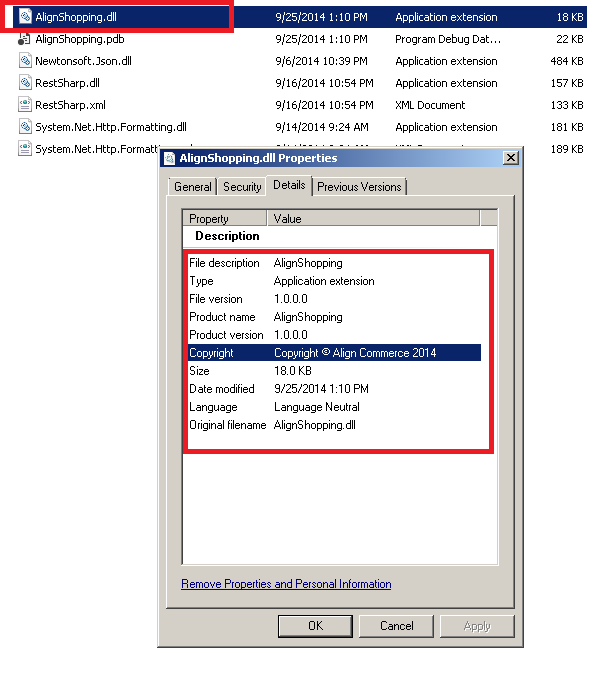
****

**Align Commerce Shopping Library**

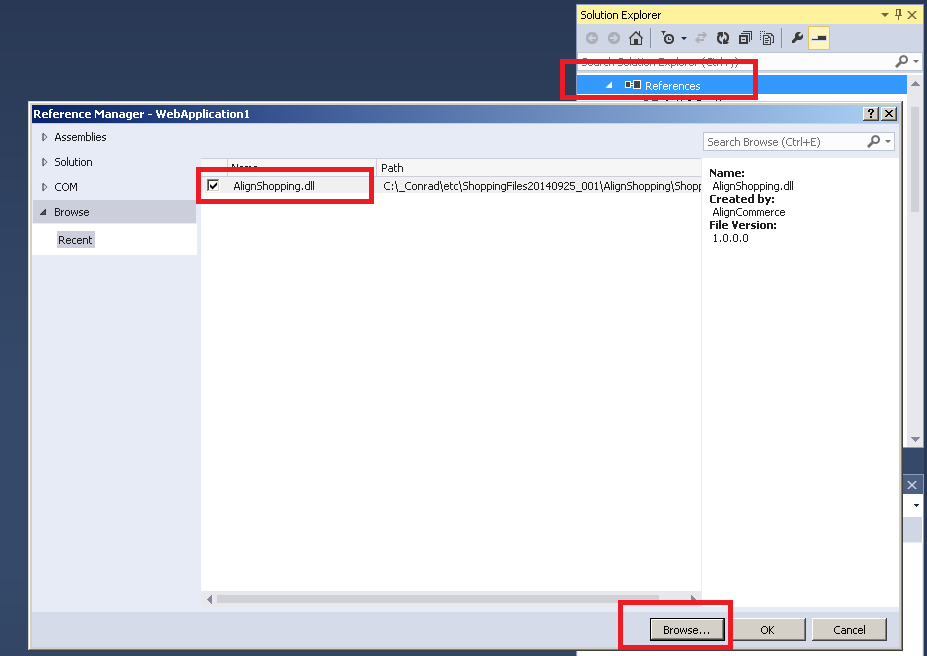
**ASP.NET 4.5**

**Visual Studio 2013**

**DLL Properties:**

****

**REFERENCE IN ASP.NET PROJECT:**

****

**EXAMPLE:**

using AlignShopping;

namespace WebApplication1

{

public partial class CreateInvoice : System.Web.UI.Page

{

protected void Button1\_Click(object sender, EventArgs e)

{

AlignAPI a = new AlignAPI();

a.SetLogin("meail@mydomain.com"

, "password"

, "client\_id"

, "client\_secret"

, "client\_credentials"

, "scope"

, "random string");

string returnvalue = a.CreateInvoice(tx\_accesstoken.Text

, tx\_checkout\_type.Text

, tx\_product\_name.Text

, Convert.ToDouble(tx\_product\_price.Text)

, Convert.ToInt32(tx\_quantity.Text)

, Convert.ToDouble(tx\_product\_shipping.Text)

, tx\_firstname.Text

, tx\_lastname.Text

, tx\_address1.Text

, tx\_address2.Text);

**AlignAPI Details**

**Note:**

* Before Call OF API Functions:
  + **SetLogin()** Must Be Called.
* Include The C# DLL file in reference.
* To call the Cart API create object of AlignAPI

AlignAPI obj = new AlignAPI();

now you can access all the functions. Call the function like this:

obj.**SetLogin**(user, pass, credentials, client\_id, client\_secret, scope, Random String);

**To Set The Login**

**Function Name With Parameter**

public void **SetLogin**(user, pass, credentials, client\_id, client\_secret, scope, Random String)

function return type is void..

**Usage:**

user: type string

your username

pass: type string

your password

credentials: type string value must be client\_credentials

client\_id: type string Your Client ID

scope: type string

value can be "product,buyer,invoice "

Random String: type string

value can be rondom string

**Function Call:**

**SetLogin**(user, pass, credentials, client\_id, client\_secret, scope, Random String);

**Get Access Token**

**Function Name With Parameter:**

string GetAccessToken()

Function return type string...function parameters are void.

**Usage:**

GetAccessToken()

It provide an access token which required for all the API TASKS.

**Functon Call:**

GetAccessToken();

It will return an access token in string format...

**List ALL Products**

**Usage**

ListProduct(AccessToken);

The function list of all products in JSON Format...

**Function Call:**

ListProduct(AccessToken);

provide Accesstoken in parameter...

**Create Product**

**Usage**

Create a product

* access\_token string Required. Provided in Oauth flow.
* product\_name string Required. Product name should be unique.
* product\_price numeric / float Required. A numeric / float value that represents the price of the product.
* product\_shipping numeric / float A numeric / float value that represents how much the cost if the product would be shipped to the buyer.
* product\_sku string Merchant's sku for the product.

**Function Call:**

CreateProduct(string access\_token,string product\_name,string product\_discription,double product\_price,double product\_shipping,string product\_sku)

string access\_token: Provide access token

string product\_name: Provide Product Name

string product\_discription : Provide Product Discription

double product\_price : Provide Product Price

double product\_shipping : Provide Product Shipping

string product\_sku : Provide Product SKU

Pass The Above Parameters

Function return type string in JSON string format.

**Update Product**

public string UpdateProduct(string access\_token,string updateid, string product\_name, string product\_discription, double product\_price, double product\_shipping, string product\_sku)

**Usage:**

* access\_token string Required. Provided in Oauth flow.
* updateid string Required. Product ID
* product\_name string Required. Product name should be unique.
* product\_price numeric / float Required. A numeric / float value that represents the price of the product.
* product\_shipping numeric / float A numeric / float value that represents how much the cost if the product would be shipped to the buyer.
* product\_sku string Merchant's sku for the product.

**Function Call:**

UpdateProduct(string access\_token,string updateid, string product\_name, string product\_discription, double product\_price, double product\_shipping, string product\_sku)

It Will Update The Product & return response in JSON string.

**List All Invoices**

GetInvoieALL(string token)

provide access token

**Function Call:**

GetInvoieALL(string token);

it will provide all the Invoices IN JSON string format...

**List Specific Product**

ListSpecificInvoice(string token, string id)

provide access token and invoice id

**Function Call**

ListSpecificInvoice(string token, string id)

provide access token and id of invoice

it will return the response in JSON string format.

**Create Invoice**

**Function Call.**

public string CreateInvoice(string access\_token,string Checkout\_Type,string product\_name,double product\_price,int quantity,double product\_shipping,string firstname,string lastname,string address1,string address2)

provide access token,checkout type,product name,product price,quantity,product shipping,first name,last name,address 1 and address 2

**List All Buyer**

provide access token

Function Call:

ListALLBuyer(string token)

return JSON string..

**List Specific Buyer**

provide access token and buyer id..

GetSpecificBuyer(string token, string id)

returns JSON string