



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Genesys SMS Aggregation Service HTTP API User Guide

Sample Java code to interact with Genesys API

12/12/2025

Sample Java code to interact with Genesys API

```
import java.io.BufferedReader;
import java.io.DataOutputStream;
import java.io.IOException;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.net.URL;
import java.net.URLEncoder;

import java.util.UUID;

public class Demo {

    public static void main(String[] args) throws IOException {

        String username      = "USERNAME";                // The
        username provided to the client by Genesys
        String password      = "PASSWORD";
        // The password provided to the client by Genesys
        String message       = "Hey";                    // The
        message to be sent.
        String messageId     = String.valueOf(UUID.randomUUID()); // A
        unique message id.
        // In this example we use UUID to generate an ID, but any value can be used
        as long as it's unique
        String phoneNumber    = "12345";                // The
        phone number to which the message is to be sent.
        String shortCode     = "99222";                // The
        shortcode on this message.
        String carrier        = "99901";                //
        The carrier ID can be found in the reach list.

        String genesysURL    = "https://smsc-api.genesyscloud.com/httpapi/
        receiver";

        // QueryString to be sent to Genesys API
        String messageBody= URLEncoder.encode( "login", "UTF-8" ) + "=" +
        URLEncoder.encode( username, "UTF-8" )
        + "&" + URLEncoder.encode( "password", "UTF-8" ) + "=" + URLEncoder.encode(
        password, "UTF-8" )
        + "&" + URLEncoder.encode( "phone_number", "UTF-8" ) + "=" +
        URLEncoder.encode( phoneNumber, "UTF-8" )
        + "&" + URLEncoder.encode( "shortcode", "UTF-8" ) + "=" + URLEncoder.encode(
        shortCode, "UTF-8" )
        + "&" + URLEncoder.encode( "carrier", "UTF-8" ) + "=" + URLEncoder.encode(
        carrier, "UTF-8" )
        + "&" + URLEncoder.encode( "message", "UTF-8" ) + "=" + URLEncoder.encode(
        message, "UTF-8" )
        + "&" + URLEncoder.encode( "message_id", "UTF-8" ) + "=" + URLEncoder.encode(
        messageId, "UTF-8" );

        URL url = new URL(genesysURL);
        HttpURLConnection conn = (HttpURLConnection)url.openConnection();
        conn.setDoOutput(true);
        conn.setRequestProperty("Content-Type", "application/x-www-form-urlencoded");
```

```
conn.setRequestProperty("Accept", "text/plain");

DataOutputStream out = new DataOutputStream(conn.getOutputStream());
out.writeBytes(messageBody);
out.flush();
out.close();

// Response Code. If not in 2XX region then the request failed
if(conn.getResponseCode() < 200 || conn.getResponseCode() > 299){
    System.out.print("The request failed. Please try again in some
time.");
    System.exit(0);
} else {
    // Printing the Response
    BufferedReader reader= new BufferedReader(new
InputStreamReader(conn.getInputStream()));
    String response= reader.readLine();

    // Getting the Result Code from the Response
    Integer resultCode= Integer.parseInt(response.split("=")[1]);

    switch(resultCode){
        case 0:
            System.out.println("Success: The message has been accepted
for processing.");
            break;
        case 1:
            System.out.println("Failure: Either one(or more) require
parameters are missing. ");
            break;
        case 2:
            System.out.println("Failure: The credentials provided are
either wrong or do not exist.");
            break;
        case 3:
            System.out.println("Failure: The carrier ID is not an Integer
value or it does not exist in the reach list.");
            break;
        case 4:
            System.out.println("Failure: The shortcode is not an
Integer.");
            break;
        case 5:
            System.out.println("Failure: The shortcode is not registered
for the client.");
            break;
        case 6:
            System.out.println("Failure: The client is trying to send
more messages in the allotted time than they are allowed.");
            break;
        case 7:
            System.out.println("Failure: This message is a duplicate.");
            break;
        case 8:
            System.out.println("Failure: The message is larger than
allowed for the client.");
            break;
        case 98:
            System.out.println("Failure: Internal Server Error. Please
try again");
            break;
        case 99:
```

Sample Java code to interact with Genesys API

```
        System.out.println("Failure: There was some error in  
processing request. Please try again");  
    }  
}  
}
```