



JSON Object Manager Package

Readme

Version 1.0

04/24/2020

Table of Contents

1. Introduction	3
1.1 Overview	3
1.2 Use Cases	3
2. Requirements & Prerequisites	4
2.1 System Requirements	4
2.2 Prerequisites	4
3. Getting Started	5
3.1 Quick Start	5
3.1.1 Setup	5
3.1.2 Configuration	5
4. Support & FAQs	7
4.1 Support	7
4.2 FAQs	7
Appendix A: Record of Changes	8
Appendix B: References	9

1. Introduction

This document contains all essential information for the user to make full use of this A2019 Package. It includes a description of the functions and capabilities and step-by-step procedures for setup & configuration of the Package.

1.1 Overview

The JSON Object Manager package is used to allow developers to parse JSON data structures which typically come as a response to REST Web services. Accompanying this package is a sample bot: JSONObjectMangerDemoBot which demonstrates using the package's actions against a sample JSON.

The included package simplifies the process of parsing responses from REST web services and JSON formatted configuration files.

The JSON Object Manager Package can be used for:

1.2 Use cases

The JSON Object Manger package is especially useful when parsing the response from a REST API or a bot configuration file. It allows for bot builders to quickly and easily extract values from a properly formatted JSON document/string and even iterate through array objects within the JSON to extract specific values.

2. Requirements & Prerequisites

2.1 System Requirements

[Enterprise A2019 \(Cloud deployed\) and Community Edition device requirements.](#)

Review the machine hardware specifications, operating system versions, and browser types supported by Automation Anywhere Enterprise for creating and running bots and command packages as an Enterprise A2019 (Cloud deployed) or Community Edition user on your local machine.

2.2 Prerequisites

No specific prerequisites required besides making sure you are using a properly formatted JSON.

3. Getting Started

3.1 Quick Start

3.1.1 Setup

1. Install the package from Bot Store into your Control Room
2. Enable the package named “JSON Object Manager” and set as default.
3. Navigate to Bot Store/JSON Object Manager - Automation Anywhere to run and examine the JSONObjectManagerDemoBot.

Note: The bot-specific logic starts on line 37 within the try block.

3.1.2 Configuration and Use

To begin using the JSON Object Manager bot, you must first enable the package

1. Navigate to the **Bots > Packages** page.
2. From the list of installed packages, select the **JSON Object Manger** by clicking the package name.
3. Next to the Versions dropdown, click the button that says **Enable and Set as Default**.
4. Once enabled, the JSON Object Manager package and its respective actions will be available in the bot building interface.
 1. This will also allow you to run the JSON Object Manager demo bot which was included in the install from Bot Store located in Bot Store\JSON Object Manager – Automation Anywhere

To begin using the package, start by using the Initialize action to establish a session of setting and extracting values.

1. In the bot building interface - from the actions pane - open the JSON Object Manager package and drag the Initialize action into your bot's workflow.
2. Initialize the session by providing a properly formatted JSON Object string, and setting a String value for the return string
 - a. The return string should be set to **success** when the initialize action is able to appropriately parse the JSON
3. Once initialized, select the Query action from the JSON Object Manger package to start querying for values on the session.

- **Accessing root objects**

- Using the name of the key itself
- Example: token

- **Accessing child objects through dot notation**

-
- Using the name of the parent and relevant child keys
 - Example: user. email
 - **Accessing the length of an array**
 - Determining the number of objects in an array
 - Example: user.roles.length()
 - Note: the returned count value will return as a string, which will need to be converted to number for use as a loop count iterator
 - **Accessing a specific object within an array**
 - Accessing specific values from within an array
 - Especially useful when used in conjunction with loops
 - Example: user.roles[0].name
 - Note: when accessing specific objects within an array, the position of the object is 0 based, whereas the count is 1 based...so when used in loops, remember that the positional reference should be -1.
 - **Returning an entire array as a string**
 - Using the dot notation to access the array itself (without a position) the entirety of the array is returned as a string
 - Example: user.roles

Note: all the examples documented here are available in the demo bot included with this package. Its sample response is mimicked from the authenticate endpoint of the Control Room API should you want to practice with another JSON.

4. Support & FAQs

4.1 Support

Free bots are not officially supported. You can get access to Community Support through the following channels:

- You can get access to Community Support, connecting with other Automation Anywhere customers and developers on [APeople](#) – the [Bot Building Forum](#), the [Bot Store Support Forum](#), or the [Developers Everywhere Group](#).
- Automation Anywhere also provides a [Product Documentation portal](#) which can be accessed for more information about our products and guidance on [Enterprise A2019](#).

4.2 FAQs

For questions relating to Enterprise A2019: See the [Enterprise A2019 FAQs](#).

Appendix A: Record of Changes

No.	Version Number	Date of Change	Author	Notes

Appendix B: References

No.	Topic	Reference Link
1	Overview of Enterprise A2019	Click here
2	Guidance: Building basic A2019 bots	Click here
3	Guidance: Building A2019 action packages	Click here
4	APeople Community Forum	Click here
5	Automation Anywhere University	Click here