



# Oracle Financial Services Know Your Customer

Utilities Guide

Release 8.0.7.0.0

## DOCUMENT CONTROL

Version Number	Revision Date	Changes Done
1.0	July 2018	Created the document.
Created by: Uttam Pandanda	Reviewed by:	Approved by:

# TABLE OF CONTENTS

<b>1</b>	<b>ABOUT THIS GUIDE.....</b>	<b>4</b>
<b>2</b>	<b>PREREQUISITES.....</b>	<b>4</b>
<b>3</b>	<b>OVERVIEW .....</b>	<b>5</b>
3.1	Table to JSON Utility.....	5
3.2	JSON to Table Utility.....	6
<b>4</b>	<b>TABLES AND TABLE ELEMENTS.....</b>	<b>6</b>
4.1	Table to JSON.....	6
4.1.1	fcc_tpg_table_json_mapping Table .....	7
4.1.2	fcc_tpg_table_json_query Table.....	9
4.1.3	fcc_tpg_table_json_query_params Table .....	9
4.2	JSON to Table.....	10
4.2.1	fcc_ob_json_table_config Table .....	11
4.2.2	fcc_ob_json_table_map Table.....	12
<b>5</b>	<b>VIEWING THE JSON .....</b>	<b>14</b>
<b>6</b>	<b>MAPPING IDs AND REST URLs FOR UTILITIES .....</b>	<b>16</b>

---

## 1 About this Guide

The Utilities Guide details the table and table elements that form a part of the Table to JSON and JSON to Table utilities' web services which can be deployed on any web server.

## 2 Prerequisites

JNDI data source configured for the ATOMIC schema. After you deploy the web service, replace the `##JNDI_NAME##` placeholder with the data source name in the following files:

- `connection.properties` file under `WEB-INF/classes` folder.
- `Web.xml` file under `WEB-INF` folder.

---

## 3 Overview

### 3.1 Table to JSON Utility

The Table to JSON utility creates a JSON according to the configurations specified by a user. The user can define the JSON structure and the JSON fields to be constructed along with an indication of where the data of each of the fields are available. Once the field values are provided, the utility creates the JSON according to the specified configurations and picks the values against each field as defined during configuration.

---

**NOTE:** The utility can support only two levels of array in the JSON. To see an example, refer [Viewing the JSON](#). Example are also provided in the [KYC Service Guide](#).

---

The Table to JSON utility is used in KYC to:

- prepare the inputs of the third party verifications to be provided as a part of onboarding service. The JSONs can vary based on data such as country, data sources, and customer type. Ready-to-use mappings have been created to use this JSON.
- prepare the input JSON for the onboarding service. This JSON is then provided as a response for KYC Onboarding.
- create risk assessments.
- prepare the input for Customer Screening (CS).
- prepare the input of the internal watch list.

The Table to JSON utility is used in Enterprise Case Management (ECM) to:

- Generate the JSON response for the Common Gateway service.
- Generate the JSON response which will be made available in the KYC setup.

---

**NOTE:** The Table to JSON utility must ONLY be used in conjunction with the Financial Crime and Compliance Management (FCCM) application packs. Usage with any external applications is not allowed.

---

---

## 3.2 JSON to Table Utility

The JSON to Table utility allows a user to define where the data of each of the fields are available. These values are captured in the database in a tabular format. This utility captures the information of individual fields of JSON as per the configuration.

---

**NOTE:** The utility can support only two levels of array in the JSON. To see an example, refer [Viewing the JSON](#). Example are also provided in the [KYC Service Guide](#).

---

The JSON to Table utility is used in KYC to:

- Save the JSON response.
- Display the JSON response on your User Interface.

---

**NOTE:** The Table to JSON utility must ONLY be used in conjunction with the Financial Crime and Compliance Management (FCCM) application packs. Usage with any external applications is not allowed.

---

## 4 Tables and Table Elements

As a part of the general configuration, there are some tables which need to be updated to get the JSONs constructed by the utility. Currently these configurations must be defined by the user in the tables. These tables and their respective columns are explained in the sections that follow.

- [Table to JSON](#)
- [JSON to Table](#)

### 4.1 Table to JSON

The following tables must be configured so that the Table to JSON utility can construct the JSON:

- [fcc\\_tpg\\_table\\_json\\_mapping Table](#)
- [fcc\\_tpg\\_table\\_json\\_query Table](#)

- [fcc\\_tpg\\_table\\_json\\_query\\_params Table](#)

The table elements are described below:

#### 4.1.1 fcc\_tpg\_table\_json\_mapping Table

This table allows you to configure the JSON that is constructed by the utility, and is the first table that needs to be configured. Use this table to:

- configure the structure of the JSON
- indicate whether the value of the JSON is from a column in the table or a user input
- configure the value of the JSON field
- configure the data type of the field/array of the JSON

To view the table values, run the following query:

```
select * from fcc_tpg_table_json_mapping where mapping_id ='parent_mapping_id value';
```

The values are described in the table below:

Mapping ID	Mapping IDs are used in queries to construct a JSON. Each table has a mapping ID assigned to it and is case sensitive. For example, TEST_ARRAY_ENH1.  <b>NOTE:</b> Mapping IDs are unique for every JSON which is constructed.
Key	A unique key is assigned for every mapping ID which is generated.
Parent Field	Parent of the key that is generated in the JSON. Parent field of the first layer of keys must be kept as 'parent' in this field. For the second layer of keys, the parent field is the fields in the first layer of keys.
Field	This field captures the exact label of the field name which is to be generated in the JSON.
Field Type	This field captures the field type. The types can be ELEMENT, OBJECT or ARRAY.

Field Data Type	This field captures the field data type of the JSON. The default is STRING. Other acceptable values are STRING, NUMBER, BOOLEAN.
Array Data Type	This field captures the array data type. The default is OBJECT (JSON object). Other acceptable values are OBJECT, STRING, NUMBER, BOOLEAN.
Data Source	<p>The data source has to be either TABLE or REQUEST.</p> <p>The data source is TABLE if the value comes from a table or expression. The data source is REQUEST if the value comes from the request parameters. For information on the request parameters, see the <a href="#">Know Your Customer API Services Guide</a>.</p>
Table Name	This field captures the name of each table which contains the JSON field values. This has to be only in the schema where the web services are deployed. If the value of the field is not available in a table, then this will be blank. If the value is a static value, then this value can be provided in the Expression column. The table name must have the physical table name as defined in the database.
Column Name	This field captures the physical name of the column corresponding to the table which contains the field values of the JSON.
Expression	<p>SQL expression that generates the field value. It overrides the Table Name and Column Name fields. It must be as is from the SQL query that generates the value.</p> <p>For example, in the case of SELECT 'SampleValue' FROM DUAL, the expression field would be 'SampleValue'. In the case of SELECT UPPER(TABLE_NAME.COLUMN_NAME) from CUST, the expression field would be UPPER(TABLE_NAME.COLUMN_NAME).</p>
Key Source	Provide the key source value as REQUEST to make a particular JSON field value dynamic. This means that the key is substituted by the value of the request parameters in the URL.
Related Mapping ID	<p>Provides the mapping ID for the child array element which is inside the parent array. For each child array value in Field, there is a corresponding Related Mapping ID.</p> <p>Default is null.</p>
Primary Key	Provides the primary key for the parent array element. For each parent array value in Parent Field, there is a corresponding Mapping ID.

	Default is null.
--	------------------

**4.1.2 fcc\_tpg\_table\_json\_query Table**

This table contains the FROM and WHERE clauses of the queries used to construct the JSON. For each mapping, this table has to be updated to reflect the FROM and WHERE clauses.

To view the table values, run the following query:

```
select * from fcc_tpg_table_json_query where mapping_id = 'parent_mapping_id value'
```

The values are described in the table below :

Mapping ID	This field captures the mapping IDs which are case sensitive. This mapping ID has to be the same as that of the previous table mapping ID.
Parent Field	Parent of the key that is generated in the JSON. Parent field of the first layer of keys must be kept as 'parent' in this field.
From Clause	FROM clause of the query that provides the data for the JSON. It should be defined at the parent level and every time we need to define a mapping ID.
Where Clause	WHERE clause of the query that provides the data for the JSON. There should only be one record being returned for OBJECT and ELEMENT types. ARRAYS may have one or more records, and each of those records will become a separate ARRAY element. It should be defined at the parent level and every time we need to pass request parameters for the URL and replace the question mark.
Foreign Key	Provides the foreign key for the child array element. For each child array value in Field, there is a corresponding Related Mapping ID. You must define the foreign key for that child array.  Default is null.

**4.1.3 fcc\_tpg\_table\_json\_query\_params Table**

This table contains the request parameters used by the queries in the fcc\_tpg\_table\_json\_query table.

To view the table values, run the following query:

---

```
select * from fcc_tpg_table_json_query_param where mapping_id = 'parent_mapping_id value';
```

The values are described in the table below :

Mapping ID	Mapping IDs are used in queries to construct a JSON. Each table has a mapping ID assigned to it and is case sensitive. For example, TEST_ARRAY_ENH1.  <b>NOTE:</b> Mapping IDs are unique for every JSON which is constructed.
Parent Field	Parent of the key that is generated in the JSON. Parent field of the first layer of keys must be kept as 'parent' in this field.
Parameter Order	Order of the parameter within the WHERE clause of the previous table. The '?' values in the WHERE clause will be replaced by these values in this order. For example, if the WHERE clause is 'where ob_cust_seq_id = ? and request_id = ?', the first '?' will be replaced by the value of the parameter with order 1.
Parameter Name	Name of the query parameter that will be passed through the URL
Parameter Source	Source of the parameter. REQUEST is the only supported value now.

After the configurations are done, enter the URL for the JSON in the POSTMAN client based on the following format:

```
http://domain:port/ TabletoJSONService/createtabletojson?mappingId= " "
```

For more information, see [Viewing the JSON](#).

## 4.2 JSON to Table

The following tables are used in this utility:

- [fcc\\_ob\\_json\\_table\\_config Table](#)
- [fcc\\_ob\\_json\\_table\\_map Table](#)

---

The table elements are described below.

#### 4.2.1 fcc\_ob\_json\_table\_config Table

This table defines the structure of the input JSON which needs to be persisted in the KYC tables.

To view the table values, run the following query:

```
select * from fcc_tpg_json_table_config where parent_mapping_id='parent_mapping_id value' order by order_used;
```

---

**NOTE:** The parent\_mapping\_id value can be one of the following:

---

- SCORING\_RESPONSE
- CS\_WLS\_RESPONSE
- INT\_WLS\_RESPONSE
- TRULIOO\_ENTITY\_RESPONSE
- TRULIOO\_RESPONSE

PARENT_MAPPING_ID	Parent mapping ID of the JSON.
JSON_ELEMENT	This field captures the JSON ID value for each JSON element. This value should match the element names in the JSON.
PARENT_JSON_ELEMENT	This field captures the name of the parent JSON element. If there is no parent element, '(null)' is displayed by default.
DATA_TYPE	This field captures the logical data type of the JSON element. The default is STRING. Other Data types are String array, Array, or JSON.
ORDER_USED	This field provides the order of the JSON elements which need to be captured among the complete JSON structure. The order of the first element to be captured is 1. The order of each subsequent element increases by 1 whenever a particular JSON element is captured.

---

## 4.2.2 fcc\_ob\_json\_table\_map Table

This table contains the mappings between the configured JSON elements and the corresponding table elements.

To view the table values, run the following query:

```
select * from fcc_tpg_json_table_map where parent_mapping_id='< parent_mapping_id value>' order by order_used;
```

---

**NOTE:** The parent\_mapping\_id value can be one of the following:

---

- SCORING\_RESPONSE
- CS\_WLS\_RESPONSE
- INT\_WLS\_RESPONSE
- IDV\_ENTITY\_RESPONSE
- IDV\_RESPONSE

The values are described in the table below:

MAPPING_ID	Child mapping ID of the parent mapping ID. For example, if the corresponding parent mapping ID is mapped to three different tables, then three unique mapping IDs will be used.
PARENT_MAPPING_ID	Mapping ID which groups all the configuration and the corresponding mappings for the JSON. This must be the same in both config and map tables.
JSON_ELEMENT	The JSON elements configured in the config table. Only captures JSON elements of type String or String array. If a JSON element of type JSON or array is defined in the config table, then that JSON element must not be displayed in the map table.
TARGET_TABLE	Table in to which data is persisted.
TARGET_COLUMN	Column in to which data is persisted.

---

EXPRESSION	Values can be the service name, system date, or another query expression. In case you do not know the actual expression value, provide the value NOTNULL.
ORDER_USED	The same order used for a particular JSON element in the config table. Only captures JSON elements of type string.

---

## 5 Viewing the JSON

To view the JSON for the Table to JSON utility:

1. Open the POSTMAN client.
2. In the Builder tab, select the POST method.
3. Enter a URL in the method field. An example is given below:  
`http://domain:port/TabletoJSONService/createtabletojson?mappingId= ' '`
4. Click **Send**.

The JSON appears in the Request Body editor.

To view the JSON for the JSON to Table utility:

1. Open the POSTMAN client.
2. In the Builder tab, select the POST method.
3. Enter a URL in the method field. An example is given below:  
`http://domain:port/JSONtoTablePersistenceUtility/persistJSON?mappingID=' '&requestid=' '&OB\_CUST\_SEQ\_ID=' '`
4. Click **Send**.

The JSON appears in the Request Body editor.

Below is a sample JSON:

The screenshot displays a REST client interface with the following details:

- URL:** `http://ofss2221076:95`
- Method:** `POST`
- Target URL:** `http://ofss2221076:9999/TabletoJSONService/createtabletojson?mappingId=TEST_UTTAM`
- Status:** `200 OK`, **Time:** `326 ms`, **Size:** `436 B`
- Response Body (JSON):**

```
1 {
2   "c": false,
3   "d": "b",
4   "obj": [
5     {
6       "requestID": 1031,
7       "custID": 1051
8     },
9     {
10      "requestID": 1031,
11      "custID": 1052
12    },
13    {
14      "requestID": 1031,
15      "custID": 1053
16    },
17    {
18      "requestID": 1032,
19      "custID": 1054
20    },
21    {
22      "requestID": 20000,
23      "custID": 1041
24    },
25    {
26      "requestID": 20000,
27      "custID": 1042
28    },
29    {
30      "requestID": 20000,
31      "custID": 1043
32    },
33    {
34      "requestID": 20000,
35      "custID": 1044
36    }
37  ],
38 }
```

If there are any errors, they are displayed in the Body editor after you send the request.

## 6 Mapping IDs and REST URLs for Utilities

The following table provides information on the mapping IDs, REST URLs and the applicable utilities. For information on the JSON preparation, see the *Input preparation/ Hitting the individual services/Output capture* section in the [Know Your Customer Service Guide](#).

Mapping ID	Description	URL	Utility
CMMN_GATEWAY_INPUT	This mapping define the structure of json to be formed for common gateway it hold the definition of each data element and its value to be picked from the table for input creation	http://#deployed_server#:#port#/TabletoJSONService/createtabletojson?mappingId=CMMN_GATEWAY_INPUT&requestId={OB_REQUEST_ID}&customerCounter={OB_CUST_COUNT}	Table to JSON
CREATE_ASSESSMENT	This mapping define the structure of json to be formed for create assesment it hold the definition of each data element and its value to be picked from the table for input creation	http://#deployed_server#:#port#/TabletoJSONService/createtabletojson?mappingId=CREATE_ASSESSMENT&requestId={OB_REQUEST_ID}	Table to JSON
CS_INPUT_IND	This mapping id define the structure of json to be formed for customer screening where the customer type is individual it hold the definition of each data element and it's value to be picked from the table	http://#deployed_server#:#port#/CommonGatewayService/createtabletojson?mappingId=CS_INPUT_IND&customerId={CUSTOMER_ID}&requestId={REQUEST_ID}	Table to JSON
CS_INPUT_NONIND	This mapping id define the structure of json to be formed for customer screening where the customer type is entity it hold the definition of each data element and it's value to be picked from the table	http://#deployed_server#:#port#/CommonGatewayService/createtabletojson?mappingId=CS_INPUT_NONIND&customerId={CUSTOMER_ID}&requestId={REQUEST_ID}	Table to JSON

INTRL_WLS_INPUT	This mapping define the structure of json to be formed for internal watchlist it hold the definition of each data element and its value to be picked from the table for input creation	<a href="http://#deployed_server#:#port#/CommonGatewayService/createtabletojson?mappingId=INTRL_WLS_INPUT&amp;requestId={REQUEST_ID}">http://#deployed_server#:#port#/CommonGatewayService/createtabletojson?mappingId=INTRL_WLS_INPUT&amp;requestId={REQUEST_ID}</a>	Table to JSON
OB_RESP_1	This mapping define the structure of json to be formed as a final response after the onboarding customer data has processed through all the services	<a href="http://#deployed_server#:#port#/TabletoJSONService/createtabletojson?mappingId=OB_RESP_1&amp;requestId=1003">http://#deployed_server#:#port#/TabletoJSONService/createtabletojson?mappingId=OB_RESP_1&amp;requestId=1003</a>	Table to JSON
SCORING_INPUT	This mapping define the structure of json to be formed for scoring service it hold the definition of each data element and its value to be picked from the table for input creation	<a href="http://#deployed_server#:#port#/TabletoJSONService/createtabletojson?mappingId=SCORING_INPUT&amp;requestId={OB_REQUEST_ID}">http://#deployed_server#:#port#/TabletoJSONService/createtabletojson?mappingId=SCORING_INPUT&amp;requestId={OB_REQUEST_ID}</a>	Table to JSON
SCORING_RESPONSE	This mapping ID defines where the values of the scoring output in JSON format has to be captured in the KYC OB tables in the form of table.column mappings for each data element of the JSON	<a href="http://#deployed_server#:#port#/JSONToTablePersistenceUtility/persistJSON?mappingID=SCORING_RESPONSE&amp;requestid={OB_REQUEST_ID}">http://#deployed_server#:#port#/JSONToTablePersistenceUtility/persistJSON?mappingID=SCORING_RESPONSE&amp;requestid={OB_REQUEST_ID}</a>	Table to JSON
CS_WLS_RESPONSE	This mapping ID defines where the values of the customer screening output in the form of JSON has to be captured in the KYC OB tables in the form of table.column	<a href="http://#deployed_server#:#port#/JSONToTablePersistenceUtility/persistJSON?mappingID=CS_WLS_RESPONSE&amp;requestId={OB_REQUEST_ID}&amp;customerId={CUST_SEQ_ID}">http://#deployed_server#:#port#/JSONToTablePersistenceUtility/persistJSON?mappingID=CS_WLS_RESPONSE&amp;requestId={OB_REQUEST_ID}&amp;customerId={CUST_SEQ_ID}</a>	JSON to table

	mappings for each data element of the JSON		
INT_WLS_RESPONSE	This mapping ID defines where the values of the internal watchlist output in the form of JSON has to be captured in the KYCOB tables in the form of table.column mappings for each data element of the JSON	http://#deployed_server#:#port#/JSONToTablePersistenceUtility/persistJSON?mappingID=INT_WLS_RESPONSE&requestid={OB_REQUEST_ID}&customerId={CUST_SEQ_ID}	JSON to table
CREATE_ASSMNT	This mapping ID defines which Table.column will be updated based on the JSON formed by Create assesment	http://#deployed_server#:#port#/JSONToTablePersistenceUtility/persistJSON?mappingID=CREATE_ASSMNT	JSON to table
KYCOB_CASE_RESPONSE	the action taken on cases gives a JSON as response this mapping ID defines in which table.column the value will be updated	http://#deployed_server#:#port#/JSONToTablePersistenceUtility/persistJSON?mappingID=KYCOB_CASE_RESPONSE	JSON to table

---

## SEND US YOUR COMMENTS

Oracle welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, indicate the title and part number of the documentation along with the chapter/section/page number (if available) and contact the Oracle Support Services.

---

**NOTE:** Before sending us your comments, you might like to ensure that you have the latest version of the document wherein any of your concerns have already been addressed. You can access My Oracle Support site which has all the revised/recently released documents.

---



Oracle Financial Services Utilities Guide

Oracle Corporation

World Headquarters

500 Oracle Parkway

Redwood Shores, CA 94065

U.S.A.

Worldwide Inquiries:

Phone: +1.650.506.7000

Fax: +1.650.506.7200

[www.oracle.com/us/industries/financial-services/](http://www.oracle.com/us/industries/financial-services/)

Copyright © 2018 Oracle Financial Services Software Limited. All rights reserved.

No part of this work may be reproduced, stored in a retrieval system, adopted or transmitted in any form or by any means, electronic, mechanical, photographic, graphic, optic recording or otherwise, translated in any language or computer language, without the prior written permission of Oracle Financial Services Software Limited.

Due care has been taken to make this Oracle Financial Services Know Your Customer 8.6.0.0 Onboarding API Elements guide and accompanying software package as accurate as possible. However, Oracle Financial Services Software Limited makes no representation or warranties with respect to the contents hereof and shall not be responsible for any loss or damage caused to the user by the direct or indirect use of this Oracle Financial Services Know Your Customer Utilities Guide and the accompanying Software System. Furthermore, Oracle Financial Services Software Limited reserves the right to alter, modify or otherwise change in any manner the content hereof, without obligation of Oracle Financial Services Software Limited to notify any person of such revision or changes.

All company and product names are trademarks of the respective companies with which they are associated.