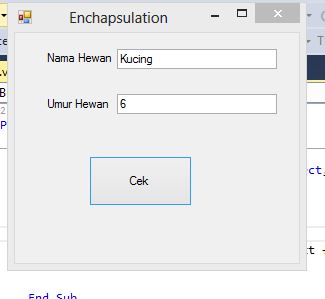
MsgBox("Nama Hewan =" + TextBox1.Text + vbLf + "Umur " + TextBox1.Text + TextBox2.Text + " tahun ")

End Sub

End Class

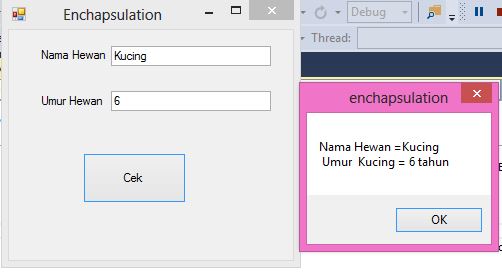
* Coba running seperti pada gambar 7.
* Kemudian isi textbox seperti pada gambar 21.

*Gambar 21 Beri input pada textbox*

**

* Dan klik button Cek seperti pada gambar 16.
* Maka akan muncul message box seperti pada gambar 22.

*Gambar 22 Tampilan Message Box*

**

# **BAB VI**

# **KONEKSI DATABASE DENGAN VB.NET**

Database dapat dikatakan sebuah tempat untuk menyimpan banyak data yang saling berkaitan. Agar data-data tersebut teratur, maka data-data tersebut dikelompokkan ke dalam beberapa tabel yang secara logika merupakan struktur dua dimensi dari baris (row atau record) dan kolom (coloum atau field). Sebuah database akan memiliki minimal satu tabel.

Database merupakan ilmu yang terpisah sendiri karena dalam modul tidak dibahas terlalu mendalam.Oleh karena itu untuk lebih jelasnya kalian dapat membaca pada buku mengenai database atau bisa mengingat kembali pelajaran database yang pernah dipelajari sebelumnya.

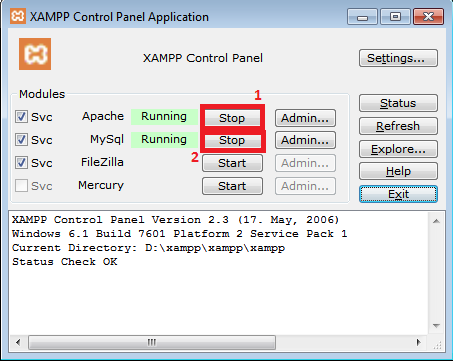
Di dalam .NET ada beberapa cara koneksi pada database, diantaranya yaitu MySqlConnection (Database MySql), SQLConnection (Database SQL Server), OLeDBConnection(Microaft Acces). Dalam modul ini akan menjelaskan cara koneksi menggunakan MySqlConnection yang menggunakan Database MySql.

Berikut : langkah-langkah membuat database pada MySql dan Cara mengkoneksikan dengan vb.net

## 6.1 Langkah-langkah membuat database pada MySql

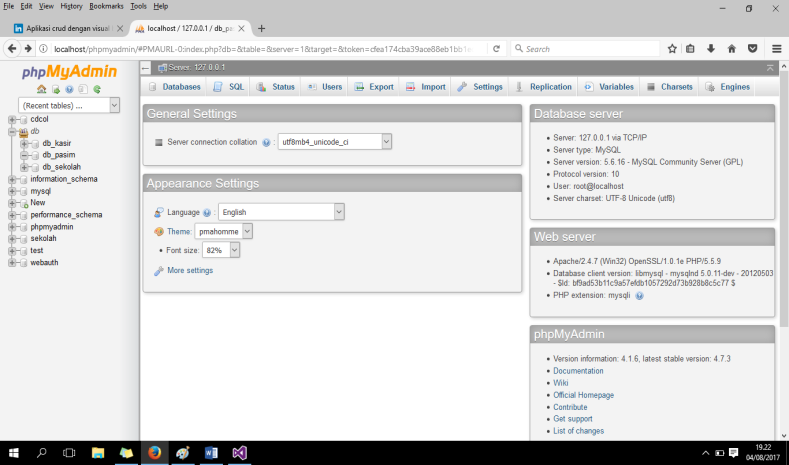
Dalam modul ini menjelaskan bagaimana membuat database tidak menggunakan query, tapi secara wizard atau bisa dikatakan secara langsung , maksudnya dalam membuat sebuah tabel kita tinggal mengisi pada kolom-kolom yang telah disediakan. Untuk lebih jelasnya mari kita ikuti langkah-langkah berikut:

1. Jalankan aplikasi server di pc/laptop (Apache/Xampp/Wampp/ yang lainnya) . Disini kita menggunakan aplikasi server Xampp.



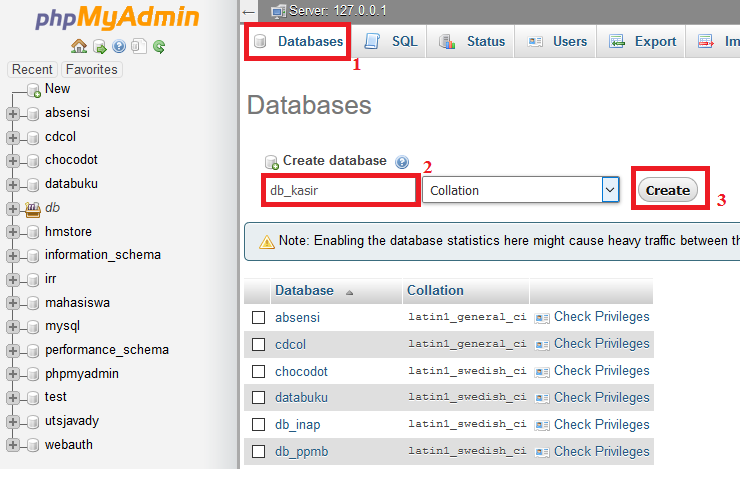
**Gambar.6.1**

1. Buka alamat “localhost/phpmyadmin” di browser (mozila firre fox/google chrome/yang lainnya) untuk membuat database. Maka akan muncul tampilan seperti di bawah ini.



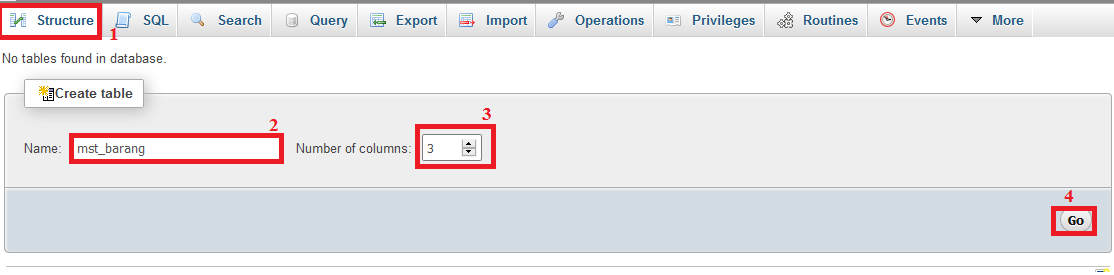
**Gambar.6.2**

1. Buat database dengan nama db\_kasir kemudian tekan create



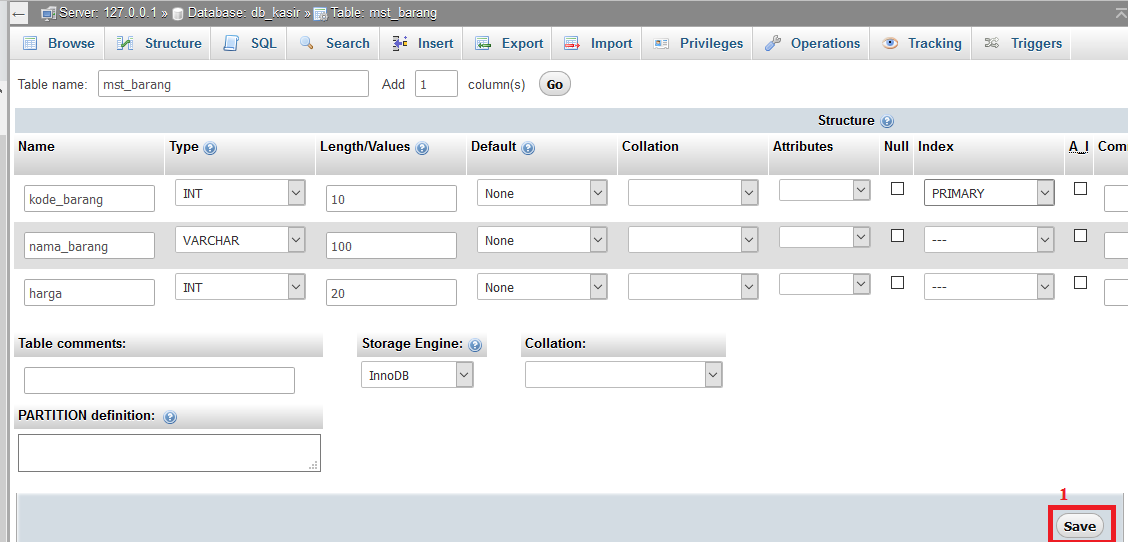
**Gambar.6.3**

1. Buat tabel dengan nama mst\_barang dengan jumlah field 3



**Gambar.6.4**

1. Kemudian desain fieldnya seperti sebagai berikut :



**Gambar. 6.5**

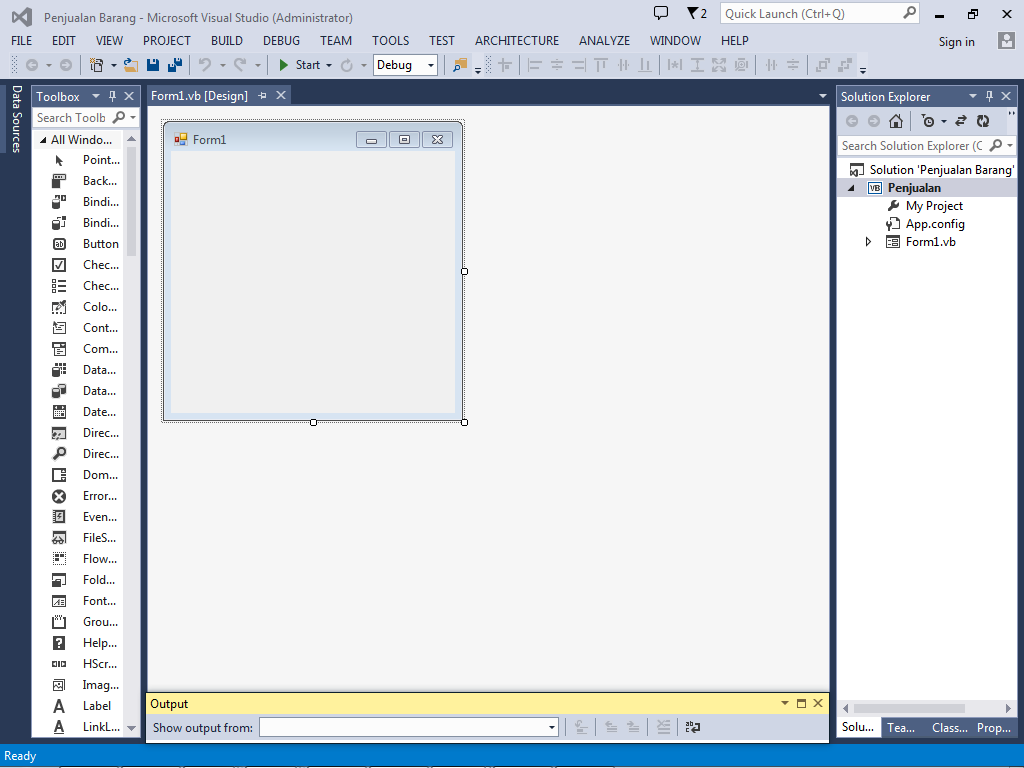
## 6.2 Menghubungkan VB.NET dengan Database MySql

Untuk mengkoneksikan vb.net denga MySql dalam modul ini menggunakan file MySql.Data. Setelah selesai membuat database di database mysql, tahap selanjutnya adalah mengkoneksikan mysql dengan vb.net . Pertama-tama adalah kita harus menginstall MySql Connector. Setelah proses penginstallan selesai, tahap selanjutnya adalah membuat project baru di visual studio 2013.

Dalam modul ini kita akan membuat project aplikasi penjualan, untuk memulai membuat project tersebut langkah-langkahnya adalah :

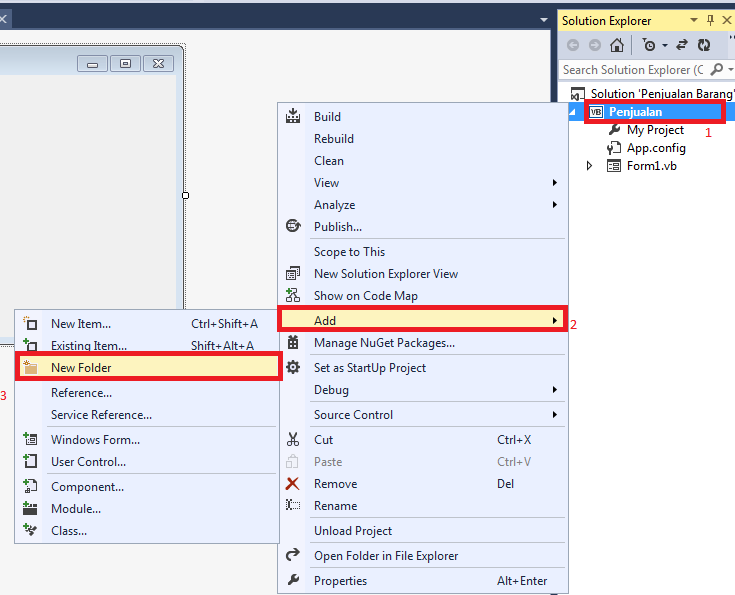
Buka microsoft visual studio 2013, pilih new project 🡪 Pilih dan klik visual basic pada bagian Installed Templates, kemudian klik Windows Form Aplication. Isi nama project yang akan dibuat di **Name**. Dalam modul ini Project ini diberi nama **Aplikasi Penjualan.**

Maka akan tampilan awal adalah sebagai berikut :



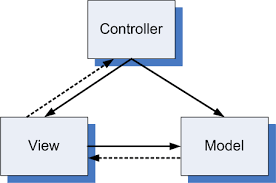
**Gambar.6.6**

Tahap selanjutnya Klik kanan pada nama project yang dibuat (Aplikasi Penjualan), pilih add 🡪 new folder , buat 3 folder, Masing-masing folder diberi nama **Model, View, dan Controller.**



**Gambar.6.7**

Ilustrasi Model, View dan Controller



**Gambar.6.7**

Penjelasan :

1. Model

Model adalah bagian kode program yang menangani query atau database. Jadi isi dari model merupakan bagian (fungsi-fungsi) yang berhubungan langsung dengan database untuk memanipulasi data seperti memasukkan data, pembaruan data, hapus data, dan lain-lain, namun tidak dapat berhubungan langsung dengan bagian view.

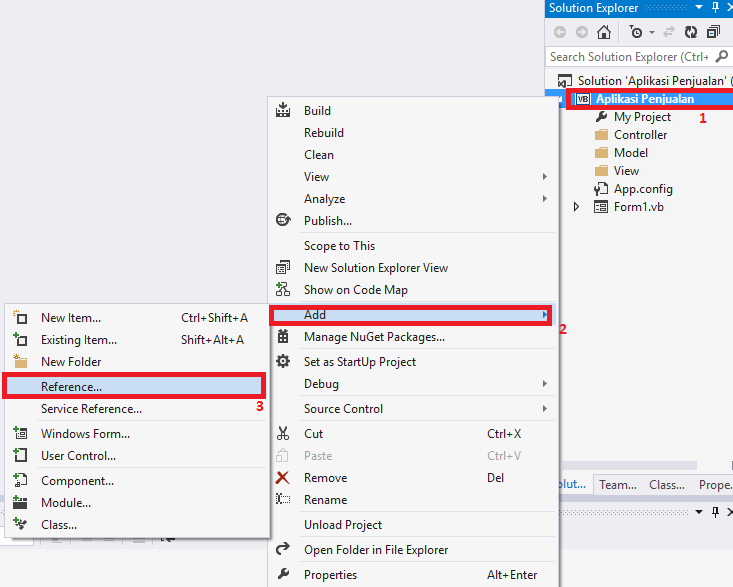
1. View

View adalah bagian kode prgram atau pun komponen  yang mengatur tampilan

1. Controller

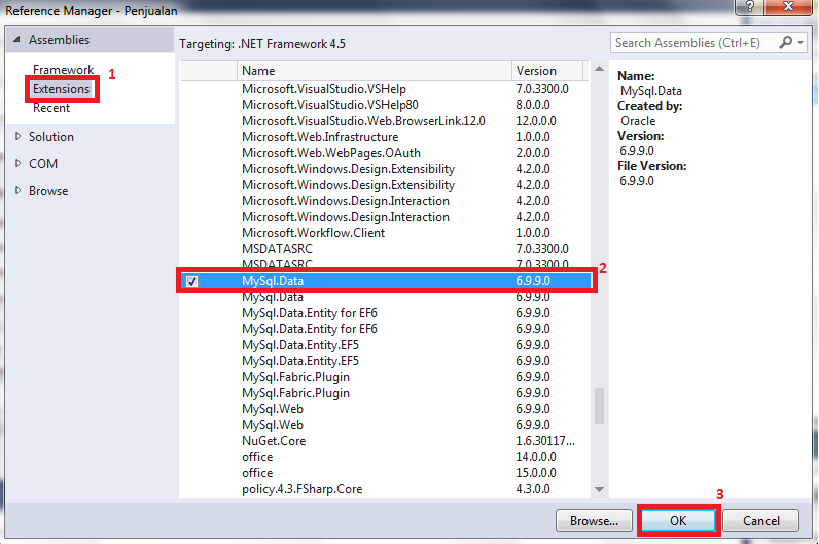
komponen yang digunakan untuk mengontrol semua alur yang diharapkan dari sebuah program. Oleh karena itu, semua logika-logika yang diperlukan berada di komponen Controller.

Untuk melakukan proses MySql kita harus menambahkan reference “MySql.Data ” dengan cara : klik kanan pada nama project(aplikasi penjualan), pilih Add 🡪 References seperti gambar berikut.



**Gambar.6.8**

Setelah itu akan muncul tampilan seperti berikut



**Gambar.6.9**

Kemudian pilih Ektensions 🡪 Mysql.Data

Dengan ditambahkannya name space MySql.Data kita dapat mengakses class-class yang terdapat dalam name space / library tersebut, class-class tersebut diantaranya adalah:

* **MySqlConnection**

untuk menghubungkan dengan database (jembatan antara aplikasi dan database). MySqlConnection berisi semua informasi yang diperlukan untuk membuat koneksi database.

* **MySqlDataAdapter**

untuk membaca data dari database atau menampilkan hasil select (read).

* **MySqlCommand**

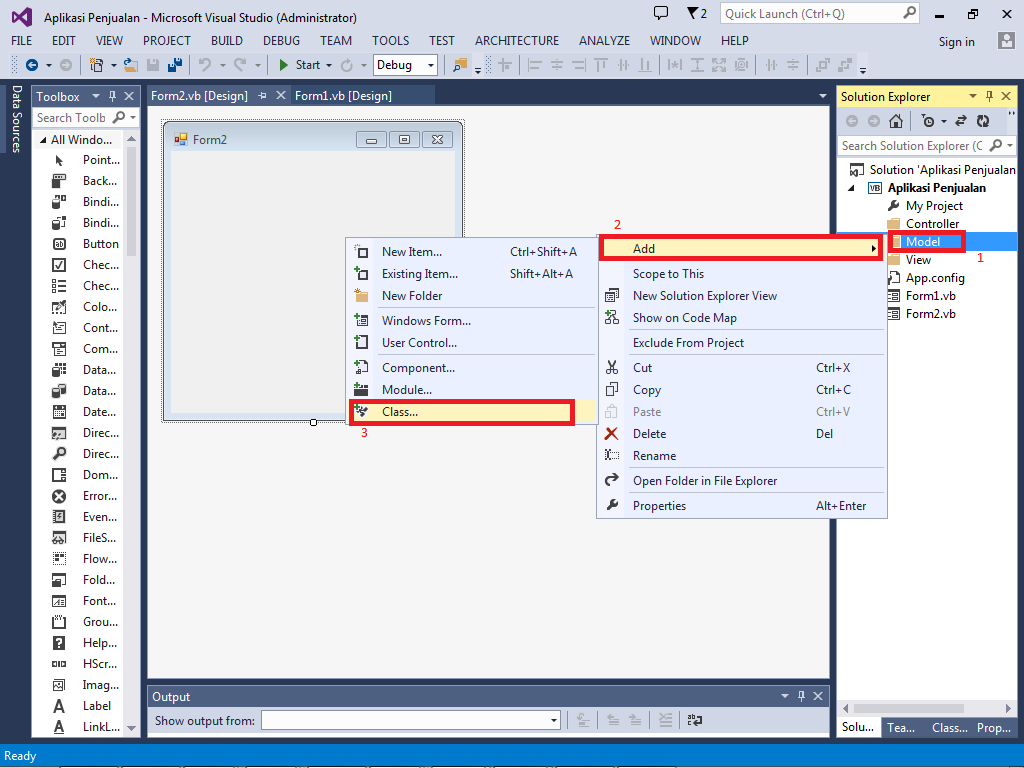
digunakan untuk menjalankan atau mengeksekusi query pada database. Dalam MysqlCommand menyediakan tiga metode yang digunakan untuk mengeksekusi perintah pada database yaitu :

1. ExcecuteNonQuery, digunakan untuk menjalankan perintah yang memiliki nilai yang tidak kembali. Seperti INSERT,UPDATE dan DELETE
2. ExcuteScalar,digunakan untuk menjalankan perintah yang mengembalikan nilai nilai tunggal dari query.
3. ExecuteReade, digunakan untuk menjalankan perintah yang memiliki nilai pengembalian hasil set dengan cara objek dataReader.

* **Data Table**

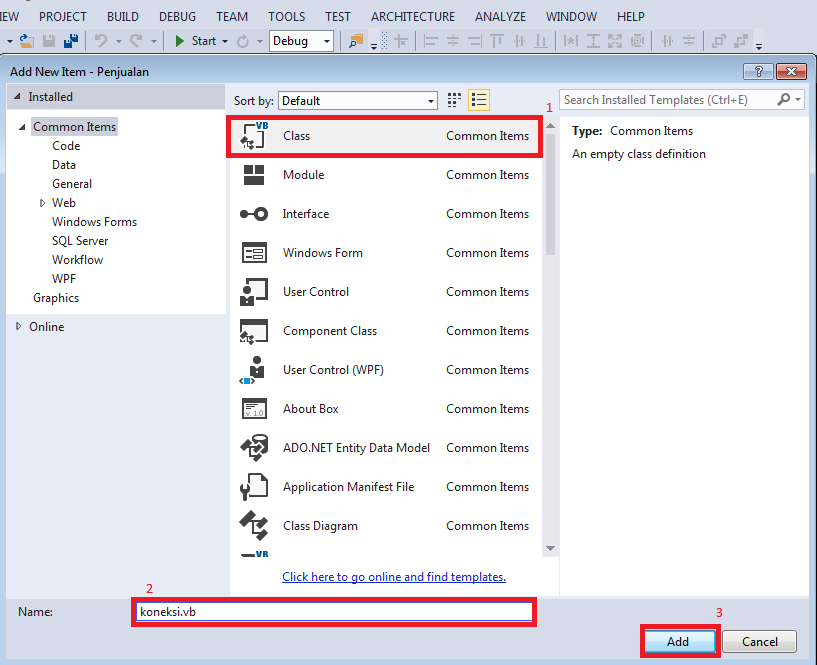
untuk menyimpan data dari database dalam bentuk tabel.

Tahap selanjutnya adalah klik kanan pada **folder Model** yang telah dibuat sebelumnya, kemudian Add Class seperti gamabar 6.10 berikut



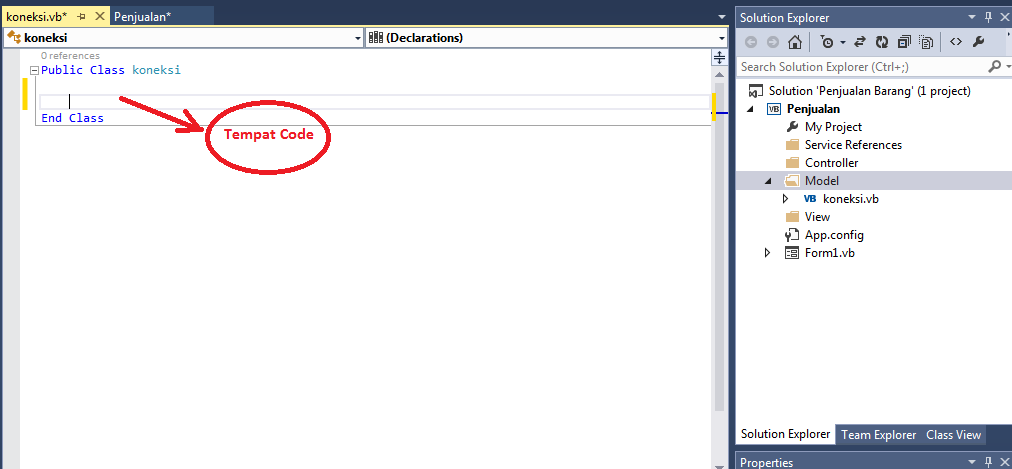
**Gambar.6.10**

Setelah itu akan muncul tampilan seperti berikut. tambahkan satu class pada folder tersebut dan ubah nama class tersebut menjadi koneksi.vb. kemudian klik Add seperti Gambar.6.11



**Gambar.6.11**

Kemudian akan muncul tampilan seperti berikut,



**Gambar.6.12**

Kemudian ketikan code dibawah ini pada class koneksi tersebut:

Dalam melakukan code dalam class ini jangan lupa ditambahkan name space: Imports MySql.Data.MySqlClient. yang di letakan di atas Class koneksi seperti kode berikut.

Dengan ditambahkan name space tersebut kita dapat mengakses clas-class yang ada dalam Mysql.data seperti yang umum digunkan adalah MySqlConnection, MySqlDataAdapter, dan MySqlCommand yang telah dibahas pada halaman sebelumnya.

Imports MySql.Data.MySqlClient

**Name Space**

Public Class koneksi

Public konek As String = "server=localhost;user id=root;database=db\_kasir"

**Objeck dari class MySqlConnection**

Private kon As New MySqlConnection(konek)

Protected cmd\_a As MySqlDataAdapter

Protected cmd\_c As MySqlCommand

Protected Function openKoneksi() As MySqlConnection

**Function openKoneksi**

Try

If kon.State = ConnectionState.Closed Then

kon.Open()

**Try**

**Statement**

**Try**

**Statement**

EndIf

Catch ex As Exception

MsgBox("Error connection in : " + ex.Message, MsgBoxStyle.Exclamation)

EndTry

Return kon

**End Function openKoneksi**

End Function

EndClass

**Penjelasan :**

Function yang bernama **openKoneksi()** diatas dibuat untuk me-return atau mengembalikan nilai koneksi ke database.

statement **If.......** yang berfungsi untuk mengecek koneksi ke database. Apabila koneksi dalam keadaan tertetutup, maka koneksi akan dibuka terlebih dahulu.

Didalam function tadi juga ditambahkan code

**Try statement** yang berfungsi untuk mengecek dan menangani apabila terjadi sebuah kesalahan / eror pada koneksi ke database, dengan adanya try catch ini, ketika terjadi eror, pada saat di running program tidak akan langsung keluar, tapi akan memunculkan informasi yang menyebabkan eror.

Pada class koneksi tersebut kita telah membuat class khusus untuk melakukan koneksi dengan mysql, dalam aplikasi ini class koneksi akan digunakan oleh class lain yang melakukan koneksi ke database.

Pada code diatas ada sebuah variabel khusus untuk menampung nilai koneksinya yaitu **string Koneksi.** Variabel tersebut di isi dengan connection string yang ada pada properties. Connection string tersebut akan muncul secara otomatis ketika telah selesai menambahkan satu database pada server explorer pada tahap sebelumnya.

## 6.3 CRUD (Create, Read, Update , Delete ) pada vb.net dengan MySql

Pada tahap awal kita telah membuat 3 folder, selanjutnya adalah membuat tampilan form yaitu dengan cara klik kanan pada folder view pilih add 🡪 windows form , berinama DataBarang.vb.

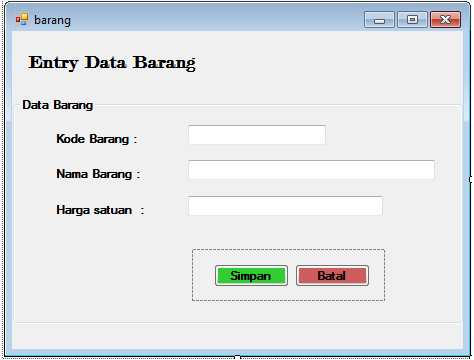
Komponen yang harus ditambahkan adalah 2 Label, 3 Button, 1 Textbox, 1 GroupBox dan 1 Data Gridview.



**Gambar.6.13**

Kemudian tambahkan satu from lagi di dalam folder view dengan nama FormBarang.vb dengan tampilan sebagai berikut :

Komponen yang harus ditambahkan adalah 1 GroupBox, 4 Label, 4 TextBox dan 2 Button.



**Gambar.6.14**

Pada tahap paling awal kita telah membuat satu database yang berisi 1 tabel, untuk menamplikan dan memanipulasi (simpan, edit, hapus, pencarian) data / isi yang ada dalam tabel tersebut pada visual basic tambahkan satu class pada folder controller yang telah dibuat sebelumnya.

Klik kanan pada folder Controller pilih add 🡪 class , beri nama crudBarang.vb.

Code dalam bab ini menerapkan konsep OOP (Object Oriented Programming), dimana konsep OOP tersebut telah dibahas pada bab sebelumnya, jadi dalam bab ini tidak dijelaskan kembali.

Ketikkan code dibawah ini pada class crudBarang:

Jangan lupa tambahkan name space “ Imports MySql.Data.MySqlClient”. pada code dibawah dibuat beberapa function yang mempunyai acces modifier public supaya dapat diakses oleh semua supaya dapat dipakai/ diakses oleh class-class lain yang membutuhkan. Beberpa function tersebut diberi nama viewData(), insertData(), updateData(), deleteData() dan searchData(). Function tersebut memiliki parameter sesuai dengan yang dibutuhkan.

**Name Space**

Imports MySql.Data.MySqlClient

PublicClasscrudBarang

**Inheritance**

Inherits koneksi

keterangan :

Dengan menginherits kelas koneksi pada class crudKategori ini, maka class crudKategori ini akan memiliki sifat yang dimiliki oleh class koneksi. Dengan itu, class crud kategori ini dapat mengakses apa yang dimiliki oleh class koneksi selama data tersebut bersifat public atau protected.

‘Dalam code dibawah di gunkan try statement untuk penanganan erornya. Untuk penjelasan bisa dilihat di halaman dan bab sebelumnya.

' untuk menampilkan data yang ada di tabel mst\_barang

Public Function viewData() As DataTable

Dim hasil As NewDataTable 🡪 untuk menampilkan data dari database dalam bentuk tabel.

Dim \_query AsString = "SELECT \* FROM mst\_barang"

Try

cmd\_a = NewMySqlDataAdapter(\_query, openKoneksi()) INI JUGA

**penjelasan :**

**untuk membaca data dari database atau menaplikan hasil select (read). Data yang di ambil adalah data yang berasal dari variabel \_query. Dalam variabel ini datanya adalah semua data yang ada dalam tabel mst\_barang.**

cmd\_a.Fill(hasil)

Catch ex AsException

MsgBox("Error : " + ex.Message, MsgBoxStyle.Exclamation)

EndTry

Return hasil

EndFunction

‘untuk tambah data

PublicFunction insertData(ByVal kode AsString, ByVal nama\_kat AsString, ByVal nama\_brg AsString, ByVal hrg AsInteger) AsBoolean

Dim hasil = False

Dim \_query AsString = "insert into mst\_barang values(@kode\_barang, @nama\_barang,@harga)"

Try

**PENJELASAN :**

**Mysqldbtype** 🡪 digunakan untuk menyatakan tipe data dari field yang ada di dalam tabel(dalam contoh di bawah kode barang yang bertipe varchar)

cmd\_c = NewMySqlCommand(\_query, openKoneksi())

cmd\_c.Parameters.Add("@kode\_barang", MySqlDbType.VarChar).Value = kode

cmd\_c.Parameters.Add("@nama\_barang",MySqlDbType.VarChar).Value = nama\_brg

cmd\_c.Parameters.Add("@harga",MySqlDbType.Int32).Value=hrg

cmd\_c.ExecuteNonQuery() ‘INI JUGA HEHE

hasil = True

Catch ex AsException

MsgBox("Error insert data barang in : "+ ex.Message, MsgBoxStyle.Exclamation)

EndTry

Return hasil

EndFunction

**Penjelasan :**

**Executenonequery() adalah perintah untuk mengeksekusi script SQL.**

**Executenonequery() adalah Eksekusi perintah SQL (Query) yang tidak menghasilkan data kembalian. sebenarnya, nilai kembalian dari fungsi ini adalah banyaknya record yang telah di rubah, contoh delete, insert, dan update.**

‘untuk edit / update data

PublicFunction updateData(ByVal kode AsString, ByVal nama\_kat AsString, ByVal nama\_brg AsString, ByVal hrg AsInteger) AsBoolean

Dim hasil = False

Dim query AsString = "update mst\_barang set nama\_barang=@nama\_barang, harga =@harga where kode\_barang='" + kode + "'"

Try

cmd\_c = NewMySqlCommand(query, openKoneksi())

cmd\_c.Parameters.Add("@nama\_barang",MySqlDbType.VarChar).Value = nama\_brg

cmd\_c.Parameters.Add("@harga",MySqlDbType.Int32).Value=hrg

cmd\_c.ExecuteNonQuery()

hasil = True

Catch ex AsException

MsgBox("salah ubah data barang di "+ex.Message, MsgBoxStyle.Exclamation)

EndTry

Return hasil

EndFunction

‘untuk hapus / delete data

‘untuk menghapus data dalam program ini diambil berdasarkan kode barangnya.

PublicFunction deleteData(ByVal kode AsString) AsInteger

Dim hasil = False

Dim \_query AsString = "delete from mst\_barang where kode\_barang =@kode\_barang"

Try

cmd\_c = NewMySqlCommand(\_query, openKoneksi())

cmd\_c.Parameters.Add("@kode\_barang", MySqlDbType.VarChar).Value = kode

cmd\_c.ExecuteNonQuery()

hasil = True

Catch ex AsException

MsgBox("Error delete data in : " + ex.Message, MsgBoxStyle.Exclamation)

EndTry

Return hasil

EndFunction

‘untuk pencarian data

‘Untuk pencarian data disini diambil berdasarkan kode barang dan nama barang

PublicFunction searchData(ByVal cari AsString) AsDataTable

Dim dt = NewDataTable()

Dim query AsString ="select kode\_barang, nama\_barang, harga from mst\_barang where kode\_barang like '%" + cari + "%' or nama\_barang like '%" + cari + "%' "

Try

cmd\_a = NewMySqlDataAdapter(query, openKoneksi())

cmd\_a.Fill(dt)

Catch ex AsException

MsgBox("salah tampil data barang " + ex.Message, MsgBoxStyle.Exclamation)

EndTry

Return dt

EndFunction

**Tahap selanjutnya adalah buka kembali FormBarang.vb yang sebelumnya telah dibuat, kemudian ketikkan code berikut:**

PublicClass FormBarang

Private \_kode AsString

Private \_aksi AsString

PublicProperty kode AsString

Get

Return \_kode

EndGet

**Enkapsulatsi/ Encapsulation**

Set(ByVal value AsString)

\_kode = value

EndSet

EndProperty

PublicProperty aksi AsString

Get

Return \_aksi

EndGet

Set(ByVal value AsString)

\_aksi = value

EndSet

EndProperty

**PENJELASAN :**

**Pada code di atas dibuat dua atribut yang bertipe string yang memiliki acces modifier Private (untuk penjelasan acces modifeir lihat pada bab sebelumnya) atribut tersebut .**

**Tahap selanjutnya adalah membuat suatu prosedur untuk memudahkan proses ketika kondisi tambah data atau ubah data dengan nama isiTextbox (nama bisa bebas).**

Untuk penjelasan prosedur sama seperti pada bahasa c, jadi dalam modul ini tidak diterangk kembali. Dalam modul ini hanya diberikan contoh membuat prosedur menurut bahasa visual basic (pendeklarasiannya). Bedanya adalah cara deklarasi atau implementasinya. Pendeklaran suatu prosedur di vb.net yaitu dengan cara :

Sub *nama\_prosedur*( *parameter* )

🡪 pernyataan

End sub

Sub isiTextbox()

If (\_aksi = "Tambah") Then

GroupBox1.Text = "Tambah Data"

tbx\_kode.ReadOnly = False

tbx\_kode.Clear()

tbx\_namabarang.Clear()

tbx\_harga.Clear()

ElseIf (\_aksi = "Ubah") Then

GroupBox1.Text = "Ubah Data"

Dim ck AsNewcrudBarang

tbx\_kode.ReadOnly = True

Dim dt AsNewDataTable

dt = ck.searchData(\_kode)

tbx\_kode.Text = dt.Rows(0).Item(0)

tbx\_namabarang.Text = dt.Rows(0).Item(1)

tbx\_harga.Text = dt.Rows(0).Item(2)

EndIf

EndSub

Selanjutnya membuat constructor class form Barang berfungsi untuk membentuk class form mau dibikin seperti apa, apakah mau dibentuk form ubah atau form tambah jadi proses ubah dan tambah tidak perlu membuat dua form.

SubNew(ByVal aksi AsString)

InitializeComponent()

**Constructorr**

\_aksi = aksi

EndSub

PrivateSub FormBarang\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) HandlesMyBase.Load

isiTextbox() ‘’prosedur

EndSub

**Penjelasan:**

**ketika form barang dipanggil / dijalankan, maka akan langsung menjalankan program sesuai dengan isi dari prosedur isTextbox().**

PrivateSub btn\_simpan\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btn\_simpan.Click

If (String.IsNullOrEmpty(tbx\_kode.Text.Trim())) Then

MsgBox("masukkan kode barangnya ", MsgBoxStyle.Exclamation)

ElseIf (String.IsNullOrEmpty(tbx\_namabarang.Text.Trim())) Then

MsgBox("masukkan nama barangnya", MsgBoxStyle.Exclamation)

ElseIf (String.IsNullOrEmpty(tbx\_harga.Text.Trim())) Then

MsgBox("masukkan harga", MsgBoxStyle.Exclamation)

Else

If (\_aksi = "Tambah") Then

Dim crk AsNewcrudBarang

crk.insertData(tbx\_kode.Text.Trim(), tbx\_kategori.Text.Trim(), tbx\_namabarang.Text.Trim(), tbx\_harga.Text.Trim())

MsgBox("Data berhasil disimpan")

ElseIf (\_aksi = "Ubah") Then

Dim crk AsNewcrudBarang

crk.updateData(tbx\_kode.Text.Trim(),tbx\_namabarang.Text.Trim(), tbx\_harga.Text.Trim())

MsgBox("Data berhasil diubah")

EndIf

**Objeck dari class Barang**

Dim frBrg As New DataBarang

frBrg.refreshForm() 🡪 prosedur refreshForm() beraasal dari form DataBarang.vb

Me.Close() 🡪 untuk keluar dari formBarang.vb

EndIf

EndSub

PrivateSub btn\_batal\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btn\_batal.Click

Dim fk AsNewDataBarang

Me.Close()🡪 untuk keluar dari formBarang.vb

End Sub

End Class

Tahap selanjutnya adalah Buka kembali form DataBarang yang telah dibuat, kemudian tekan F7 dan tambahkan code berikut :

PublicClass DataBarang

'membuat suatu prosedur supaya dapat langsung ngeselect (menampilkan data) ketika insert, update dan delete

Sub refreshForm()

Dim ck AsNewcrudBarang

DataGridView1.DataSource = ck.viewData()

EndSub

Buka kembali form Data Barang, selanjutnya klik 2 kali pada form DataBarang dan ketikkan code dibawah ini:

PrivateSub DataBarang\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) HandlesMyBase.Load

'pertama kali dijalankan/ di running maka akan menampilkan

'apa yang ada dalam sub refreshForm()

refreshForm()

EndSub

**penjelasan :**

**prosedur refrehForm() ini akan menampilkan seluruh data yang ada dalam database.**

PrivateSub btn\_tambah\_Click(ByVal sender As

System.Object, ByVal e As System.EventArgs) Handles

btn\_tambah.Click

Dim frBrg As New FormBarang("Tambah")

frBrg.ShowDialog()

refreshForm()

EndSub

PrivateSub btn\_edit\_Click(ByVal sender As

System.Object, ByVal e As System.EventArgs) Handles btn\_edit.Click

Dim frBrg AsNewFormBarang("Ubah")

frBrg.kode =

DataGridView1.CurrentRow.Cells(0).Value.ToString()

frBrg.ShowDialog()

refreshForm()

EndSub

Kemudian double klik pada textbox pencarian yang ada pada form DataBarang yang telah dibuat, dan ketikkan code di bawah ini. Textbox ini digunakan untuk melakukan searching / pencarian, dimana pencarian dalam aplikasi ini berdasarkan kode barang dan nama barang.

PrivateSub tbx\_cari\_TextChanged(ByVal sender As

System.Object, ByVal e As System.EventArgs) Handles tbx\_cari.TextChanged

If(String.IsNullOrEmpty(tbx\_cari.Text.Trim()))Then

refreshForm()

Else

Dim ck AsNewcrudBarang

DataGridView1.DataSource =

ck.searchData(tbx\_cari.Text.Trim())

EndIf

EndSub

tahap selanjutnya double klik pada button refresh yang tadi dibuat dan ketikkan code dibawah ini. Button refresh ini digunakan untuk merefresh kembali data yang sebelumnya telah diubah/ ditambah maupun dicari.

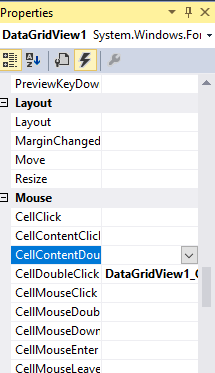
PrivateSub btn\_refresh\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btn\_refresh.Click

refreshForm()

tbx\_cari.Clear()

EndSub

Tambahkan Event DataGridView1\_CellDoubleClick yang ada dalam properties window dengan cara seperti gambar dibawah ini :



**Gambar.6.15**

Event tersebut digunakan untuk melukan tindakan untuk menghapus data. Jadi untuk menghapus data dalam aplikasi ini cukup dengan cara double klik pada cell maupun kolom yang akan dihapus.

ketikkan code dibawah ini :

PrivateSub DataGridView1\_CellDoubleClick(ByVal sender As System.Object, ByVal e As

System.Windows.Forms.DataGridViewCellEventArgs) Handles DataGridView1.CellDoubleClick

'membuat variabel untuk mengambil nilai didatagridview pada indek kolom ke-0

Dim kode AsString =

DataGridView1.CurrentRow.Cells(0).Value.ToString()

Dim dr AsDialogResult = MsgBox("apakah yakin ingin menghapus data ini ?",MsgBoxStyle.OkCancel)

Memunculkan kotak dialog ok dan cancel

If (dr = DialogResult.OK) Then

Dim ck AsNewcrudBarang

If (ck.deleteData(kode) = True) Then

MsgBox("Data berhasil dihapus")

refreshForm()

EndIf

Else

MsgBox("Data Batal dihapus")

EndIf

EndSub

Untuk mengetahui hasilnya, silahkan coba di running dengan cara **klik start debuging yang ada pada tool bar atau dengan cara tekan tombol F5 pada keyboard.**

**Daftar Isi**

[**BAB I** 1](#_Toc490493268)

[**MENGENAL .NET FRAMEWORK** 1](#_Toc490493269)

[1.1 Penjelasan Tentang .Net Framework 1](#_Toc490493270)

[1.2 .Net Enterprise Server 3](#_Toc490493271)

[**BAB II** 10](#_Toc490493272)

[**DASAR – DASAR OOP (OBJECT ORIENTED PROGRAMMING)** 10](#_Toc490493273)

[2.1 Pengertian OOP 10](#_Toc490493274)

[2.2 Kelebihan dan Kekurangan konsep OOP 11](#_Toc490493275)

[**BAB III** 14](#_Toc490493276)

[**KOMPONEN – KOMPONEN OOP** 14](#_Toc490493277)

[3.1 Class 14](#_Toc490493278)

[3.2 Objek 18](#_Toc490493279)

[3.3 Method 19](#_Toc490493280)

[3.4 Attribute 20](#_Toc490493281)

[**BAB IV** 22](#_Toc490493282)

[**PENANGANAN ERROR (ERROR HANDLING)** 22](#_Toc490493283)

[4.1 Try – Catch 22](#_Toc490493284)

[**BAB V** 29](#_Toc490493285)

[**FITUR – FITUR DALAM OOP** 29](#_Toc490493286)

[5.1 Inheritance 29](#_Toc490493287)

[5.2 Polymorphism 37](#_Toc490493288)

[5.3 Encapsulation 48](#_Toc490493289)

[**BAB VI** 52](#_Toc490493290)

[**KONEKSI DATABASE DENGAN VB.NET** 52](#_Toc490493291)

[6.1 Langkah-langkah membuat database pada MySql 52](#_Toc490493292)

[6.2 Menghubungkan VB.NET dengan Database MySql 55](#_Toc490493293)

[6.3 CRUD (Create, Read, Update , Delete ) pada vb.net dengan MySql 64](#_Toc490493294)