

iCredit

התממשקות לדפי חיוב
מאתר מסחר אלקטרוני

גרסא : 1.0.2.3
עדכון אחרון : 03/08/2014

הקדמה :

מערכת iCredit לסליקת כרטיסי אשראי הינה מערכת מוסמכת בתקן PCI DSS Service Provider Level 1.

באמצעות iCredit יכול כל בית עסק לסלוק כרטיסי אשראי באתר האינטרנט ולשלב אותו עם מערכת חנות המסחר האלקטרוני (E-Commerce) בה האתר משתמש.

יתרונות דפי הסליקה ב-iCredit :

- חיבור טבעי למערכת הנהלת החשבונות ריווחית \ Online ויכולת להפיק חשבונית מס קבלה בצורה פשוטה וקלה.
- הטמעת המערכת קלה ביותר.
- גמישות מירבית בעיצוב דפי החיוב (הגדרת שדות, הערות, מבנה הטופס וכד').

תוכנת ריווחית \ Online הינה הגרסה ה-web-ית של תוכנת ריווחית \ ניהול עסק, מערכת ה-ERP המתקדמת והידידותית. מערכת ריווחית מהווה Back Office מלא לפעילות אתר האינטרנט.

הממשק שיתואר בהמשך מסביר כיצד להטמיע את המערכת באתר המסחר באתרך.

סביבת Test :

בכל שלבי הפיתוח יש להשתמש בסביבת ה-Test בלבד.

אין להזין בסביבת ה-Test מספרי כרטיס אשראי אמיתיים, יש להשתמש בכרטיסי בדיקות בלבד.

להלן רשימת כרטיסי טסט לשימוש :

מספר כרטיס : 4580000000000000 (ויזה)

מספר כרטיס : 5326000000000000 (מסטרקארד)

במערכת הסליקה ישלח אליך במייל ערך פרטי לפרמטר GroupPrivateToken ודוגמת קוד לשימוש (בד"כ בJSON).

בממשק לתוכנת ריווחית \ Online יש להשתמש בעסק הבדיקות בלבד עפ"י פרטי ההתחברות הבאים :

שם משתמש : demo

סיסמה : 123

מספר ע.מ. : 8888

הטמעת קוד באתר :

על מנת לשלב את דפי החיוב של iCredit באתר מסחר אלקטרוני יש לבצע את השלבים הבאים :

1. הגדרת קבוצת מכירה בממשק הניהול – קבוצת המכירה מכילה את מבנה ועיצוב דף החיוב, והפריטים הנמכרים בכל דף.

2. בקבוצת המכירה יש לשים לב לפרטים הבאים :

א. יש לשמור את שדה מזהה קבוצה – קוד זה ישמש אותנו להזדהות כאשר נבקש לעבור לדף חיוב ממערכת המסחר האלקטרוני.

ב. יש להגדיר עבור קבוצת המכירה את הפריטים שאנו מצפים שישלחו לרכישה בדף חיוב זה.

ג. יש להגדיר את כתובת ה-Redirect להצלחה אליה יועבר הלקוח בסיום העסקה.

3. במידה ומעוניינים להפיק חשבוניות באמצעות ריווחית \ Online :

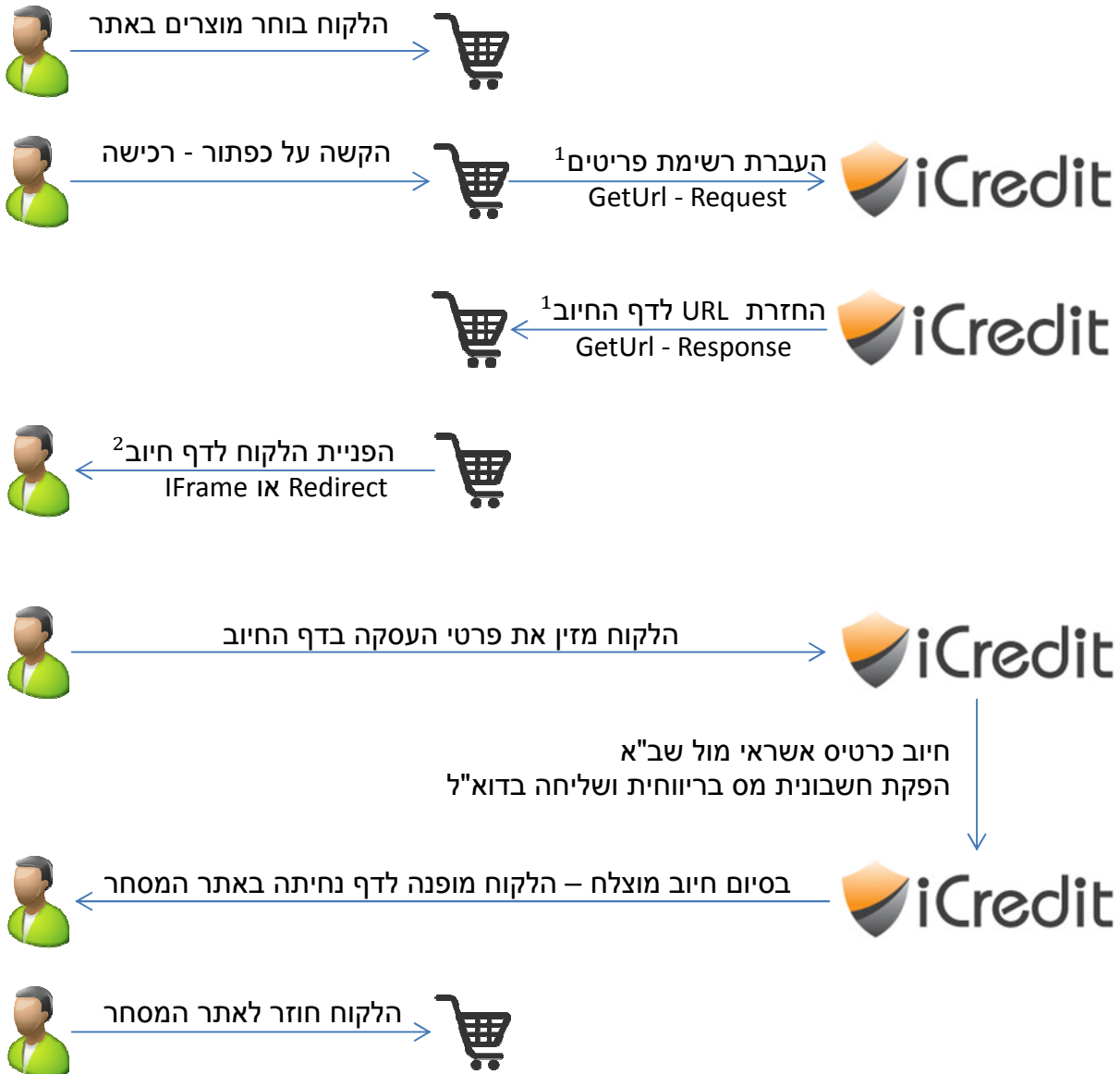
א. יש להגדיר את ה-api_token של ריווחית בתפריט ניהול משתמש ← עריכת פרטי משתמש ← הגדרות מתקדמות / ריווחית ← מזהה API. כמו כן יש להגדיר במסך זה את שאר השדות.

ב. יש להגדיר עבור כל פריט בדף המכירה את קוד הפריט בריווחית. (ניתן לבצע סינכרון אוטומטי של כל הפריטים מריווחית).

מימוש פניה לדף חיוב :

לאחר השלמת הגדרות המערכת ניתן להטמיע קוד באתר למימוש הפניות לדף החיוב.

להלן התהליך המלא :



השלבים שבהם יש לבצע מימוש באתר בית העסק הם :

1. העברת מזהה קבוצת מכירה ורשימת פריטים לחיוב באמצעות המתודה GetURL.
2. קבלת URL לדף החיוב והפניית הלקוח ל-URL זה באמצעות Redirect או IFrame.

: GetURL

תיאור המתודה : קבלת דף חיוב עבור רשימת פריטים.

REST	
POST	Method
https://testicredit.rivhit.co.il/API/PaymentPageRequest.svc/GetUrl	URL Test
https://icredit.rivhit.co.il/API/PaymentPageRequest.svc/GetUrl	URL Prod
XML, JSON	Request
XML, JSON	Response

Request Parameters			
שדה חובה	string	מזהה קבוצה פרטי	GroupPrivateToken
שדה חובה		מערך פריטים במבנה הבא :	Items[]
	int	קוד פריט בריווחית	Id
	string	מספר קטלוגי בריווחית	CatalogNumber
שדה חובה	decimal	כמות לרכישה	Quantity
שדה חובה	decimal	מחיר ליחידה	UnitPrice
	string	תיאור הפריט	Description
	string	כתובת להפניית הלקוח	RedirectURL
הנחיות ל-IPN בהמשך	string	כתובת IPN Listener	IPNURL
	bool	האם המוצרים פטורים ממע"מ	ExemptVAT
	int	מספר תשלומים מקסימלי	MaxPayments
	int	עסקת קרדיט מעל מספר תשלומים	CreditFromPayment
	int	אסמכתא	Reference
	string	מספר הזמנה	Order
	string	כתובת דוא"ל לשליחת החשבונית	EmailAddress
	string	כתובת דוא"ל נסתר	EmailBcc
	string	שם פרטי של הלקוח	CustomerLastName
	string	שם משפחה של הלקוח	CustomerFirstName
	string	כתובת	Address
	int	תא דואר	POB
	string	עיר מגורים	City
	int	מיקוד	Zipcode
	string	מספר טלפון	PhoneNumber
	string	מספר טלפון נוסף	PhoneNumber2
	string	מספר פקס	FaxNumber
	int	מספר ת"ז	IdNumber
	int	מספר עוסקה מורשה	VatNumber
	int	מס' כרטיס לקוח בריווחית	CustomerId
	Int	מס' סוכן בריווחית	AgentId
	string	הערות לחשבונית	Comments
	Int	קוד פרוייקט בריווחית	ProjectId
	bool	הסתר את רשימת המוצרים בדף	HideItemList
אנגלית = en	string	שפת מסמך החשבונית	DocumentLanguage
	string	האם לייצר טוקן בעסקה מוצלחת	CreateToken
	decimal	סכום הנחה	Discount
	string	שדה בהתאמה אישית 1	Custom1
	string	שדה בהתאמה אישית 2	Custom2
	string	שדה בהתאמה אישית 3	Custom3
	string	שדה בהתאמה אישית 4	Custom4
	string	שדה בהתאמה אישית 5	Custom5
	string	שדה בהתאמה אישית 6	Custom6

	string	שדה בהתאמה אישית 7	Custom7
	string	שדה בהתאמה אישית 8	Custom8
	string	שדה בהתאמה אישית 9	Custom9

Request Example - XML

```
<PaymentPageURLRequestDTO xmlns="https://icredit.rivhit.co.il/api">
  <GroupPrivateToken>1627aea5-8e0a-4371-9022-9b504344e724</GroupPrivateToken>
  <Items>
    <Item>
      <Id>1</Id>
      <Quantity>3</Quantity>
      <UnitPrice>55.9</UnitPrice>
      <Description>שרשרת פנינים</ Description >
    </Item>
  </Items>
  <RedirectURL>http://www.rivhit.co.il</RedirectURL>
  <ExemptVAT>true</ExemptVAT>
  <MaxPayments>12</MaxPayments>
  <CustomerFirstName>ישראל</CustomerFirstName>
  <CustomerLastName>ישראלי</CustomerLastName>
</PaymentPageURLRequestDTO>
```

Request Example - JSON

```
{
  "GroupPrivateToken": "1627aea5-8e0a-4371-9022-9b504344e724",
  "Items": [{
    "Id": 1,
    "Quantity": 3,
    "UnitPrice": 55.9,
    "Description": "שרשרת פנינים"
  }],
  "RedirectURL": "http://www.rivhit.co.il",
  "ExemptVAT": true,
  "MaxPayments": 12,
  "CustomerFirstName": "ישראל",
  "CustomerLastName": "ישראלי"
}
```

			Response Values
ראה הערה 1	string	נתיב דף החיוב להפניית הלקוח	URL

Response Example - XML

```
<PaymentPageURLResponseDTO xmlns="https://icredit.rivhit.co.il/api">
  <URL>http://testicredit.rivhit.co.il/payment/PaymentItems.aspx?GroupId=f54179da-c34f-4fc7-964d-d4275571113a&Token=8c2cd593-8a66-4e77-ac86-267b96a94473</URL>
</PaymentPageURLResponseDTO>
```

Response Example - JSON

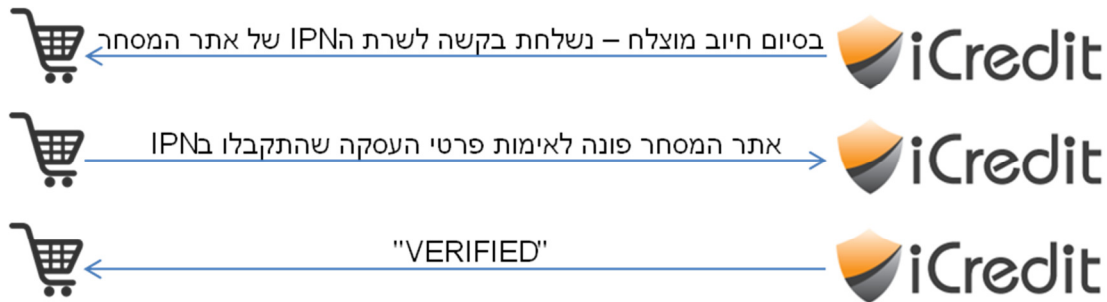
```
{
  "URL": "http://testicredit.rivhit.co.il/payment/PaymentItems.aspx?GroupId=f54179da-c34f-4fc7-964d-d4275571113a&Token=5662145b-535e-4773-a0ea-3c23814cf0db"
}
```

הערות :

- במידה וישנה שגיאה תוחזר מחרוזת ריקה (URL=""), במידה והכל תקין יוחזר URL לדף החיוב אליו יש להפנות את הלקוח להמשך הרכישה.
- ברשימת הפריטים יש להעביר בנוסף מזהה לפריט באחד השדות Id או CatalogNumber.
- שדות אשר מוגדרים כלא מוצגים יופיעו בחשבונית המופקת בריווחית.
- שדות שלא הועברו כפרמטרים ומוגדרים לתצוגה בדף החיוב, יוצגו למילוי ע"י הלקוח בדף החיוב.

מימוש Instant Payment Notification (IPN) :

ניתן לשלב מנגנון IPN אשר מיידע את בית העסק בסיום חיוב מוצלח באופן הבא :



יש להעביר את כתובת שירות ה-IPN Listener בבקשת GetURL (לעיל) בפרמטר IPNURL.

ה-IPN Listener הנמצא בשרת אתר המסחר יקבל בקשת HTTP POST עם פרמטרים המכילים את נתוני העסקה :

פרמטרים	
SaleId	מזהה המכירה
GroupPrivateToken	מזהה קבוצה פרטי
NumberOfItems	מספר הפריטים במכירה
ItemIdN	קוד פריט בריווחית לפריט ה-N
ItemCatalogNumberN	מספר קטלוגי בריווחית לפריט ה-N
ItemQuantityN	כמות לרכישה לפריט ה-N
ItemUnitPriceN	מחיר ליחידה לפריט ה-N
ItemDescriptionN	תיאור הפריט לפריט ה-N
Reference	אסמכתא
Order	מספר הזמנה
CustomerLastName	שם פרטי של הלקוח
CustomerFirstName	שם משפחה של הלקוח
Address	כתובת
POB	תא דואר
City	עיר מגורים
Zipcode	מיקוד
PhoneNumber	מספר טלפון
PhoneNumber2	מספר טלפון נוסף
FaxNumber	מספר פקס
IdNumber	מספר ת"ז
VatNumber	מספר עוסק מורשה
Custom1	שדה בהתאמה אישית 1
Custom2	שדה בהתאמה אישית 2
Custom3	שדה בהתאמה אישית 3
Custom4	שדה בהתאמה אישית 4
Custom5	שדה בהתאמה אישית 5
Custom6	שדה בהתאמה אישית 6
Custom7	שדה בהתאמה אישית 7
Custom8	שדה בהתאמה אישית 8
Custom9	שדה בהתאמה אישית 9
CustomerId	מס' כרטיס לקוח בריווחית
DocumentURL	כתובת המסמך שהופק
DocumentNum	מספר המסמך שהופק

	סוג המסמך שהופק	DocumentType
	סכום העסקה	TransactionAmount
	מספר אישור מחברת האשראי	TransactionAuthNum
	שם מותג כרטיס האשראי	TransactionCardName
	מספר הכרטיס	TransactionCardNum
	קוד תנאי אשראי	TransactionCreditTerms
	שם תנאי אשראי	TransactionCreditTermsDescription
בעסקת תשלומים	סכום תשלום ראשון	TransactionFirstAmount
	סכום תשלומים קבועים נוספים	TransactionNonFirstAmount
	מספר תשלומים קבועים	TransactionNumOfPayment
	ציון לכרטיס חו"ל	TransactionForeignSign
	ספק סולק	TransactionSolekSapak
	סטטוס עסקה	TransactionStatus
	שם המסוף בשב"א	TransactionTerminalName
	מספר המסוף בשב"א	TransactionTerminalNum
	סכום העסקה (במט"ח)	TransactionTransAmount
	מועד העסקה	TransactionDateTime
	קוד סוג העסקה	TransactionType
	שם סוג העסקה	TransactionTypeDescription
1 – ש"ח 2 – דולר אמריקאי אחר – קוד ISO מט"ח	מטבע העסקה	TransactionCurrency
	שם מטבע העסקה	TransactionCurrencyDescription
	מזהה הטוקן שנוצר	TransactionToken

המימוש שיש לבצע באתר בית העסק:

1. בעת הפניה למתודה GetURL יש להעביר את נתיב ה-IPN Listener בפרמטר IPNURL.
2. יש לממש את ה-Listener על מנת שיבצע אימות לעסקה שהתקבלה. ראה נספח א' – דוגמא למימוש IPN Listener.
3. יש לבצע אימות לפרטי העסקה שהתקבלו מול שרת ה-iCredit באמצעות המתודה Verify.

: Verify

תיאור המתודה : אימות פרטי העסקה שהתקבלו ב-IPN.

REST	
POST	Method
https://testicredit.rivhit.co.il/API/PaymentPageRequest.svc/Verify	URL Test
https://icredit.rivhit.co.il/API/PaymentPageRequest.svc/Verify	URL Prod
XML, JSON	Request
XML, JSON	Response

Request Parameters			
שדה חובה	string	מזהה קבוצה פרטי	GroupPrivateToken
שדה חובה	string	מזהה המכירה	SaleId
שדה חובה	decimal	סכום העסקה	TotalAmount

Request Example - XML

```
<VerifyRequestDTO xmlns="https://icredit.rivhit.co.il/api">
  <GroupPrivateToken>1627aea5-8e0a-4371-9022-9b504344e724</GroupPrivateToken>
  <SaleId>ec52aaf8-743f-4e89-a74b-6646e11610f6</SaleId>
  <TotalAmount>150.90</TotalAmount>
</VerifyRequestDTO>
```

Request Example - JSON

```
{
  "GroupPrivateToken":"1627aea5-8e0a-4371-9022-9b504344e724",
  "SaleId":"ec52aaf8-743f-4e89-a74b-6646e11610f6",
  "TotalAmount":150.90
}
```

Response Values			
או "VERIFIED" "NOTVERIFIED"	string	תוצאת אימות העסקה	Status

Response Example - XML

```
<VerifyResponseDTO xmlns="https://icredit.rivhit.co.il/api">
  <Status> VERIFIED </Status>
</VerifyResponseDTO>
```

Response Example - JSON

```
{
  "Status":"VERIFIED"
}
```

נספח א' – דוגמא למימוש IPN Listener באמצעות דף .aspx

```

<%@ Page Language="C#" %>

<%@ Import Namespace="System" %>
<%@ Import Namespace="System.Collections.Generic" %>
<%@ Import Namespace="System.Linq" %>
<%@ Import Namespace="System.Web" %>
<%@ Import Namespace="System.Web.UI" %>
<%@ Import Namespace="System.Web.UI.WebControls" %>
<%@ Import Namespace="System.IO" %>
<%@ Import Namespace="System.Net" %>
<%@ Import Namespace="System.Runtime.Serialization.Json" %>
<%@ Import Namespace="System.Text" %>
<%@ Import Namespace="System.Runtime.Serialization" %>
<script language="C#" option="Explicit" runat="server">
    [DataContract]
    public class VerifyRequestDTO
    {
        [DataMember]
        public Guid GroupPrivateToken { get; set; }
        [DataMember]
        public Guid SaleId { get; set; }
        [DataMember]
        public decimal TotalAmount { get; set; }
    }

    [DataContract]
    public class VerifyResponseDTO
    {
        [DataMember]
        public string Status { get; set; }
    }

    protected void Page_Load(object sender, EventArgs e)
    {
        if (!string.IsNullOrEmpty(Request.Params["SaleId"]))
        {
            string status = VerifyResponse(Guid.Parse(Request.Params["SaleId"]),
            Guid.Parse(Request.Params["GroupPrivateToken"]),
            decimal.Parse(Request.Params["TransactionAmount"]));

            if (status == "VERIFIED")
            {
                // TODO: Check that GroupId belongs to me
                // TODO: Check that transaction has not been previously processed
                // TODO: Check that amounts are correct

                // Example: writing all parameters to file on C:\temp\{SALEID}.txt
                using (FileStream fs = File.Create(@"C:\temp\" + Request.Params["SaleId"] +
                ".txt"))
                {
                    using (StreamWriter sw = new StreamWriter(fs))
                    {
                        sw.WriteLine("{0}", status);
                        foreach (string key in Request.Params.AllKeys)
                        {
                            sw.WriteLine("{0}: {1}", key, Request.Params[key]);
                        }
                    }
                }
            }
        }
    }
}
    
```

```

private string VerifyResponse(Guid saleId, Guid privateGroupToken, decimal amount)
{
    VerifyRequestDTO parameters = new VerifyRequestDTO()
    {
        GroupPrivateToken = privateGroupToken,
        SaleId = saleId,
        TotalAmount = amount,
    };
    try
    {
        string VERIFY_TEST =
@"https://testicredit.rivhit.co.il/API/PaymentPageRequest.svc/Verify";
        string VERIFY_PROD =
@"https://icredit.rivhit.co.il/API/PaymentPageRequest.svc/Verify";
        HttpRequest req = WebRequest.Create(new Uri(VERIFY_TEST)) as HttpWebRequest;
        req.Method = "POST";
        req.ContentType = "application/json";
        req.Accept = "application/json";
        MemoryStream stream1 = new MemoryStream();
        DataContractJsonSerializer ser = new
DataContractJsonSerializer(typeof(VerifyRequestDTO));
        ser.WriteObject(stream1, parameters);
        stream1.Position = 0;
        StreamReader sr = new StreamReader(stream1);

        byte[] requestData = UTF8Encoding.UTF8.GetBytes(sr.ReadToEnd());
        req.ContentLength = requestData.Length;

        // Send the request:
        using (Stream post = req.GetRequestStream())
        {
            post.Write(requestData, 0, requestData.Length);
        }

        // Pick up and deserialize the response:
        DataContractJsonSerializer responseSerializer = new
DataContractJsonSerializer(typeof(VerifyResponseDTO));
        VerifyResponseDTO response;
        try
        {
            using (HttpWebResponse resp = req.GetResponse() as HttpWebResponse)
            {
                if (resp.StatusCode == HttpStatusCode.OK)
                {
                    response =
(VerifyResponseDTO)responseSerializer.ReadObject(resp.GetResponseStream());
                    return response.Status;
                }
                else
                {
                    // The data is empty
                    response = null;
                    return "EXCEPTION";
                }
            }
        }
        catch (WebException wex)
        {
            using (HttpWebResponse errorResponse = (HttpWebResponse)wex.Response)
            {
                response =
(VerifyResponseDTO)responseSerializer.ReadObject(errorResponse.GetResponseStream());
            }
            return "EXCEPTION";
        }
        catch (Exception)
        {
            return "EXCEPTION";
        }
    }
}
</script>

```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head id="Head1" runat="server">
  <title></title>
</head>
<body>
  <form id="form1" runat="server">
    <div>
      Load this first to check the syntax of your page
    </div>
  </form>
</body>
</html>
```