

## Practical No 20 & 21

### VIII. Resources required (Additional)

→ If any web resources required.

### X. Resources used (Additional)

<https://www.tutorialspoint.com/vb.net/vb.net>

### XI. Program Code

1. Write a program to implement the concept of method overloading & overriding.

➤ **Overloading:**

Module Module1

Sub Main()

Dim res As New addition

Console.WriteLine("Overloaded Values of Class addition")

Console.WriteLine(res.add(10))

Console.WriteLine(res.add(35, 20))

Console.ReadLine()

End Sub

End Module

Public Class addition

Public i, j As Integer

Public Function add(ByVal i As Integer) As Integer

Return i

End Function

Public Function add(ByVal i As Integer, ByVal j As Integer) As Integer

Return i + j

End Function

End Class

### Output:

Overloaded Values of Class addition

10

55

### Overriding:

Module Module1

```
Sub Main()
    Dim obj As New child
    Dim result As Integer
    result = obj.add(10, 5)
    Console.WriteLine("Overloaded Values of Class addition")
    Console.WriteLine("Result =" & result)
    Console.ReadLine()
End Sub
End Module
Public Class parent
    Public Overridable Function add(ByVal i As Integer, ByVal j As Integer)
        Return (i + j)
    End Function
End Class

Public Class child
    Inherits parent
    Public Overrides Function add(ByVal i As Integer, ByVal j As Integer)
        Console.WriteLine("Result of Addition =" & MyBase.add(12, 18))
        Return (i + j)
    End Function
End Class
```

### Output:

Result of Addition =30  
Overloaded Values of Class addition  
Result =15

### XII. Results (output of the program)

In the above overloading example the same function add is called to perform different operations based on different arguments.

In the above overriding the parent class function add is overridden in the child class using the `MyBase.add(12, 18)` statement. So first the overridden value is displayed, then the value from child class is display.

### XIII. Practical related Questions

1. Find output of following Code.

- Area of the Circle : 31.181246
- Area of the Rectangle : 20

2. Implement windows application for employee details using overriding method.

- Module Module1

```
Sub Main()  
    Dim obj As New EmpInfo  
    obj.ShowInfo()  
    Console.ReadLine()  
End Sub  
  
End Module  
Public Class EmpPersonalDetails  
    Dim name As String  
    Dim address As String  
    Public Overridable Function ShowInfo()  
        Console.WriteLine("Employee Name" & name)  
        Console.WriteLine("Employee Address" & address)  
    End Function  
End Class  
Public Class EmpInfo  
    Inherits EmpPersonalDetails  
    Dim EmpId As Integer  
    Dim sallary As Integer  
    Dim JoinDate As Date  
    Overloads Function ShowInfo()  
        MyBase.ShowInfo()  
        Console.WriteLine("Employee ID" & EmpId)  
        Console.WriteLine("Employee Sallary" & sallary)  
        Console.WriteLine("Employee Joining Date" & JoinDate)  
    End Function  
End Class
```

OUTPUT:

Employee Name  
Employee Address  
Employee ID0  
Employee Sallary0  
Employee Joining Date12:00:00 AM

#### XIV. Exercise

1. Implement a windows application for show string concatenation using overload method



Public Class Form1

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

Dim str1, str2, str3 As String

str1 = TextBox1.Text

str2 = TextBox2.Text

str3 = str1 + str2

TextBox3.Text = str3

End Sub

End Class

Output: