



## *Bermain Logika*

Komputer bisa berpikir dan memutuskan sesuatu apabila kita memasukkan unsur logika dalam sebuah pemrograman. Logika bisa sangat sederhana, misalnya: jika berat badan si pengguna lebih besar dari 70 kg, maka munculkan tulisan: “Kurangi porsi makannya...” Tapi logika juga bisa sangat kompleks, misalnya memutuskan untuk menggerakkan bidak-bidak catur jika lawan merespons lewat serangan-serangan tertentu.

Agar logika itu bisa kita buat di dalam sebuah pemrograman, maka paling tidak ada beberapa unsur yang mesti dilibatkan, yaitu:

- Syntax untuk menulis logika.
- Variabel yang dinilai menggunakan operator.
- Operator logika yang digunakan untuk menilai variabel terhadap parameter tertentu.
- Kondisi yang akan dijalankan.

### *Syntax Penulisan Logika*

Secara umum, logika dapat “dimainkan” dalam script pemrograman menggunakan kontrol ***If ... Then***. Gampangnya, kita bisa perintahkan komputer untuk menilai: jika *begini-begitu*, maka *lakukan seperti itu*. Nanti jika logika tersebut kompleks, Anda bisa mengembangkannya dengan tambahan ***If ... Then ... ElseIf*** atau ***If ... Then ... Else***.

Agar syntax di atas berjalan dengan baik, maka Anda perlu memanfaatkan operator yang nanti digunakan untuk membandingkan variabel-variabel.

## Operator Logika

Operator berfungsi untuk membandingkan variabel-variabel sesuai dengan kondisi tertentu. Pada akhirnya, hasil perbandingan itu berfungsi untuk menjalankan perintah. Ada dua jenis operator yang bisa digunakan, yaitu conditional operator dan logical operator.

### Conditional Operator

Conditional Operator digunakan untuk membandingkan variabel-variabel yang umumnya bertipe data angka sehingga sering disebut dengan operator matematika. Tapi walaupun disebut operator matematika, namun Anda bisa memanfaatkannya untuk membandingkan variabel berjenis data teks, misalnya Anda bisa membuat syntax yang kira-kira berbunyi: *jika nama pengguna adalah Antok maka beri hadiah 1 juta rupiah*. Dalam hal ini, "Antok" adalah teks dan bukan angka.

Berikut conditional operator yang bisa Anda gunakan:

<i>Conditional Operator</i>	<i>Operator</i>	<i>Arti</i>
	=	<i>Sama dengan</i>
	>	<i>Lebih besar daripada</i>
	<	<i>Lebih kecil daripada</i>
	>=	<i>Lebih besar atau sama dengan</i>
	<=	<i>Lebih kecil atau sama dengan</i>
	<>	<i>Tidak sama dengan</i>

### Logical Operator

Logical Operator pada dasarnya membuat proses logika menjadi lebih kompleks. Dengan menggunakan Logical Operator Anda bisa menggabungkan beberapa kondisi. Misalnya: *jika si A dari Yogyakarta DAN nilainya lebih tinggi dari 75, maka beri ucapan kelulusan menggunakan Bahasa Jawa*. Pada contoh di atas, kata "dan" (yang sebenarnya adalah AND), merupakan Logical Operator.

Berikut beberapa logical operator yang bisa Anda gunakan:

<i>Logical Operator</i>	<i>Operator</i>	<i>Arti</i>
	AND	<i>Keduanya harus dalam kondisi TRUE</i>
	OR	<i>Salah satu harus dalam kondisi TRUE</i>
	XOR	<i>Salah satu (bukan keduanya) dalam kondisi TRUE</i>
	NOT	<i>Keduanya tidak ada yang TRUE</i>

## Contoh Kasus 1: Pengecek Waktu

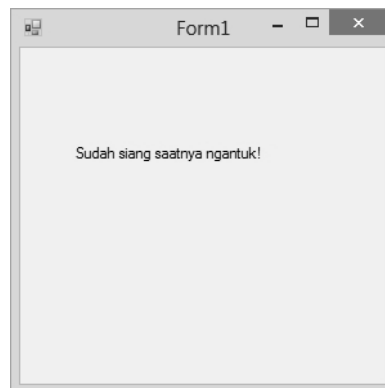
Untuk mendapatkan pemahaman yang lebih baik tentang cara pembuatan alur logika menggunakan pemrograman Visual Basic, maka berikut disajikan sebuah aplikasi untuk mengecek waktu. Berikut langkah-langkahnya:

1. Buatlah project baru terlebih dulu.
2. Lantas, klik **Label** dari toolbox dan drag ke dalam jendela aplikasi.
3. Kosongkan **Text** di dalam panel **Properties**.
4. Selanjutnya, klik-ganda jendela aplikasi dan ketik script di bawah ini:

```
Public Class Form1
    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        If (Now.Hour >= 12) Then
            Label1.Text = "Sudah siang saatnya ngantuk!"
        End If
    End Sub
End Class
```

5. Klik ikon **Start** untuk meluncurkan aplikasi tersebut. Apabila sekarang sudah di atas pukul 12 siang atau lebih, maka akan muncul

tulisan di dalam jendela aplikasi. Kalau belum pukul 12 siang, maka tak ada reaksi apa pun.



*Tulisan yang akan muncul jika menemukan kondisi tertentu*

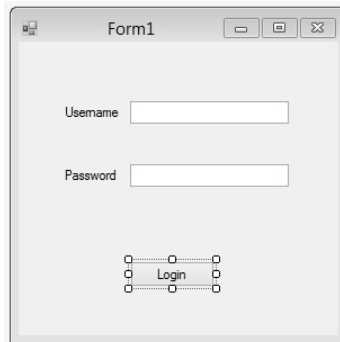
Inti dari script di atas terletak pada penggunaan If ... Then di mana dilakukan pengujian apakah hari ini jam sudah menunjukkan angka di atas 12 siang atau belum. Jika sudah, maka akan muncul tulisan di tempat Label1 diletakkan. Untuk mengetahui sistem waktu saat ini, gunakanlah function Now (menghasilkan informasi tanggal dan waktu) dan Hour (secara spesifik digunakan untuk mengetahui sistem jam).

## Contoh Kasus 2: Pengecekan Username dan Password

Anda bisa menggunakan If ... Then untuk pengecekan username dan password. Jadi, jika username yang ditulis adalah "Bejo" dan password-nya "fansJKT48", maka aplikasi akan menampilkan hak akses. Tapi jika salah, maka aplikasi akan memunculkan pesan kesalahan.

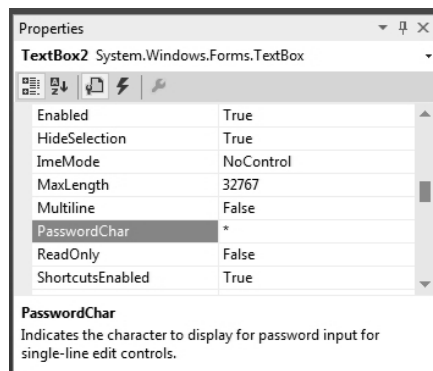
Untuk membuat aplikasi seperti itu, maka lakukanlah langkah-langkah sebagai berikut:

1. Buatlah project baru dan di dalam jendela aplikasi, pasang dua buah textbox dan satu tombol.
2. Lengkapi dengan label bertuliskan **Username** dan **Password**.



*Jendela aplikasi yang dilengkapi username dan password*

3. Selanjutnya, klik textbox di samping kanan Password itu dan masuklah ke dalam panel **Properties**. Carilah **PasswordChar**.
4. Isi PasswordChar dengan tanda bintang (\*). Dengan begitu, teks apa pun yang ditulis di dalam textbox ini akan menjadi tanda bintang tersebut.



*Mengisi karakter di dalam isian PasswordChar*

5. Klik-ganda tombol Login di atas dan tulis script lengkap sebagai berikut:

```
Public Class Form1
```

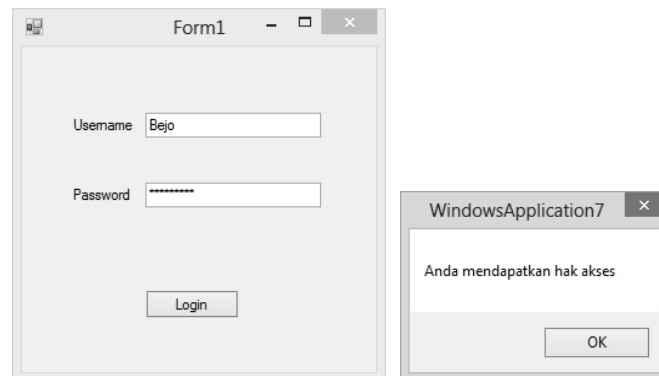
```
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
```

```

Dim Username, Password As String
Username = TextBox1.Text
Password = TextBox2.Text
If Username = "Bejo" And Password = "fansJKT48"
Then
    MsgBox("Anda mendapatkan hak akses")
End If
End Sub
End Class

```

Luncurkan dengan menekan tombol **F5** dan amati apa yang terjadi jika Anda mengisi kotak teks Username dan Password dengan tulisan "Bejo" serta "fansJKT48", dan jika bukan.



*Pengisian username dan password serta reaksi yang terjadi setelah tombol Login ditekan*

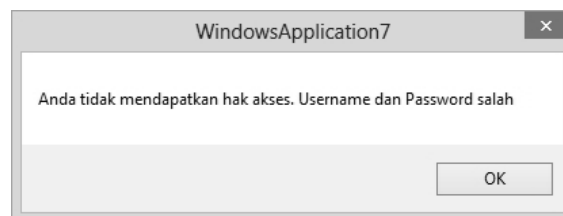
### **Menggunakan Logical Operator untuk Menilai Dua Kondisi yang Berbeda**

Pada script di atas, kita menggunakan AND untuk menggabungkan dua kondisi yang harus terpenuhi semuanya. Jika hanya terpenuhi satu saja, misalnya penulisan Username yang benar atau Password yang benar sementara lainnya salah, maka script tidak akan memunculkan kotak dialog.

## Menggunakan Else untuk Melakukan Sebuah Tindakan Ketika dalam Kondisi Salah

Apa jadinya jika dalam contoh di atas penulisan username dan password ternyata salah? Menurut script di atas, tidak akan terjadi apa pun. Semestinya, akan terasa lebih lengkap kalau ada tindakan yang dijalankan saat penulisan username dan password mengalami kesalahan. Inilah fungsi dari Else.

Jadi, kita bisa melengkapi script di atas dengan script di bawah ini:



*Kotak dialog yang akan terlihat ketika username dan password ditulis secara salah*

```
Public Class Form1

    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        Dim Username, Password As String
        Username = TextBox1.Text
        Password = TextBox2.Text
        If Username = "Bejo" And Password = "fansJKT48"
        Then
            MsgBox("Anda mendapatkan hak akses")
        Else
            MsgBox("Anda tidak mendapatkan hak akses. Username dan Password salah")
        End If
    End Sub
End Class
```

## Menggunakan ElseIf

ElseIf digunakan ketika ada beberapa alternatif yang ingin kita tempuh. Pada contoh-contoh di atas, kita hanya memiliki dua pilihan saja. Misalnya:

- Pada contoh kasus pertama, kita hanya menguji dua pilihan apakah sekarang sudah di atas jam dua belas atau belum.
- Pada contoh kasus kedua, kita hanya menguji dua pilihan apakah username dan password ditulis dengan benar atau tidak.

Lantas bagaimana jika ada tiga atau lebih pilihan, misalnya:

- Jika di bawah jam 10, maka akan muncul tulisan "Selamat pagi", di atas jam sepuluh namun di bawah jam 3 akan muncul tulisan "Selamat siang", dan di atas jam 3 muncul tulisan "Selamat Sore".

Untuk mengakomodasi pilihan-pilihan yang lebih banyak, maka Anda dapat memanfaatkan ElseIf. Lihat contohnya di bawah ini:

```
Public Class Form1
```

```
    Private Sub Form1_Load(sender As Object, e As  
    EventArgs) Handles MyBase.Load
```

```
        If (Now.Hour <= 9) Then  
            Label1.Text = "Masih pagi. Wajib ngantuk"  
        ElseIf (Now.Hour > 9 And Now.Hour <= 15) Then  
            Label1.Text = "Sudah siang. Saatnya kerja"  
        ElseIf (Now.Hour > 15 And Now.Hour <= 18) Then  
            Label1.Text = "Sudah sore. Saatnya mandi"  
        Else  
            Label1.Text = "Malam telah tiba. Tidur!"  
        End If
```

```
    End Sub
```

```
End Class
```

Script di atas mendeteksi jam sistem yang berlaku di komputer dan memicu munculnya tulisan yang tepat berdasarkan kondisi yang berlaku. Ada empat kemungkinan yang akan terjadi sesuai dengan kondisi-kondisi di atas.



## Memilih Banyak Opsi Menggunakan Select Case

If...Then memang bisa digunakan untuk memilih. Namun jika pilihannya sangat banyak, maka akan menyebabkan baris-baris script menjadi terlalu berlimpah. Lantas? Gunakan Select Case kalau begitu. Script ini ideal jika pilihan yang Anda tawarkan cukup banyak.

Berikut contoh penggunaan Select Case. Kita ingin membuat aplikasi bunga deposito. Jika nasabah mendepositokan uangnya selama 1 bulan, maka akan mendapat bunga 5%, 3 bulan sebesar 7%, 6 bulan sebesar 8%, 12 bulan sebesar 10%, dan 24 bulan sebesar 24%.

Langkah-langkahnya adalah sebagai berikut:

1. Rancang terlebih dulu jendela aplikasi seperti berikut ini. Untuk keterangan Label, ikuti panduan berikut:
  - Label1: Jangka Waktu
  - Label2: Pilihan: 1, 3, 6, 12, 24
  - Label3: Jumlah
  - Label4: Bunga Deposito
  - Label5: Total Uang Anda
  - Label6: Rp.
  - Label7: Rp.



*Jendela aplikasi untuk menghitung deposito*

2. Klik-ganda tombol **Hitung!** dan ketik script sebagai berikut:

```
Public Class Form1
```

```
    Private Sub Button1_Click(sender As Object, e As  
EventArgs) Handles Button1.Click  
        Dim JangkaWaktu As Integer  
        Dim Uang, Bunga, TotalUang, Persen As Decimal  
        JangkaWaktu = TextBox1.Text  
        Uang = TextBox2.Text  
        Select Case JangkaWaktu  
            Case 1  
                Persen = 5 / 100  
            Case 3  
                Persen = 7 / 100  
            Case 6  
                Persen = 8 / 100  
            Case 12  
                Persen = 10 / 100  
            Case 24  
                Persen = 24 / 100  
        End Select  
        Bunga = Persen * Uang  
        TotalUang = Uang + Bunga  
        Label6.Text = "Rp. " & Bunga  
        Label7.Text = "Rp. " & TotalUang  
    End Sub  
End Class
```

3. Tekan tombol **F5** untuk menjalankan aplikasi tersebut.



*Aplikasi penghitung deposito yang sudah dibuat*



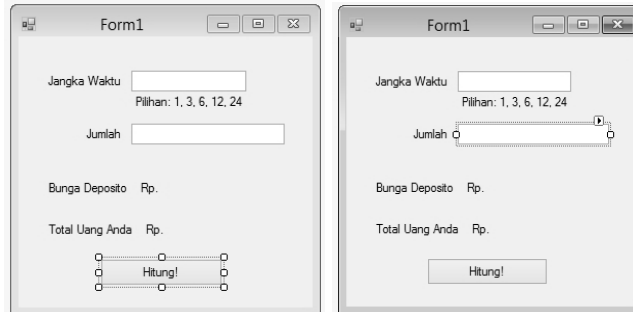
# *Bersenang- senang dengan Control*

Visual Basic 2013 merupakan pemrograman yang diarahkan untuk membuat aplikasi berbasis Graphical User Interface (GUI), Itu artinya, pengguna akan mengoperasikan aplikasi Anda menggunakan tampilan-tampilan visual seperti textbox, tombol, checkbox, radio button, dan kontrol-kontrol lain di dalam sebuah jendela aplikasi. Kontrol-kontrol itu dapat Anda tambahkan lewat panel Toolbox.

Oleh karena itu, marilah kita santai sejenak dengan belajar mengenal kontrol-kontrol yang bisa dimanfaatkan untuk membuat aplikasi yang indah dan fungsional.

## *Pointer*

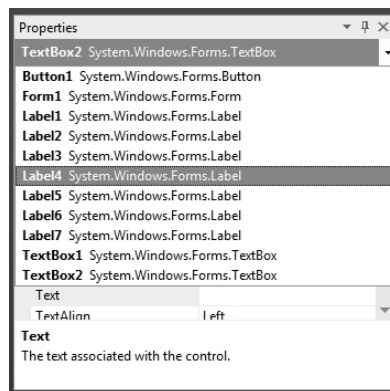
Pointer digunakan untuk memilih kontrol-kontrol yang sudah Anda masukkan ke dalam jendela aplikasi. Begitu kontrol dalam kondisi terseleksi, maka kontrol itu akan dikelilingi garis seleksi dengan beberapa handle di sekitarnya.



***Menyeleksi kontrol-kontrol di dalam jendela aplikasi***

Namun adakalanya, Anda akan kesulitan memilih kontrol di dalam jendela aplikasi saat kontrol itu “tidak terlihat”. Sebagai contoh, jika Anda memasang Label dan Anda menghapus isian Text pada panel Properties. Karena tak ada teks yang ditampilkan, maka Label itu pun sulit ditemukan dan diklik. Lantas? Sebagai alternatif menyeleksi kontrol lakukan langkah-langkah sebagai berikut:

1. Di bagian atas pada panel Properties terdapat menu drop-down. Klik sehingga menu drop-down itu terbuka.
2. Klik pada nama kontrol yang akan Anda seleksi.



***Menyeleksi control menggunakan panel Properties***

## Button

Secara umum, fungsi button adalah meletakkan tombol di dalam jendela aplikasi. Tombol dipakai untuk memproses input-input atau data yang dimasukkan oleh pengguna ke dalam kontrol-kontrol yang telah disediakan. Oleh karena itu, banyak script yang ditulis untuk mengatur apa yang terjadi saat tombol ini ditekan.

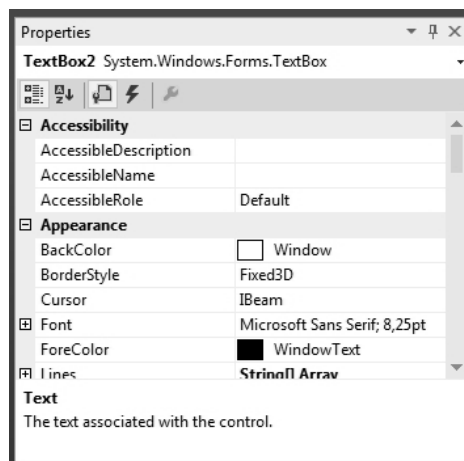
Berikut properties yang bisa digunakan untuk mengatur tombol:

- **BackColor:** berfungsi mengatur warna latar belakang tombol.
- **BackgroundImage:** mengatur gambar latar belakang pada tombol.
- **BackgroundImageLayout:** mengatur penataan gambar latar belakang pada tombol itu.
- **Cursor:** mengatur bentuk kursor yang akan terlihat saat berada di atas tombol.
- **FlatAppearance - Border Color:** mengatur warna garis border pembentuk tombol.
- **FlatAppearance - Border Size:** mengatur ukuran garis border.
- **FlatAppearance - MouseDownBackColor:** mengatur warna latar belakang pada tombol saat tombol ditekan.
- **FlatAppearance - MouseOverBackColor:** mengatur warna latar belakang pada tombol ketika kursor mouse berada di atasnya.
- **FlatStyle:** mengatur bentuk model tombol.
- **DialogResult:** berfungsi untuk mengatur sifat tombol. Jika Anda menulis script pada tombol ini (dengan cara mengklik-ganda tombol itu dan menulis script seperti yang kita lakukan selama ini), maka pilih opsi None. Tapi kalau Anda ingin agar tombol tersebut berfungsi seperti tombol OK, Retry, Ignore, Cancel, Yes, dan No, maka pilih salah satu opsi yang tersedia.

## TextBox

Ini adalah control yang paling sering kita gunakan dalam contoh-contoh pembuatan aplikasi. Pada intinya digunakan untuk menginput data teks maupun angka. Ukuran textbox bisa diperbesar maupun diperkecil sesuai kebutuhan dengan cara mengklik-drag kotak-kotak handle yang mengelilinginya sampai ditemukan ukuran yang pas.

Di dalam panel Properties, Anda akan menemukan beberapa opsi untuk membantu mengonfigurasi tampilan dan fungsi textbox itu. Berikut beberapa yang penting:



*Properties untuk mengatur bagaimana tampilan textbox dan fungsinya*

- **BackColor:** untuk mengatur warna textbox
- **BorderStyle:** untuk mengatur garis tepi yang membentuk textbox.
- **Cursor:** untuk mengatur bentuk kursor saat berada di dalam textbox.
- **Font:** untuk mengatur jenis font ketika pengguna mengetik di dalam textbox.
- **ForeColor:** untuk mengatur warna teks.

- **Lines:** untuk menulis teks di dalam textbox yang mendukung beberapa baris (multiline).
- **RightToLeft:** mengatur arah teks dari kanan ke kiri jika dalam kondisi Yes.

**TIP:**

*Opsi RightToLeft biasanya ideal digunakan untuk karakter-karakter di kebudayaan yang penulisannya dari kanan ke kiri seperti tulisan Arab.*

- **Scrollbars:** untuk menampilkan scrollbar di tepi textbox apabila diaktifkan, yaitu: Horizontal (hanya scrollbar mendatar), Vertical (hanya scrollbar vertikal), dan Both (keduanya).
- **Text:** untuk mengatur tulisan yang muncul di dalam textbox tanpa menunggu input dari pengguna.
- **TextAlign:** mengatur perataan teks.
- **UseWaitCursor:** jika dalam kondisi True, maka akan muncul ikon Wait (berbentuk jam pasir).
- **AcceptsReturn:** dalam kondisi Yes, maka Anda bisa menekan tombol Enter untuk membuat baris baru pada textbox multiline (textbox berukuran besar yang dapat menampung beberapa baris teks).
- **AcceptsTab:** dalam kondisi Yes, maka Anda bisa menekan tombol Tab untuk menciptakan jarak pada teks saat menulis di dalam textbox multiline.
- **AllowDrop:** dalam kondisi Yes, maka Anda bisa klik-drag teks dari sumber manapun (misal: dari Notepad) langsung ke dalam textbox.
- **CharacterCasing:** ada tiga pilihan, yaitu Normal: penggunaan huruf kapital dan kecil disesuaikan keinginan pengguna, Upper: teks menjadi huruf kapital semuanya, dan Lower: teks menjadi huruf kecil semuanya.
- **Enabled:** untuk menentukan apakah textbox ini bisa dipakai oleh pengguna (Yes) atau tidak (No).
- **HideSelection:** berfungsi untuk menyembunyikan seleksi.
- **MaxLength:** berfungsi untuk mengatur jumlah karakter maksimal yang bisa dimasukkan ke dalam textbox.

**TIP:**

*Mengapa karakter harus dibatasi? Ingat aplikasi penghitung deposito sebelumnya? Apakah wajar jika pengguna memasukkan lebih dari 3 angka untuk penentuan bulan (misal: 100)? Tidak, bukan? Kalau begitu, batasi saja sampai 2 angka sehingga di bagian `MaxLength` isi dengan angka: 2.*

- **PasswordChar:** jika ingin membuat textbox pengisian password, tentukan karakter yang akan terlihat saat pengguna mengetik apa pun. Biasanya, diisi dengan karakter: \*.
- **ReadOnly:** itu artinya, Anda bisa menulis teks secara langsung di dalam textbox dan teks itu tidak bisa diubah oleh pengguna.
- **ShortcutsEnabled:** jika dalam kondisi True, maka untuk mengaktifkan textbox ini pengguna bisa menekan tombol shortcut.
- **TabIndex:** kalau pengguna menekan tombol Tab untuk berpindah dari satu kontrol ke kontrol lainnya, maka di urutan keberapa textbox ini akan diseleksi? Atur lewat `TabIndex` ini. Jika Anda mengisi dengan angka 4, misalnya, maka textbox akan aktif jika pengguna menekan Tab sebanyak 4 kali.
- **TabStop:** dalam kondisi True, maka textbox bisa aktif (langsung ada kursor di dalamnya) saat pengguna menekan tombol Tab.
- **UseSystemPasswordChar:** dalam kondisi True, maka karakter yang akan dipilih oleh sistem menjadi karakter pengganti teks saat pengguna menulis password.
- **Visible:** dalam kondisi True, maka textbox akan terlihat di dalam jendela aplikasi.
- **WordWrap:** dalam kondisi True, jika panjang teks melebihi panjang textbox, maka teks akan dipotong ke baris berikutnya.
- **(Name):** nama textbox yang nanti sangat penting manfaatnya ketika menulis script pemrograman VisualBasic. Anda diijinkan untuk mengganti nama textbox dengan nama yang lebih informatif.
- **CausesValidation:** dalam kondisi True maka Anda bisa memvalidasi data yang ditulis di dalam textbox begitu kursor mouse pindah ke kontrol yang lain.



- **Layout:** di dalam kelompok ini, Anda akan temukan Anchor, Dock, Location, Margin, dan sebagainya yang digunakan untuk mengatur posisi dan ukuran textbox di dalam jendela aplikasi.

## Label

Kalau Anda ingin menulis teks di dalam jendela aplikasi, entah teks yang sifatnya informatif maupun teks hasil dari pemrograman, maka gunakanlah Label. Anda bisa memasukkan banyak label ke dalam jendela aplikasi sesuai kebutuhan.

Berikut property-property yang bisa Anda kelola untuk mengatur label:

- **Cursor:** berfungsi untuk mengatur bentuk kursor ketika kursor mouse berada di atas label.
- **FlatStyle:** mengatur bentuk label saat kursor mouse berada di atasnya dan pengguna mengklik label itu.
- **Font:** mengatur font, ukuran, warna, dan efek-efek teks lainnya pada label.
- **Image:** berfungsi untuk memunculkan gambar di dalam label.
- **ImageAlign:** berfungsi mengatur perataan gambar.
- **UseMnemonic:** huruf pertama pada teks label yang Anda tulis, jika diberi awalan tanda "&", akan digarisbawahi. Dengan begitu, pengguna bisa menekan tombol Alt diikuti huruf bergaris bawah itu untuk mengaktifkan label tersebut.

## LinkLabel

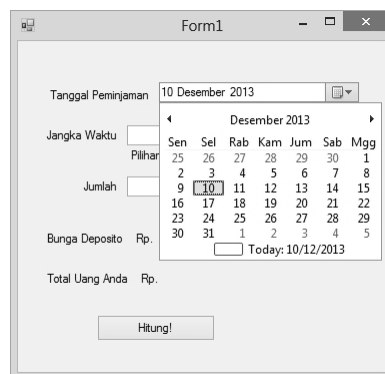
Kalau Anda ingin menulis informasi dalam bentuk teks yang mengandung link, maka gunakan LinkLabel. Dengan demikian, di dalam jendela aplikasi Anda bisa menulis alamat email, alamat website, atau link lainnya. LinkLabel memiliki beberapa property sebagai berikut:

- **ActiveLinkColor:** mengatur warna link ketika link itu diklik oleh pengguna.
- **DisabledLinkColor:** mengatur warna link saat link itu tidak aktif.

- **LinkColor:** mengatur warna link.
- **LinkVisited:** ketika dalam kondisi True, maka warna link itu menunjukkan kalau link itu pernah diklik (lihat pada warna VisitedLinkColor).
- **VisitedLinkColor:** mengatur warna link yang pernah dikunjungi.

## DateTimePicker

DateTimePicker digunakan untuk membantu pengguna memilih tanggal dan waktu secara cepat dan akurat. Nanti akan terlihat menu kombo di dalam jendela aplikasi dan selanjutnya pengguna tinggal mengklik tanggal dan waktu yang diinginkan (misal: untuk memilih tanggal lahir).



*Jendela aplikasi yang dilengkapi dengan pemilihan tanggal*

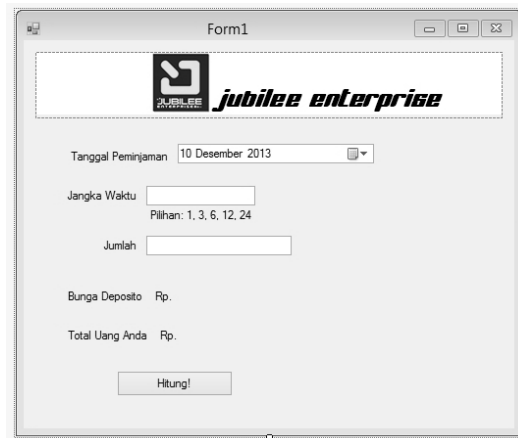
Berikut properties yang bisa digunakan untuk memodifikasi pemilihan tanggal tersebut:

- **CalendarFont:** mengatur jenis font untuk menampilkan teks tanggal.
- **CalendarForeColor:** mengatur warna teks pada kalender.
- **CalendarMonthBackground:** mengatur warna kalender secara umum.
- **CalendarTitleBackColor:** mengatur warna latar belakang pada judul kalender.

- **CalendarTitleForeColor:** mengatur warna teks judul kalender.
- **CalendarTrailingForeColor:** mengatur warna teks untuk trailing day, yaitu hari sebelum dan sesudah bulan saat ini.
- **DropDownAlign:** mengatur arah menu drop-down ketika menu ini diklik oleh pengguna. Bisa Right atau Left.
- **Format:** mengatur format penulisan tanggal dan waktu. Jika memilih Long, maka tanggal dan waktu akan ditulis lengkap (misal: 10 Januari 2014).
- **RightToLeftLayout:** jika dalam kondisi Yes, maka layout kalender akan menggunakan sistem dari kanan ke kiri.
- **ShowCheckBox:** dalam kondisi Yes, maka akan ada checkbox yang muncul di dalam menu drop-down.
- **ShowUpDown:** dalam kondisi Yes, maka tampilan menu drop-down akan digantikan model Spin saat memilih tanggal.
- **Checked:** kalau Anda mengaktifkan ShowCheckBox, maka apabila opsi Checked ini dalam kondisi True, maka kotak cek itu akan tercentang.
- **MaxDate:** menentukan tanggal maksimal yang bisa ditampilkan/dipilih oleh pengguna.
- **MinDate:** menentukan tanggal minimal yang bisa ditampilkan/dipilih oleh pengguna.
- **Value:** menentukan tanggal dan sistem waktu yang saat ini dalam kondisi aktif.

## PictureBox

PictureBox berfungsi untuk memasang gambar atau citra bitmap ke dalam jendela aplikasi. Bayangkanlah jika aplikasi tersebut ingin Anda pasang logo, maka kontrol PictureBox ini bisa dimanfaatkan.



*Logo di dalam jendela aplikasi yang dibuat memakai PictureBox*

Berikut properties yang bisa dimanfaatkan:

- **BackColor:** mengatur warna latar belakang pada komponen.
- **BackgroundImage:** berfungsi untuk menentukan gambar latar belakang.
- **BackgroundImageLayout:** berfungsi untuk menentukan penataan gambar yang akan dijadikan latar belakang itu. Pilih Tile jika gambar akan dipasang berulang-ulang. Center dipakai jika gambar latar belakang diletakkan di tengah-tengah jendela aplikasi. Stretch dipakai apabila gambar latar belakang benar-benar memenuhi seluruh area jendela aplikasi. Atau pilih Zoom untuk memperbesar ukuran gambar latar belakang.
- **BorderStyle:** berfungsi untuk mengatur apakah ada garis border di sekeliling gambar. Jika ada, maka tentukan pula pola garisnya.
- **Image:** berfungsi untuk memasang gambar yang bisa dipindah-pindahkan sesuai kebutuhan (misal: logo).
- **ErrorImage:** mengatur tampilan gambar yang akan muncul jika gambar yang Anda buat ternyata gagal ditampilkan.
- **ImageLocation:** mengatur di mana file gambar itu berada.
- **InitialImage:** mengatur gambar yang akan terlihat saat gambar yang Anda pilih sedang dimuat (loading).

- **WaitOnLoad:** ketika dalam kondisi True, maka kontrol-kontrol lainnya tak bisa digunakan sampai gambar muncul secara sempurna.

## CheckBox

Fungsinya untuk memasang kontrol berbentuk kotak cek (checkbox) ke dalam jendela aplikasi. Checkbox akan menghasilkan salah satu dari dua kondisi, yaitu True atau False. Ketika tercentang, maka dalam kondisi True. Dengan menggunakan checkbox Anda bisa membuat pilihan-pilihan kepada pengguna.

*Dalam aplikasi pencatat deposito, checkbox bisa digunakan untuk menandai apakah pengguna nasabah lama atau tidak*

Berikut properties yang bisa digunakan:

- **Appearance:** mengatur bentuk checkbox. Ada dua pilihan, yaitu Normal seperti yang kita lihat sekarang ini (berbentuk kotak cek) dan Button (berbentuk tombol).
- **CheckAlign:** mengatur posisi kotak cek terhadap teks. Ada beberapa pilihan yang bisa diklik salah satunya.
- **Checked:** dalam kondisi True maka kotak cek akan tercentang secara otomatis.

- **CheckState:** mengatur kondisi ketika kotak cek ini pertama kali ditampilkan di dalam jendela aplikasi. Unchecked artinya, kotak cek ini tidak dalam kondisi tercentang.
- **ThreeState:** dalam kondisi True, maka kotak cek itu bisa diklik tiga kali, yaitu untuk mencentang, memberi titik, dan menghilangkan centang.

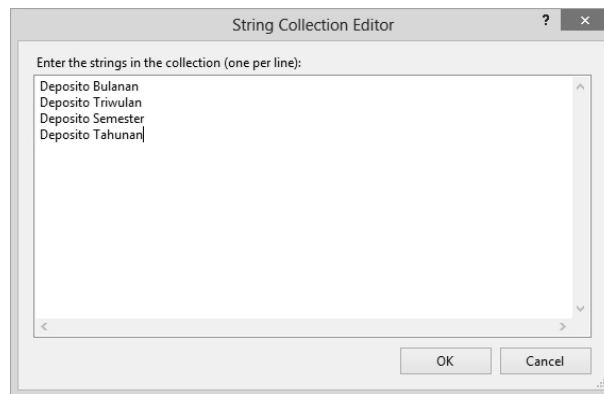
## RadioButton

Berbeda dengan kotak cek yang bisa ditampilkan satu saja dalam jendela aplikasi, radio button umumnya dihadirkan dalam bentuk satu set (lebih dari satu) sehingga pengguna bisa memilih salah satu dari pilihan yang disediakan. Secara umum, properties checkbox dan radiobutton tidaklah berbeda.

## CheckBox

Berapa jumlah checkbox yang Anda butuhkan dalam satu jendela aplikasi? Lebih dari dua? Kalau begitu, untuk efisiensi, gunakan CheckListBox. Anda bisa membuat banyak checkbox secara efektif menggunakan kontrol ini. Untuk membuatnya, lakukan langkah-langkah sebagai berikut:

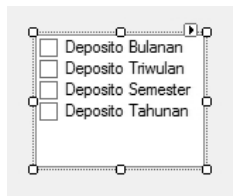
1. Klik-drag **CheckListBox** dari toolbox ke dalam jendela aplikasi.
2. Di dalam panel **Properties**, klik persegi empat kecil yang ada di sisi kanan **Items**.
3. Isikan opsi-opsi yang akan diberi checkbox satu demi satu. Setiap opsi ditulis dalam baris-baris terpisah.



*Membuat opsi-opsi untuk setiap checkbox*

4. Tekanlah tombol **OK**.

Di dalam jendela aplikasi akan terlihat beberapa checkbox sesuai dengan yang telah Anda buat itu.



*Deretan checkbox yang sudah dibuat*

Berikut deretan property yang bisa diatur:

- **MultiColumn:** jika Anda memilih True, maka checkbox itu bisa ditata secara horizontal apabila ukuran kontrol lebih kecil dari jumlah checkbox.
- **ScrollAlwaysVisible:** jika memilih True maka akan muncul scrollbar saat ukuran kontrol lebih kecil daripada jumlah checkbox.
- **Sorted:** kalau memilih Yes, maka checkbox itu akan diurutkan.

## NumericUpDown

NumericUpDown berfungsi untuk menentukan value dalam bentuk angka. Dengan begitu, Anda tidak perlu khawatir bilamana si pengguna keliru memasukkan data teks. Cara kerja NumericUpDown sederhana. Gunakan anak panah atas dan bawah, baik yang ada di keyboard atau di dalam kontrol itu, untuk menambah atau mengurangi angka.

Berikut property yang bisa diatur untuk NumericUpDown:

- **Hexadecimal:** dalam kondisi True, maka yang muncul di dalam kontrol ini adalah versi Hexadecimal dari sebuah angka.
- **UpDownAlign:** berfungsi mengatur posisi anak panah atas-bawah untuk penambahan serta pengurangan angka.
- **InterceptArrowKeys:** dalam kondisi True, maka Anda bisa menambah atau mengurangi angka lewat anak panah atas dan bawah pada keyboard.
- **ReadOnly:** dalam kondisi True, maka pengguna tidak bisa memasukkan angka dengan mengetikkannya secara langsung. Ideal untuk mengatur angka-angka yang sudah fixed (misal: jumlah bulan penyimpanan deposito yang tidak bisa diatur secara bebas oleh pengguna).
- **DecimalPlaces:** mengatur jumlah angka desimal (setelah koma) yang akan terlihat di dalam kontrol.
- **Increment:** mengatur banyaknya kelipatan angka ketika ditambahkan atau dikurangi. Kalau Anda isi 5, misalnya, maka angka akan bertambah dalam kelipatan 5.
- **Maximum:** mengatur angka maksimum yang bisa dicapai.
- **Minimum:** mengatur angka minimum yang bisa dipilih.
- **ThousandsSeparator:** berfungsi untuk memecah angka ribuan agar lebih mudah dibaca.

## RichTextBox

Kontrol ini berfungsi untuk menyediakan tempat bagi pengguna yang ingin menulis teks dalam jumlah banyak. Berikut property yang bisa digunakan untuk pengaturan RichTextBox itu:

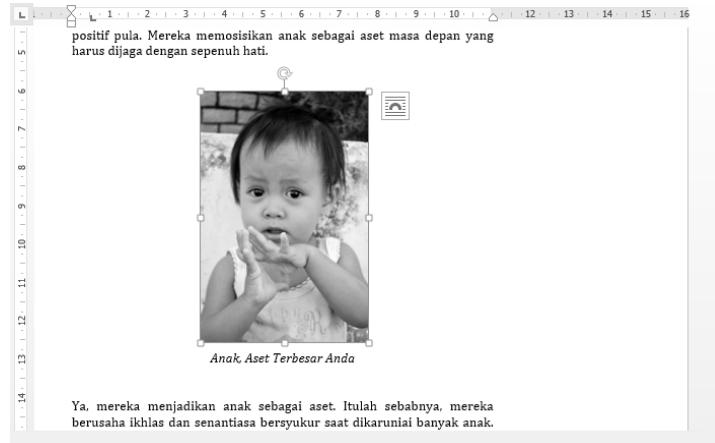


- **AcceptsTab:** jika dalam kondisi True, maka pengguna bisa menekan tombol Tab untuk menambah spasi pada teks. Tapi jika False, maka penekanan tombol Tab berfungsi untuk berpindah dari satu kontrol ke kontrol berikutnya.
- **AutoWordSelection:** jika dalam kondisi True, maka pengguna bisa menyeleksi teks secara otomatis.
- **BulletIndent:** berfungsi untuk mengatur jarak bullet.
- **DetectUrls:** dalam kondisi True, maka jika menulis alamat URL (alamat website), maka akan langsung diubah menjadi link.
- **EnableAutoDragDrop:** dalam kondisi True, maka Anda bisa klik-drag teks atau gambar ke dalam kontrol ini.
- **MaxLength:** berfungsi untuk mengatur total jumlah karakter yang bisa ditulis ke dalam kontrol ini.
- **RightMargin:** mengatur margin kanan di dalam kontrol.
- **ShowSelectionMargin:** dalam kondisi True maka akan selection margin.
- **ZoomFactor:** berfungsi untuk mengatur seberapa besar teks yang terlihat di dalam kontrol ini. Jika diisi 1 maka akan terlihat tampilan normal.

Selain menggunakan RichTextBox, Anda juga bisa memanfaatkan TextBox untuk membantu pengguna menulis teks dalam jumlah besar. Tapi alasan mengapa menggunakan RichTextBox adalah, si pengguna dapat memasukkan gambar. Bahkan ketika opsi EnableAutoDragDrop aktif, si pengguna bisa memasukkan gambar dari satu dokumen ke dalam RichTextBox dengan metode klik-drag.

Contohnya seperti ini:

1. Andaikan gambar itu ada di dalam dokumen MS Word. Klik gambar tersebut sehingga terseleksi.



*Klik foto atau gambar yang akan dipindahkan*

2. Lantas, drag ke dalam RichTextBox pada aplikasi Anda.
3. Gambar atau foto itu pun akan masuk ke dalam RichTextBox.



*Gambar yang ada di dalam RichTextBox*

## ComboBox

ComboBox biasa dikenal dengan istilah menu drop-down karena bisa diklik untuk melihat opsi-opsi yang ada di dalamnya. ComboBox dipilih karena relatif hemat tempat. Berikut property-property yang bisa Anda atur untuk mendapatkan ComboBox yang diinginkan:

- **DropDownStyle:** berfungsi untuk menentukan bentuk combobox. DropDown dan DropDownList memiliki bentuk yang relatif sama. Namun jika memilih Simple, maka ukuran combobox akan melebar ke bawah sebanyak jumlah opsi yang nanti akan Anda tampilkan.
- **MaxDropDownItems:** berfungsi mengatur jumlah maksimal opsi-opsi yang bisa ditampung oleh menu drop-down ini.
- **MaxLength:** berfungsi mengatur jumlah karakter maksimal yang bisa dimasukkan ke dalam menu drop-down.
- **Items:** berfungsi untuk memasukkan opsi-opsi ke dalam menu drop-down tersebut. Setiap opsi dibuat dalam baris-baris terpisah.

## ListBox

Tampilan ListBox lebih memanjang ke bawah sehingga untuk penataan ruang, ListBox lebih memakan tempat. Namun, pengguna bisa memilih lebih dari satu opsi yang Anda tawarkan jika disajikan menggunakan ListBox. Berikut property yang bisa digunakan untuk mengatur ListBox:

- **HorizontalScrollbar:** jika dalam kondisi True, maka akan muncul scrollbar horizontal pada listbox.
- **HorizontalExtent:** untuk mengatur seberapa banyak listbox akan bergeser jika scrollbar horizontal diklik oleh pengguna. Masukkan nilai ke dalam HorizontalExtent ini dalam satuan pixel.
- **SelectionMode:** untuk menentukan apakah pengguna bisa memilih lebih dari satu atau hanya satu.