

TELKOM SCHOOLS

**BIDANG KEAHLIAN: TEKNOLOGI INFORMASI DAN
KOMUNIKASI
PROGRAM KEAHLIAN : TEKNIK KOMPUTER DAN INFOMATIKA
PAKET KEAHLIAN: REKAYASA PERANGKAT LUNAK
MATA PELAJARAN: PEMROGRAMAN DESKTOP XI
SEMESTER : GANJIL**

**Nomor Dokumen : RPL C3-19-2015
Versi : 1.0
Tanggal : 31 Desember 2015**

Diterbitkan oleh:
TELKOM SCHOOLS

Directorate of Primary & Secondary Education (PSE)
Telkom Education Foundation
Gedung Sate -- West Wing @ 2nd Floor -- Room #7
Jl. Cisanggarung No.2 Bandung 40115, West Java Indonesia
t: ++62 22 7272 078, 22 7106 043 ext 134, f: ++62 22 7102 4444

@Hak Cipta TELKOM SCHOOLS 2015
Dilarang memperbanyak dokumen ini dalam bentuk apapun, sebagian atau keseluruhan tanpa ijin tertulis dari penerbit.

TELKOM SCHOOLS

**BIDANG KEAHLIAN: TEKNOLOGI INFORMASI DAN
KOMUNIKASI
PROGRAM KEAHLIAN : TEKNIK KOMPUTER DAN INFOMATIKA
PAKET KEAHLIAN: REKAYASA PERANGKAT LUNAK
MATA PELAJARAN: PEMROGRAMAN DESKTOP XI
SEMESTER : GANJIL**

**Nomor Dokumen : RPL C2-19-2015
Versi : 1.0
Tanggal : 10 Nopember 2015**

Dengan diberlakukannya Buku Ajar ini, maka Buku Ajar versi sebelumnya dinyatakan tidak berlaku lagi. (*)

**Ditetapkan di : Bandung
Pada tanggal : 31 Desember 2015**

**Director of PSE
Telkom Education Foundation**

NIK.

DAFTAR ISI

DAFTAR ISI	III
DAFTAR TABEL	V
DAFTAR GAMBAR	VI
KEGIATAN BELAJAR 1	1
Pengenalan Pengembangan Aplikasi Desktop	1
A. TUJUAN PEMBELAJARAN	1
B. DESKRIPSI MATERI	1
1. <i>Teknologi Desktop</i>	2
2. <i>Instalasi Aplikasi Program Desktop</i>	9
C. UJI KEMAMPUAN	15
KEGIATAN BELAJAR 2	17
Desain Aplikasi Desktop	17
A. TUJUAN PEMBELAJARAN	17
B. DESKRIPSI MATERI	17
1. <i>Pembuatan Form</i>	18
2. <i>Menambah Kontrol pada Form</i>	26
3. <i>Event</i>	29
4. <i>Properti DialogResult</i>	31
5. <i>DataGridView</i>	32
C. UJI KEMAMPUAN	34
KEGIATAN BELAJAR 3	35
Desain Aplikasi Multiwindow	35
A. TUJUAN PEMBELAJARAN	35
B. DESKRIPSI MATERI	35
1. <i>Prinsip Desain Multiwindow</i>	36
2. <i>Navigasi antar window</i>	36
3. <i>Multiwindow dengan Tabulasi</i>	39
4. <i>Navigasi Splash dengan Timer</i>	42
C. UJI KEMAMPUAN	45
KEGIATAN BELAJAR 4	46
Pengkodean Alur Program	46
A. TUJUAN PEMBELAJARAN	46
B. DESKRIPSI MATERI	46
1. <i>Tipe Data, Variabel, Dan Operator</i>	47
2. <i>Array</i>	51
3. <i>Struktur Kontrol Percabangan</i>	51
4. <i>Struktur Kontrol Perulangan</i>	53
C. UJI KEMAMPUAN	55
KEGIATAN BELAJAR 5	57
Pengolahan Input User	57
A. TUJUAN PEMBELAJARAN	57
B. DESKRIPSI MATERI	57

1. <i>Konversi Data</i>	58
2. <i>Pengelolaan Input User Pada Operasi Aritmatika dan Logika</i>	58
3. <i>Pengelolaan Input User Pada Operasi String</i>	61
C. Uji KEMAMPUAN	70
DAFTAR PUSTAKA	71

DAFTAR TABEL

TABLE 1 CONTOH PROGRAM KONTROL PADA FORM	26
TABLE 2 TIPE DATA NUMERIK	47
TABLE 3 TIPE DATA NON-NUMERIK	48
TABLE 4 DEKLARASI EMPLISIT	49
TABLE 5 OPERATOR.....	49
TABLE 6 OPERATOR PERBANDINGAN.....	50
TABLE 7 OPERATOR LOGIKA.....	50
TABLE 8 OPERATOR PENUGASAN	50
TABLE 9 OPERATOR PENGgabungan.....	50
TABLE 10 KONVERSI DATA	58

DAFTAR GAMBAR

GAMBAR 1 PENGENALAN APLIKASI DESKTOP	1
GAMBAR 2 MACAM-MACAM APLIKASI PROGRAM DESKTOP	2
GAMBAR 3 SEJARAH VISUAL BASIC.....	2
GAMBAR 4 SEJARAH DELPHI.....	3
GAMBAR 5 SEJARAH TURBO C++	4
GAMBAR 6 SEJARAH BORLAND C++	4
GAMBAR 7 SEJARAH XCODE.....	5
GAMBAR 8 DAEMON TOOLS LITE	10
GAMBAR 9 PILIHAN INSTAL VS2010	11
GAMBAR 10 PERSETUJUAN PEMBERIAN INFORMASI VS2010.....	12
GAMBAR 11 FITUR INSTALASI VS2010	13
GAMBAR 12 PROSES INSTALASI VS2010.....	14
GAMBAR 13 SUKSES INSTALASI VS2010.....	15
GAMBAR 14 STIMULUS DESAIN APLIKASI DESKTOP.....	17
GAMBAR 15 FORM NEW PROJECT VS.....	18
GAMBAR 16 TOOLBOX.....	20
GAMBAR 17 SEJARAH DELPHI.....	22
GAMBAR 18 FILE NEW PROJECT VB	23
GAMBAR 19 NEW PROJECT VB.....	24
GAMBAR 20 PROPERTIS BAR VB.....	25
GAMBAR 21 ERROR MESSAGE VB	26
GAMBAR 22 CONTOH FORM SEDERHANA	27
GAMBAR 23 CONTOH FORM PROGRAM EVENT.....	29
GAMBAR 24 CONTOH FORM MENU	30
GAMBAR 25 CONTOH PROGRAM DIALOGRESULT	32
GAMBAR 26 CONTOH GRIDVIEW	32
GAMBAR 27 CONTOH MENAMBAH KOLOM PADA GRIDVIEW	33
GAMBAR 28 CONTOH HASIL PENGISIAN GRIDVIEW	33
GAMBAR 29 STIMULUS APLIKASI MULTIWINDOW	35
GAMBAR 30 MENAMBAH FORM.....	36
GAMBAR 31 PILIHAN ADD WINDOWS FORM.....	37
GAMBAR 32 HALAMAN 1 MULTI FORM.....	37
GAMBAR 33 HALAMAN 2 MULTI FORM.....	38

GAMBAR 34 MULTIWINDOW DENGAN TABULASI	39
GAMBAR 35 PROPERTIS TABCONTROL	39
GAMBAR 36 CONTOH MULTIWINDOW DENGAN TABULASI	40
GAMBAR 37 CONTOH MENGHILANGKAN TULISAN TABPAGE	41
GAMBAR 38 PENAMBAHAN MENU PADA TABPAGE	41
GAMBAR 39 HASIL MULTIWINDOW DENGAN MODIFIKASI TABPAGE	42
GAMBAR 40 FORM MENAMBAH SPLASHSCREEN	43
GAMBAR 41 NEW PROJECT SPLASHSCREEN.....	43
GAMBAR 42 FORM SPLASHSCREEN TERMODIFIKASI	44
GAMBAR 43 STIMULUS PENGKODEAN ALUR PROGRAM	46
GAMBAR 44 CONTOH UJI KEMAMPUAN PENGKODEAN ALUR PROGRAM	56
GAMBAR 45 STIMULUS PENGOLAHAN INPUT USER	57
GAMBAR 46 CONTOH FORM MENGHITUNG LUAS	61



KEGIATAN BELAJAR 1

PENGENALAN PENGEMBANGAN APLIKASI DESKTOP

A. Tujuan Pembelajaran

Setelah melakukan kegiatan pembelajaran siswa dapat :

1. Memahami teknologi pengembangan
2. Memahami Instalasi Program desktop
3. Menyajikan teknologi pengembangan Teknologi desktop
4. Melakukan Instalasi Program desktop pertama

B. Deskripsi Materi

Pilih	Nama	SN	No Mesin	Koneksi	IP Address / URL	Port / Password WEB	Tipe	Password	Serial Port	Baud Rate	Auto Download	Layer
<input type="checkbox"/>	1RA	0331133400080	1	Ethernet	192.168.50.102	4370		0				TFT
<input type="checkbox"/>	1RB	03311334001368	1	Ethernet	192.168.50.101	4370		0				TFT
<input type="checkbox"/>	1RC	0331133400082	1	Ethernet	192.168.50.124	4370		0				TFT
<input type="checkbox"/>	1TA	3315473281252	1	Ethernet	192.168.50.103	4370		0				TFT
<input type="checkbox"/>	1TB	3315473280626	1	Ethernet	192.168.50.104	4370		0				TFT
<input type="checkbox"/>	2RA	0331133400276	1	Ethernet	192.168.50.105	4370		0				TFT
<input type="checkbox"/>	2RB	0331133400520	1	Ethernet	192.168.50.106	4370		0				TFT
<input type="checkbox"/>	2TA	0331133400497	1	Ethernet	192.168.50.107	4370		0				TFT
<input type="checkbox"/>	2TB	0331133400544	1	Ethernet	192.168.50.108	4370		0				TFT
<input type="checkbox"/>	3RA	0331133400119	1	Ethernet	192.168.50.121	4370		0				TFT
<input type="checkbox"/>	3RB	0331133400056	1	Ethernet	192.168.50.110	4370		0				TFT
<input type="checkbox"/>	3RC	0331133400195	1	Ethernet	192.168.50.109	4370		0				TFT
<input type="checkbox"/>	3TA	0331133400173	1	Ethernet	192.168.50.111	4370		0				TFT
<input type="checkbox"/>	3TB	0331133400492	1	Ethernet	192.168.50.112	4370		0				TFT
<input type="checkbox"/>	unused	0331133400784	1	Ethernet	192.168.50.124	4370		0				TFT

Gambar 1 Pengenalan Aplikasi Desktop

Tugas 1

Amati Gambar diatas

Aplikasi apa yang terdapat pada gambar?

Apakah basis Aplikasi tersebut?

Ceritakan pendapat anda, sampaikan dalam kegiatan pembelajaran!

1. Teknologi Desktop

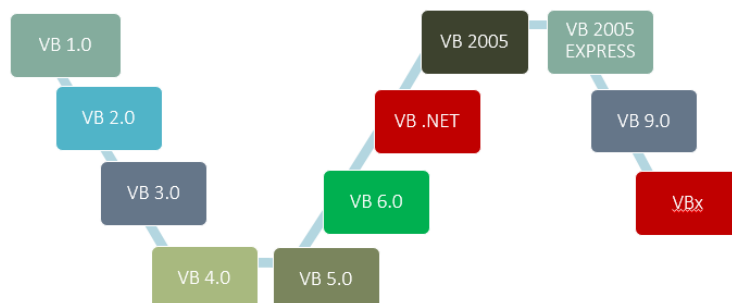
Sejarah Aplikasi Program Desktop



Gambar 2 Macam-macam Aplikasi Program Desktop

Microsoft Visual Basic (sering disingkat sebagai VB saja) merupakan sebuah bahasa pemrograman yang menawarkan Integrated Development Environment (IDE) visual untuk membuat program perangkat lunak berbasis sistem operasi Microsoft Windows dengan menggunakan model pemrograman COM.

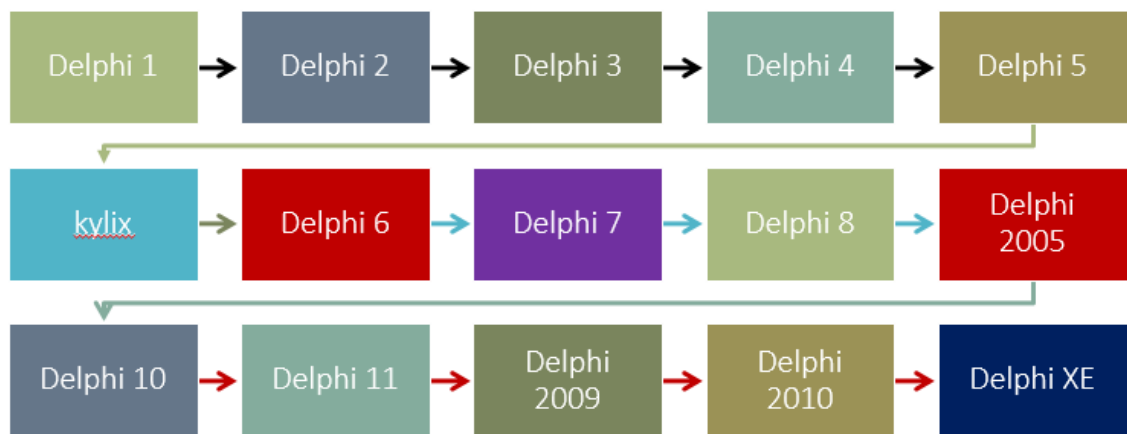
VERSI VISUAL BASIC



Gambar 3 Sejarah Visual Basic

Delphi adalah sebuah Integrated Development Environment (IDE) untuk mengembangkan aplikasi konsol, desktop, web, ataupun perangkat mobile. Bahasa Delphi, atau dikenal pula sebagai object pascal pada mulanya ditujukan hanya untuk Microsoft Windows, namun saat ini telah mampu digunakan untuk mengembangkan aplikasi untuk Mac OS X, iOS, Microsoft .NETframework

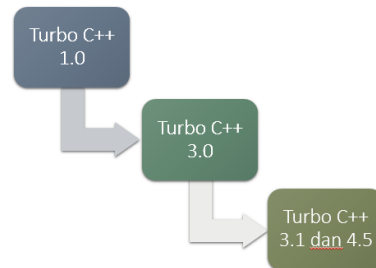
VERSI DELPHI



Gambar 4 Sejarah Delphi

Turbo C++ merupakan kompiler C++ dengan IDE yang terintegrasi yang dikembangkan oleh Borland, terkenal karena kecepatannya dalam kompilasi dan linking. Produk ini merupakan bagian dari keluarga kompiler borland yang sangat populer termasuk Turbo Pascal, Turbo Basic, Turbo Prolog, dan Turbo C.

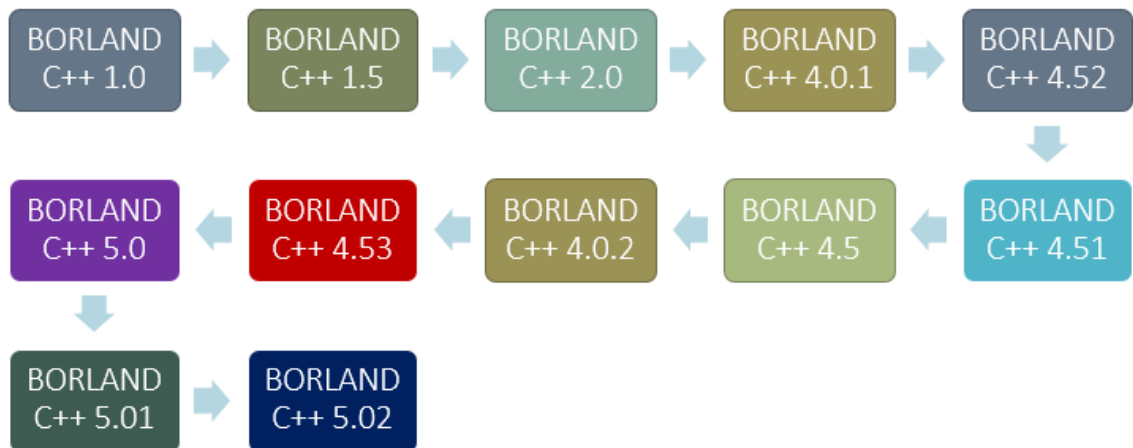
VERSI TURBO C++



Gambar 5 Sejarah Turbo C++

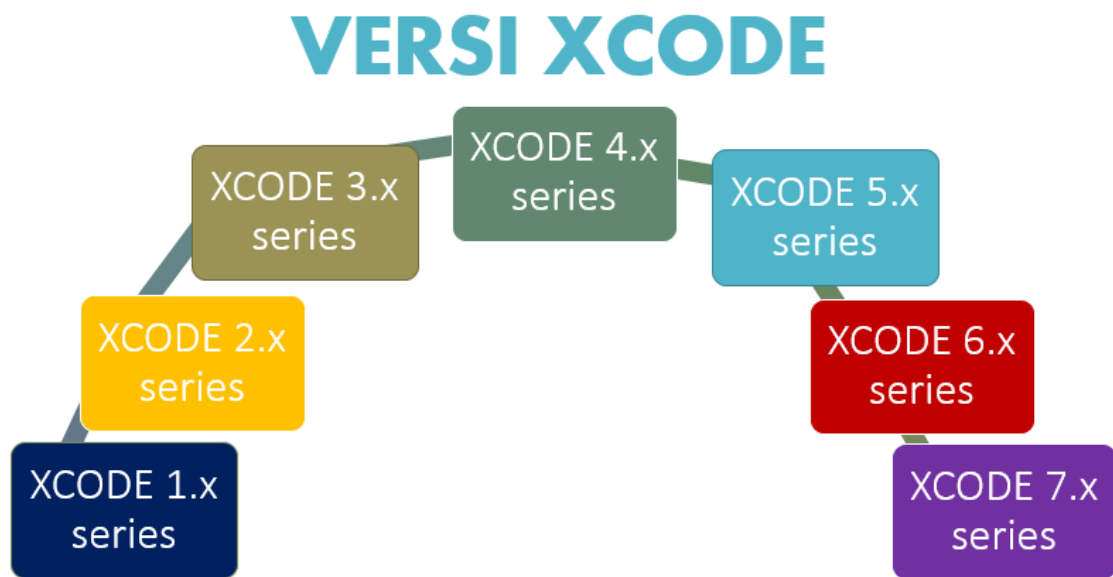
Pada akhirnya Turbo C++ digantikan oleh Borland C++ yang lebih lengkap dan kaya akan fitur, namun pada akhirnya menghilangkan fitur yang menjadi kesuksesan jajaran produk Turbo, seperti kecepatan kompilasi dan fasilitas dari IDE yang sangat baik.

VERSI BORLAND C++



Gambar 6 Sejarah Borland C++

Xcode adalah aplikasi pemrograman desktop yang di develop oleh Apple untuk mendvelop software untuk OS X dan IOS . Pertama kali dirilis pada tahun 2003,yaitu versi terbaru 6.4 yang stabil dan tersedia melalui Mac App Store secara gratis untuk pengguna OS X Yosemite.



Gambar 7 Sejarah XCode

Unity Technologies dibangun pada tahun 2004 oleh David Helgason (CEO), Nicholas Francis (CCO), dan Joachim Ante (CTO) di Copenhagen, Denmark setelah game pertama mereka GooBall gagal dalam meraih sukses. Ketiganya menyadari nilai sebuah engine dan tool dalam sebuah pengembangan game dan berencana untuk membuat sebuah engine yang dapat digunakan oleh semua dengan harga terjangkau. Unity Technologies mendapat bantuan dana dari Sequoia Capital, WestSummit Capital, and iGlobe Partners.

Pada tahun 2008, Unity melihat kebangkitan iPhone dan menjadi game engine pertama yang melakukan dukungan penuh pada platform tersebut. Unity sekarang di gunakan oleh 53.1% developers (termasuk mobile game developer) dengan ratusan game yang dirilis baik untuk iOS maupun Android.

Pada tahun 2009, Unity mulai meluncurkan produk mereka secara gratis. Jumlah developer yang mendaftar melonjak drastic sejak pengumuman tersebut. Pada April 2012, Unity mencapai popularitas yang sangat tinggi dengan lebih dari 1 juta developer.

C# (C sharp)

Merupakan sebuah bahasa pemrograman yang berorientasi objek yang dikembangkan oleh Microsoft sebagai bagian dari inisiatif kerangka .NET Framework. Bahasa pemrograman ini dibuat berbasis bahasa C++ yang telah dipengaruhi oleh aspek-aspek ataupun fitur bahasa yang terdapat pada bahasa-bahasa pemrograman lainnya seperti Java, Delphi, Visual Basic, dan lain-lain) dengan beberapa penyederhanaan.

JavaScript/Unity Script

Bahasa terkustomisasi yang terinspirasi dari syntax ECMAScript, dalam bentuk JavaScript

BooScript

Bahasa yang terinspirasi dari syntax bahasa pemrograman python

Sejarah Visual Basic

Bill Gates, pendiri Microsoft, memulai bisnis perangkat lunak dengan mengembangkan interpreter bahasa Basic untuk Altair 8800, untuk kemudian ia ubah agar dapat berjalan di atas IBM PC dengan sistem operasi DOS. Perkembangan berikutnya ialah diluncurkannya BASICA (basic-advanced) untuk DOS. Setelah BASICA, Microsoft meluncurkan Microsoft QuickBasic dan Microsoft Basic (dikenal juga sebagai Basic Compiler).

Visual Basic adalah pengembangan dari bahasa komputer BASIC (Beginner's All-purpose Symbolic Instruction Code). Bahasa BASIC diciptakan oleh Professor John Kemeny dan Thomas Eugene Kurtz dari Perguruan Tinggi Dartmouth pada pertengahan tahun 1960-an. Bahasa program tersebut tersusun mirip dengan bahasa Inggris yang biasa digunakan oleh para programmer untuk menulis program-program komputer sederhana yang berfungsi sebagai pembelajaran bagi konsep dasar pemrograman komputer.

Sejak saat itu, banyak versi BASIC yang dikembangkan untuk digunakan pada berbagai platform komputer, seperti Microsoft QBASIC, QUICKBASIC, GWBASIC, IBM BASICA, Apple BASIC dan lain-lain.

Apple BASIC dikembangkan oleh Steve Wozniak, mantan karyawan Hewlett Packard dan teman dekat Steve Jobs (pendiri Apple Inc.). Steve Jobs pernah bekerja dengan Wozniak sebelumnya (mereka membuat game arcade "Breakout" untuk Atari). Mereka mengumpulkan uang dan bersama-sama merakit PC, dan pada tanggal 1 April 1976 mereka secara resmi mendirikan perusahaan komputer Apple. Popularitas dan pemakaian BASIC yang luas dengan berbagai jenis komputer turut berperan dalam mengembangkan dan memperbaiki bahasa itu sendiri, dan akhirnya berujung pada lahirnya Visual Basic yang berbasis GUI (Graphic User Interface) bersamaan dengan Microsoft Windows. Pemrograman Visual Basic begitu mudah bagi pemula dan programmer musiman karena ia menghemat waktu pemrograman dengan tersedianya komponen-komponen siap pakai.

Hingga akhirnya Visual Basic juga telah berkembang menjadi beberapa versi, sampai yang terbaru, yaitu Visual Basic 2010. Bagaimanapun juga Visual Basic 6.0 tetap menjadi versi yang paling populer karena mudah dalam membuat programnya dan ia tidak menghabiskan banyak memori.

Sejarah BASIC di tangan Microsoft sebagai bahasa yang diinterpretasi (BASICA) dan juga bahasa yang dikompilasi (BASCOM) membuat Visual Basic diimplementasikan sebagai gabungan keduanya. Programmer yang menggunakan Visual Basic bisa memilih kode bahasa pemrograman yang dikompilasi atau kode yang harus bahasa pemrograman yang diinterpretasikan sebagai hasil porting dari kode VB. Sayangnya, meskipun sudah terkompilasi jadi bahasa mesin, DLL bernama MSVBVMxx.DLL tetap dibutuhkan. Namun karakteristik bahasa terkompilasi tetap muncul (ia lebih cepat dari kalau kita pakai mode terinterpretasi).

Visual Basic 1.0 dikenalkan pada tahun 1991. Konsep pemrograman dengan metode drag-and-drop untuk membuat tampilan aplikasi Visual Basic ini diadaptasi dari prototype generator form yang dikembangkan oleh Alan Cooper dan perusahaannya, dengan nama Tripod. Microsoft kemudian mengontrak Cooper dan perusahaannya untuk mengembangkan Tripod menjadi sistem form yang dapat diprogram untuk Windows 3.0, di bawah kode nama Ruby.

Tripod tidak memiliki bahasa pemrograman sama sekali. Ini menyebabkan Microsoft memutuskan untuk mengkombinasikan Ruby dengan bahasa pemrograman Basic untuk membuat Visual Basic.

Dari waktu ke waktu:

20 Mei 1991: Microsoft merilis Visual Basic versi 1.0 untuk Windows di Windows World 1991, Atlanta, Amerika Serikat. Programmer dapat membuat antar muka pengguna dengan mudah. Pembuatan aplikasi bisa lebih cepat dibandingkan sebelumnya. Kode ditulis oleh programmer untuk memberikan aksi terhadap Event dari pengguna (bagaimana pengguna merespon kepada object yang terdapat di antar muka).

1 September 1992: Microsoft mengumumkan Microsoft Visual Basic for MS-DOS® dalam edisi Standard dan Professional. Seperti Visual Basic untuk Windows, versi ini mengkombinasikan kemudahan pendisainan secara grafis dengan kekuatan dan keunggulan dalam banyak hal di pemrograman

2 November 1992: Microsoft mengumumkan kemampuan dari Visual Basic versi 2.0 untuk Windows pada edisi Professional dan Standard. Versi ini menyertakan lebih dari 300 fitur baru dan peningkatan untuk pengembangan aplikasi yang lebih cepat, pengaksesan ke fitur-fitur tingkat lanjut di Windows, dan produktivitas pengembang yang lebih besar lagi. Di antaranya: MDI Forms, ODBC, dan variabel

14 Mei 1993: Microsoft mengumumkan Visual Basic versi 3.0 untuk edisi Standard dan Professional. Versi 3.0 menyediakan kemudahan akses ke berbagai sumber data yang banyak dengan mengintegrasikan mesin database Microsoft Access Database for Windows 1.1 dan kemampuan di bidang aplikasi melalui Object Linking and Embedding (OLE) 2.0. Juga ditambahkan tools baru berupa controls baru, penggunaan yang lebih mudah, dan sebuah peningkatan standarisasi control bagi

29 Juni 1993: Microsoft mengumumkan bahwa Microsoft Visual Basic for Applications (VBA) akan diintegrasikan ke dalam Microsoft Excel 5.0 dan Microsoft Project 4.0. Visual Basic akan melayani bahasa Macro yang umum di versi yang akan datang dari aplikasi-aplikasi untuk Windows dan Macintosh. Fitur Macro yang umum untuk aplikasi dan OLE 2.0, keduanya merupakan elemen penting di visi dan strategi Microsoft untuk aplikasi yang dapat diprogram —yang akan memberikan kemampuan bagi pengguna untuk mengintegrasikan aplikasi-aplikasi, tugas-tugas otomatis, dan pembuatan solusi yang bersifat kostumais. VBA memiliki mesin pengembangan tangguh dan fleksibel yang sama yang membuat Visual Basic menjadi terkenal, teknologi pemrograman yang memperoleh

14 November 1994: Visual Basic versi 4.0 untuk Windows didemonstrasikan di Fall/COMDEX '94 di Las Vegas. Demonstrasi ini berfokus pada perannya sebagai aplikasi berbasis Windows 32-bit pertama yang menggunakan kontrol kostumais OLE (OCXs) — komponent perangkat lunak yang dapat digunakan kembali yang ditetapkan dengan spesifikasi OLE. Pada versi ini diperkenalkan: Class, OXC's, dan programmer dapat membuat add-ins sendiri dengan

12 September 1995: Visual Basic versi 4.0 untuk MS-DOS, Microsoft Windows NT®, dan Windows 95 diumumkan. Untuk meningkatkan edisi Standard dan Professional, maka Edisi Enterprise yang baru diperkenalkan untuk memenuhi kebutuhan perusahaan dan team pengembang. Semua versi menyertakan dukungan teknologi OLE yang lebih dikembangkan, perbaikan

akses data, sebuah lingkungan pengembangan yang bersifat visual, dan migrasi ke Windows 95 dan Windows NT untuk aplikasi-aplikasi yang sudah

7 Desember 1995: Microsoft mengumumkan Microsoft Visual Basic Scripting Edition (VBScript), sebuah bahasa skrip Internet berbasis pada Visual Basic. VBScript merupakan sebuah skrip dengan performansi yang tinggi yang didisain untuk membuat isi yang bersifat aktif pada suatu situs. VBScript membolehkan pengembang untuk mengaitkan dan mengotomatiskan berbagai jenis object di halaman situs, termasuk object OLE. VBScript menjadi sebuah spesifikasi yang diusulkan berlisensi gratis yang tersedia bagi komunitas

3 Februari 1997: Microsoft membuka sistem pemrograman Visual Basic versi 5.0, Edisi Professional — tool versi terakhir yang paling terkenal di dunia untuk membangun aplikasi yang tangguh (Rapid Application Development RAD) tool. Fitur-fitur Visual Basic versi 5.0 terdiri dari sejumlah performansi yang signifikan dan peningkatan produktivitas yang akan membuat para pengembang lebih efisien dan fleksibel serta cepat dalam memberikan solusi bagi pengguna akhir. Fitur-fitur tingkat lanjut — seperti kompilasi berjenis native code, akses database berkecepatan tinggi, dan sebuah lingkungan pengembangan yang ditingkatkan — membuat Visual Basic versi 5.0 menjadi versi yang paling hebat. Sebagai tambahan, kemampuan teknologi pembuatan komponen Microsoft ActiveX® di Visual Basic versi 5.0 akan membolehkan lebih dari 3 juta pengembang menggunakan Visual Basic dalam membuat aplikasi yang berbasis pada komponen yang bertujuan untuk digunakan di Internet, intranets, dan lingkungan tradisional client/

10 Maret 1997: Versi terakhir yang dirilis dari Visual Basic 5.0, Edisi Pembuatan Kontrol tersedia. Edisi Pembuatan Kontrol adalah cara mudah untuk membuat ActiveX Controls bagi Internet, intranets, dan aplikasi client/

15 Juni 1998: Microsoft mengumumkan Visual Basic versi 6.0, dan dimasukkan ke dalam Microsoft Visual Studio® versi 6.0. Fitur-fitur Visual Basic versi 6.0 menyediakan pengaksesan data secara terintegrasi dan bersifat grafis ke sumber data (data source) ODBC atau OLE DB manapun, dan perangkat tambahan database yang didisain untuk database Oracle dan Microsoft SQL Server™. Fitur unggulan di versi ini adalah: ActiveX Data Objects (ADO) untuk memanipulasi dan membuat database. Fitur Pengembangan Situs membawa kemudahan dalam penggunaan, model pemrograman berbasis komponen dari Visual Basic untuk membuat HTML — dan Dynamic HTML (DHTML) — berbasis aplikasi. Fitur-fitur baru ini — dikombinasikan dengan optimisasi performansi, pengembangan aplikasi yang disederhanakan dan debugging, dan dukungan untuk Microsoft teknologi server — membuat Visual Basic versi 6.0 sebuah pilihan yang ideal untuk membangun aplikasi berskala

2 September 1998: Visual Studio 6.0, solusi tool pengembangan lengkap berskala perusahaan tersedia. Visual Studio 6.0 menyertakan Visual Basic versi 6.0, Microsoft Visual C++®, Microsoft Visual FoxPro®, Microsoft Visual InterDev® Web, dan Microsoft Visual J++®. Bersamaan dengan diluncurkannya Developer Days '98, lebih dari 115 perusahaan papan atas mengumumkan dukungan terhadap Visual Studio 6.

28 April 1999: Lingkungan pengembangan VBA versi 6.0 dan Software Development Kit (SDK) VBA (SDK) versi 6.0 tersedia bagi vendor perangkat lunak pihak ketiga melalui program lisensi VBA. VBA 6.0 merupakan sebuah teknologi pengembangan yang tangguh untuk aplikasi yang dikemas secara kostumais dan merupakan sebuah komponen utama di Microsoft Office

15 Februari 2000: Steve Ballmer, presiden yang baru saja ditunjuk dan CEO dari Microsoft Corporation, berbicara kepada lebih dari 2.000 pengembang dalam pidatonya pada Visual Basic Insiders Technical Summit (VBITS), menyatakan visi Microsoft terhadap situs yang dapat diprogram dan memaksa kembali komitmen perusahaan yang dalam terhadap para pengembang Visual

11 Juli 2000: Microsoft menyampaikan Tool Platform .NET untuk XML Web Services. Dalam pidatonya yang ditujukan pada acara Microsoft Professional Developers Conference (PDC) 2000 yang kedelapan, Paul Maritz, wakil presiden kelompok dari Platforms Group di Microsoft, memperlihatkan Microsoft .NET Framework dan Microsoft Visual Studio .NET. Ini merupakan versi terakhir dari perangkat pengembangan yang digunakan di seluruh dunia, Visual Studio .NET menyediakan dukungan untuk pengembangan drag-and-drop terhadap XML Web Services. Secara bersama, kedua produk tersebut menyediakan produktivitas yang tinggi kepada para pengembang, suatu lingkungan yang terdiri dari banyak bahasa untuk pembangunan, pengantaran, dan pengintegrasian XML Web Services pada platform Microsoft .

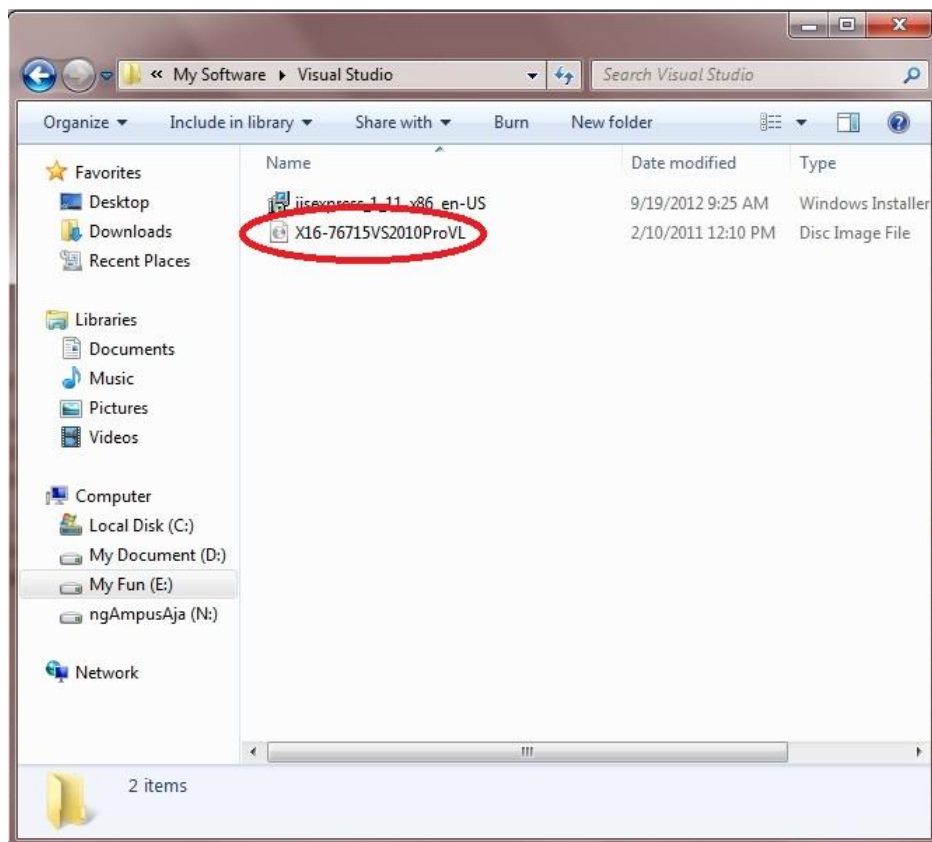
13 November 2000: Microsoft mengumumkan kemampuan Visual Studio .NET versi Beta 1. Dalam pidato acara COMDEX/Fall 2000, Bill Gates mengumumkan kemampuan versi beta pertama Visual Studio.NET dan .NET Framework, dua teknologi kunci untuk memungkinkan para pengembang membangun XML Web services pada platform .NET. XML Web services merupakan aplikasi dan komponen yang dibuat tersedia melewati situs dengan menggunakan XML dan Simple Object Access Protocol (SOAP), dan kunci yang dapat diprogram untuk membangun generasi masa depan Internet. Microsoft membuat Visual Studio .NET Beta 1 dan .NET Framework tersedia bagi jutaan pelanggan dan rekanan

13 Februari 2002: Microsoft mengumumkan kemampuan dari Visual Studio .NET versi akhir.

2. Instalasi Aplikasi Program Desktop

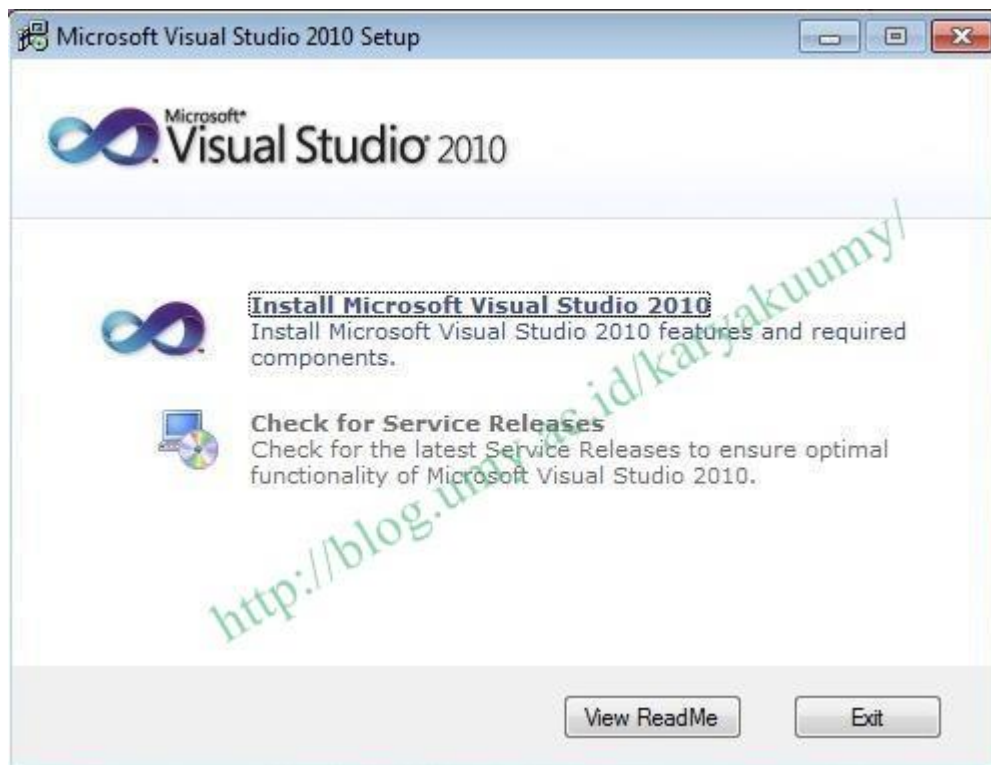
Langkah pertama untuk menginstal Visual Studio 2010 ini ialah mempersiapkan masternya terlebih dahulu dan OS yang cocok dengan Aplikasi ini. Karena di IT UMY mewajibkan memakai OS windows 7 maka saya akan menjelaskan tentang menginstal Visual Studio 2010.

1. Persiapkan master yang akan diinstal, dan persiapkan **Daemon Tools Lite** untuk men-Mount Image kan file iso yang ada.



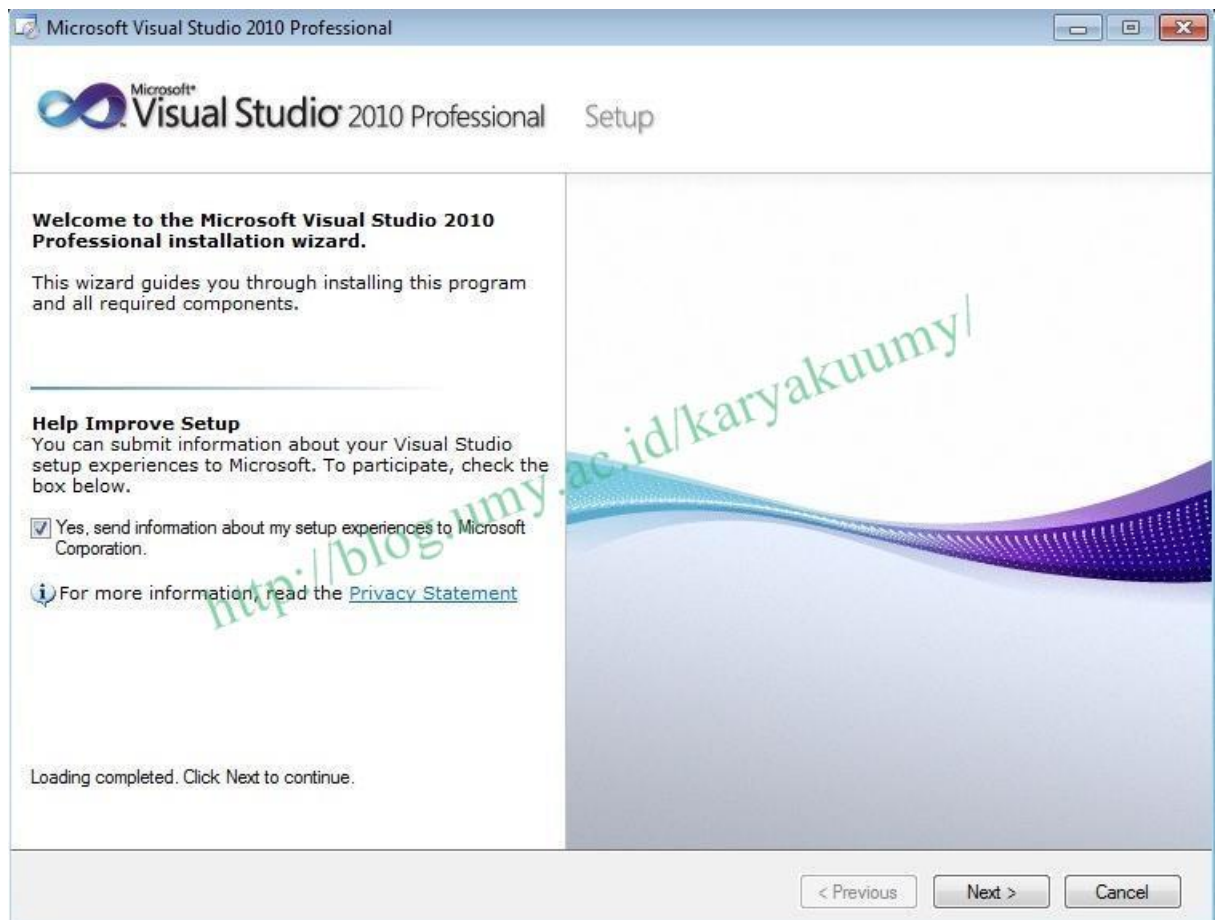
Gambar 8 Daemon Tools Lite

2. Setelah di Mount Image kalian bisa menginstal Visual Studio 2010 dan klik **Install Microsoft Visual Studio 2010**, dengan tampilan awal seperti dibawah ini :



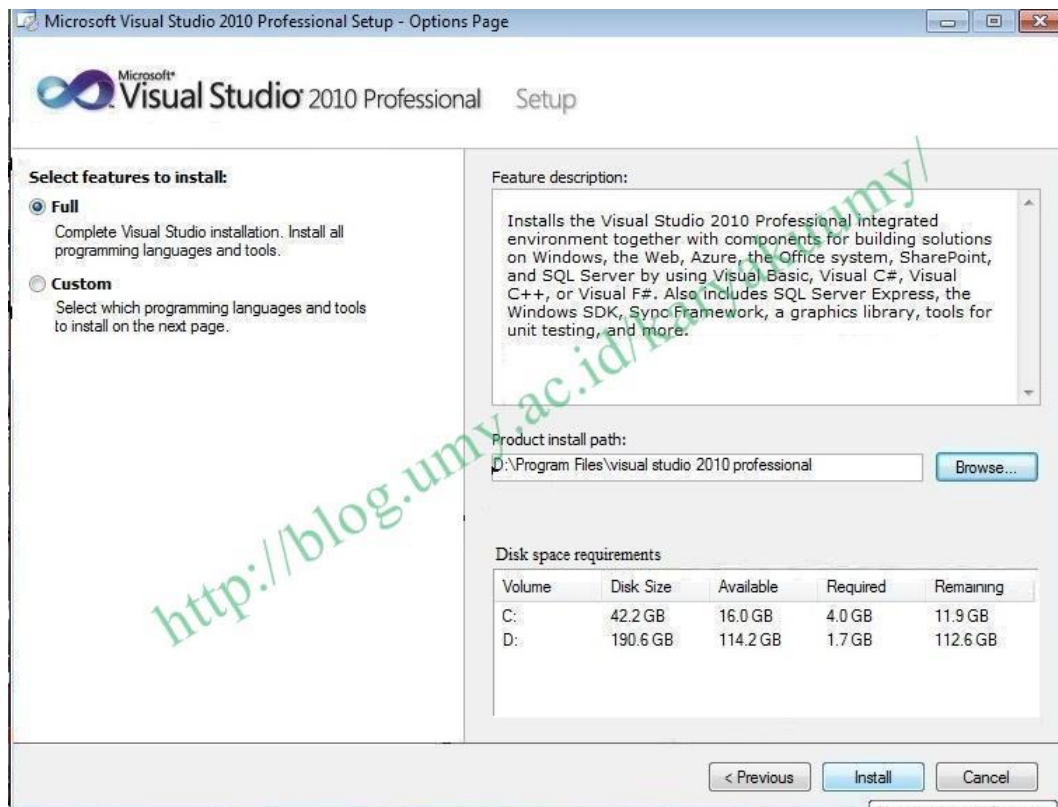
Gambar 9 Pilihan Instal VS2010

3. Setelah itu langsung muncul jendela baru seperti yang tampil dibawah ini, lalu klik next untuk melanjutkan.



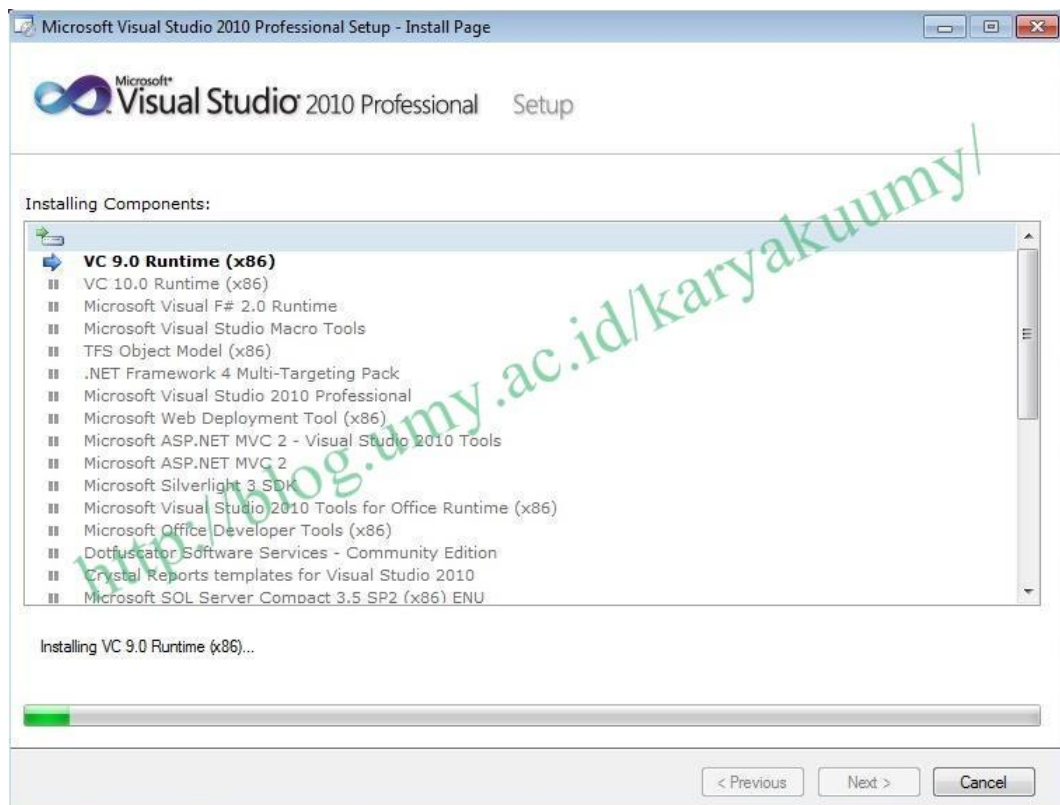
Gambar 10 Persetujuan Pemberian Informasi VS2010

4. Setelah next akan muncul tampilan seperti dibawah ini dan langsung akan tampil tempat yang akan kita simpan untuk menginstal Visual Studio 2012. Biasanya langsung manual akan di tempatkan di program files di part C.



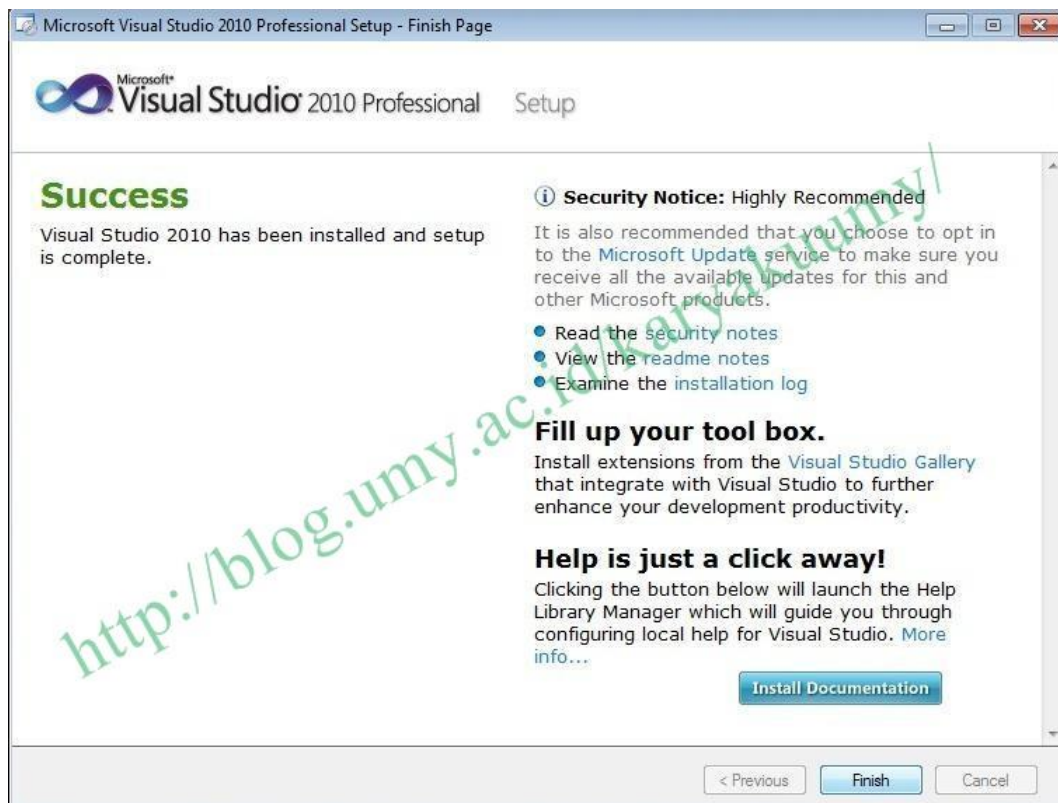
Gambar 11 Fitur Instalasi VS2010

5. Setelah di klik next, maka tunggu saja instalasi sampai selesai dan proses instalasi akan tampak seperti gambar dibawah ini :



Gambar 12 Proses Instalasi VS2010

6. Setelah selesai maka klik next saja untuk mengakhiri proses instalasi. Maka dengan ini, kamu telah berhasil menginstalasi Visual Studio 2010.



Gambar 13 Sukses Instalasi VS2010

C. Uji Kemampuan

Kerjakan praktikum berikut secara mandiri

Jalankan Program Anda untuk pertama kalinya!



KEGIATAN BELAJAR 2

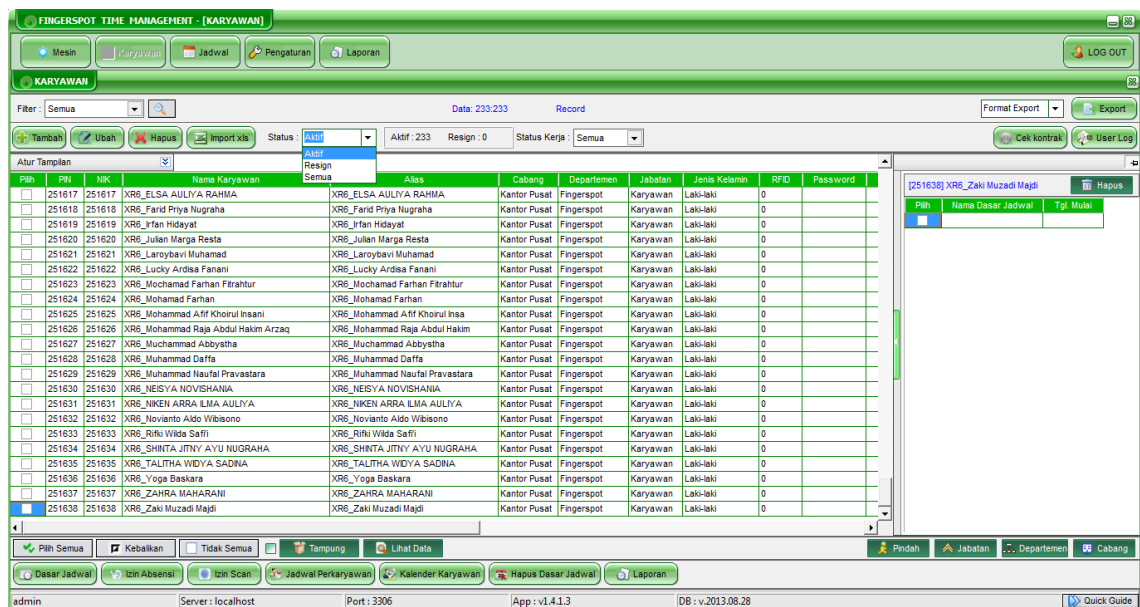
DESAIN APLIKASI DESKTOP

A. Tujuan Pembelajaran

Setelah melakukan kegiatan pembelajaran siswa dapat :

1. Memahami teknik desain aplikasi desktop
2. Menyajikan desain aplikasi berbasis desktop

B. Deskripsi Materi



Gambar 14 Stimulus Desain Aplikasi Desktop

Tugas 2

Amati Gambar diatas!

Komponen apa saja yang terdapat pada gambar tersebut?

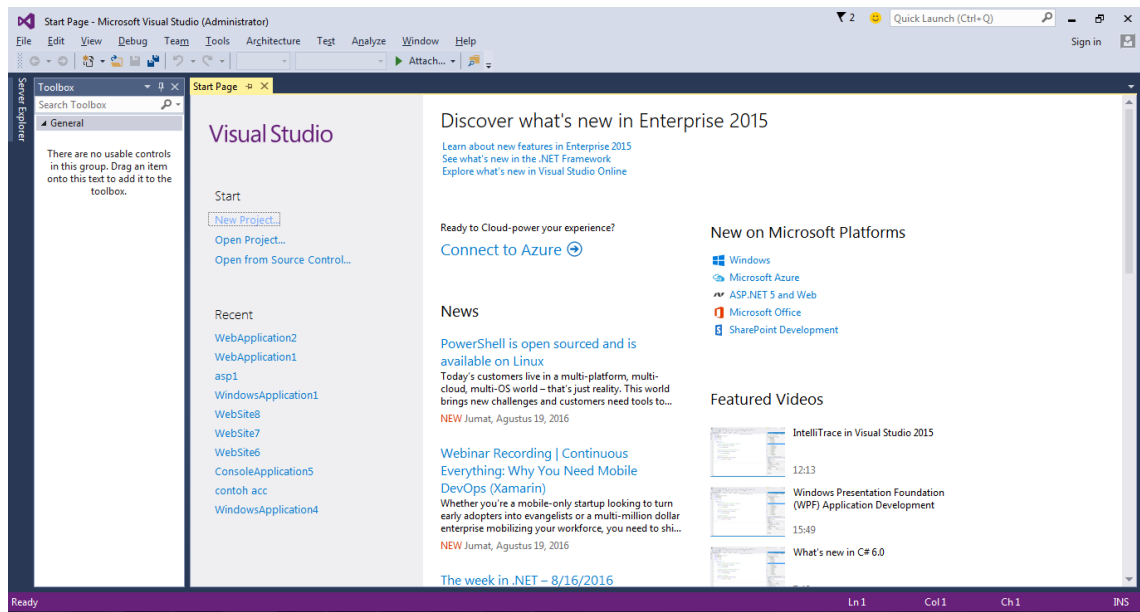
Ceritakan pendapat anda, sampaikan dalam kegiatan pembelajaran!

1. Pembuatan Form

Visual Basic merupakan salah satu tool untuk pengembangan aplikasi yang banyak diminati oleh orang. Disini Visual Basic menawarkan kemudahan dalam pembuatan aplikasi dan dapat menggunakan komponen-komponen yang telah tersedia.

Pengenalan Tampilan Awal Visual Basic 2010

Pada awal Visual Basic dibuka, akan muncul gambar seperti berikut :






















Gambar 15 Form New Project VS






Menu Bar

Menu bar digunakan untuk melakukan perintah – perintah tertentu. Untuk menggunakan menu bar, dapat dilakukan dengan dua cara, yaitu :

1. Dengan mouse, klik mouse pada menu dan sub menu.
2. Dengan keyboard, tekan Alt dan karakter bergaris bawah.

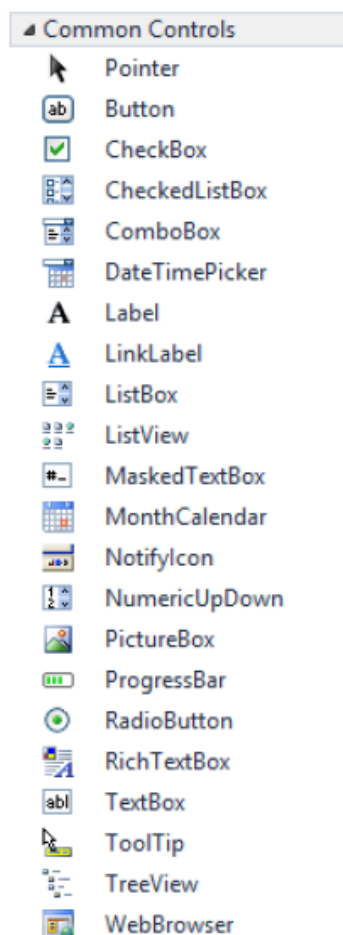
Standard Bar

1.  *Add Project* → menambah project baru.
2.  *Open Project* → membuka project yang pernah dibuka sebelumnya.
3.  *Add New Item* → menambahkan item di dalam project yang sudah ada.
4.  *Save Selected Item* → menyimpan hanya salah satu form yang diinginkan.
5.  *Save All* → menyimpan semua data yang terkait di dalam satu project.
6.  *Cut* → memotong elemen / tools yang ada pada layar.
7.  *Copy* → menyalin elemen / tools yang dipilih pada layar.
8.  *Paste* → menyalin elemen / tools yang sebelumnya sudah dipotong atau disalin.
9.  *Find* → mencari text pada jendela code.
10.  *Comment Out The Selected Lines* → memberikan tanda komentar pada baris yang dipilih.
11.  *Uncomment Out The Selected Lines* → menghilangkan tanda komentar pada baris yang dipilih.
12.  *Undo* → membatalkan suatu perintah yang pernah dijalankan sebelumnya.
13.  *Redo* → mengulangi perintah yang pernah dibatalkan sebelumnya.
14.  *Start Debugging* → menjalankan project yang dibuat pada *Visual Studio*.
15.  *Stop Debugging* → menghentikan *running* program.
16.  *Step Into* → mengarahkan ke *source code* apabila kita klik pada program yang sedang *start debugging*.
17.  *Step Over* → mengarahkan ke *source code* dan menghentikan program sementara pada saat di klik pada program yang sedang *start debugging*.
18.  *Step Out* → menampilkan *output* dari *start debugging* pada *form* yang sedang dijalankan.
19.  *Solution Explorer* → menyimpan informasi mengenai *solution*, *project*, *file*, *form* ataupun *resource* yang digunakan pada program.


















20.  *Properties Window* → menampilkan jendela *properties*.
21.  *Toolbox* → menampilkan jendela *toolbox*.
22.  *Extension Manager* → menampilkan jendela *extension*.
23.  *Error List* → menampilkan jendela kesalahan pada program atau *error list*.
24.  *Immediate* → menampilkan jendela *immediate* pada *error list*.



Toolbox

Jendela kecil yang biasanya terletak di sebelah kiri ini berisi semua kontrol yang hendak digunakan dalam pengembangan aplikasi berbasis windows form. Pada toolbox terdapat beberapa kelompok control yang disusun berdasarkan kegunaannya. Jendela ini dapat diperluas (expand[+]) atau diperkecil (collapse[-]).



Gambar 16 Toolbox

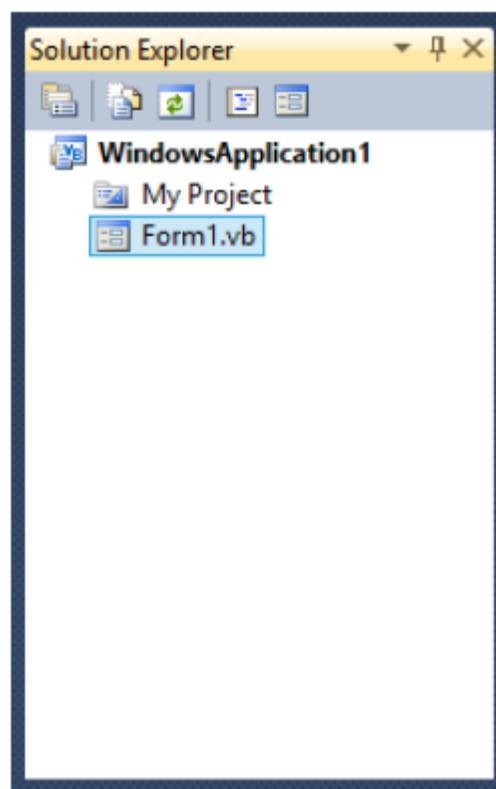
1.  *Button* → *tool* yang berupa tombol, biasa digunakan untuk melakukan perintah seperti *save, delete, submit*, dll.
2.  *Check Box* → memilih item yang terdiri dari satu item dan bisa memilih lebih dari satu item.
3.  *Checked List Box* → fungsinya sama seperti nomor 2 hanya saja *tool* ini berbentuk list dalam satu *frame*.
4.  *Combo Box* → digunakan untuk memilih item dalam bentuk *combo*. Kita hanya dapat memilih satu item.
5.  *Date Time Picker* → untuk menampilkan data tanggal.
6.  *Label* → untuk memberikan label / *text* pada kontrol yang ada.
7.  *Link Label* → dengan control ini apabila label tersebut di klik akan mengarah ke alamat *website* di internet.
8.  *List Box* → untuk menampilkan beberapa item.
9.  *List View* → untuk menampilkan data dalam bentuk tabel yang mempunyai baris dan kolom.
10.  *Masked Text Box* → untuk menampilkan data dengan format tertentu.
11.  *Month Calendar* → untuk menampilkan tanggal dalam satu bulan dengan tampilan penuh.
12.  *Notify Icon* → untuk menampilkan *icon system* di *Task Bar window* bagian kanan.
13.  *Numeric Up Down* → untuk menyajikan nilai numerik saja.
14.  *Picture Box* → untuk menampilkan gambar
15.  *Progress Bar* → digunakan untuk menampilkan status proses.
16.  *Radio Button* → untuk menampilkan beberapa item yang bisa dipilih, tetapi hanya satu item saja yang bisa dipilih.
17.  *Rich Text Box* → untuk menampilkan teks / paragraph.
18. *Text Box* → untuk menginput / menampilkan teks.
19. *Tool Tip* → kontrol ini digunakan untuk menampilkan informasi mengenai kegunaan tombol – tombol yang biasanya muncul apabila mouse didekatkan atau diarahkan pada tombol tersebut.

- 20.  *Tree View* → untuk menampilkan data atau item – item dengan bentuk cabang.
- 21.  *Web Browser* → digunakan untuk membuat halaman *browsing* di internet.

Solution Explorer

Solution Explorer adalah jendela yang menyimpan informasi mengenai solution, project, file, form, ataupun resource yang digunakan pada program aplikasi. Satu project aplikasi biasanya terdiri dari beberapa form dan komponen lainnya, untuk mempermudah berpindah antara komponen dalam Visual Basic dapat menggunakan jendela project. Pada solution explorer dapat menambahkan class, module, windows form baru dan sebagainya. Jika pada saat mengaktifkan Visual Studio 2010 jendela solution explorer tidak ada, kita dapat menampilkannya pada menu bar View, Solution Explorer atau menggunakan tombol Ctrl + Alt + L.

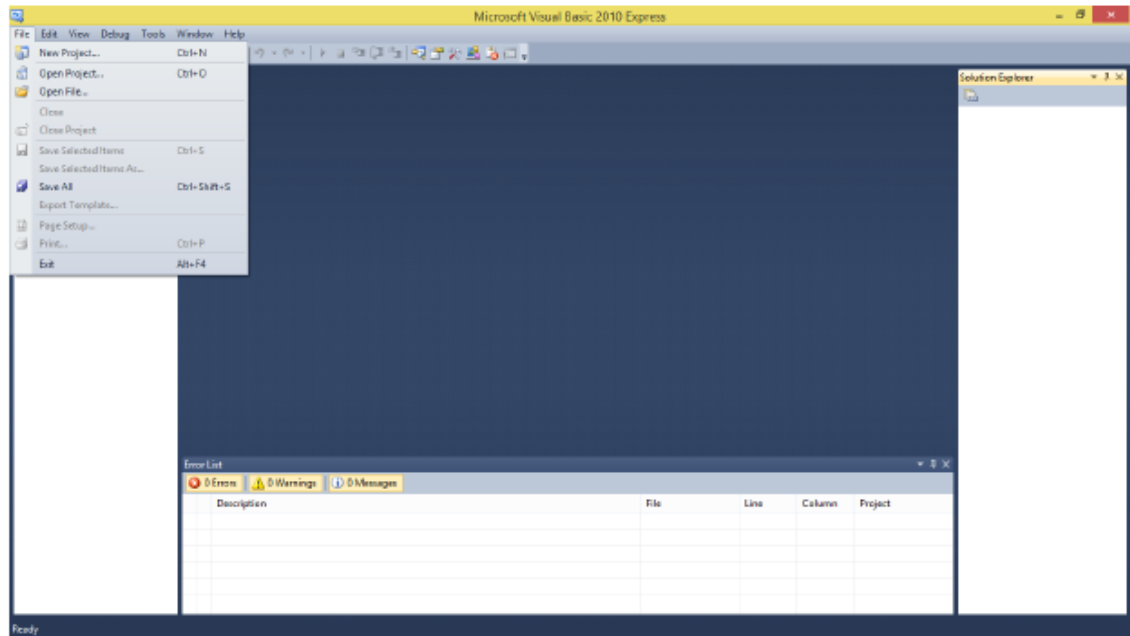
Pada satu Solution kita dapat mengembangkan project dengan bahasa pemrograman yang lain.



Gambar 17 Sejarah Delphi

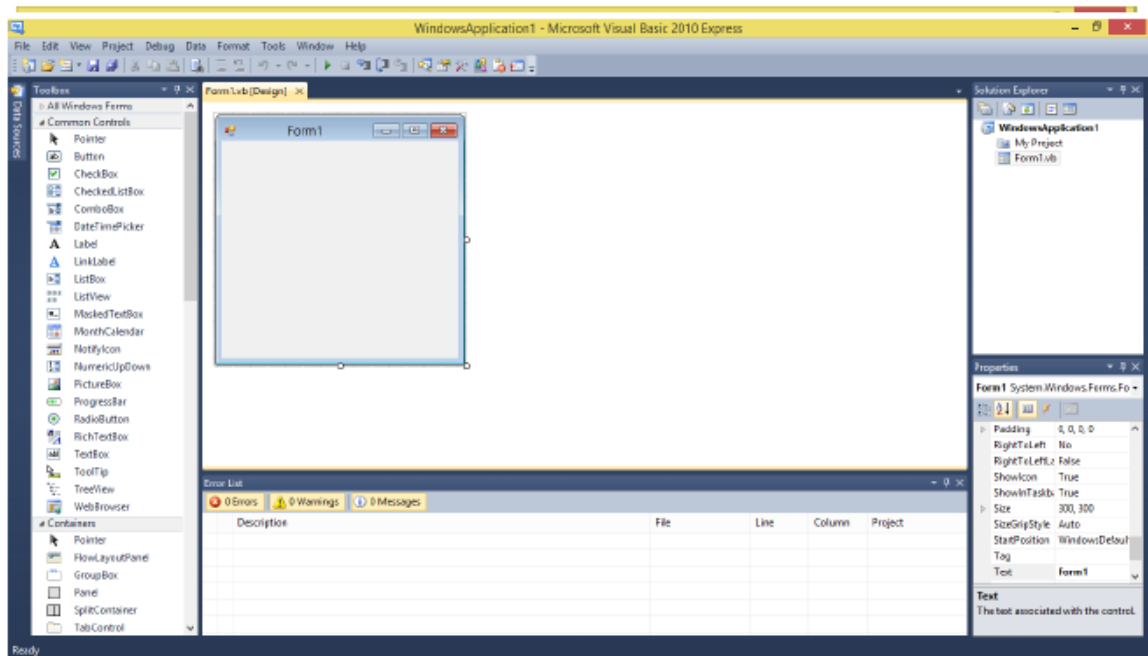
Memulai Project VB.net

Klik Menu File New Project



Gambar 18 File New Project VB

Akan muncul halaman seperti berikut

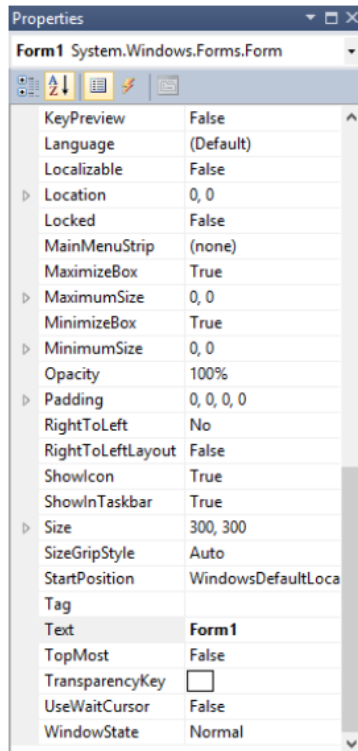


Gambar 19 New Project VB

Pada menu types pilih Windows Form Application, maka akan tampil halaman desain seperti berikut.

Properties Bar

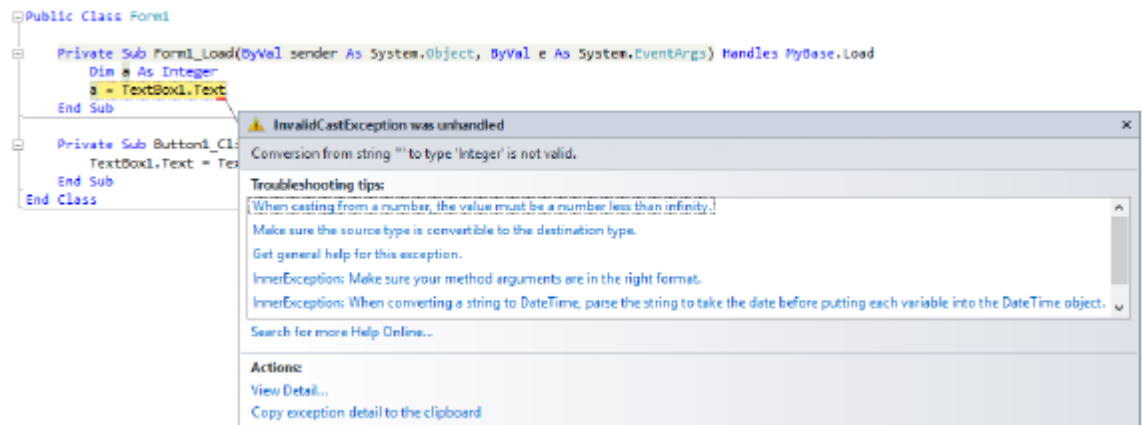
Fungsi dari jendela Properties adalah untuk mengubah properti setiap elemen yang terdapat pada form dan memberikan informasi mengenai objek yang sedang aktif. Untuk menampilkan jendela Properties pada layar, klik ikon jendela Properties pada Toolbox.



Gambar 20 Propertis Bar VB

Error Message

Apabila terjadi kesalahan maka akan muncul tampilan error message seperti berikut.



Gambar 21 Error Message VB

2. Menambah Kontrol pada Form

Contoh pembuatan form pertama kali. Disain form dan propertiis komponen sesuai tabel dan gambar.

Table 1 Contoh Program Kontrol pada Form

Nama Control	Propertis	Nilai
Form3	Text	Form
Label1	Text	Nama
Label2	Text	Kosongkan
Button1	Name	btnproses
	Text	Proses
Button1	Name	Btntutup
	Text	Tutup
Grupbox1	Text	Warna
Grupbox2	Text	Font Style
Radiobutton1	Text	Merah
Radiobutton2	Text	Kuning
Radiobutton3	Text	Hijau
Radiobutton4	Text	Biru
Checkbox1	Text	Bold
Checkbox2	Text	Italic
Checkbox3	Text	Underline
Checkbox4	Text	Strikeout

The image shows a Windows application window titled "Form3". Inside the window, the text "Program Sederhana" is centered at the top. Below this, there are three main sections. On the left, there is a label "Nama" followed by a text input field containing the text "MARHAS". Below the input field is a preview of the text "MARHAS" in a red, bold, italic, and underlined font. In the center, there is a section labeled "Warna" (Color) with four radio buttons: "Merah" (selected), "Kuning" (Yellow), "Hijau" (Green), and "Biru" (Blue). On the right, there is a section labeled "Font Style" with four checkboxes: "Bold" (checked), "Italic" (checked), "Underline" (checked), and "Strikeout" (unchecked). At the bottom of the form, there are two buttons: "Proses" (Process) and "Tutup" (Close).

Gambar 22 Contoh Form Sederhana

Kode programnya :

```
Public Class Form3

    Private Sub btnproses_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnproses.Click
        Label2.Text = txtnama.Text
    End Sub

    Private Sub btntutup_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btntutup.Click
        End
    End Sub

    Private Sub RadioButton1_CheckedChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles RadioButton1.CheckedChanged
        Label2.ForeColor = Color.Red
    End Sub

    Private Sub RadioButton2_CheckedChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles RadioButton2.CheckedChanged
        Label2.ForeColor = Color.Yellow
    End Sub

    Private Sub RadioButton3_CheckedChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles RadioButton3.CheckedChanged
        Label2.ForeColor = Color.Green
    End Sub

    Private Sub RadioButton4_CheckedChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles RadioButton4.CheckedChanged
        Label2.ForeColor = Color.Blue
    End Sub

    Private Sub Form3_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load

        End Sub

    Private Sub CheckBox1_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles CheckBox1.CheckedChanged
        Label2.Font = New Font("Microsoft Sans Serif", Label2.Font.Size,
Label2.Font.Style Xor FontStyle.Bold)
    End Sub

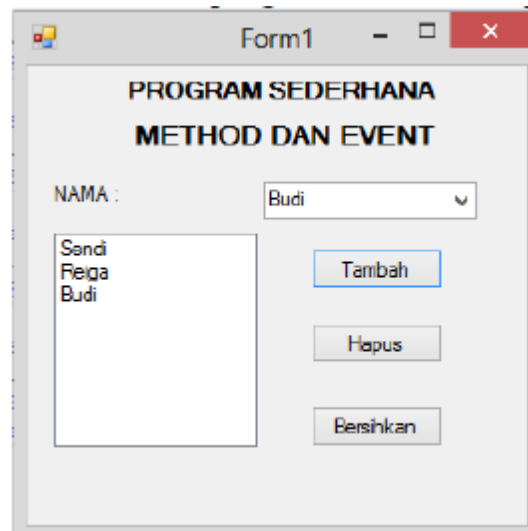
    Private Sub CheckBox2_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles CheckBox2.CheckedChanged
        Label2.Font = New Font("Microsoft Sans Serif", Label2.Font.Size,
Label2.Font.Style Xor FontStyle.Italic)
    End Sub

    Private Sub CheckBox3_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles CheckBox3.CheckedChanged
        Label2.Font = New Font("Microsoft Sans Serif", Label2.Font.Size,
Label2.Font.Style Xor FontStyle.Underline)
    End Sub

    Private Sub CheckBox4_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles CheckBox4.CheckedChanged
        Label2.Font = New Font("Microsoft Sans Serif", Label2.Font.Size,
Label2.Font.Style Xor FontStyle.Strikeout)
    End Sub
End Class
```

3. Event

Buatlah program seperti gambar



Gambar 23 Contoh Form Program Event

Kode programnya :

```
Public Class Form1

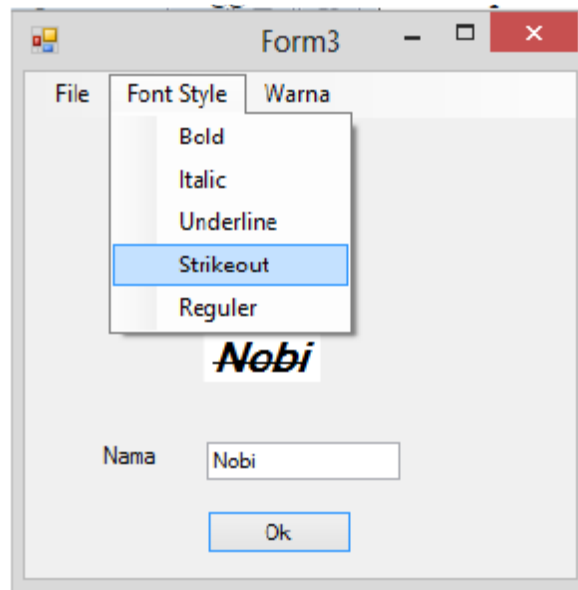
    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        ComboBox1.Items.Add("Budi")
        ComboBox1.Items.Add("Sendi")
        ComboBox1.Items.Add("Kiki")
        ComboBox1.Items.Add("Reiga")
        ComboBox1.Items.Add("Fauzi")
    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        ListBox1.Items.Add(ComboBox1.Text)
    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
        ListBox1.Items.Remove(ListBox1.Text)
    End Sub

    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button3.Click
        ListBox1.Items.Clear()
    End Sub
End Class
```

Buatlah program seperti gambar



Gambar 24 Contoh Form Menu

Kode programnya :

```
Public Class Form3

    Private Sub ExitToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles ExitToolStripMenuItem.Click
        End
    End Sub

    Private Sub BoldToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles BoldToolStripMenuItem.Click
        Label2.Font = New Font("Microsoft Sans Serif", Label2.Font.Size, Label2.Font.Style Xor FontStyle.Bold)
    End Sub

    Private Sub FontStyleToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles FontStyleToolStripMenuItem.Click
        End Sub

    Private Sub ItalicToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles ItalicToolStripMenuItem.Click
        Label2.Font = New Font("Microsoft Sans Serif", Label2.Font.Size, Label2.Font.Style Xor FontStyle.Italic)
    End Sub

    Private Sub UnderlineToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles UnderlineToolStripMenuItem.Click
        Label2.Font = New Font("Microsoft Sans Serif", Label2.Font.Size, Label2.Font.Style Xor FontStyle.Underline)
    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
        Label2.Text = TextBox1.Text
    End Sub
```

```
Private Sub StrikeoutToolStripMenuItem_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles StrikeoutToolStripMenuItem.Click
    Label2.Font = New Font("Microsoft Sans Serif", Label2.Font.Size,
Label2.Font.Style Xor FontStyle.Strikeout)
End Sub

Private Sub RegularToolStripMenuItem_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles RegularToolStripMenuItem.Click
    Label2.Font = New Font("Microsoft Sans Serif", Label2.Font.Size,
Label2.Font.Style Xor FontStyle.Regular)
End Sub

Private Sub MerahToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles MerahToolStripMenuItem.Click
    Label2.ForeColor = Color.Red
End Sub

Private Sub KuningToolStripMenuItem_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles KuningToolStripMenuItem.Click
    Label2.ForeColor = Color.Yellow
End Sub

Private Sub HijauToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles HijauToolStripMenuItem.Click
    Label2.ForeColor = Color.Green
End Sub

Private Sub BiruToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles BiruToolStripMenuItem.Click
    Label2.ForeColor = Color.Blue
End Sub

Private Sub KuningToolStripMenuItem1_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles KuningToolStripMenuItem1.Click
    Label2.ForeColor = Color.Yellow
End Sub
End Class
```

4. Properti DialogResult

Klik New Project > Windows Form Application

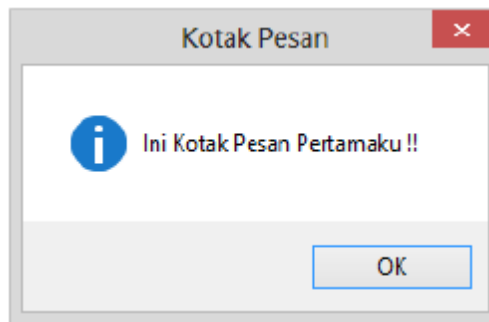
Kemudian ketikkan Peraktek pertama pada kotak Name setelah itu klik ok

Kemudian klik dua kali pada kotak form dan ketikkan kode di bawah ini :

```
Public Class Form1

    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        MessageBox.Show("Ini Kotak Pesan Pertamaku !!", "Kotak Pesan",
        MessageBoxButtons.OK, MessageBoxIcon.Information)
    End Sub
End Class
```

Setelah itu klik run maka hasilnya akan terlihat seperti gambar di bawah ini :



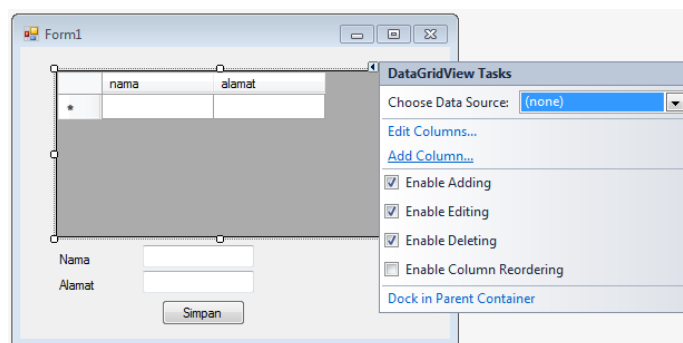
Gambar 25 Contoh Program DialogResult

5. DataGridView

Klik New Project > Windows Form Application

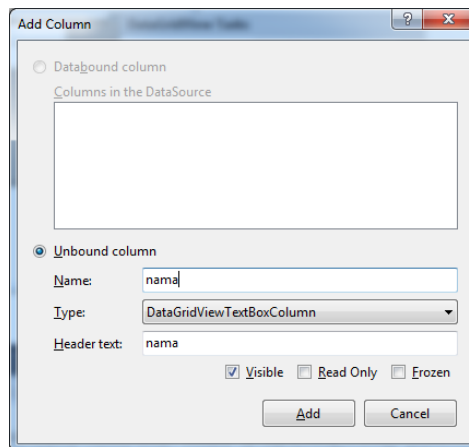
Pada Form, tambahkan DataGridView, Label, Textbox, Button sesuai gambar, dan ikuti petunjuk :

Klik pada panah



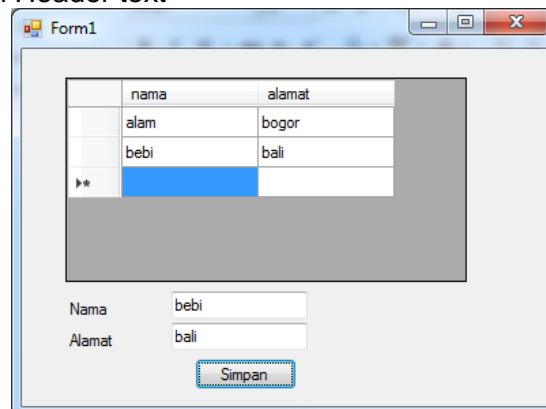
Gambar 26 Contoh GridView

Klik Add Colum...



Gambar 27 Contoh Menambah Kolom pada GridView

Isi Name dan Header text



Gambar 28 Contoh Hasil Pengisian GridView

Kemudian double klik pada button, ketik program seperti dibawah ini:

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles  
    Button1.Click  
        DataGridView1.Rows.Add(TextBox1.Text, TextBox2.Text)  
End Sub
```

Setelah itu klik run maka hasilnya akan terlihat seperti gambar di bawah ini :

C. Uji Kemampuan

Kerjakan praktikum berikut secara mandiri

Buatlah Program yang telah dicontohkan pada kegiatan belajar ini jadikan masing-masing contoh dalam 1 form, dan jadikan dalam 1 project. (Judul Project : UI_Kelas_No.absen_Nama, masing-masing judul Form diberi text: Kelas No absen Nama)



KEGIATAN BELAJAR 3

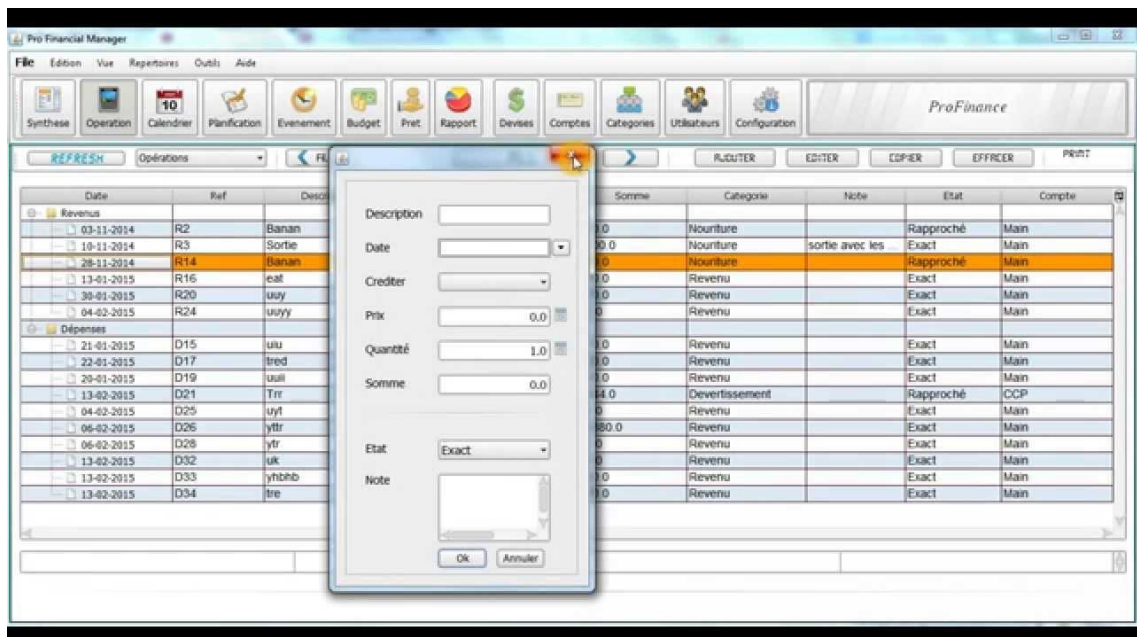
DESAIN APLIKASI MULTIWINDOW

A. Tujuan Pembelajaran

Setelah melakukan kegiatan pembelajaran siswa dapat :

1. Memahami teknik desain multiwindow
2. Memahami teknik Navigasi antar window
3. Memahami teknik Multiwindow dengan Tabulasi
4. Memahami teknik Navigasi Splash dengan timer
5. Menyajikan teknik desain multiwindow
6. Memahami teknik Navigasi antar window
7. Menyajikan teknik Multiwindow dengan Tabulasi
8. Menyajikan teknik Navigasi Splash dengan timer

B. Deskripsi Materi



Gambar 29 Stimulus Aplikasi MultiWindow

Tugas 3

Amati Gambar tersebut!

Dicetak atau dihasilkan menggunakan aplikasi apa?

Apakah memungkinkan dihasilkan oleh VB.NET?

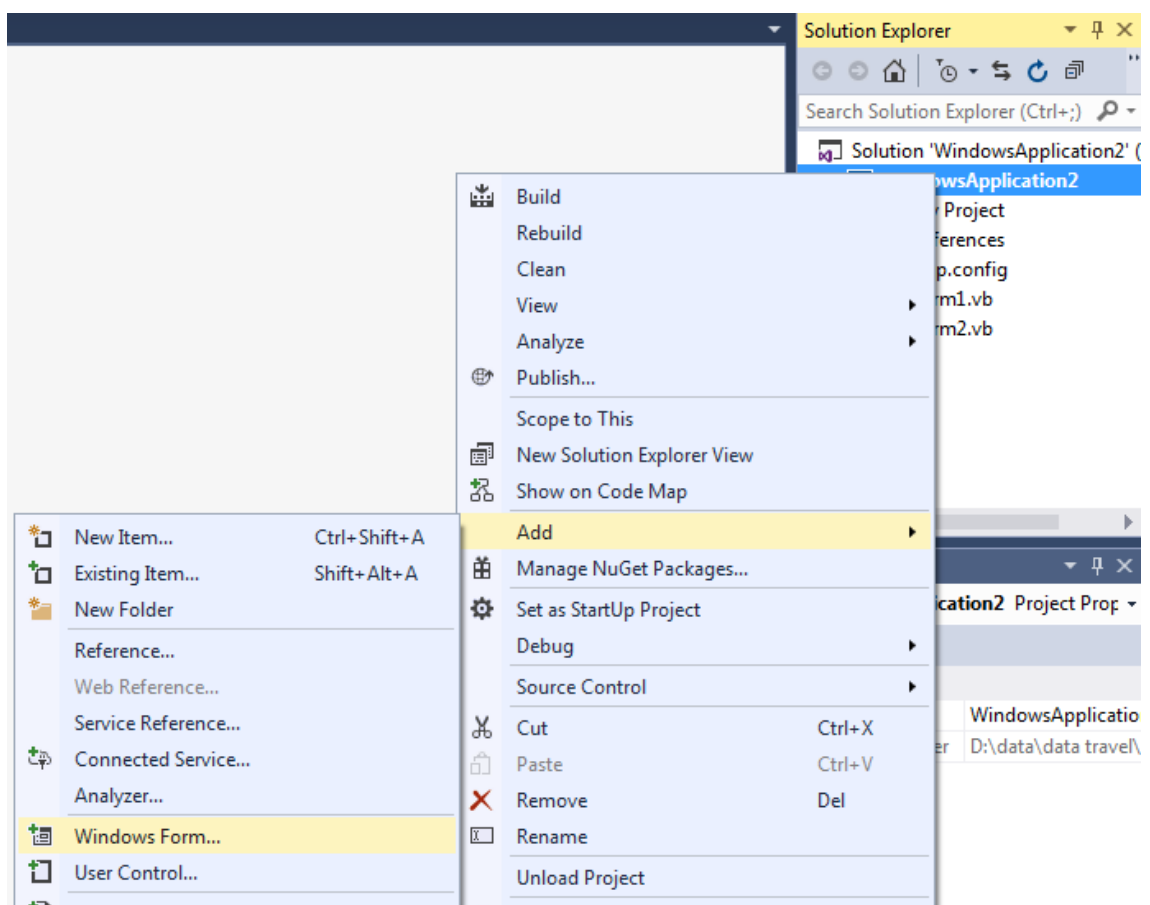
Ceritakan pendapat anda, sampaikan dalam kegiatan pembelajaran!

1. Prinsip Desain Multiwindow

Disain Multiwindow diperlukan bila kita ingin berganti halaman baik dalam satu form maupun beberapa form

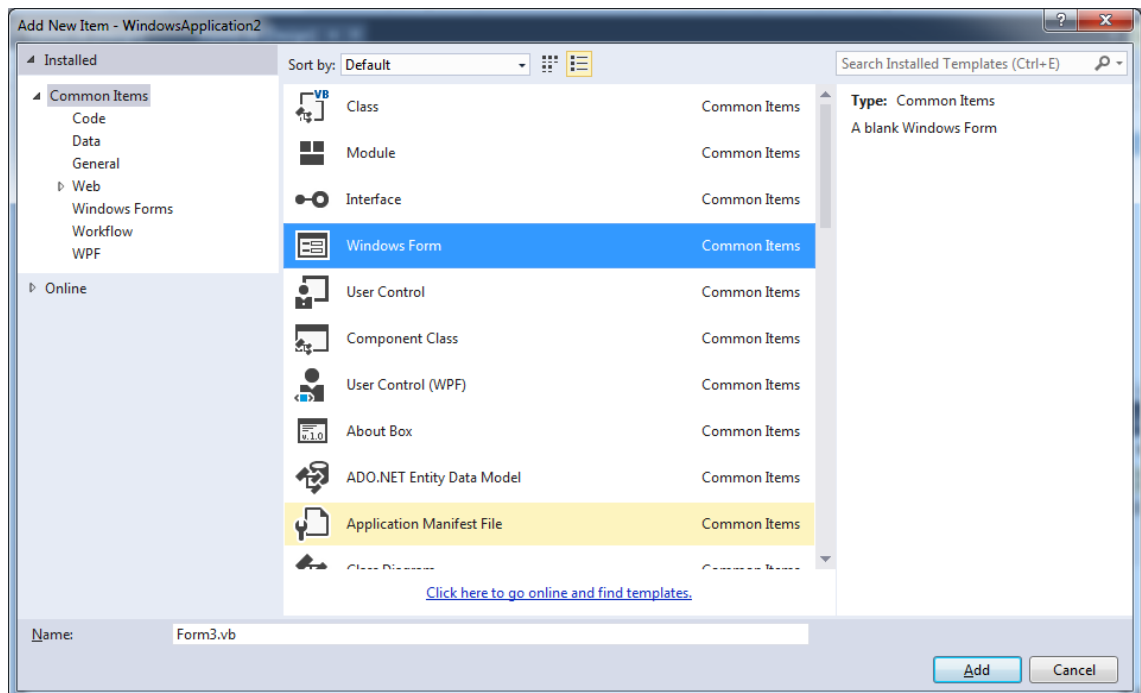
2. Navigasi antar window

Untuk membuat program multiwindow dengan banyak form, kita hanya memerlukan menambah form baru, dengan cara klik kanan pada project yang telah kita buat pada window solution explorer, pilih Add, lalu Windows Form.



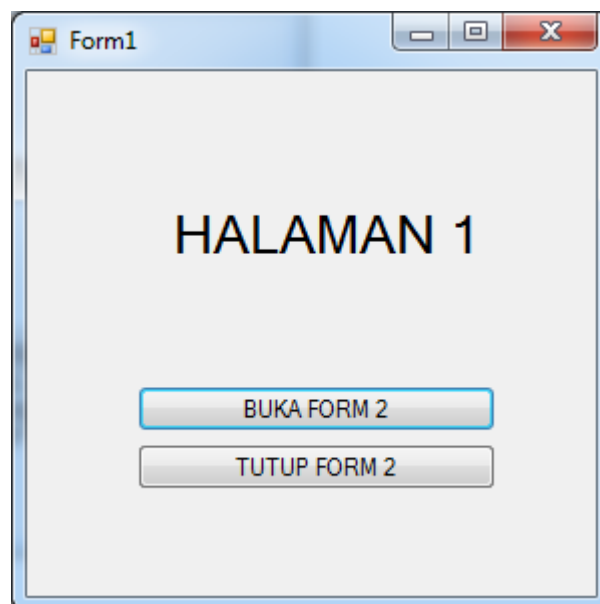
Gambar 30 Menambah Form

lalu akan muncul window dibawah ini, lalu pilih Windows Form

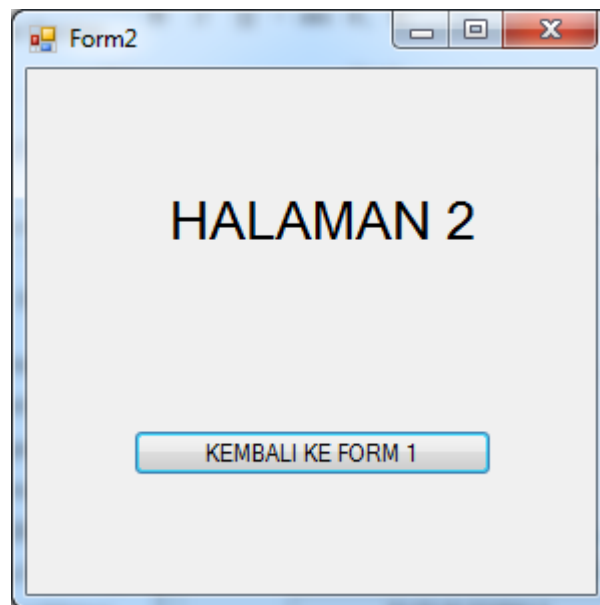


Gambar 31 Pilihan Add Windows Form

Cara berkomunikasi antar form dengan cara sebagai berikut :
Buatlah disain seperti contoh



Gambar 32 Halaman 1 Multi Form



Gambar 33 Halaman 2 Multi Form

Dengan kode program pada form 1 sebagai berikut

```
Public Class Form1
    Private Sub Label1_Click(sender As Object, e As EventArgs) Handles
Label1.Click

        End Sub

    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
        Form2.Visible = True
    End Sub

    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
Button2.Click
        Form2.Close()
    End Sub
End Class
```

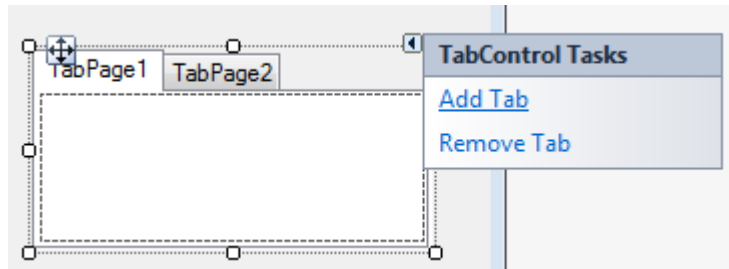
kode program pada form 2 sebagai berikut

```
Public Class Form2
    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
Button2.Click
        Close()
    End Sub
End Class
```

3. Multiwindow dengan Tabulasi

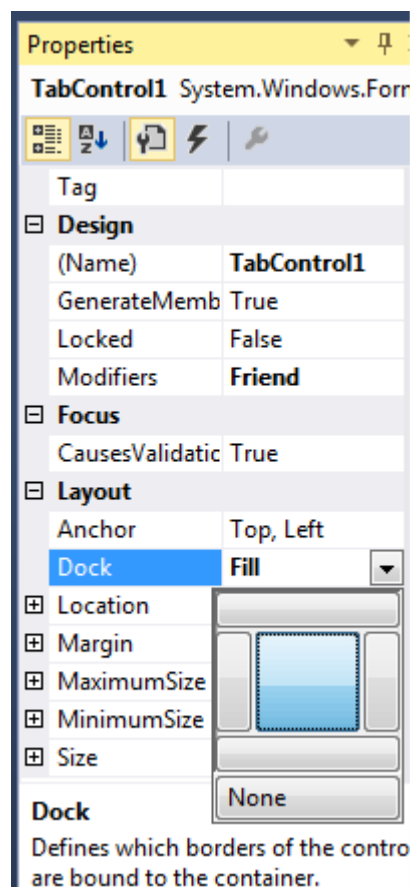
Untuk membuat program multiwindow dengan banyak tabulasi (tab), kita hanya memerlukan menambah komponen TabControl pada window Toolbox.

Untuk menambahkan halaman baru, klik kiri pada tanda panah di ujung kanan atas komponen TabControl dan pilih Add Tab, bila ingin menghapus halaman pilih Remove Tab.



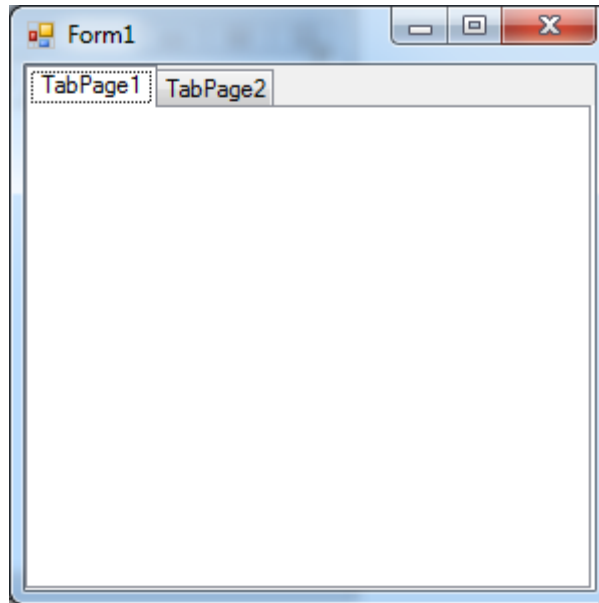
Gambar 34 Multiwindow dengan Tabulasi

Untuk membuat TabControl memenuhi halaman atur properti dari TabControl, Dock



Gambar 35 Propertis TabControl

Maka tampilan dari halaman akan menjadi seperti berikut ini.

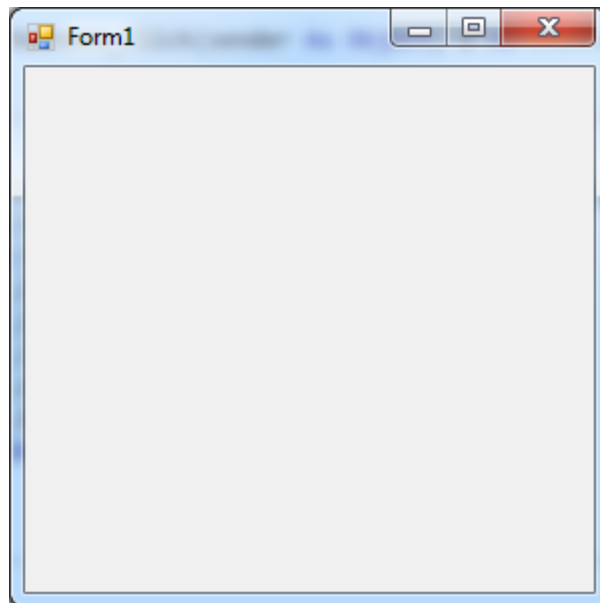


Gambar 36 Contoh Multiwindow dengan Tabulasi

Untuk menghilangkan tulisan TabPage dapat digunakan kode program berikut yang dimasukkan pada event Form 1 > Load

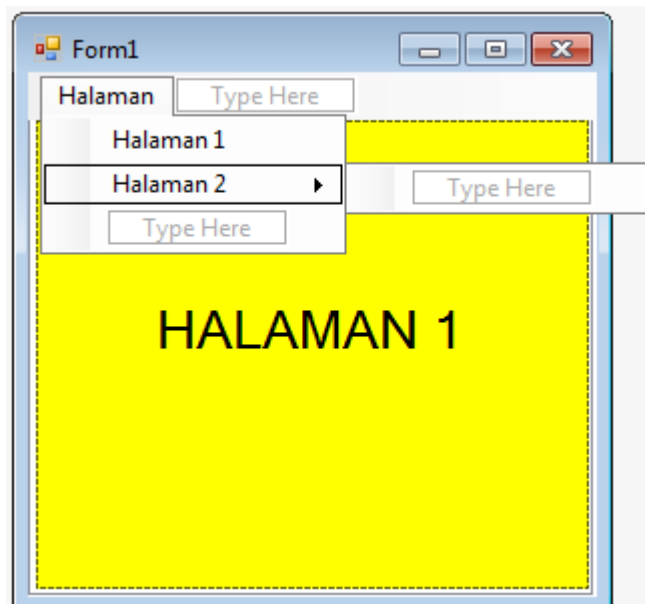
```
Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    For Each tc In {TabControl1}
        With tc
            .SuspendLayout()
            .SizeMode = TabSizeMode.Fixed
            .ItemSize = New Size(0, 1)
            .Appearance = TabAppearance.Buttons
            .ResumeLayout()
        End With
    Next
End Sub
```

Maka tampilan program akan menjadi seperti ini:

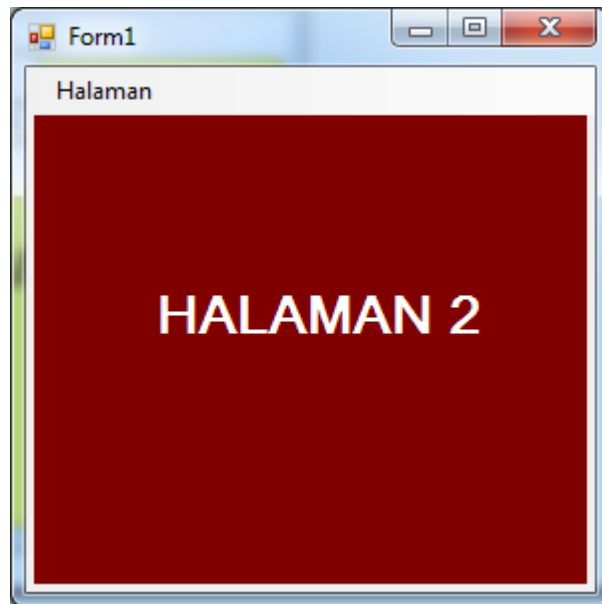


Gambar 37 Contoh Menghilangkan Tulisan tabPage

Kemudian tambahkan MenuStrip sedemikian rupa sehingga menu ada pada form, bukan tab page dan disain tab page sesuai contoh



Gambar 38 Penambahan Menu pada tabPage



Gambar 39 Hasil MultiWindow dengan Modifikasi TabPage

Dan tambahkan program berikut, agar dapat berpindah dari halaman 1 ke halaman 2

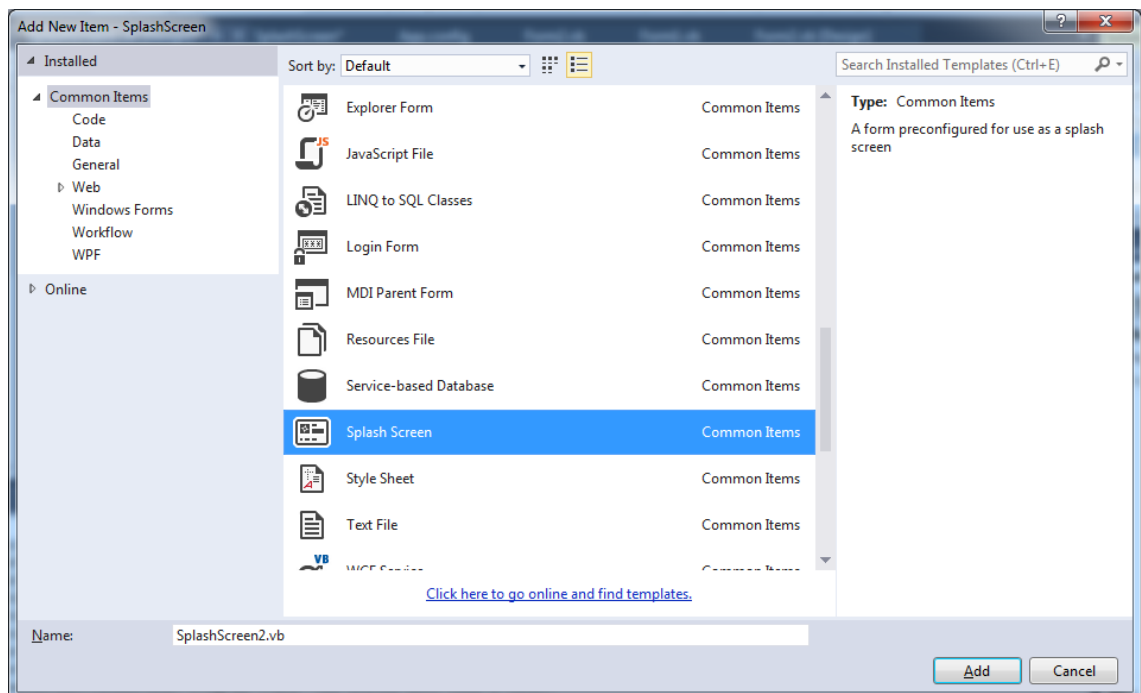
```
Private Sub Halaman1ToolStripMenuItem1_Click(sender As Object, e As
EventArgs) Handles Halaman1ToolStripMenuItem1.Click
    TabControl1.SelectedIndex = 0
End Sub

Private Sub Halaman2ToolStripMenuItem_Click(sender As Object, e As EventArgs)
Handles Halaman2ToolStripMenuItem.Click
    TabControl1.SelectedIndex = 1
End Sub
```

4. Navigasi Splash dengan Timer

Ada kalanya kita menginginkan cover dari produk kita, kalau dalam bentuk software salah satunya dengan membuat splash screen. Membuat splash screen di VB.NET bukanlah hal yang sulit, karena telah disediakan default form untuk splash screen

Buatlah project baru dengan dua form form 1 kita jadikan form utama dan tambahkan form baru dengan memilih jenis form yang Splash Screen



Gambar 40 Form Menambah SplashScreen

dan akan muncul default splash screen sebagai berikut



Gambar 41 New Project SplashScreen

Untuk mengubah background di splashscreen pada properti pilih komponen MainLayoutPanel, lalu cari BackgroundImage ubah lah background sesuai keinginan anda, maka tampilan splash screen akan menjadi sebagai berikut



Gambar 42 Form SplashScreen Termodifikasi

Pengaturan Splash Screen agar dibuka pertama kali, dengan double klik pada My Project pada Solution Explorer, ubah StartUp form pada menu Application menjadi Splash Screen.

Untuk memanggil Form 1, kita gunakan timer, yang berfungsi setelah splash screen tampil maka satu detik kemudian tampilah form 1. Atur interval menjadi 1000, Enabled menjadi True. Kemudian tambahkan kode berikut pada timer dengan cara double klik pada komponen timer

```
Private Sub Timer1_Tick(sender As Object, e As EventArgs) Handles Timer1.Tick
    Me.Hide()
    Form1.Visible = True
    Timer1.Enabled = False
End Sub
```

C. Uji Kemampuan

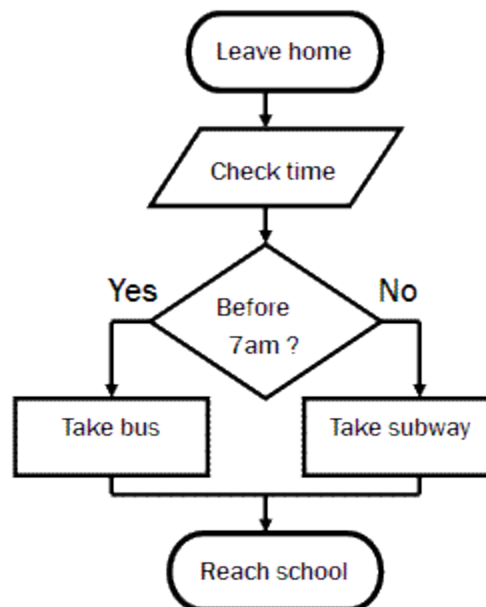
Kerjakan praktikum berikut secara kelompok

Buatlah Program yang telah dicontohkan pada kegiatan belajar ini, dan jadikan dalam 1 project. (Judul Project : Multi_Kelas_No.absen_Nama, masing-masing Judul Form diberi text : Kelas No.absen Nama)

**KEGIATAN BELAJAR 4****PENGKODEAN ALUR PROGRAM****A. Tujuan Pembelajaran**

Setelah melakukan kegiatan pembelajaran siswa dapat :

1. Memahami Tipe data, variabel, dan operator
2. Memahami Array
3. Memahami Struktur kontrol percabangan
4. Memahami Struktur kontrol perulangan
5. Mengolah data menggunakan Tipe data, variabel, dan operator
6. Mengolah data menggunakan Array
7. Mengolah data menggunakan Struktur kontrol percabangan
8. Mengolah data menggunakan Struktur kontrol perulangan

B. Deskripsi Materi

Gambar 43 Stimulus Pengkodean Alur Program

Tugas 4

Amati gambar tersebut!

Apa yang dapat anda dapat pelajari dari gambar tersebut

Apakah mungkin aplikasi itu dibuat menggunakan bahas basic?

Ceritakan pendapat anda, sampaikan dalam kegiatan pembelajaran!

1. Tipe Data, Variabel, Dan Operator

Tipe data adalah suatu bentuk penggolongan jenis data berdasarkan kategori data, ukuran dan kegunaan data yang dapat ditampung oleh sebuah variabel

Tipe data pada Visual Basic ada dua macam, yaitu :

Table 2 Tipe Data Numerik

Tipe Data	Penjelasan	Rentangan Nilai	Banyaknya RAM
Byte	Berupa bilangan bulat positif	0 – 255	1 byte
Integer	Berupa bilangan bulat	(-32.768) – 32.767	2 byte
Long	Berupa bilangan bulat (nilai jauh lebih banyak dari Integer)	(-2.147.483.648) – 2.147.483.648	4 byte
Single	Berupa bilangan pecahan	Untuk nilai negatif : (-3,402823E+38)–(-1,401298E-45) Untuk nilai positif : 1,401298E-45 – 3,402823E+38	4 byte
Double	Berupa Bilangan Pecahan	Untuk nilai negatif : (-1,79769313486232E+308) – (-4,94065645841247E-324) Untuk nilai positif : 4,94065645841247E-324 – 1,79769313486232E+308	8 byte
Currency	Untuk menyimpan nilai mata uang	(-922.337.203.685.477,5808) – 922.337.203.685.477,5807	8 byte
Decimal	Untuk menyimpan nilai desimal	Jika tidak menggunakan angka desimal : +/- 79.228.162.514.264.337.593.543.950.335 28 angka desimal : +/- 7,9228162514264337593543950335	12 byte

Table 3 Tipe Data Non-numerik

Tipe Data	Penjelasan	Rentangan Nilai	Banyaknya RAM
String (Fixed Length)	Untuk menyimpan nilai dalam bentuk karakter	1 – 65.400 Karakter	Sepanjang string
String (Variable Length)	Untuk menyimpan nilai dalam bentuk karakter	0 – 2 Milyar Karakter	10 byte + panjang string
Date	Untuk menyimpan tanggal dan jam	1 Januari 100 – 31 Desember 9999	8 byte
Boolean	Untuk pendeklarasian benar atau salah	Nilai (Benar dan Salah)	2 byte
Object	Untuk menyimpan objek	Semua objek yang dipakai	4 byte
Variant (Numeric)	Merupakan tipe data default	Semua numerik sesuai batasan pada jenis data double	16 byte
Variant (Text)	Merupakan tipe data default	Sama dengan variable string	22 byte + panjang string

Variabel

Variabel adalah sebuah tempat penyimpanan data yang bersifat sementara data yang bersifat sementara dan biasa berubah, artinya data yang tersimpan pada sebuah variable dapat terhapus setelah program dihentikan.

Cara mendeklarasikan variabel

Dim (nama variable) AS (Type Data) "nama_variable adalah nama variable yang dideklarasikan, sedangkan type data adalah tipe data yang tersimpan dalam variable." Pendeklarasikan variable di bagi menjadi 2 kategori antara lain:

a. Deklarasi eksplisit

Suatu pendeklarasian umum yang sangat sering dilakukan dengan memberikan kata kunci As sebagai bentuk umum.

Macam-macam kode penulisan variable pada VB 10 :

- Privat : digunakan untuk membuat variable local
- Dim : digunakan untuk mendaftarkan variable tertentu
- Public : digunakan agar variable tersebut dapat di akses dari prosedur lain
- Static : digunakan agar isi datanya tetap ada di memori setiap kali di gunakan

Contoh :

Dim HARGA As Currency

Dim JUMLAH As Integer

b. Deklarasi implisit

Deklarasi yang tidak memberikan kata kunci As pada semua variable tersebut. Melainkan memberikan simbol di akhir variable.

Table 4 Deklarasi Emplisit

karakter / simbol	type data
%	integer
\$	string
&	long
@	currency
!	single
#	double

Jenis-Jenis Variabel

Ada 2 jenis dari variabel yaitu :

- Variabel local (local variable) adalah variable yang diletakkan didalam prosedur.
- Variabel global (global variable) adalah variable yang diletakkan di luar prosedur atau general.

Ketentuan penulisan nama variable

Ketentuan untuk menulis nama variable adalah sebagai berikut :

- Harus diawali dengan huruf
- Dapat berisi angka, huruf, karakter kusus, misalnya “_”
- Maksimum 255 huruf dan minimum 1 huruf
- Tidak boleh ada spasi
- Tidak boleh menggunakan nama variable yang sama dalam satu bagian

Operator**1. Operator Aritmatika**

Operator Aritmatika adalah merupakan suatu operator perhitungan matematika yang digunakan untuk proses perhitungan matematika. Operator aritmatika dapat dilihat pada table berikut ini :

Table 5 Operator

Operator	Fungsi
+	Penjumlahan
-	Pengurangan
*	Perkalian
/	Pembagian
Mod	Sisa hasil pembagian
^	Perpangkatan

2. Operator Perbandingan

Operator perbandingan adalah merupakan suatu operator yang menyatakan suatu kondisi itu benar atau salah. Operator perbandingan dapat dilihat pada table berikut ini

Table 6 Operator Perbandingan

Operator	Fungsi
<>	Tidak sama dengan
<	Kurang dari
>	Lebih dari
>=	Lebih dari sama dengan
<=	Kurang dari sama dengan
=	Sama dengan
Like	Ciri yang sama
Is	Keterangan

3. Operator Logika

Operator Logika adalah merupakan suatu operator yang dapat membandingkan dua ekspresi atau lebih data logika (Boolean) dan menghasilkan data logika (Boolean) baru. Operator logika dapat dilihat pada table berikut ini:

Table 7 Operator Logika

Operator	Fungsi
NOT	Tidak
AND	Dan
OR	Atau
XOR	Eksklusif atau

Operator Penugasan

Operator penugasan adalah merupakan operator yang bertugas untuk menentukan nilai yang dikeluarkan oleh suatu variabel terhadap kondisi tertentu.

Table 8 Operator Penugasan

Operator	Fungsi
=	Sama dengan

4. Operator Penggabungan

Operator Penggabungan adalah merupakan suatu operator yang dapat digunakan untuk menggabungkan dua buah string atau lebih. Operator Penggabungan dapat dilihat pada table berikut ini :

Table 9 Operator Penggabungan

Operator	Fungsi
+	Menggabungkan tipe data yang sama
&	Menggabungkan tipe data yang tidak sama

2. Array

Array adalah sekumpulan variabel yang memiliki tipe data yang sama dan dinyatakan dengan nama yang sama pula. Array merupakan konsep yang penting dalam pemrograman karena array memungkinkan untuk menyimpan data dalam jumlah banyak dan terindeks.

Array menggunakan indeks integer untuk menentukan urutan elemen-elemen didalamnya, indeks tersebut dimulai dari 0, 1, 2, dan seterusnya

Mendeklarasikan variabel array

Untuk mendeklarasikan variabel array anda bisa menggunakan script dibawah ini

```
Dim strarray() As String
' string menentukan tipe data array
' () menentukan ukuran array
' strarray menentukan nama array
```

Mendefinisikan array

Setelah mendeklarasikan anda perlu untuk mendefinisikan dalam artian anda perlu menentukan ukuran array tersebut

```
strarray = New String((5) - 1) {}
' 5 adalah ukuran array
```

Menetapkan nilai array

Anda perlu menetapkan nilai array yang anda buat tadi dengan script dibawah ini

```
strarray(0) = "satu"    ' Mengisi array pertama dengan satu
strarray(1) = "dua"     ' Mengisi array kedua dengan dua
strarray(2) = "tiga"    ' Mengisi array ketiga dengan tiga
strarray(3) = "empat"   ' Mengisi array keempat dengan empat
strarray(4) = "Lima"    ' Mengisi array kelima dengan lima
```

3. Struktur Kontrol Percabangan

Struktur kontrol percabangan, digunakan untuk memutuskan kode program mana yang akan dikerjakan berdasarkan suatu kondisi

If Then

Struktur Keputusan merupakan suatu blok pernyataan khusus yang urutan pelaksanaan pernyataanya dikendalikan oleh suatu pernyataan bersyarat. bentuk penulisan, yaitu :

IF <kondisi> THEN <pernyataan>

dimana kondisi adalah pernyataan bersyarat pernyataan adalah pernyataan program Visual Basic yang sah. Bila <kondisi> bernilai benar maka <pernyataan> akan dikerjakan. jika menggunakan multiple-line sebagai berikut:

```
IF <syarat kondisi> then
<pernyataan pertama>
```

```
<pernyataan kedua>  
....  
<pernyataan ke-n>  
End if
```

Sebagai contoh,

```
If nilai >= 80 Then  
Textbox1.Text = A  
End if
```

Pernyataan program diatas akan mendeteksi nilai dari variable angka jika nilai adalah besar atau sama dengan 80 maka pada textbox1 bernilai A .Jika nilainya kecil atau tidak sama dengan 80 maka pernyataan tersebut di abaikan

```
If Then Else  
IF <kondisi> THEN  
<pernyataan 1>  
ELSE  
<pernyataan 2>  
END IF
```

Dimana Bila <kondisi1> bernilai True maka <pernyataan 1> akan dikerjakan, tetapi bila <kondisi1> bernilai False maka <pernyataan 2> yang akan dikerjakan.

Contoh :

```
If nilai >=60 then  
Label1.text="Anda LULUS"  
Else  
Label1.text="Maaf Anda Belum Lulus"  
Endif
```

If Then Elseif

```
IF  
<kondisi1> THEN  
<pernyataan jika kondisi 1 bernilai True> ELSEIF  
<kondisi 2>  
<pernyataan jika kondisi 2 bernilai True> ELSEIF  
<kondisi 3>  
<pernyataan jika kondisi 3 bernilai True> ELSE  
<pernyataan jika tidak ada kondisi yang bernilai True>  
END IF
```

Dimana <kondisi> yang merupakan suatu ekspresi logika akan diuji perintah if yang pertama yang ada disebelah kirinya. Jika <kondisi> benar maka blok perintah yang terletak di bawahnya akan dikerjakan sampai menemui dan jika tidak, maka perintah Elself yang dikerjakan dan seterusnya hingga menemui atau Else lalu langsung melompat ke pernyataan End If.

Select Case

Select Case sebenarnya serupa dengan struktur If...Then...Else, hanya saja struktur ini lebih efisien, mudah dibaca dan mudah digunakan terutama jika percabangan dibuat berdasarkan satu variabel kunci, atau uji kasus (test case). Struktur ini biasanya digunakan untuk percabangan dengan pilihan lebih dari 2 (percabangan dengan banyak pilihan).

Tata penulisan untuk struktur Select Case adalah sebagai berikut:

```
SELECT CASE
<pilihan> CASE
<pilihan 1>
<pernyataan program yang dilaksanakan jika pilihan = pilihan1>

CASE <pilihan 2>
<pernyataan program yang dilaksanakan jika pilihan = pilihan2>

CASE <pilihan n>
<pernyataan program yang dilaksanakan jika pilihan = pilihann>

[CASE ELSE]
<pernyataan program yang dilaksanakan jika pilihan1 sampai n tidak ada yang
sesuai maka pernyataan ini akan dikerjakan>
END SELECT
```

4. Struktur Kontrol Perulangan

Struktur pengulangan digunakan untuk mengulang perintah, sehingga kita tidak perlu menulis ulang perintah sebanyak pengulangan yang diinginkan. Beberapa jenis struktur kontrol pengulangan adalah sebagai berikut :

FOR NEXT

Bentuk ini digunakan untuk melakukan pengulangan (iterasi) suatu blok program beberapa kali yang ditentukan nilai awal dan nilai akhir (banyak pengulangannya sudah pasti). Pada struktur kontrol For...Next, kita tidak perlu menuliskan sebuah kondisi untuk diuji oleh Visual Basic. Kita hanya perlu menuliskan nilai awal dan akhir variabel penghitung. Nilai variabel penghitung ini akan secara otomatis bertambah atau berkurang setiap kali sebuah pengulangan (loop) dilaksanakan. Standar penulisan struktur ini adalah

```
For variabel as tipeData(angka) = mulai to akhir [STEP]
    Proses()
Next
```

- Variabel : adalah kode simbol yang mewakili sebuah angka dari tipeData
- Tipe Data : adalah untuk memberitahu sistem jenis variabel angka yang akan digunakan, karena didalam pemograman banyak variabel angka yang tersedia, secara keseluruhan semua jenis adalah sama hanya berbeda adalah pada setiap jenis

variabelnya jumlah angka yang tersedia berbeda-beda contohnya adalah : byte, integer, double .

- Mulai : adalah nilai awal perulangan yang ingin diproses, seharusnya dari 0 tapi dalam penerapannya terkadang value yang diperlukan lebih dari 0.
- Akhir : adalah nilai akhir dari sebuah perulangan itu.
- Step : ini menjelaskan berapa besar langkah yang diinginkan untuk menjalankan looping

Untuk contoh perulangan tersebut dalam program lihatlah kode dibawah ini :

```
For i As Integer = 0 To 10
    ListBox3.Items.Add("Loop : " & i)
Next
```

Dari perintah diatas dapat dipahami bahwa perulangan akan dilakukan sebanyak 11 kali mulai dari 0 sampai dengan 10

Atau menggunakan langkah / step pada proses pengulangannya. Kodingnya seperti ini:

```
For i As Integer = 0 To 10 Step 3
    ListBox3.Items.Add("Loop : " & i)
Next
```

Dari perintah diatas dapat difahami bahwa perulangan akan dilakukan sebanyak 11 kali mulai dari 0 sampai dengan 10

DO WHILE LOOP

Bentuk perulangan Do While Loop adalah bentuk perulangan yang memerintahkan program untuk menjalankan proses yang berulang apabila kondisi persyaratan yang ada masih bernilai true. Jika kondisi persyaratan sudah bernilai False maka program akan melanjutkan proses ke kode program berikutnya. Pengulangan ini memiliki 2 bentuk sintaks. Perbedaan kedua bentuk tersebut terletak pada kondisi perulangan ditempatkan.

Bentuk 1	Bentuk 2
DO WHILE <kondisi>	DO WHILE <kondisi>
Proses	Proses
LOOP	LOOP while <kondisi>

Keterangan :

<Proses> akan diulang selama <kondisi> bernilai TRUE. Pengulangan berhenti bila <kondisi> sudah bernilai FALSE

Untuk contoh perulangan ini lihatlah sintaks dibawah ini:

```
Dim i As Integer = 0
Do While
    ListBox2.Items.Add("Loop : " & i)
    i += 1
Loop While i < 10
```

DO UNTIL LOOP

Perulangan ini akan menjalankan proses perulangan sampai kondisi perulangan dipenuhi. Apabila kondisi perulangan masih bernilai False, maka proses akan dijalankan terus menerus. Perulangan ini memiliki 2 bentuk sintaks sebagai berikut :

Bentuk 1	Bentuk 2
DO UNTIL <kondisi>	DO <kondisi>
<proses>	<proses>
LOOP	LOOP UNTIL

Keterangan :

<Proses> akan diulang selama <kondisi> bernilai TRUE. Pengulangan berhenti bila <kondisi> sudah bernilai TRUE.

Untuk contoh perulangan Do-Until lihatlah code dibawah ini :

```
Dim i As Integer = 0
Do
    ListBox1.Items.Add("Loop : " & i)
    i += 1
Loop Until i = 10
```

C. Uji Kemampuan

Kerjakan praktikum berikut secara mandiri

Buatlah Program yang berisi form :

1. Splash Screen (Berisi : Kelas, No. Absen, Nama)
2. Form Login (data username dan password disimpan dalam array, data diisi saat form load)
 - Jika username dan password benar maka akan tampil form utama yang menampilkan home (berisi menu, menampilkan data dalam listbox berisi no, nama, alamat, tanggal lahir data berasal dari array, tab page berikutnya berisi entry data)
 - Jika pengisian username dan password salah 3x maka form menutup sendiri.
3. Form Pengisian data (username, password, nama, tempat tanggal lahir, gender(combobox), alamat, nomor telrpon, hoby(CheckedListBox), data disimpan dalam array, data dalam array ditampilkan di DataGridView)

Contoh tampilan :

nama	alamat	gender	hobi
»			

Gambar 44 Contoh Uji Kemampuan Pengkodean Alur Program

(Nama Project : Algo_Kelas_no.absen_Nama, Judul Form : Kelas No.absen Nama)



KEGIATAN BELAJAR 5

PENGOLAHAN INPUT USER

A. Tujuan Pembelajaran

Setelah melakukan kegiatan pembelajaran siswa dapat :

1. Memahami teknik Konversi data
2. Memahami teknik Pengelolaan input user pada operasi aritmatika dan logika
3. Memahami teknik Pengelolaan input user pada operasi string
4. Menggunakan teknik Konversi data
5. Menggunakan teknik Pengelolaan input user pada operasi aritmatika dan logika
6. Menggunakan teknik Pengelolaan input user pada operasi string

B. Deskripsi Materi

The image shows a Windows-style login form. At the top, a grey header bar contains the text 'Login enter your credentials'. Below this, the form has two input fields: 'Username:' and 'Password:'. A 'Login' button is positioned below the password field. At the bottom of the form, a message box with a red 'X' icon displays the text 'Please enter a correct username and password'.

Gambar 45 Stimulus Pengolahan Input User

Tugas 5

Amati gambar!

Apa yang sedang anda lihat?

Mungkinkah tampilan dan program gambar diatas dibuat di VB.NET?

Ceritakan pendapat anda, sampaikan dalam kegiatan pembelajaran!

1. Konversi Data

Visual Basic menyediakan beberapa fungsi untuk mengubah tipe data. Tabel berikut adalah fungsi-fungsi tersebut:

Table 10 Konversi Data

Conversion To	Function
Boolean	Cbool
Byte	Cbyte
Currency	Ccur
Date	Cdate
Decimals	Cdec
Double	CDbl
Integer	Cint
Long	CLng
Single	CSng
String	CStr
Variant	Cvar
Error	CVErr

Dalam beberapa kasus tipe data dapat terkonversi secara otomatis (implicit). Sebagai contoh memberikan nilai numerik pada variable string.

Contoh:

```
Dim intNomor as Integer
```

```
Dim strMsg as String
```

```
intNomor = 2253
```

```
strMsg = intNomor
```

```
Print strMsg
```

2. Pengelolaan Input User Pada Operasi Aritmatika dan Logika

Kadang kala konversi harus secara eksplisit menggunakan fungsi konversi.

CStr

Fungsi Cstr mengubah suatu nilai ke tipe data string. Salah satu penggunaanya adalah saat kita ingin memasukan nilai numerik menjadi nilai dari sebuah textbox.

Contoh:

```
Dim dblNilai As Double
```

```
Text1.Text = CStr(dblNilai)
```

Str

Fungsi Str juga mengkonversi nilai ke tipe data string, samahalnya dengan CStr.

Contoh:

```
Dim str1 As String
Dim int1 As Integer
str1 = Str(int1)
```

Saat membuat program biasanya kita berkali-kali mempresentasikan nilai numerik dalam string.

CDbl

Fungsi CDbl mengubah nilai ke tipe data Double. Juga membuat nilai numerik dari string dengan karakter numerik. It also extracts the numeric value from a string.

Contoh 1: Jika kita tidak menggunakan fungsi CDbl, program akan menghasilkan error overflow.

```
Dim m As Integer, v As Double
m = 30887
v = CDbl(m) * 31880
Print v
```

Contoh 2:

```
Dim dblValue As Double
dblValue = CDbl(Text1.Text)
Print dblValue + 1
```

Contoh 3:

```
Dim dblValue As Double
dblValue = CDbl("55")
Print dblValue + 1
```

Val

Terkadang kita dapat menggunakan fungsi Val function selain fungsi CDbl. Fungsi Val hanya mengubah nilai numerik yang ada dalam string menjadi Double. Tetapi tidak dapat mengubah dari nilai bertipe data numerik.

Contoh:

```
Dim l As Double
l = Val(Text1.Text)
```

CLnt

Fungsi CLnt function mengubah nilai numerik ke integer.

Contoh:

Dim m As Integer

m = CLnt(856.878) 'menghasilkan 857 setelah pembulatan.

m = CLnt(-4.7) 'menghasilkan -5 setelah pembulatan.

Nilai yang ingin diubah harus dalam range tipe data integer.

856.878 and -5.7 masih dalam range tipe data integer.

CLng

Fungsi CLng mengubah ke tipe Long.

Contoh:

Dim lng As Long

lng = CLng(1156593647.87656) 'membulatkan nilai ke atas

Debug.Print lng

CBool

Fungsi CBool mengubah nilai menjadi tipe data Boolean, menjadi True atau False.

Nilai 0 berubah jadi False, sedangkan semua nomor lainnya menjadi True. Sementara untuk strings "True" menjadi True dan "False" menjadi False.

Contoh:

Dim blnX as Boolean

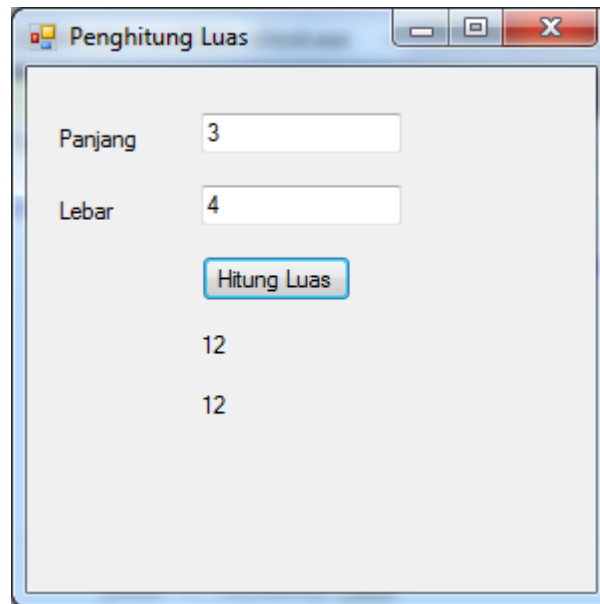
blnX = cbool(0) 'menghasilkan False

blnX = cbool(1) 'menghasilkan true

blnX = cbool(345) 'menghasilkan True

Untuk fungsi CByte, CSng, CVar, CCur, CDate dan CDecimal, kita dapat menggunakannya dijelaskan pada contoh-contoh diatas.

Tetapi beberapa konversi sudah dilakukan secara langsung tanpa perlu fungsi pengubah tipe variabel. Contoh program dibawah menunjukkan hasil yang sama, baik diubah tipe data maupun tidak.



Gambar 46 Contoh Form Menghitung Luas

Kode programnya ada dibawah ini.

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles  
Button1.Click  
    Dim panjang, lebar, luas As Integer  
  
    panjang = CInt(TextBox1.Text)  
    lebar = CInt(TextBox2.Text)  
    luas = panjang * lebar  
  
    LHasil1.Text = luas  
  
    panjang = TextBox1.Text  
    lebar = TextBox2.Text  
    luas = panjang * lebar  
  
    LHasil2.Text = luas.ToString  
End Sub
```

3. Pengelolaan Input User Pada Operasi String

Kelas String dari kerangka .NET menyediakan banyak built-in metode untuk memfasilitasi perbandingan dan manipulasi string. Ini sekarang masalah sepele untuk mendapatkan data tentang string, atau untuk membuat string baru dengan memanipulasi string saat ini. Visual Basic .NET bahasa. Juga memiliki metode yang melekat yang menduplikasi banyak dari fungsionalitas.
Jenis Metode Manipulasi String

Pada bagian ini Anda akan membaca tentang beberapa cara berbeda untuk menganalisis dan memanipulasi string Anda. Beberapa metode adalah bagian dari bahasa Visual Basic, dan lain-lain yang melekat dalam kelas String.

Visual Basic NET metode. Yang digunakan sebagai fungsi yang melekat dari bahasa. Mereka dapat digunakan tanpa kualifikasi dalam kode Anda. Contoh berikut menunjukkan penggunaan umum dari perintah Visual Basic NET string manipulasi.

```
Dim aString As String = "someString"  
Dim bString As String  
bString = Mid(aString, 3, 3)
```

Dalam contoh ini, fungsi Mid melakukan operasi langsung pada aString dan memberikan nilai ke bString.

Anda juga dapat memanipulasi string dengan metode dari kelas String. Ada dua jenis metode dalam String: metode bersama dan metode instan.

Sebuah metode berbagi adalah metode yang berasal dari kelas String itu sendiri dan tidak memerlukan sebuah instance dari kelas itu untuk bekerja. Metode-metode ini dapat dikualifikasikan dengan nama kelas (String) bukan dengan sebuah instance dari kelas String. Sebagai contoh:

```
Dim aString As String  
bString = String.Copy("A string literal")
```

Dalam contoh sebelumnya, metode String.Copy adalah metode statis, yang bertindak atas ekspresi itu diberikan dan memberikan nilai yang dihasilkan untuk bString.

Metode Instance, sebaliknya, berasal dari contoh tertentu dari String dan harus berkualitas dengan nama contoh. Sebagai contoh:

```
Dim aString String = "String A"  
Dim bString As String  
bString = aString.Substring(2, 6) 'bString = "String"
```

Dalam contoh ini, metode Substring adalah metode dari contoh dari String (yaitu, aString). Ia melakukan sebuah operasi pada aString dan memberikan nilai bahwa untuk bString.

Tidak ada dan String

Runtime Visual Basic dan NET Framework. Tidak ada yang berbeda mengevaluasi ketika datang ke string. Perhatikan contoh berikut:

```
Dim MyString As String = "Ini adalah string saya"  
Dim StringLength As Integer  
'Secara eksplisit mengatur string ke Nothing.  
MyString = vbNull  
  
'StringLength = 0
```

```
StringLength = Len (MyString)
'Garis ini, bagaimanapun, menyebabkan pengecualian untuk dilempar.
StringLength = MyString.Length
```

. Runtime Visual Basic NET mengevaluasi Tidak sebagai string kosong, yaitu, "". NET Framework., Bagaimanapun, tidak, dan akan melemparkan pengecualian setiap kali dilakukan usaha untuk melakukan operasi string pada apa-apa.

Membandingkan String

Anda dapat membandingkan dua string dengan menggunakan metode `String.Compare`. Ini adalah metode, statis overload dari kelas string dasar. Dalam bentuk yang paling umum, metode ini dapat digunakan untuk secara langsung membandingkan dua string berdasarkan urutan abjad semacam mereka. Hal ini mirip dengan fungsi `Fungsi StrComp Visual Basic`. Contoh berikut menggambarkan bagaimana metode ini digunakan:

Salinan

```
Dim myString As String = "abjad"
SecondString Dim As String = "Order"
Dim hasil As Integer
hasil = String.Compare (myString, secondString)
```

Metode ini mengembalikan sebuah integer yang menunjukkan hubungan antara dua string dibandingkan berdasarkan urutan penyortiran. Sebuah nilai positif bagi hasilnya menunjukkan bahwa string pertama lebih besar dari string kedua. Sebuah hasil negatif menunjukkan string pertama lebih kecil, dan nol menunjukkan kesetaraan antara string. Setiap string, termasuk string kosong, mengevaluasi ke yang lebih besar daripada referensi null.

Overloads tambahan dari metode `String.Compare` memungkinkan Anda untuk menunjukkan apakah atau tidak untuk mengambil format kasus atau budaya ke account, dan untuk membandingkan substring dalam string yang disediakan. Untuk informasi lebih lanjut tentang cara untuk membandingkan string, lihat Metode `String.Compare`. Metode terkait termasuk Metode `String.CompareOrdinal` dan Metode `String.CompareTo`.

Mencari String Dalam Strings Anda

Ada saat-saat itu berguna untuk memiliki data tentang karakter dalam string Anda dan posisi karakter-karakter di dalam string Anda. Sebuah string dapat dianggap sebagai array dari karakter (contoh `Char`); Anda dapat mengambil karakter tertentu dengan referensi indeks dari karakter yang melalui properti Karakter. Sebagai contoh:

Salinan

```
Dim myString As String = "ABCDE"
Dim myChar Sebagai Char
myChar = myString.Chars (3) 'myChar = "D"
```

Anda dapat menggunakan metode `String.IndexOf` untuk mengembalikan indeks di mana karakter tertentu ditemui, seperti dalam contoh berikut:

Salinan

```
Dim myString As String = "ABCDE"
Dim myInteger As Integer
myInteger = myString.IndexOf ("D") 'myInteger = 3
```

Dalam contoh sebelumnya, metode `indexOf` dari `myString` digunakan untuk mengembalikan indeks yang sesuai dengan contoh pertama dari karakter "C" dalam string. `IndexOf` adalah metode kelebihan beban, dan overloads lain memberikan metode untuk mencari salah satu dari set karakter, atau untuk mencari string dalam string Anda, antara lain. Visual Basic `InStr` perintah. .NET juga memungkinkan Anda untuk melakukan fungsi yang sama. Untuk informasi lebih lanjut dari metode ini, lihat Metode `String.IndexOf` dan Fungsi `InStr`. Anda juga dapat menggunakan Metode `String.LastIndexOf` untuk mencari kejadian terakhir dari sebuah karakter dalam string Anda.

Membuat String Baru dari Old

Bila menggunakan string, Anda mungkin ingin memodifikasi string Anda dan membuat yang baru. Anda mungkin ingin melakukan sesuatu yang sederhana seperti mengubah seluruh string menjadi huruf besar, atau trim off spasi tambahan, atau Anda mungkin ingin melakukan sesuatu yang lebih kompleks, seperti penggalan substring dari string Anda. Kelas `System.String` menyediakan berbagai macam pilihan untuk memodifikasi, memanipulasi, dan membuat string baru dari yang lama Anda.

Untuk menggabungkan beberapa string, Anda dapat menggunakan operator Rangkaian (& atau +). Anda juga dapat menggunakan Metode `String.Concat` untuk menyatukan serangkaian string atau string yang terkandung dalam obyek. Sebuah contoh dari metode `String.Concat` berikut:

Salinan

```
Dim aString As String = "A"
Dim bString As String = "B"
Dim cString As String = "C"
Dim dString As String = "D"
Dim myString As String
' MyString = "ABCD"
myString = String.Concat (aString, bString, cString, dString)
```

Anda dapat mengkonversi string ke semua huruf besar atau huruf kecil semua string baik menggunakan Visual Basic .NET fungsi `UCase` Fungsi dan Fungsi `lcase` atau Metode `String.ToUpper` dan metode `String.ToLower` Metode. Sebuah contoh ditunjukkan di bawah ini:

Salinan

```
MyString Dim As String = "atas atau huruf kecil"
Dim newString As String
' NewString = "UPPER CASE ATAU RENDAH"
newString = UCase (myString)
' NewString = "atas atau huruf kecil"
newString = lcase (myString)
```

```
'NewString = "UPPER CASE ATAU RENDAH"  
newString = myString.ToUpper  
'NewString = "atas atau huruf kecil"  
newString = myString.ToLower
```

Metode `String.Format` dan perintah Dasar. Format Visual NET dapat menghasilkan string baru dengan menerapkan format ke string yang diberikan. Untuk informasi tentang perintah-perintah ini, lihat Fungsi Format atau Metode `String.Format`.

Anda mungkin pada waktu perlu menghapus trailing atau memimpin spasi dari string Anda. Misalnya, Anda mungkin parsing string yang dimasukkan ruang untuk tujuan keselarasan. Anda dapat menghapus spasi menggunakan fungsi Metode `String.Trim`, atau Visual Basic fungsi `Trim`. NET. Contoh ditunjukkan: Salinan

```
Dim spaceString As String = _  
"String ini akan memiliki ruang dihapus"  
Dim oneString As String  
Dim twoString As String  
'Ini akan menghapus semua trailing dan ruang terkemuka.  
oneString = spaceString.Trim  
'Ini juga menghapus semua trailing dan ruang terkemuka.  
twoString = Trim (spaceString)
```

Jika Anda hanya ingin menghapus spasi, Anda dapat menggunakan Metode `String.TrimEnd` atau fungsi `RTRIM`, dan untuk ruang terkemuka Anda dapat menggunakan Metode `String.TrimStart` atau fungsi `LTRIM`. Untuk rincian lebih lanjut, lihat `LTRIM`, `RTRIM`, dan Potong Fungsi fungsi.

Fungsi `String.Trim` dan fungsi terkait juga memungkinkan Anda untuk menghapus contoh karakter tertentu dari ujung-ujung tali Anda. Trim Contoh berikut semua leading dan trailing karakter contoh "#": Salinan

```
MyString Dim As String = "##### Hapus !#####" mereka  
Dim oneString As String  
OneString = myString.Trim ("#")
```

Anda juga dapat menambahkan terkemuka atau trailing karakter dengan menggunakan Metode `String.PadLeft` atau Metode `String.PadRight`.

Jika Anda memiliki karakter berlebih dalam tubuh string Anda, Anda dapat cukai mereka dengan menggunakan Metode `String.Remove`, atau Anda dapat menggantinya dengan karakter lain menggunakan Metode `String.Replace`. Sebagai contoh: Salinan

```
AString Dim As String = "Ini adalah Str @ o @ saya o @ ing"  
Dim myString As String
```



```
Dim anotherString As String
'MyString = "Ini adalah String Saya"
myString = aString.Remove (14, 5)
'AnotherString = "Ini adalah String lain"
anotherString = myString.Replace ("saya", "lain")
```

Anda dapat menggunakan metode `String.Replace` untuk menggantikan salah satu karakter individu atau string karakter. Visual Basic Pernyataan Mid. NET juga dapat digunakan untuk menggantikan string interior dengan string lainnya.

Anda juga dapat menggunakan Metode `String.Insert` untuk memasukkan string dalam string lain, seperti dalam contoh berikut:
Salinan

```
AString Dim As String = "Ini adalah Stng saya"
Dim myString As String
"Hasil dalam nilai" Ini adalah String Saya ".
myString = aString.Insert (13, "ri")
```

Parameter pertama bahwa metode `String.Insert` dibutuhkan adalah indeks dari karakter string yang akan disisipkan setelah, dan parameter kedua adalah string yang akan dimasukkan.

Anda dapat menggabungkan array dari string sama dengan string pemisah dengan menggunakan Metode `String.Join`. Berikut adalah suatu contoh:
Salinan

```
Dim shoppingItem (2) As String
Dim shoppingList As String
shoppingItem (0) = "Susu"
shoppingItem (1) = "Telur"
shoppingItem (2) = "Roti"
shoppingList = String.Join (" , ", shoppingItem)
```

Nilai `shoppingList` setelah menjalankan kode ini adalah "Susu, Telur, Roti". Perhatikan bahwa jika array Anda memiliki anggota kosong, metode masih menambahkan string pemisah antara semua kasus kosong dalam array Anda.

Anda juga dapat membuat array dari string dari string tunggal dengan menggunakan Metode `String.split`. Contoh berikut menunjukkan kebalikan dari contoh sebelumnya: dibutuhkan daftar belanja dan mengubahnya menjadi sebuah array dari item belanja. Pemisah dalam hal ini adalah sebuah contoh dari tipe data `Char`, oleh karena itu ditambahkan dengan karakter literal jenis `c`.
Salinan

```
Dim shoppingList As String = "Susu, Telur, Roti"
Dim shoppingItem (2) As String
shoppingItem = shoppingList.Split (" , " c)
```

Visual Basic Fungsi Mid. NET dapat digunakan untuk menghasilkan substring dari string Anda. Contoh berikut menunjukkan fungsi ini digunakan:

Salinan

```
AString Dim As String = "Kanan Kiri Pusat"  
Dim rString, lString, mString As String  
'RString = "Kanan"  
rString = Mid (aString, 13)  
'LString = "Kiri"  
lString = Mid (aString, 1, 4)  
'MString = "Pusat"  
mString = Mid (aString, 6,6)
```

Substring dari string Anda juga dapat dihasilkan dengan menggunakan Metode String.Substring. Metode ini membutuhkan dua argumen: indeks karakter mana substring adalah untuk memulai, dan panjang substring. Metode String.Substring beroperasi seperti fungsi Mid. Sebuah contoh ditunjukkan di bawah ini:

Salinan

```
AString Dim As String = "Kanan Kiri Pusat"  
Dim As String substring  
'Substring = "Pusat"  
substring = aString.SubString (5,6)
```

Ada satu perbedaan yang sangat penting antara metode String.SubString dan fungsi Mid. Fungsi Mid membutuhkan sebuah argumen yang menunjukkan posisi karakter untuk substring untuk memulai, dimulai dengan posisi 1. Metode String.SubString mengambil indeks dari karakter dalam string di mana substring adalah untuk memulai, dimulai dengan posisi 0. Jadi, jika Anda memiliki string "ABCDE", karakter individu nomor 1,2,3,4,5 untuk digunakan dengan fungsi Mid, tetapi 0,1,2,3,4 untuk digunakan dengan fungsi System.String

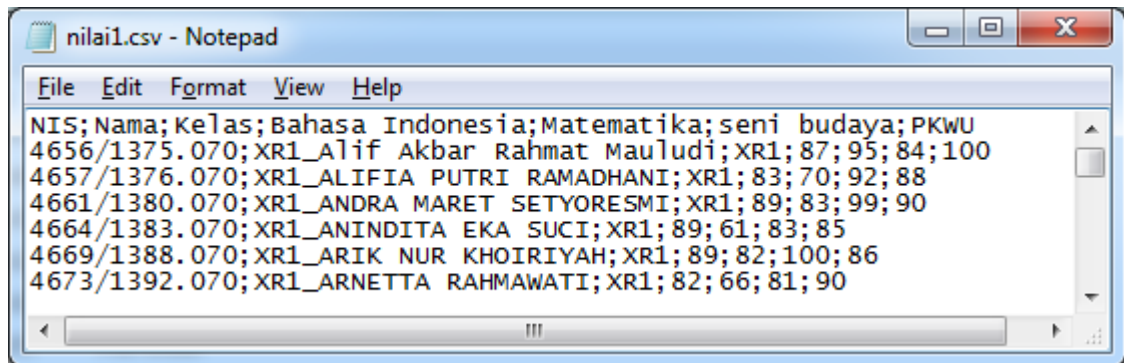
Import Data

Visual Basic mampu untuk mengolah data dari aplikasi lain, salah satunya excel dengan format .csv. Dengan menambahkan komponen OpenFileDialog yang berguna untuk membuka file dari luar VB

Tambahkan kode program dibawah untuk membuka file .csv yang diinginkan

```
OpenFileDialog1.ShowDialog()  
If OpenFileDialog1.FileName <> "" Then  
    buf = My.Computer.FileSystem.ReadAllText(OpenFileDialog1.FileName)
```

Pastikan file .csv yang anda buka dapat dibuka melalui notepad dengan tampilan sebagai berikut :



Berikut diberikan contoh program untuk mengimpor data dari csv dan mengolahnya untuk diurutkan (dengan metode merge sort) dan menampilkan data pada RichTextBox

```
Public Class Form6
```

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
    Dim buf As String
    Dim nama(200), nis(200), kelas(200) As String
    Dim bIndonesia(200), matematika(200), seniBudaya(200), pkwu(200),
rata(200) As Integer
    Dim i, j, k, l, jmlData As Integer
    Dim sortMatematika(200), noIndex(200), sortIndex(200) As Integer
    Dim bawah, atas, tengah, t, b As Integer

    OpenFileDialog1.ShowDialog()
    If OpenFileDialog1.FileName <> "" Then
        buf = My.Computer.FileSystem.ReadAllText(OpenFileDialog1.FileName)
        RichTextBox1.Text = buf

        i = 0
        While buf.IndexOf(";") > 0
            noIndex(i) = i
            buf = buf.Remove(0, buf.IndexOf(";") + 1)
            nis(i) = buf.Substring(0, buf.IndexOf(";"))
            buf = buf.Remove(0, buf.IndexOf(";") + 1)
            nama(i) = buf.Substring(0, buf.IndexOf(";"))
            buf = buf.Remove(0, buf.IndexOf(";") + 1)
            kelas(i) = buf.Substring(0, buf.IndexOf(";"))
            buf = buf.Remove(0, buf.IndexOf(";") + 1)
            bIndonesia(i) = buf.Substring(0, buf.IndexOf(";"))
            buf = buf.Remove(0, buf.IndexOf(";") + 1)
            matematika(i) = buf.Substring(0, buf.IndexOf(";"))
            buf = buf.Remove(0, buf.IndexOf(";") + 1)
            seniBudaya(i) = buf.Substring(0, buf.IndexOf(";"))
            buf = buf.Remove(0, buf.IndexOf(";") + 1)
            pkwu(i) = buf.Substring(0, buf.IndexOf(Chr(Keys.Return)))
            buf = buf.Remove(0, buf.IndexOf(Chr(Keys.Return)) + 2)
        End While
    End If
End Sub
```

```
RichTextBox1.Text = buf
i += 1
End While

jmlData = i
MessageBox.Show("Jumlah data = " + Convert.ToString(jmlData))

j = 1
While j <= jmlData
    j = j * 2
    k = 0
    RichTextBox1.Text = RichTextBox1.Text + "tahap " +
        Convert.ToString(j) + vbCrLf
    For i = 0 To jmlData Step j
        bawah = i
        b = bawah

        atas = i + j - 1
        If atas > jmlData Then
            atas = jmlData - 1
        End If

        tengah = bawah + (j - 1) \ 2
        t = tengah + 1

        While b <= tengah And t <= atas

            If matematika(b) <= matematika(t) Then
                sortMatematika(k) = matematika(b)
                sortIndex(k) = noIndex(b)
                b += 1
            Else
                sortMatematika(k) = matematika(t)
                sortIndex(k) = noIndex(t)
                t += 1
            End If

            k += 1
        End While

        If b > tengah Then
            For l = t To atas
                sortMatematika(k) = matematika(l)
                sortIndex(k) = noIndex(l)
                k += 1
            Next
        Else
            For l = b To tengah
                sortMatematika(k) = matematika(l)
                sortIndex(k) = noIndex(l)
                k += 1
            Next
        End If
    Next
Next

For i = 0 To jmlData - 1
    matematika(i) = sortMatematika(i)
```

```
        noIndex(i) = sortIndex(i)
        RichTextBox1.Text = RichTextBox1.Text +
Convert.ToString(matematika(i)) + vbCrLf
    Next

    MessageBox.Show("Tahap " + Convert.ToString(j))

End While
RichTextBox1.Text = RichTextBox1.Text + "Proses Akhir" + vbCrLf
For i = 0 To jmlData - 1
    RichTextBox1.Text = RichTextBox1.Text + nama(noIndex(i)) + vbTab
+
    Convert.ToString(matematika(i)) + vbCrLf
Next
End If
End Sub
End Class
```

C. Uji Kemampuan

Kerjakan praktikum berikut secara mandiri

Buatlah Program kalkulator yang terdiri dari (penjumlahan, pengurangan, perkalian, pembagian, pangkat, akar untuk bilangan tipe data real dan konversi bilangan biner, desimal, heksadesimal). (Nama Project : String_Kelas_no.absen_Nama, Judul Form : Kelas No absen Nama)

DAFTAR PUSTAKA

- Sianipar, R.H., 2014, *Pemrograman Visual Basic .Net*, Bandung, INFORMATIKA
- Halvorson, Michael, Microsoft Visual Basic 2013 Step by Step
<https://jurisma1992.files.wordpress.com/2F2015%2F03%2Fms-microsoft-visual-basic-2013-step-by-step.pdf>. 17 Juni 2015
- Hendra, ST, Dasar Pemrograman VB.NET
<http://www.kemendag.go.id/files/regulasi/2012/05/Dasar%20Pemrograman%20VB.Net.pdf>. 26 Desember 2015
- Hendra, ST. Pemrograman Berorientasi Objek pada VB.Net
<https://dataiman.googlecode.com/files/Pemrograman%20Berorientasi%20Objek%20pada%20VB.Net.pdf>. 26 Desember 2015
- Hendra, ST, Pemrograman GDI+ pada VB.Net
http://ocw.stikom.edu/course/download/2013/05/07-12-2011.10.38.15_970230_410113007_Pemrograman-Visual-I-SI-KA_P1_Pert12_1.pdf. 26 Desember 2015
- Junaedi, Danang, Setup & Deployment VB.NET
<https://danangjunaedi.files.wordpress.com/2009/08/sesi-15-package-deployment.pdf>. 26 Desember 2015
- Ramez Debbas, How to publish your Windows 8 App to the Store
<http://blogs.msdn.com/b/cdnstudents/archive/2012/11/30/publishing-windows-8-app-to-the-windows-store-how-to.aspx> 29/12/2015. 26 Desember 2015
- Events and Event Handlers [https://msdn.microsoft.com/en-us/library/2z7x8ys3\(v=vs.90\).aspx](https://msdn.microsoft.com/en-us/library/2z7x8ys3(v=vs.90).aspx). 26 Desember 2015
- Mengenal Bahasa Pemrograman ASP .NET
<http://www.alipromadoni.com/download/Web-Programming-dengan-Visual-Basic-2010.pdf>. 26 Desember 2015