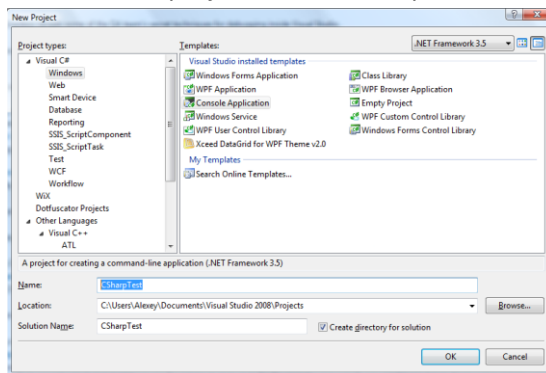


DlpLib HowTo

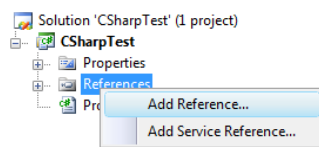
This article describes how to incorporate ID Parsing component (developed by IDScan.net)
You can download free Example projects from our web-site <http://www.IDScan.net/dlplib> or <http://www.IDScan.net/downloads>

C#

1. Install a Microsoft Visual Studio. You can download free Express Edition here <http://www.microsoft.com/express/download/default.aspx>
2. Create a new project. In this example it is Console Application:



3. Open Package Manager Console
View->Other Windows->Package Manager Console
4. Select preferred method
 - a. Type or copy and paste the following command
Install-Package dlp-sdk -s <https://www.myget.org/F/dlp-sdk/api/v2>
 - b. Add reference to the library DlpLib.dll if you prefer to use a local copy.
Under tab **Browse** select the file DlpLib.dll.



5. Now you are ready to write an application:

```
static void Main(string[] args)
{
    // Load a driver license text from a file. Don't forget to put testdl.txt next to your exe file.
    var scannedText = File.ReadAllText(@"testdl.txt");
    // Parse the text and create an instance of DriverLicense.
    // All fields in DriverLicense are strings.
    // When the parser is not able to parse the text it returns null. Parser doesn't throw exceptions.
    var license = DriverLicense.ParseText(scannedText);

    if (license != null)
    {
        //This functions demonstrates a simple example
        Console.WriteLine("-----Basic Usage-----");
        BasicUsage(license);

        // If you need to enumerate all field you can use DataElementAttribute and reflection
        Console.WriteLine("-----Print All Fields-----");
        PrintAllFields(license);
    }

    // However, if you need typed variables or you need to know what was found in the license you should use DriverLicenseEx
    // It does the same job but the field types can be string, DateTime?, Char? or bool?
    // If a field was not present in the license string this field is null.
    var licenseEx = DriverLicenseEx.ParseText(scannedText);
    if (licenseEx != null)
    {
        Console.WriteLine("-----Print All Found Fields-----");
        PrintAllFoundFields(licenseEx);
    }
}

private static void BasicUsage(DriverLicense license)
{
    // Basic usage
    Console.WriteLine(license.FirstName);
    Console.WriteLine(license.LastName);
    Console.WriteLine();
}

private static void PrintAllFields(DriverLicense license)
{
    //Enumerate all fields
    var properties = license.GetType().GetProperties();
    var foundFields = (from info in properties
        let element = (DataElementAttribute[])info.GetCustomAttributes(typeof(DataElementAttribute), false)
        where element.Length > 0
        select info).ToList();

    foundFields.Sort((x, y) => string.Compare(x.Name, y.Name, StringComparison.Ordinal));

    for (var i = 0; i < foundFields.Count; i++)
    {
        var fieldName = foundFields[i].Name;
        var fieldValue = foundFields[i].GetValue(license, null);
        Console.ForegroundColor = ConsoleColor.White;
        Console.Write("{0,-3}{1,-30}", i + 1, fieldName);
        Console.ForegroundColor = ConsoleColor.Yellow;
        if (fieldName.EndsWith("Date") && !string.IsNullOrEmpty(fieldValue.ToString()))
        {
            DateTime dt;
            var res = DateTime.TryParseExact(fieldValue.ToString(), "MMdyyyy", CultureInfo.InvariantCulture, DateTimeStyles.None, out dt);
            Console.WriteLine("{0}", res ? dt.ToString("MM/dd/yyyy") : fieldValue);
        }
        else
        {
            Console.WriteLine("{0}", fieldValue);
        }
    }
}

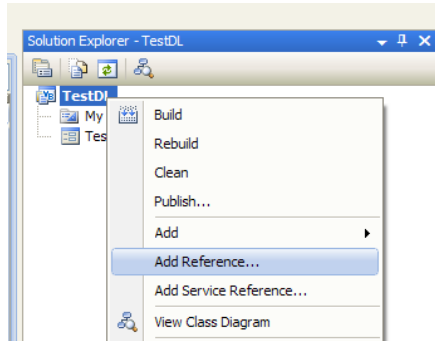
private static void PrintAllFoundFields(DriverLicenseEx license)
{
    //Enumerate all fields
    var properties = license.GetType().GetProperties();
    var foundFields = (from info in properties
        let element = (DataElementAttribute[])info.GetCustomAttributes(typeof(DataElementAttribute), false)
        where element.Length > 0
        select info).ToList();

    foundFields.Sort((x, y) => string.Compare(x.Name, y.Name, StringComparison.Ordinal));

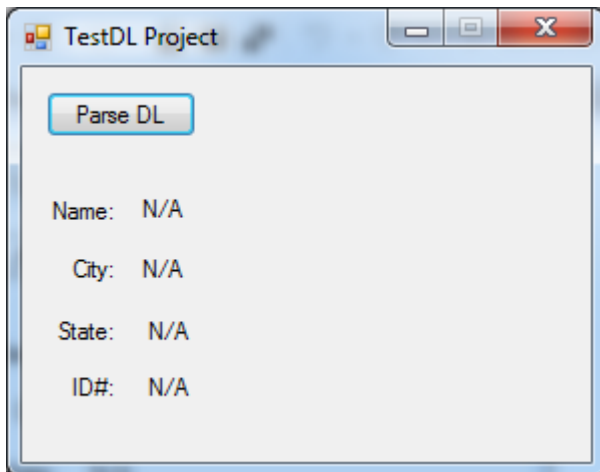
    for (var i = 0; i < foundFields.Count; i++)
    {
        var fieldName = foundFields[i].Name;
        var fieldValue = foundFields[i].GetValue(license, null);
        Console.ForegroundColor = ConsoleColor.White;
        Console.Write("{0,-3}{1,-30}", i + 1, fieldName);
        Console.ForegroundColor = ConsoleColor.Yellow;
        Console.WriteLine("{0}", fieldValue?.ToString() ?? "N/A");
    }
}
```

VB.NET

1. Install a Microsoft Visual Studio. You can download free Express Edition here <http://www.microsoft.com/express/download/default.aspx>
2. Create a new project. In this example it is Windows Application:
3. Add reference to the library DlpLib.dll. Right click on your project and select "Add Reference..."
Select tab **Browse** and then select the file DlpLib.dll



4. Now you are ready to write an application. Place a few labels and a button on the form.



5. Create an event for the Button click(Double click on your new button to create an event).

```
Imports System.Collections.Generic
Imports System.IO
```

```
Public Class TestDLForm
```

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

```
    'Path to the Example File
    Dim fileName = "c:\TestDL.txt"
    Dim lineList As String = File.ReadAllText(fileName)

    'Make sure you have Reference to DlpLib.dll (use command Add Reference...)
    Dim dl = Nautilus.DriverLicense.ParseText(lineList)
    If IsDBNull(dl) = True Then
        MsgBox("Can't Recognize DL", MsgBoxStyle.Information)
    Else
        Me.lbl_Name.Text = dl.FirstName + " " + dl.LastName
        Me.lbl_state.Text = dl.JurisdictionCode
        Me.lbl_idnum.Text = dl.LicenseNumber
        Me.lbl_City.Text = dl.City
    End If
```

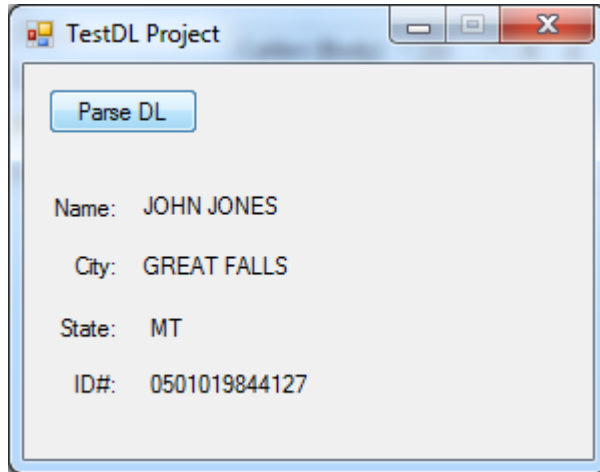
```
End If
```

End Sub

End Class

6. Run the application and press the “Parse DL button”

You should see results similar to the screen below



How to use your IDScan.net™ SDK License Key

If you don't have a Valid License Key please contact IDScan.net™ or authorized reseller where you purchased the IDScan.net™ SDK.

With your purchase you should receive an email with license file *dlplib.lic*

You need to place *dlplib.lic* file in one of the following locations:

- In the same folder with *dlplib.dll* file
- In the same folder with your executable file
- In a folder specified in the Environment variable **PATH**

You can update/renew your license without making any changes to the source code.