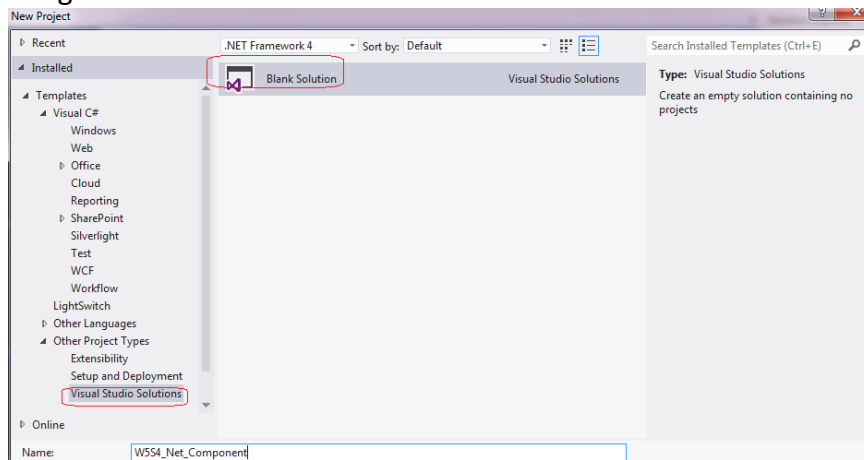


# IE3213323-IE420323 Pemrograman Berbasis Komponen dan Framework

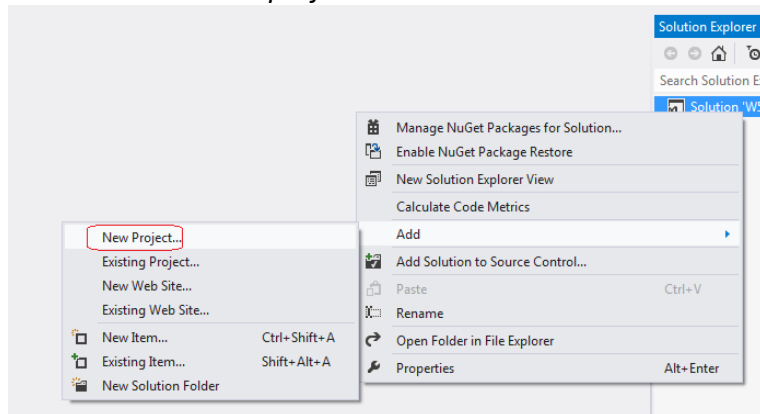
<b>Minggu/Sesi</b>	:	13/4
<b>Topik</b>	:	.Net Component
<b>Tanggal Pengumpulan</b>	:	
<b>Direktori</b>	:	
<b>Instruktur</b>	:	RJS, BPS
<b>Contents</b>		
A. Build Component .....		2
B. Build Client Consume Component.....		4
C. Using Third Party Component: Read pdf content .....		6
D. Using COM: Text to Spech .....		10
E. Using COM: Office.....		12
F. Daftar Pustaka.....		15

## A. Build Component

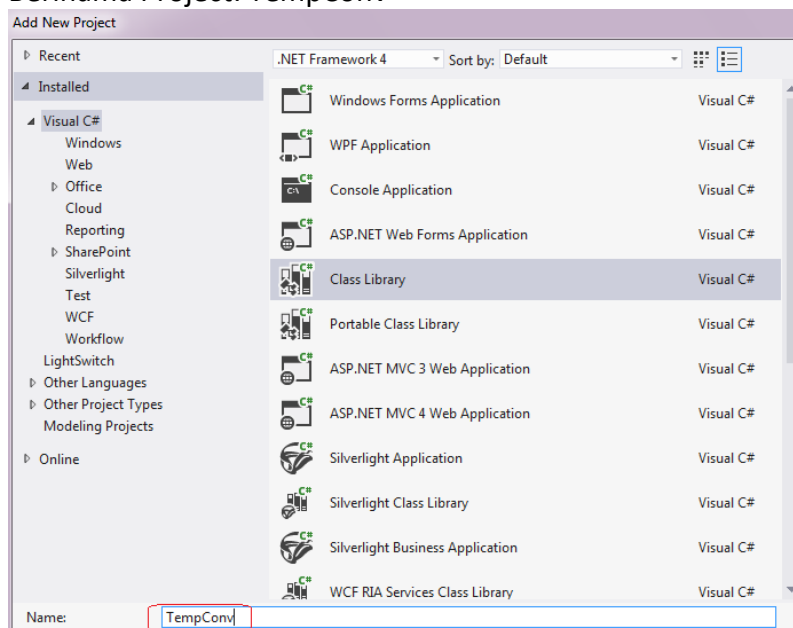
Bangunlah sebuah BlankSolution di Visual Studio:



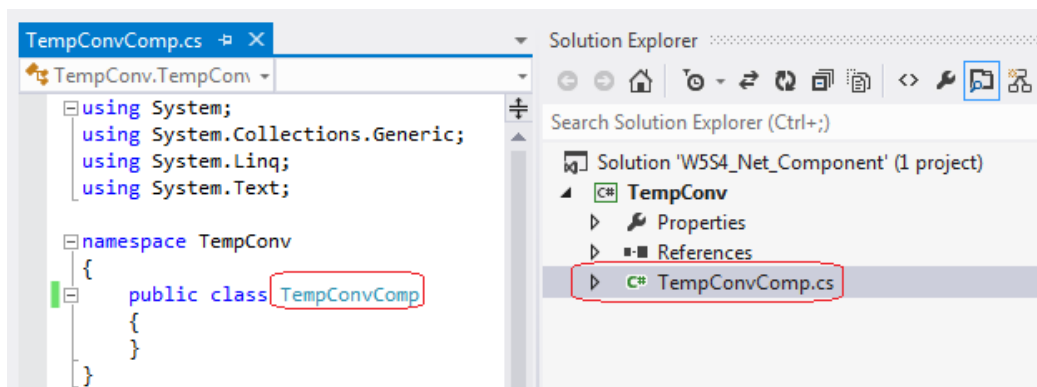
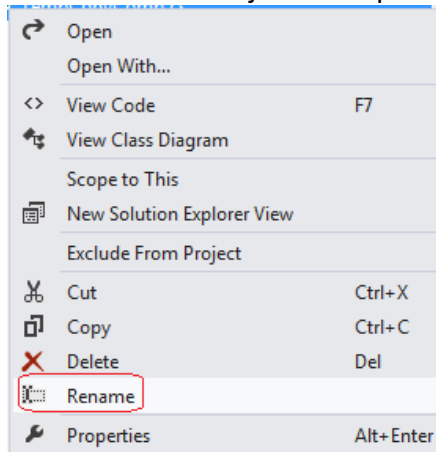
Tambahkan sebuah *project*:



Berinama Project: TempConv



Rename *class* menjadi : TempConvComp



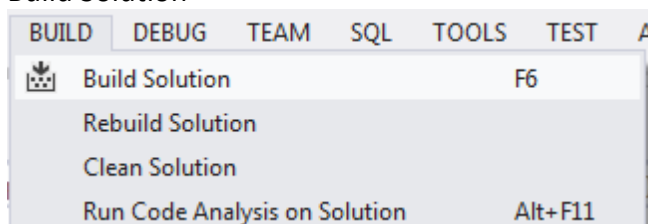
Tambahkan code berikut:

```
public class TempConvComp
{
    public TempConvComp() { }

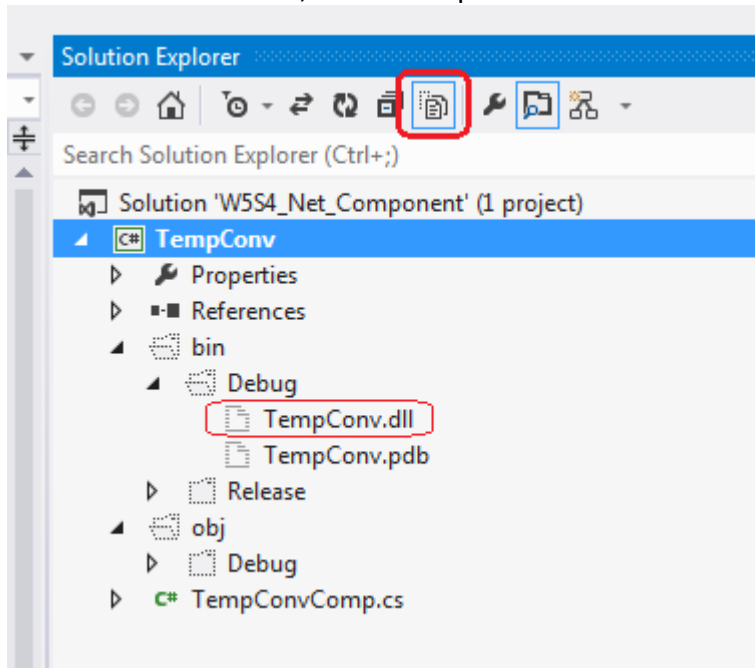
    public double cToF(double c)
    {
        return (int)((c * 9 / 5.0 + 32) * 100) / 100.0;
    }

    public double fToC(double f)
    {
        return (int)((f - 32) * 5 / 9.0 * 100) / 100.0;
    }
}
```

Build Solution

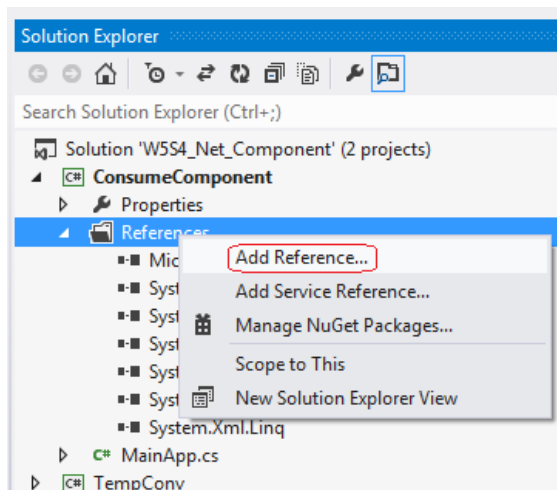


Pastikan Show All Files, maka .dll dpat dilihat di folder Debug (TempConv.dll)

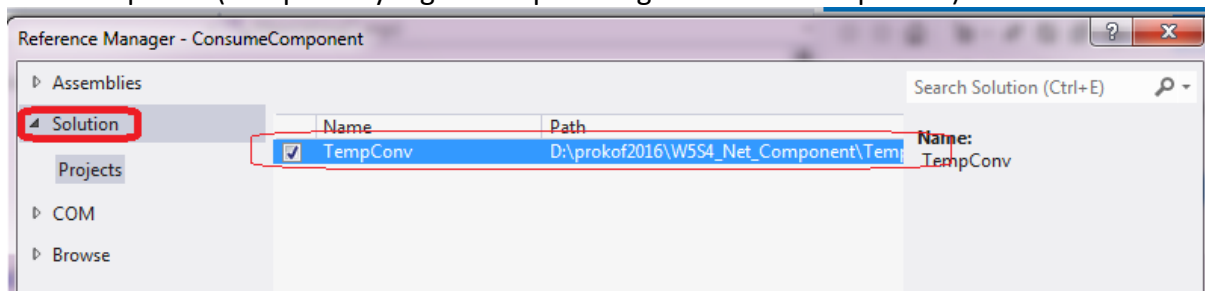


## B. Build Client Consume Component

Buatlah sebuah project console, kemudian tambahkan component yang telah dibuat sebagai reference:



Pilih TempConv (komponen yang dibuat pada bagian A.Build Component)



Pada bagian directive, tambahkan code berikut:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using TempConv;
```

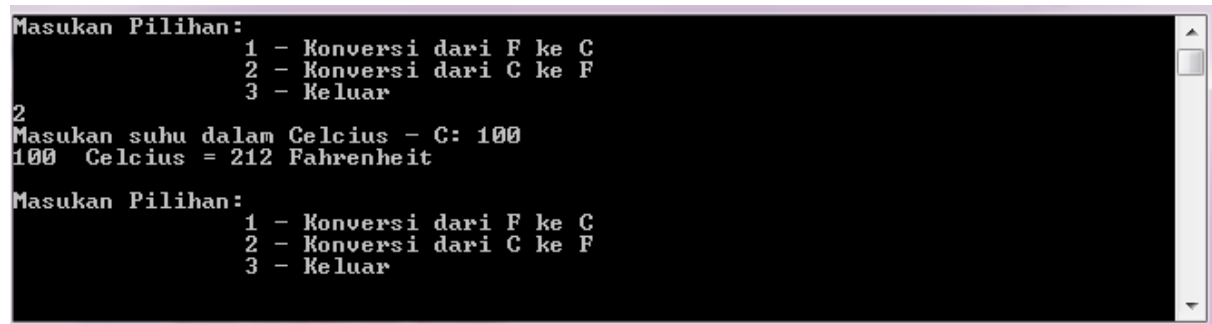
Tujuan code diatas supaya TempConv dapat digunakan.

Tambahkan Code Berikut:

```
static void Main()
{
    TempConv.TempConvComp konversiTemperatur = new TempConv.TempConvComp();
    double choice;
    double input;
    double output;
    bool next = true;

    while (next)
    {
        Console.WriteLine(@"Masukan Pilihan:
        1 - Konversi dari F ke C
        2 - Konversi dari C ke F
        3 - Keluar");
        choice = Double.Parse(Console.ReadLine());

        if (choice == 1)
        {
            Console.Write("Masukan Suhu dalam Fahrenheit-F: ");
            input = Double.Parse(Console.ReadLine());
            output = konversiTemperatur.fToC(input);
            Console.WriteLine("{0} Fahrenheit = {1} Celcius", input, output);
            Console.WriteLine();
        }
        else if (choice == 2)
        {
            Console.Write("Masukan suhu dalam Celcius - C: ");
            input = Double.Parse(Console.ReadLine());
            output = konversiTemperatur.cToF(input);
            Console.WriteLine("{0} Celcius = {1} Fahrenheit", input, output);
            Console.WriteLine();
        }
        else
        {
            next = false; Console.WriteLine("Sampai Jumpa"); }
    }
}
```



```
Masukan Pilihan:
    1 - Konversi dari F ke C
    2 - Konversi dari C ke F
    3 - Keluar
2
Masukan suhu dalam Celcius - C: 100
100 Celcius = 212 Fahrenheit
Masukan Pilihan:
    1 - Konversi dari F ke C
    2 - Konversi dari C ke F
    3 - Keluar
```

## C. Using Third Party Component: Read pdf content

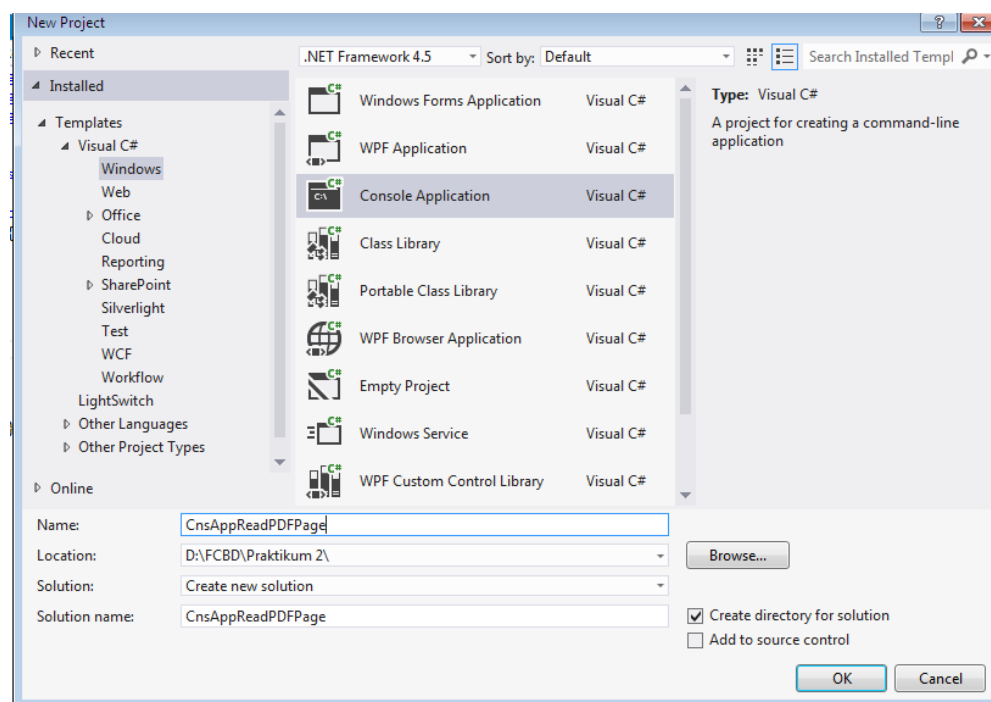
Aplikasi adakalanya membutuhkan fitur yang dapat membaca jumlah halaman file, misalnya suatu aplikasi upload file membutuhkan informasi jumlah halaman pdf yang sedang diupload.

Pada praktikum kali ini kita akan menggunakan komponen yang dibuat oleh pihak ketiga (bukan Microsoft) untuk digunakan dalam lingkungan visual studio. Komponen yang akan kita gunakan dibuat oleh Itext sebuah open source library (<http://itextpdf.com/>), library tersebut dapat didownload di <http://sourceforge.net/projects/itext/files/>, untuk praktikum ini akan disediakan sebuah .dll : **itextsharp.dll**

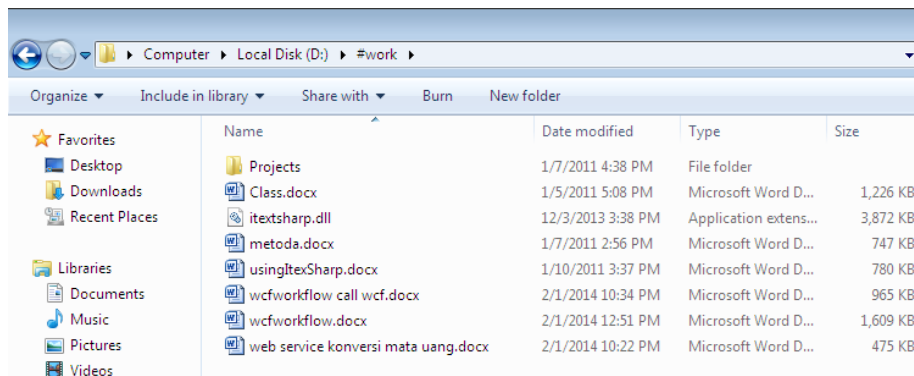
Langkah – langkah praktikum:

Bangunlah sebuah aplikasi Console sebagai berikut:

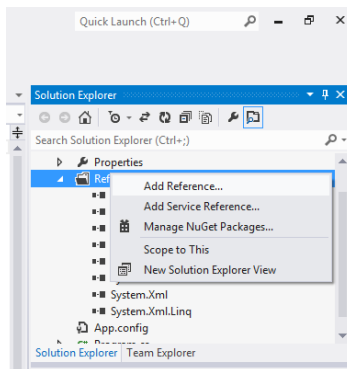
*Name: CnsAppReadPDFPage*



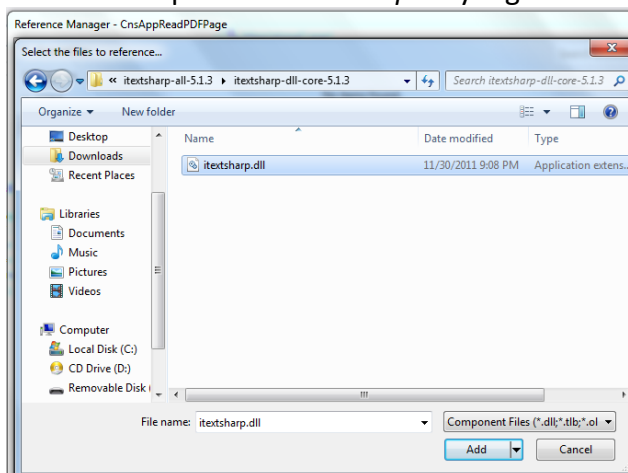
Pastikan file **itextsharp.dll** sudah ada di local hardisk



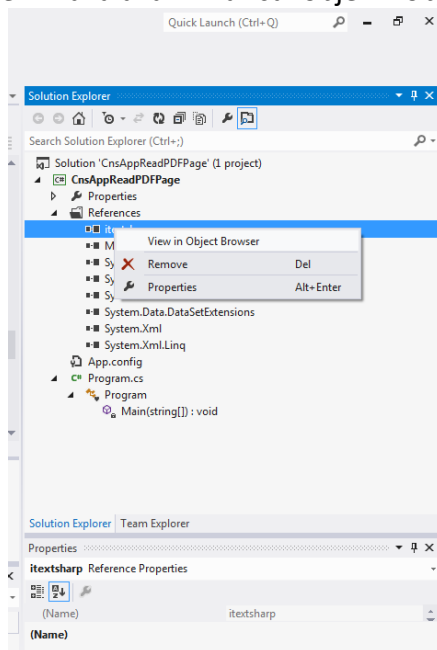
Kemudian pergi ke Solution Explorer dan klik kanan *Reference* dan *Add reference*



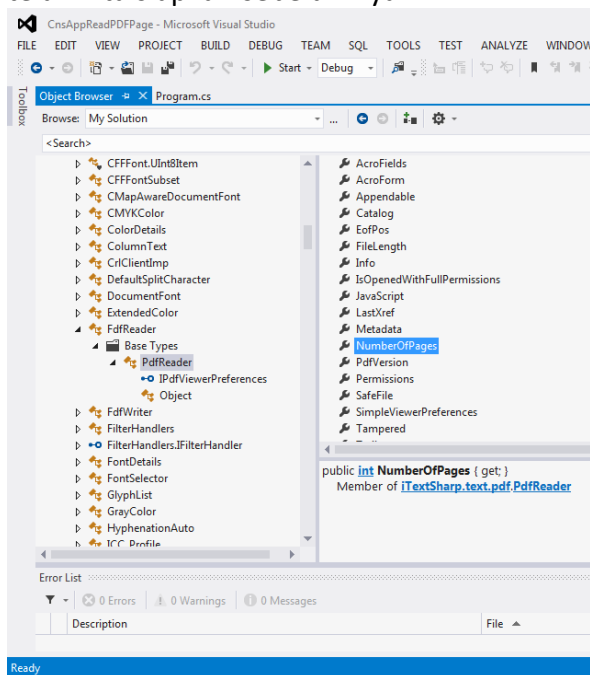
Arahkan dan pilih file **itextsharp.dll** yang telah disimpan di hardisk



Dibawah Reference akan muncul *itextsharp* klik kanan disini dan pilih *View in Object Browser* maka akan muncul objek – objek yang disediakan library tersebut.



Arahkan ke *iTextSharp.text.pdf* kemudian ke *PdfReader* dan *NumberOfPages*. Dalam praktikum ini kita akan menggunakan method *NumberOfPages* untuk membaca file pdf yang telah kita siapkan sebelumnya



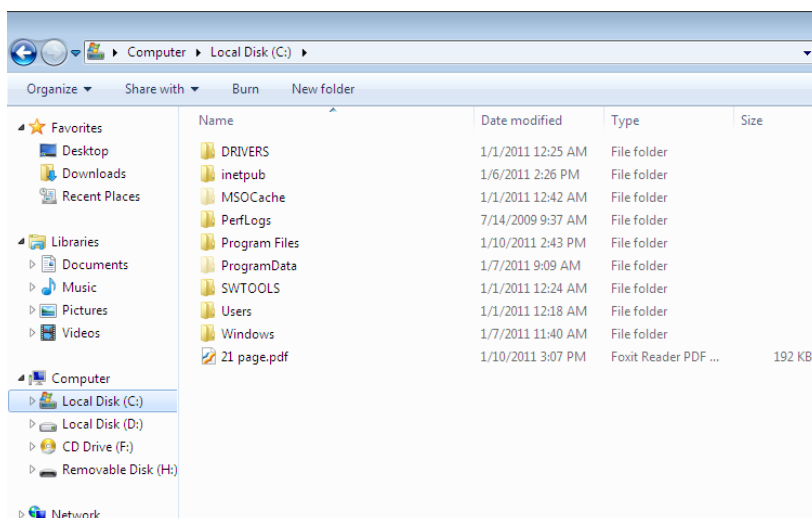


Ubah code di class Program menjadi :

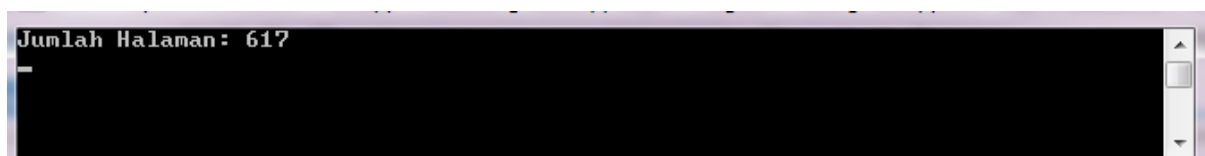
```
using System;
using iTextSharp.text.pdf;

namespace CnsAppReadPDFPage
{
    class Program
    {
        static void Main(string[] args)
        {
            // Lokasi file pdf yang akan dibaca
            string path = "C:/21 page.pdf";
            PdfReader bacaPDF = new PdfReader(path);
            int jumHalaman = bacaPDF.NumberOfPages;
            Console.WriteLine("Jumlah Halaman: {0}", jumHalaman);
            Console.ReadLine();
        }
    }
}
```

Dan pastikan file '*21 page.pdf*' sudah ada di hardisk dan terletak di folder c , file pdf tersebut berisi 21 halaman.



'Start Debugging' dengan menekan tombol F5, akan muncul angka 21 yang menunjukkan halaman file pdf yang dibaca oleh aplikasi yang dibangun.

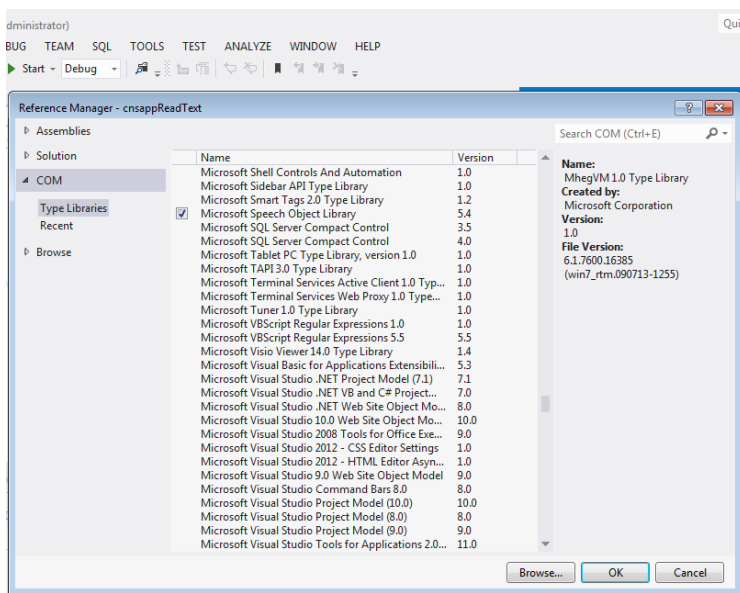
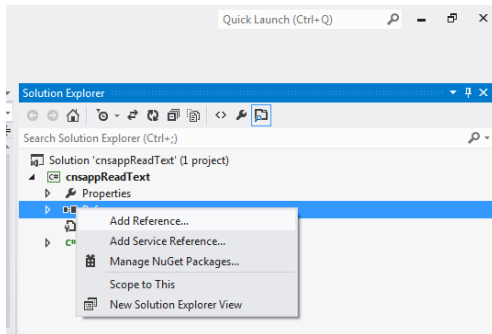


## D. Using COM: Text to Speech

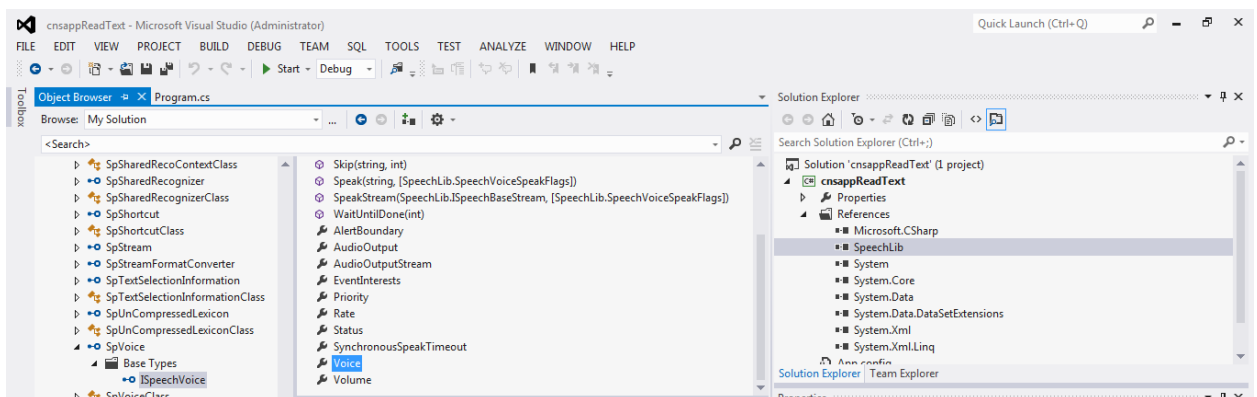
COM (Component Object Model), penjelasannya tentang COM dapat dilihat di

<http://www.microsoft.com/com/default.mspx>

Bangun sebuah aplikasi console dan tambahkan reference dari 'COM' dan tambahkan *Microsoft Speech Object Library*



Kita akan menggunakan objek dari *SpeechLib*

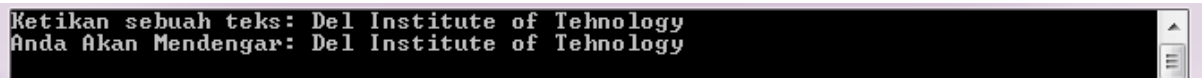


Tambahkan code berikut ke class Program

```
using System;
using SpeechLib;

namespace cnsappReadText
{
    class Program
    {
        static void Main(string[] args)
        {
            SpVoice voice = new SpVoice();
            Console.Write("Ketikan sebuah teks: ");
            string teks = Console.ReadLine();
            Console.WriteLine("Anda Akan Mendengar: {0}", teks);
            voice.Speak(teks, SpeechVoiceSpeakFlags.SVSFDefault);
            Console.ReadLine();
        }
    }
}
```

Tekan tombol F5



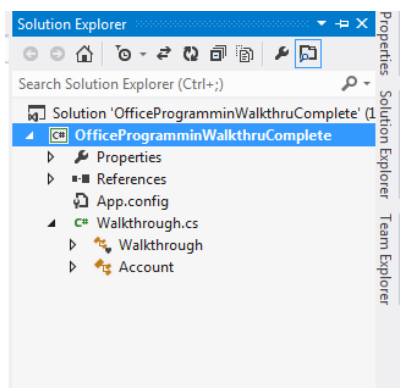
Ketikan sebuah teks: Del Institute of Tehnology  
Anda Akan Mendengar: Del Institute of Tehnology

## E. Using COM: Office

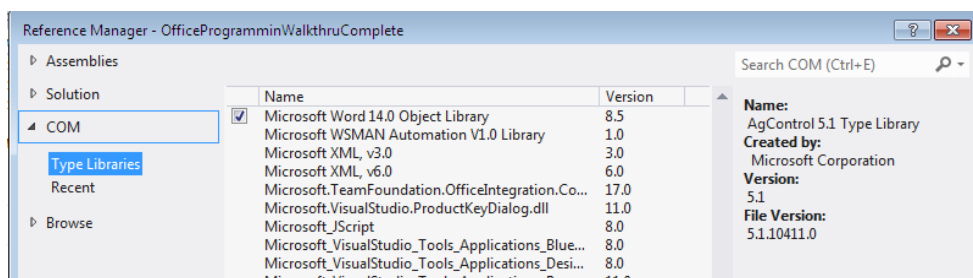
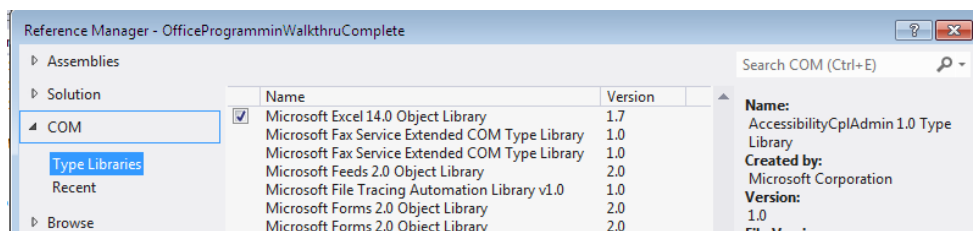
Pada Bagian ini akan ditunjukkan bagaimana memanfaatkan library microsoft office yang ada di visual studio. Aplikasi console yang akan dibangun adalah mengirim teks ke Microsoft excel dan membuat icon yang di Microsoft Word yang menjadi link ke table excel yang telah dibuat oleh aplikasi.

Langkah – langkah praktikum:

Bangunlah sebuah aplikasi console dengan nama : OfficeProgramminWalkthruComplete dan ubah nama program.cs menjadi Walkthrough.



Tambahkan reference Microsoft Excel 14.0 Object Library dan Microsoft Word 14.0 Object Library



Code di program utama *class Walkthrough* sebagai berikut:

```
using System;
using System.Collections.Generic;
using System.Linq;
using Excel = Microsoft.Office.Interop.Excel;
using Word = Microsoft.Office.Interop.Word;

namespace OfficeProgramminWalkthruComplete
{
    class Walkthrough
    {
        static void Main(string[] args)
        {
            var bankAccounts = new List<Account>
            {
                new Account {
                    ID = 345678,
                    Balance = 541.27
                },
                new Account {
                    ID = 1230221,
                    Balance = -127.44
                }
            };

            DisplayInExcel(bankAccounts);

            CreateIconInWordDoc();
        }
    }
}
```

```

static void DisplayInExcel(IEnumerable<Account> accounts)
{
    var excelApp = new Excel.Application();

    excelApp.Visible = true;

    excelApp.Workbooks.Add();

    Excel._Worksheet workSheet = excelApp.ActiveSheet;

    workSheet.Cells[1, "A"] = "ID Number";
    workSheet.Cells[1, "B"] = "Current Balance";

    var row = 1;
    foreach (var acct in accounts)
    {
        row++;
        workSheet.Cells[row, "A"] = acct.ID;
        workSheet.Cells[row, "B"] = acct.Balance;
    }

    workSheet.Columns[1].AutoFit();
    workSheet.Columns[2].AutoFit();

    workSheet.Range["A1", "B3"].AutoFormat(
        Excel.XlRangeAutoFormat.xlRangeAutoFormatClassic2);

    workSheet.Range["A1:B3"].Copy();
}

static void CreateIconInWordDoc()
{
    var wordApp = new Word.Application();
    wordApp.Visible = true;

    wordApp.Documents.Add();

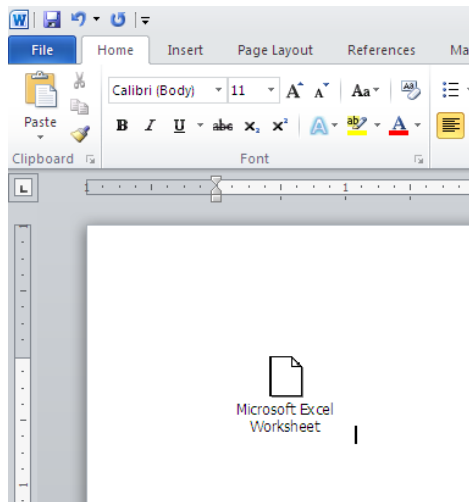
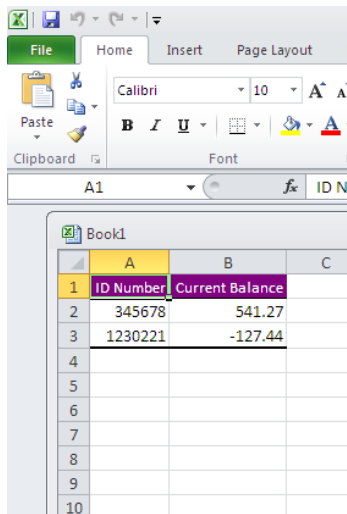
    wordApp.Selection.PasteSpecial(Link: true, DisplayAsIcon: true);
}

public class Account
{
    public int ID { get; set; }
    public double Balance { get; set; }
}

```

Tekan tombol F5 (Start Debugging)

Akan membuka Microsoft Excel dan Microsoft Word sebagai berikut



Sumber: <http://msdn.microsoft.com/en-us/library/dd264733.aspx>

## F. Daftar Pustaka

Andy Ju, Wang Kai Qian, Component Oriented Programming, Jhon Willey and Son 2005