**GYMBUDDY-CODE SNIPPETS**

**JAVA**

import java.io.\*;  
import java.net.MalformedURLException;  
import java.util.Base64;  
import java.util.HashMap;  
import java.util.Map;  
import java.net.HttpURLConnection;  
import java.net.URL;  
import java.nio.charset.StandardCharsets;  
public class HttpClientTest {  
public static void main(String[] args) throws IOException {  
  
// NOTE: you must manually set API\_KEY below using information retrieved from your IBM Cloud account. (<https://dataplatform.cloud.ibm.com/docs/content/wsj/analyze-data/ml-authentication.html?context=wx>)  
  
String API\_KEY = "<your API key>";  
  
HttpURLConnection tokenConnection = null;  
HttpURLConnection scoringConnection = null;  
BufferedReader tokenBuffer = null;  
BufferedReader scoringBuffer = null;  
try {  
// Getting IAM token  
URL tokenUrl = new URL("<https://iam.cloud.ibm.com/identity/token?grant_type=urn:ibm:params:oauth:grant-type:apikey&apikey=>" + API\_KEY);  
tokenConnection = (HttpURLConnection) tokenUrl.openConnection();  
tokenConnection.setDoInput(true);  
tokenConnection.setDoOutput(true);  
tokenConnection.setRequestMethod("POST");  
tokenConnection.setRequestProperty("Content-Type", "application/x-www-form-urlencoded");  
tokenConnection.setRequestProperty("Accept", "application/json");  
  
if (tokenConnection.getResponseCode() == 200) { // Successful response  
tokenBuffer = new BufferedReader(new InputStreamReader(tokenConnection.getInputStream()));  
} else { // Error response  
tokenBuffer = new BufferedReader(new InputStreamReader(tokenConnection.getErrorStream()));  
}  
  
            String line;  
StringBuffer jsonString = new StringBuffer();  
            while ((line = tokenBuffer.readLine()) != null) {  
                jsonString.append(line);  
            }  
            System.out.println("Token response body:\n" + jsonString);  
// Scoring request  
URL scoringUrl = new URL("<https://us-south.ml.cloud.ibm.com/ml/v4/deployments/59c8e33b-95c8-45e3-9e3c-47258c545faf/ai_service_stream?version=2021-05-01>");  
String iam\_token = "Bearer " + jsonString.toString().split(":")[1].split("\"")[1];  
scoringConnection = (HttpURLConnection) scoringUrl.openConnection();  
scoringConnection.setDoInput(true);  
scoringConnection.setDoOutput(true);  
scoringConnection.setRequestMethod("POST");  
scoringConnection.setRequestProperty("Accept", "application/json");  
scoringConnection.setRequestProperty("Authorization", iam\_token);  
scoringConnection.setRequestProperty("Content-Type", "application/json; charset=UTF-8");  
OutputStreamWriter writer = new OutputStreamWriter(scoringConnection.getOutputStream(), "UTF-8");  
  
// NOTE:  manually define and pass the array(s) of values to be scored in the next line  
String payload = {"messages":[{"content":"","role":""}]};  
  
writer.write(payload);  
writer.close();  
  
if (scoringConnection.getResponseCode() == 200) { // Successful response  
scoringBuffer = new BufferedReader(new InputStreamReader(scoringConnection.getInputStream()));  
} else { // Error response  
scoringBuffer = new BufferedReader(new InputStreamReader(scoringConnection.getErrorStream()));  
}  
  
            String lineScoring;  
StringBuffer jsonStringScoring = new StringBuffer();  
            while ((lineScoring = scoringBuffer.readLine()) != null) {  
                jsonStringScoring.append(lineScoring);  
            }  
            System.out.println("Scoring response body:\n" + jsonStringScoring);  
} catch (IOException e) {  
System.out.println("The request was not valid.");  
System.out.println(e.getMessage());  
}  
finally {  
if (tokenConnection != null) {  
tokenConnection.disconnect();  
}  
if (tokenBuffer != null) {  
tokenBuffer.close();  
}  
if (scoringConnection != null) {  
scoringConnection.disconnect();  
}  
if (scoringBuffer != null) {  
scoringBuffer.close();  
}  
}  
}  
}

**PYTHON**

import requests  
  
# NOTE: you must manually set API\_KEY below using information retrieved from your IBM Cloud account (<https://dataplatform.cloud.ibm.com/docs/content/wsj/analyze-data/ml-authentication.html?context=wx>)  
API\_KEY = "<your API key>"  
token\_response = [requests.post](http://requests.post/" \t "_blank)('<https://iam.cloud.ibm.com/identity/token>', data={"apikey": API\_KEY, "grant\_type": 'urn:ibm:params:oauth:grant-type:apikey'})  
mltoken = token\_response.json()["access\_token"]  
  
header = {'Content-Type': 'application/json', 'Authorization': 'Bearer ' + mltoken}  
  
# NOTE:  manually define and pass the array(s) of values to be scored in the next line  
payload\_scoring = {"messages":[{"content":"","role":""}]}  
  
response\_scoring = [requests.post](http://requests.post/)('<https://us-south.ml.cloud.ibm.com/ml/v4/deployments/59c8e33b-95c8-45e3-9e3c-47258c545faf/ai_service_stream?version=2021-05-01>', json=payload\_scoring,  
 headers={'Authorization': 'Bearer ' + mltoken})  
  
print("Scoring response")  
try:  
    print(response\_scoring.json())  
except ValueError:  
    print(response\_scoring.text)  
except Exception as e:  
    print(f"An unexpected error occurred: {e}")