***Method Overloading***

*Overloading* adalah membuat dua atau lebih *method* dengan nama yang sama

dalam satu *class*. *Method-method* tersebut harus dapat dibedakan antara satu dengan yang lain dalam jumlah dan atau tipe argumennya.

Atau *Overloading* juga dapat dikatakan pemakaian nama *class* yang sama tetapi dengan daftar parameter yang berbeda.

**Penerapan *Method Overloading***

*Polymorphism* adalah pemakian *method* dengan nama yang sama pada *class-*

*class* yang berbeda dan memungkinkan *method* yang tepat dieksekusi berdasarkan konteks yang memanggilnya. *Polymorphism* dapat diimplementasikan menggunakan *method Overloading* dan *method Overriding*.

Kita dapat memperluas fungsional *base class* dengan menambahkan *method* pada *subclass* yang memiliki nama *method* dan *property* yang sama dari *base class* tetapi memiliki daftar parameter yang berbeda.

*Method overloading* dari *base class* dilakukan menggunakan kata kunci

*Overloads*.

***Overloading* Pada Poperty**

Untuk contoh, class **Karyawan** memiliki property **Nama**. Pekerja selain

memiliki **Nama**, mungkin juga memiliki **Nama Panggilan** dan **Nama Lengkap**. Untuk memperluas **Nama**, kita ubah *Class* Pekerja dengan menambahakan *method overloading* untuk *property* **Nama**.

Buka file Pekerja.Vb dan tambahkan enum berikut sebelum deklarasi *class*.

Public Enum JenisNama

Lengkap = 1

Panggilan =2

End Enum

Public Class Pekerja

Kita dapat menambahkan *Overloading* pada *property* **Nama** didalam *class*

Pekerja.

Public Class Pekerja

Inherits Karyawan

Private idJamkerja As Double Private idUpahPerjam As Double Private idKolNama As New Hashtable

Public Overloads Property Nama ( ByVal Type As JenisNama) As

String

Get

Return idKolNama (Type)

End Get

Set(ByVal Value As String)

idKolNama.Add (Type,Value) End Set

End Property

…….. Kode lainnya

*Property* nama sebenarnya merupakan *array property* yang membolehkan kita memasukkan banyak nilai melalui *property* yang sama. Dalam contoh kita memasukkan nilai melalui objek *Hash table* yang diindeks menggunakan nilai Enum yang sudah dibuat sebelumnya.

*Property* nama pada *subclass* memiliki nama yang sama pada *baseclass*, tetapi memiliki daftar parameter yang berbeda menggunakan *Overloading*.

***Overloading* Pada Method**

Seperti halnya pada property, method juga dapat kita overload, guna

mendefinisikan fungsionalitas lain. Sebagai contoh:

‘Method dengan parameter harga dan diskon

Public Function Haga(ByVal h As Integer, ByVal diskon As Integer) As Double h = h – (h \* (diskon / 100))

Return h

End Function

Seignatur untuk sebuah method meliputi nama, jumlah parameter dan tipenya, serta tipe kembalian (jika ada). Pada prinsipnya, keyword Overloads digunakan dalam konteks pernyataan **Function**, **Sub**, dan **Property**.

‘Overloading method harga

Public Function Haga(ByVal h As Integer, ByVal diskon As **Double**) As Double h = h – (h \* (diskon / 100))

Return h

End Function

Pada saat property dan method sama-sama diimplementasikan sebagai prosedur yang menerima argumen, keduanya akan terlihat mirip sekali. Secara umum, dapat dibedakan bahwa property meyimpan data untuk suatu objek, sedangkan method adalah aksisuatu objek yang dapat diminta untuk dilaksanakan.

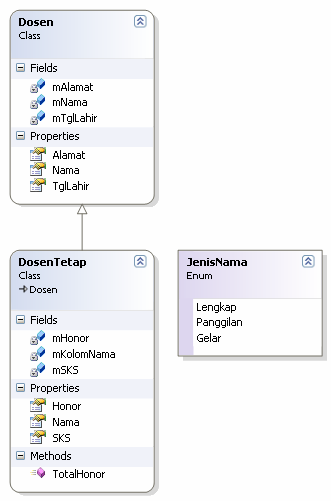
**Praktikum *Method Overloading***

**Praktikum**

Pada praktikum ini akan membuat program dosen. Analisalah *class diagram* berikut.

**Class**

Analisa *class* berikut, kita akan melakukan *overloading* pada field Nama, dimana Nama bisa memiliki NamaLengakap, NamaPanggilan, dan Gelar.



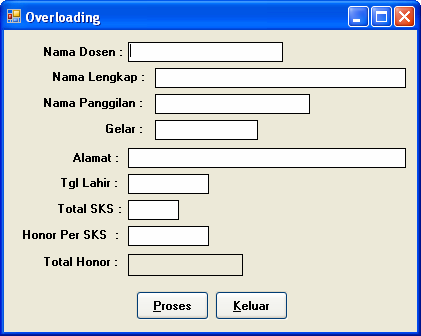
**Gambar.** *Class diagram*

**Methods**

TotalHonor = SKS \* Honor

**Form**

Rancanglah sebuah form seperti pada **Gambar**



**Gambar 9.2.** *Form overloding*

**Setting property**

|  |  |  |
| --- | --- | --- |
| **Nama Objek** | **Property** | **Setting Property** |
| Form1 | Name  Text | Form1  Dosen |
| Label1 | Name  Text | Label1  Nama Dosen : |
| Label2 | Name  Text | Label2  Alamat : |
| Label3 | Name  Text | Label3  Tgl Lahir : |
| Label4 | Name  Text | Label4  Total SKS : |
| Label5 | Name  Text | Label5  Honor Per SKS : |
| Label6 | Name  Text | Label6  Total Honor : |
| TextBox1 | Name  Text | txtNama  Kosongkan |
| TextBox2 | Name  Text | txtAlamat  Kosongkan |
| TextBox3 | Name  Text | txtTglLahir  Kosongkan |
| TextBox4 | Name  Text | txtSKS  Kosongkan |
| TextBox5 | Name  Text | txtHonor  Kosongkan |
| Label7 | Name  Text AutoSize BorderStyle | lblTotalHonor  Kosongkan False FixedSingle |
| Label8 | Name  Text | Jangan diganti  Nama Lengkap : |
| Label9 | Name | Jangan diganti |

|  |  |  |
| --- | --- | --- |
|  | Text | Nama Panggilan : |
| Label10 | Name  Text | Jangan diganti  Gelar : |
| TextBox6 | Name  Text | txtNamaLengkap  Kosongkan |
| TextBox7 | Name  Text | txtNamaPanggilan  Kosongkan |
| TextBox8 | Name  Text | txtGelar  Kosongkan |
| Button1 | Name  Text | cmdProses  &Proses |
| Button2 | Name  Text | cmdKeluar  &Keluar |

**Program**

Buat **base class** dengan nama Dosen, buat program berikut.

'Super class / Base Class

Public Class Dosen

Private mNama As String Private mAlamat As String Private mTglLahir As Date

Public Property Nama() As String

Get

Return mNama

End Get

Set(ByVal value As String)

mNama = value

End Set

End Property

Public Property Alamat() As String

Get

Return mAlamat

End Get

Set(ByVal value As String)

mAlamat = value

End Set

End Property

Public Property TglLahir() As Date

Get

Return mTglLahir

End Get

Set(ByVal value As Date)

mTglLahir = value

End Set

End Property

End Class

Buat sub **class** dengan nama DosenTetap, dan buat program berikut

'Atribut yang akan dioverload ditaruh dalam has tabel

Public Enum JenisNama

Lengkap = 1

Panggilan = 2

Gelar = 3

End Enum

Public Class DosenTetap

Inherits Dosen

Private mKolomNama As New Hashtable'memanggilnilai has tabel

Private mSKS As Integer

Private mHonor As Single

Public Overloads Property Nama(ByVal type As JenisNama) As

String

Get

Return mKolomNama(type)

End Get

Set(ByVal value As String)

mKolomNama.Add(type, value) End Set

End Property

Public Property SKS() As Integer

Get

Return mSKS End Get

Set(ByVal value As Integer)

mSKS = value

End Set

End Property

Public Property Honor() As Single

Get

Return mHonor

End Get

Set(ByVal value As Single)

mHonor = value

End Set

End Property

'Method menghitung TotalHonor

Public Function TotalHonor(ByVal iSks As Integer, ByVal iHonorSks As Single) As Single

Return iSks \* iHonorSks

End Function

End Class

Double click **cmdProses**, buat program berikut.

Private Sub cmdProses\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles cmdProses.Click

Dim objekDosenTetap As New DosenTetap

With objekDosenTetap

.Nama = txtNama.Text

.Nama(JenisNama.Lengkap) = txtNamaLengkap.Text

'atribut yang dioverloading

.Nama(JenisNama.Panggilan) = txtNamaPanggilan.Text

'atribut yang dioverloading

.Nama(JenisNama.Gelar) = txtGelar.Text 'atribut yang dioverloading

.Alamat = txtAlamat.Text

.TglLahir = txtTglLahir.Text

.SKS = txtSKS.Text

.Honor = txtHonor.Text

lblTotalHonor.Text = Format(.TotalHonor(.SKS,

.Honor), "Rp 0,00")

End With

End Sub

Double click **cmdCancel**, buat program berikut.

Private Sub cmdKeluar\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles cmdKeluar.Click

End

End Sub