

Code Examples for sending SMS

Get started instantly. You can copy & paste the code examples below for use with our SMS API Gateway.

JavaPHPC#/.NetC++/CLICVB.NETASPRUBY

Code Example for sending SMS - Java

```
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
import java.io.UnsupportedEncodingException;
import java.net.URL;
import java.net.URLConnection;
import java.net.URLEncoder;

// Simple send SMS programm
public class SendSMS {
    public static String sendSMS(String username, String password, String to, String
message, String originator) {
        String url;
        StringBuilder inBuffer = new StringBuilder();
        try {
            url = "http://api.textmarketer.co.uk/gateway/" +
                "?username=" + username + "&password=" + password + "&option=xml" +
                "&to=" + to + "&message=" + URLEncoder.encode(message, "UTF-8") +
                "&orig=" + URLEncoder.encode(originator, "UTF-8");
        } catch (UnsupportedEncodingException e) {
            return null;
        }
        try {
            URL tmUrl = new URL(url);
            URLConnection tmConnection = tmUrl.openConnection();
            tmConnection.setDoInput(true);
            BufferedReader in = new BufferedReader(new
InputStreamReader(tmConnection.getInputStream()));
            String inputLine;
            while ((inputLine = in.readLine()) != null)
                inBuffer.append(inputLine);
            in.close();
        } catch (IOException e) {
            return null;
        }
        return inBuffer.toString();
    }
    public static void main(String[] args) {
        // Example of use
        String response = sendSMS("myUsername", "myPassword", "4477777777", "My test
message", "TextMessage");
        System.out.println(response);
    }
}
```

Code Example for sending SMS - PHP

```
<?php
    // Simple SMS send function
    function sendSMS($username, $password, $to, $message, $originator) {
        $URL =
'http://api.textmarketer.co.uk/gateway/'. "?username=$username&password=$password&opt
ion=xml";
        $URL .=
"&to=$to&message=".urlencode($message).'&orig='.urlencode($originator);
        $fp = fopen($URL, 'r');
        return fread($fp, 1024);
    }
    // Example of use
    $response = sendSMS('myUsername', 'myPassword', '4477777777', 'My test message',
'TextMessage');
    echo $response;?>
```

Code Example for sending SMS - C#/NET

```
using System;
using System.IO;
using System.Net;
using System.Text;
using System.Web;

namespace SendSMS {
    class Program {
        public static string SendSMS(string username, string password, string to,
string message, string originator) {
            StringBuilder sb = new StringBuilder();
            byte[] buf = new byte[1024];
            string url = "http://api.textmarketer.co.uk/gateway/" +
                "?username=" + username + "&password=" + password + "&option=xml" +
                "&to=" + to + "&message=" + HttpUtility.UrlEncode(message) +
                "&orig=" + HttpUtility.UrlEncode(originator);
            HttpWebRequest request = (HttpWebRequest)WebRequest.Create(url);
            HttpWebResponse response = (HttpWebResponse)request.GetResponse();
            Stream resStream = response.GetResponseStream();
            string tempString = null;
            int count = 0;
            do {
                count = resStream.Read(buf, 0, buf.Length);
                if (count != 0) {
                    tempString = Encoding.ASCII.GetString(buf, 0, count);
                    sb.Append(tempString);
                }
            } while (count > 0);
            return sb.ToString();
        }
        static void Main(string[] args) {
            string respXML = SendSMS("myUsername", "myPassword", "4477777777", "My
test message", "TextMessage");
            Console.WriteLine(respXML);
        }
    }
}
```

Code Example for sending SMS - C++

```
#include <iostream>
#include <string>
#include <System.Dll>
#include <System.Web.Dll>

using namespace std;
using namespace System;
using namespace System::Web;
using namespace System::Net;
using namespace System::IO;
using namespace System::Runtime::InteropServices;

ref class SMSSender
{
private:
    static String^ Username = "myUsername";
    static String^ Password = "myPassword";

public:
    SMSSender()
    {}
    String^ SendSMS(String^ To, String^ Message, String^ From)
    {
        Message = HttpUtility::UrlEncode(Message);
        From = HttpUtility::UrlEncode(From);
        String^ URL =
"http://api.textmarketer.co.uk/gateway/?&option=xml&username=" + Username +
"&password=" + Password + "&message=" + Message + "&orig=" + From + "&to=" + To;
        WebRequest^ Handle = WebRequest::Create(URL);
        WebResponse^ HTTPResponse = Handle->GetResponse();
        StreamReader^ Stream = gcnew
StreamReader(HTTPResponse->GetResponseStream());
        String^ Response = Stream->ReadToEnd()->Trim();
        HTTPResponse->Close();
        return Response;
    }
};

int main() {
    SMSSender^ test = gcnew SMSSender();
    String^ resp = test->SendSMS("4477777777", "My test message",
"TextMessage");
    Console::WriteLine(resp);
    return 0;
}
```

Code Example for sending SMS - C

```
/* * Send SMS C/C++ example need curl lib download from http://curl.haxx.se/ */

#include <stdio.h>
#include <tchar.h>
```

```

#include <string.h>
#include <curl/curl.h>
#define URLSIZE 256

struct MemoryStruct {
    char *memory;
    size_t size;
};

/* Converts a hex character to its integer value */

char from_hex(char ch) {
    return isdigit(ch) ? ch - '0' : tolower(ch) - 'a' + 10;
}

/* Converts an integer value to its hex character*/

char to_hex(char code) {
    static char hex[] = "0123456789abcdef";
    return hex[code & 15];
}

/* Returns a url-encoded version of str */

char *url_encode(char *str) {
    char *pstr = str, *buf = (char *)malloc(strlen(str) * 3 + 1), *pbuf = buf;
    while (*pstr) {
        if (isalnum(*pstr) || *pstr == '-' || *pstr == '_' || *pstr == '.'
|| *pstr == '~')
            *pbuf++ = *pstr;
        else if (*pstr == ' ')
            *pbuf++ = '+';
        else
            *pbuf++ = '%', *pbuf++ = to_hex(*pstr >> 4), *pbuf++ =
to_hex(*pstr & 15);
        pstr++;
    }
    *pbuf = '\0';
    return buf;
}

/* CURL Callback write function */

static size_t WriteMemoryCallback(void *contents, size_t size, size_t nmemb, void
*userp) {
    size_t realsize = size * nmemb;
    struct MemoryStruct *mem = (struct MemoryStruct *)userp;
    mem->memory = (char *)realloc(mem->memory, mem->size + realsize + 1);
    if (mem->memory == NULL) {
        /* out of memory! */
        printf("not enough memory (realloc returned NULL)\n");
        exit(EXIT_FAILURE);
    }
    memcpy(&(mem->memory[mem->size]), contents, realsize);
    mem->size += realsize;
    mem->memory[mem->size] = 0;
    return realsize;
}

```

```

/* Send SMS */

char * sendSMS(const char *username, const char *password, const char *to, char
*message, char *originator) {
    static char url[URLSIZE] =
"http://api.textmarketer.co.uk/gateway/?option=xml";
    char *encoded;
    CURL *curl;
    CURLcode res;
    struct MemoryStruct chunk;
    chunk.memory = (char *)malloc(1); /* will be grown as needed by the realloc
above */
    chunk.size = 0; /* no data at this point */
    curl = curl_easy_init();
    if(curl) {
        strcat_s(url, URLSIZE, "&username=");
        strcat_s(url, URLSIZE, username);
        strcat_s(url, URLSIZE, "&password=");
        strcat_s(url, URLSIZE, password);
        strcat_s(url, URLSIZE, "&to=");
        strcat_s(url, URLSIZE, to);
        strcat_s(url, URLSIZE, "&message=");
        encoded = url_encode(message);
        strcat_s(url, URLSIZE, encoded);
        free(encoded);
        encoded = url_encode(originator);
        strcat_s(url, URLSIZE, "&orig=");
        strcat_s(url, URLSIZE, encoded);
        free(encoded);
        curl_easy_setopt(curl, CURLOPT_URL, url);
        /* send all data to this function */
        curl_easy_setopt(curl, CURLOPT_WRITEFUNCTION, WriteMemoryCallback);
        /* we pass our 'chunk' struct to the callback function */
        curl_easy_setopt(curl, CURLOPT_WRITEDATA, (void *)&chunk);
        if((res = curl_easy_perform(curl)) != CURLE_OK) {
            return NULL;
        }
        curl_easy_cleanup(curl);
    }
    return chunk.memory;
}

int main(void) {
    char *response;
    response = sendSMS("myuser", "mypass", "4477777777", "test test test",
"me");
    if(response != NULL) {
        printf(response);
        free(response);
    }
}

```

```
    getChar();
    return 0;
}
```

Code Example for sending SMS - VB.NET

```
Imports System.Web
```

```
Module Module1
```

```
    Public Function SendSMS(ByVal username As String, ByVal password As String,
ByVal toPhone As String,
```

```
ByVal message As String, ByVal originator As String)
```

```
        Dim webClient As New System.Net.WebClient
```

```
        Dim url As String = "http://api.textmarketer.co.uk/gateway/?username=" &
        username & "&password=" & password & "&option=xml" &
```

```
        "&to=" & toPhone & "&message=" &
```

```
System.Web.HttpUtility.UrlEncode(message) &
```

```
        "&orig=" & System.Web.HttpUtility.UrlEncode(originator)
```

```
        Dim result As String = webClient.DownloadString(url)
```

```
        SendSMS = result
```

```
    End Function
```

```
    Sub Main()
```

```
        Dim result As String = SendSMS("myUsername", "myPassword", "4477777777", "My
test message", "TextMessage")
```

```
        Console.WriteLine(result)
```

```
        Console.ReadKey()
```

```
    End Sub
```

```
End Module
```

Code Example for sending SMS - ASP

```
<%@language=JScript%><%
```

```
    username = "myUsername";
```

```
    password = "myPassword";
```

```
    number = "4477777777";
```

```
    message = Server.URLEncode("My test message");
```

```
    originator = Server.URLEncode("TextMessage");
```

```
    url = "http://www.textmarketer.biz/gateway/?&option=xml" +
        "&username=" + username + "&password=" + password +
        "&number=" + number + "&message=" + message +
        "&orig=" + originator;
```

```
    var objSrvHTTP;
```

```
    objSrvHTTP = Server.CreateObject("Msxml2.ServerXMLHTTP");
```

```
    objSrvHTTP.open(url, false);
```

```
    objSrvHTTP.send();
```

```
    Response.ContentType = "text/xml";
```

```
    xmlResp = objSrvHTTP.responseXML.xml;
```

```
    Response.Write(xmlResp);%>
```

Code Example for sending SMS - RUBY

```
require 'net/http'
require 'uri'

def send_sms(username, password, to, message, originator)
  requested_url = 'http://api.textmarketer.co.uk/gateway/?option=xml' +
    "&username=" + username + "&password=" + password +
    "&to=" + to + "&message=" + URI.escape(message) +
    "&orig=" + URI.escape(originator)
  url = URI.parse(requested_url)
  full_path = (url.query.blank?) ? url.path : "#{url.path}?#{url.query}"
  the_request = Net::HTTP::Get.new(full_path)
  the_response = Net::HTTP.start(url.host, url.port) { |http|
    http.request(the_request)
  }
  raise "Response was not 200, response was #{the_response.code}" if
the_response.code != "200"
  return the_response.bodyend
resp = send_sms('myUsername', 'myPassword', '4477777777', 'My test message',
'TextMessage')

puts(resp)
```

Our example code is an illustration of how you might integrate with our systems and is not certified for production environments. You are responsible for testing and QA.

- [Next: API Responses](#)
- [Home: Text Marketer Developers Documentation](#)