

IRAS API SERVICES INTERFACE SPECIFICATIONS

TAX TREATY RATE

Last updated on: 1 Apr 2019

Version No.: 1.1

Table of Contents

1.	Introduction	3
2.	Registration at API Portal	3
3.	API Services	4
3.1	General Information	4
3.1.1	Production Usage	4
3.1.2	Common Interface Information	4
3.1.3	Common Response Payload	5
3.2	Tax Treaty Calculator	6
3.2.1	Request Payload	6
3.2.2	Response Payload	8
3.3	Sandbox Testing	9
3.3.1	Tax Treaty Service	9
4.	Sample Code (C#)	12

1. Introduction

The Inland Revenue Authority of Singapore (IRAS) provides application programming interface (API) services to allow application developers to submit and retrieve tax related matters using HTTP requests. Most of the APIs will be in the form of a JSON web service which reduces client/server coupling and thus enabling easier integration between IRAS' service with external developers.

There will be a variety of services available in due time. While some services require a simple GET, others may be secured and require credentials that can be passed via HTTP header parameters which are as follows :

X-IBM-Client-Id	String containing the client ID of the application invoking IRAS API. This value will be provided to the application vendor by IRAS. E.g. a1234b5c-1234-abcd-efgh-a1234b5cdef
X-IBM-Client-Secret	String containing the client secret of the application invoking IRAS API. This value will be provided to the application vendor by IRAS. E.g. a12345bC67e8fG9a12345bC67e8fG9a12345bC67e8fG9

This document serves to help developers consume the API services provided by IRAS.

2. Registration at API Portal

Application developers are required to register for an account at <https://apisandbox.iras.gov.sg/> to subscribe to IRAS API services for Sandbox Testing and at <https://apiservices.iras.gov.sg/> for Production. A computer-generated email will be sent to the subscriber's email account for account activation of the API Portal.

3. API Services

The table below shows the list of Tax Treaty API services currently available in IRAS.

S/No	Name of API Services	Description	URL
1	Tax Treaty Calculator	This API can be used to retrieve the applicable DTA Taxrate for FORM IR37.	For Sandbox Testing: https://apisandbox.iras.gov.sg/iras/sb/TaxTreatyRate/retrieve
			For Production Usage: https://apiservices.iras.gov.sg/iras/prod/TaxTreatyRate/retrieve

3.1 General Information

3.1.1 Production Usage

Approval is **NOT** required to use this service.

The following parameters must be populated in the HTTP header:

X-IBM-Client-Id	String containing the client ID of the application invoking IRAS API. This value will be provided to the application vendor by IRAS.
X-IBM-Client-Secret	String containing the client secret of the application invoking IRAS API. This value will be provided to the application vendor by IRAS.

3.1.2 Common Interface Information

- JSON is case sensitive by specifications.
- All date strings are to be represented in compliance to the [ISO-8601](#) standard.
- All string fields are subject to validation of the following acceptable characters that is allowed (in red)
 - [a-zA-Z0-9'@#() \-./&+ _] (**Note:** whitespace is included)
- All properties follow the camel-case convention.
- Unless stated as optional, all JSON object properties must be specified.
- Unless otherwise specified, all JSON services are invoked using HTTP verb POST.
- All input data format are as specified like the following:



3.1.3 Common Response Payload

All response payloads share the following common fields:

data	Object	The data property will be populated differently based on the API that is being invoked.
returnCode	Integer	10 : Success - The request was successfully processed. 20 : Warning - The request was successfully processed. However, there are non-fatal issues. Please refer to the “info” object for diagnostic information. 30 : Failure – The request was not processed. Refer to “info” object for error information.
info	Object	This complex object holds any diagnostic information that will allow developers to debug their failed requests.
info.message	String	Diagnostic message in the event of warning or error.
Info.messageCode	Integer	Integer code signifying the type of error or warning. 850300 : Request object is null – The incoming JSON request is null. 850301 : Arguments error – There is an error with one of the arguments provided. 850302 : Generic error – There is an exception within the service. 850303 : Service is inactive. 850304 : Service is not authorized for usage based on the provided credentials. 850305 : Invalid test user – The input fields provided are not valid for sandbox testing.
info.fieldInfoList	Array	An array for FieldInfo objects.
info.fieldInfoList.field	String	Name of the field that resulted in a warning / error.
Info.fieldInfoList.message	String	Diagnostic message provided to aid consumer’s developers.
Info.fieldInfoList.recordID	Number	The unique identification number of the affected record. <i>Note: This is only present if the field resulting in an error belongs to a TaxTreatyrDtl object.</i>

3.2 Tax Treaty Calculator

3.2.1 Request Payload

TaxTreatyDtl	Object	This complex object holds the Requests for tax treaty calculations.																				
recordID	Number (4)	The running number assigned should start from 1 and be unique.																				
countryCode	Number (3)	The country code of the non resident. Pls refer to the attached document for the country code.  Country_code.pdf																				
natureOfPayment	Number (3)	Nature of payments code as per the tax treaties. Pls refer to the attached document for the payment code.  Payment_mode.pdf																				
payeeType	String (1)	Profile of the payee, must be set to the following. <table border="1" data-bbox="821 1061 1445 1464"> <tr><td>1</td><td>INDIVIDUAL</td></tr> <tr><td>2</td><td>PARTNERSHIP</td></tr> <tr><td>3</td><td>CORPORATE COMPANY</td></tr> <tr><td>4</td><td>GOVERNMENT BODY/GOVT RELATED</td></tr> <tr><td>5</td><td>UNIVERSITY</td></tr> <tr><td>6</td><td>CLUB</td></tr> <tr><td>7</td><td>ASSOCIATION</td></tr> <tr><td>8</td><td>SOCIETY</td></tr> <tr><td>9</td><td>OTHERS</td></tr> <tr><td>10</td><td>FINANCIAL INSTITUTION</td></tr> </table>	1	INDIVIDUAL	2	PARTNERSHIP	3	CORPORATE COMPANY	4	GOVERNMENT BODY/GOVT RELATED	5	UNIVERSITY	6	CLUB	7	ASSOCIATION	8	SOCIETY	9	OTHERS	10	FINANCIAL INSTITUTION
1	INDIVIDUAL																					
2	PARTNERSHIP																					
3	CORPORATE COMPANY																					
4	GOVERNMENT BODY/GOVT RELATED																					
5	UNIVERSITY																					
6	CLUB																					
7	ASSOCIATION																					
8	SOCIETY																					
9	OTHERS																					
10	FINANCIAL INSTITUTION																					
serviceRenderedPeriodFrom	String (8)	The period the service was rendered from. The format should be YYYYMMDD. It should not be future date.																				
serviceRenderedPeriodTo	String (8)	The period the service was rendered to. The format should be YYYYMMDD.																				
isAmtDerivedFromSgp	String (1)	Indicator Y or N if the amount is derived from any trade, business, profession or vocation carried on through the non-resident's fixed place of operation (e.g. branch) in Singapore.																				

Sample JSON request payload

```
{
  "TaxTreatyDtl": [
    {
      "recordID": 1,
      "countryCode": 104,
      "natureOfPayment": 11,
      "payeeType": "4",
      "serviceRenderedPeriodFrom": "20170117",
      "serviceRenderedPeriodTo": "20171010",
      "isAmtDerivedFromSgp": "N"
    },
    {
      "recordID": 2,
      "countryCode": 104,
      "natureOfPayment": 12,
      "payeeType": "4",
      "serviceRenderedPeriodFrom": "20170117",
      "serviceRenderedPeriodTo": "20171010",
      "isAmtDerivedFromSgp": "N"
    }
  ]
}
```

3.2.2 Response Payload

data	Object	The object payload containing the reference code of successful submission.
data.recordID	Number	The corresponding recordID from the submission.
data.taxRate	String	Applicable DTA Tax Rate
data.submissionDueDate	String	Submission due date in YYYYMMDD
data.remarks	String	<ol style="list-style-type: none"> 1. "Please submit the Certificate of Residence by 31-03-YYYY." Or 2. "Please submit your Certificate of Residence within 3 months from the date of submission of Form IR37" Or 3. "Further Information is required to determine your tax rate" Or 4. "Invalid selection. Please try again." Or 5. "DTA tax rate is not applicable" Or 6. Special highlight messages to take note of.
returnCode	As per Section 3.1.3	
info		
info.message		
info.messageCode		
info.fieldInfoList		
info.fieldInfoList.field		
info.fieldInfoList.message		
info.fieldInfoList.recordID		

Sample success JSON response payload

```

{
  "returnCode": 10,
  "info": {
    "fieldInfoList": []
  },
  "data": {
    {
      "TaxTreatyDtl": [
        {
          "recordID": 1,
          "taxRate": "8",
          "submissionDueDate": "20180331",
          "remarks": ""
        }
      ],
    }
  }
}

```



```

        "recordID": 2,
        "taxRate": "0",
        "submissionDueDate": "",
        "remarks": "Further Information is required to determine tax rate."
    }
]
}

```

Sample error JSON response payload

```

{
  "returnCode": "30",
  "info": {
    "message": "Arguments Error",
    "messageCode": "850301",
    "fieldInfoList": [
      {
        "field": "countryCode",
        "message": "Value is not valid",
        "RecordID": "1"
      }
    ]
  },
  "data": {
    "TaxTreatyDtl": []
  }
}

```

3.3 Sandbox Testing

As explained in [section 2](#), developers can first create an account in the Sandbox environment to make API calls to our Sandbox URL. This allows the developers to mimic the characteristics of our production environment and create a simulated response from our API.

3.3.1 Tax Treaty Service

Input	Expected Output
<pre> { "TaxTreatyDtl": [{ "recordID": "1", "countryCode": "302", "natureOfPayment": "2", "payeeType": "1", }], "serviceRenderedPeriodFrom": "20170101", "serviceRenderedPeriodTo": "20180101", "isAmtDerivedFromSgp": "N" } </pre>	<pre> { "returnCode": "10", "info": { "fieldInfoList": [] }, "data": { "TaxTreatyDtl": [{ "recordID": "1", "remarks": "Period must be within a calendar year, please try again" }] } } </pre>

IRAS API Services Interface Specifications

<pre> }, { "recordID": "2", "countryCode": "36", "natureOfPayment": "26", "payeeType": "10", "serviceRenderedPeriodFrom": "20170515", "serviceRenderedPeriodTo": "20170630", "isAmtDerivedFromSgp": "N" }, { "recordID": "3", "countryCode": "101", "natureOfPayment": "1", "payeeType": "4", "serviceRenderedPeriodFrom": "20060815", "serviceRenderedPeriodTo": "20061231", "isAmtDerivedFromSgp": "N" }, { "recordID": "4", "countryCode": "102", "natureOfPayment": "26", "payeeType": "10", "serviceRenderedPeriodFrom": "20170401", "serviceRenderedPeriodTo": "20170616", "isAmtDerivedFromSgp": "N" }, { "recordID": "5", "countryCode": "331", "natureOfPayment": "2", "payeeType": "5", "serviceRenderedPeriodFrom": "20160101", "serviceRenderedPeriodTo": "20160624", </pre>	<pre> "SubmissionDueDate": "20180331", "TaxRate": "5.00", "recordID": "2", "remarks": "Please submit the Certificate of Residence by 3/31/2018" }, { "TaxRate": "10.00", "recordID": "3", "remarks": "Please submit your Certificate of Residence within 3 months from the date of submission of Form IR37" }, { "recordID": "4", "remarks": "Further Information is required to determine your tax rate" }, { "recordID": "5", "remarks": "DTA tax rate is not applicable" }, { "SubmissionDueDate": "20190331", "TaxRate": "10.00", "recordID": "6", "remarks": "Please submit the Certificate of Residence by 3/31/2019" }, { "SubmissionDueDate": "20190331", "TaxRate": "7.00", "recordID": "7", "remarks": "Please submit the Certificate of Residence by 3/31/2019" }, { "SubmissionDueDate": "20190331", "TaxRate": "10.00", "recordID": "8", "remarks": "Please submit the Certificate of Residence by 3/31/2019" }, { "SubmissionDueDate": "20190331", </pre>
--	---

<pre> "isAmtDerivedFromSgp": "Y" }, { "recordID": "6", "countryCode": "312", "natureOfPayment": "11", "payeeType": "3", "serviceRenderedPeriodFrom": "20180101", "serviceRenderedPeriodTo": "20180331", "isAmtDerivedFromSgp": "N" }, { "recordID": "7", "countryCode": "336", "natureOfPayment": "26", "payeeType": "10", "serviceRenderedPeriodFrom": "20180101", "serviceRenderedPeriodTo": "20180331", "isAmtDerivedFromSgp": "N" }, { "recordID": "8", "countryCode": "336", "natureOfPayment": "26", "payeeType": "3", "serviceRenderedPeriodFrom": "20180101", "serviceRenderedPeriodTo": "20180331", "isAmtDerivedFromSgp": "N" }, { "recordID": "9", "countryCode": "336", "natureOfPayment": "26", "payeeType": "4", "serviceRenderedPeriodFrom": </pre>	<pre> "TaxRate": "0.00", "recordID": "9", "remarks": "Please submit the Certificate of Residence by 3/31/2019" }] } } </pre>
--	---

<pre>"20180101", "serviceRenderedPeriodTo": "20180331", "isAmtDerivedFromSgp": "N" }]</pre>	
<pre>{ "TaxTreatyDtl": [{ "recordID": "1", "countryCode": "3021", "natureOfPayment": "2", "payeeType": "1", "serviceRenderedPeriodFrom": "20170101", "serviceRenderedPeriodTo": "20180101", "isAmtDerivedFromSgp": "N" }] }</pre>	<pre>{ "returnCode":"30", "info":{ "message":"Arguments Error", "messageCode":"850301", "fieldInfoList":[{ "field":"countryCode", "message":"Value is not valid", "RecordID":"1" }] }, "data":{ "TaxTreatyDtl":[] } }</pre>

4. Sample Code (C#)

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

// jsonData – contains data from Section 3.1.1 of this document
public static void callWebAPI(string jsonData, string url)
{
    try
    {
        var httpWebRequest = (HttpWebRequest)WebRequest.Create(url);
        httpWebRequest.ContentType = "application/json";
        httpWebRequest.Method = "POST";

        //Step 1: Enter the Client-Id given by IRAS
        httpWebRequest.Headers["Client-Id"] = "{YOUR_CLIENT_ID}";
        //Step 2: Enter the Client-Secret given by IRAS
        httpWebRequest.Headers["Client-Secret"] = "{YOUR_CLIENT_SECRET}";

        // Step 3: Call API using POST
        using (var streamWriter = new StreamWriter(httpWebRequest.GetRequestStream()))
        {
            streamWriter.Write(jsonData);
            streamWriter.Flush();
            streamWriter.Close();
        }
    }
}
```

```
// Step 3a: Output response
var httpResponse = (HttpWebResponse)httpWebRequest.GetResponse();
using (var streamReader = new StreamReader(httpResponse.GetResponseStream()))
{
    var result = streamReader.ReadToEnd();
    //print the received response
    Console.WriteLine(result);
}
}
catch (WebException e)
{
    if (!string.IsNullOrEmpty(e.Message))
    {
        // Step 3b: Print general errors
        Console.WriteLine("Exception - ");
        Console.WriteLine(e.Message);
    }

    if (e.Response != null)
    {
        // Step 3c: Print Output response exception
        Stream receiveStream = e.Response.GetResponseStream();
        StreamReader readStream = new StreamReader(receiveStream, Encoding.UTF8);
        // print the error received from Server
        Console.WriteLine("Response error received - ");
        Console.WriteLine(readStream.ReadToEnd());
    }
}
}
```

The information provided is intended for better general understanding and is not intended to comprehensively address all possible issues that may arise. The contents are correct as at 13 Oct 2017 and are provided on an "as is" basis without warranties of any kind. IRAS shall not be liable for any damages, expenses, costs or loss of any kind however caused as a result of, or in connection with your use of this document.

While every effort has been made to ensure that the above information is consistent with existing policies and practice, should there be any changes, IRAS reserves the right to vary our position accordingly.