

SRP_Json

Creates and parses JSON.

Syntax

```
Result = SRP_Json(Handle, Service, Param1, Param2, Param3)
```

Returns

The service result.

Parameters

Parameter	Description
Handle	Handle to JSON Entity. Required.
Service	The service to execute against the given JSON Entity Handle. Required.
Param1	Generic parameter show purpose depends upon the Service being used. Optional.
Param2	Generic parameter show purpose depends upon the Service being used. Optional.
Param3	Generic parameter show purpose depends upon the Service being used. Optional.

Remarks

With RESTful WebAPI services becoming more prominent, the need to build and parse JSON is ever increasing. SRP_Json provides an API that not only makes this task easier, it also ensures the operations will happen quickly. Like many of our SRP Utilities, this single method provides multiple functions through services.

To get started, here are some terminologies you should become familiar with:

- **JSON:** JavaScript Object Notation. This is a lightweight interchange format utilized by many modern WebAPIs.
- **JSON Entity:** SRP_Json creates and manipulates JSON entities. A JSON entity might be one of the following types: Object, Array, Number, String, or Boolean.
- **JSON Entity Handle:** SRP_Json creates or returns JSON Entity Handles, which are simply numbers that point to a specific entity in memory. These handles are also passed to SRP_Json to manipulate a specific entity. **You must never alter a handle yourself.**
- **JSON object:** A JSON Entity that is of type "Object".
- **JSON array:** A JSON Entity that is of type "Array".
- **value:** A number, string, or boolean value as opposed to a JSON Entity Handle.
- **member:** A JSON entity that is a child of a JSON object. A member is identified by a name. For example, a JSON object can have a member whose name is "id."
- **element:** A JSON entity that is a child of a JSON array. An element is identified by an index. The first element in an array is always 1.

QuickStart Guide

The full list of services are provided below, but first lets get a bird's eye view of what it's like to use the API. We'll start with building a JSON from scratch, then we'll show how to parse JSON and extract values. In both cases, we'll use the following sample JSON:

```
{ "employees": [
  { "firstName": "John", "lastName": "Doe" },
  { "firstName": "Anna", "lastName": "Smith" },
  { "firstName": "Peter", "lastName": "Jones" }
]}
```

Here is how you parse JSON. The following example assumes we have the entire JSON contents above in a variable called SampleJSON:

```

// Parse the SampleJSON string
ParseResult = SRP_Json(EmployeesHandle, "Parse", SampleJSON)

// The result is "" if parsing was successful
If ParseResult EQ "" then

    // Success! Now get the employee array
    EmployeeArrayHandle = SRP_Json(EmployeesHandle, "Get", "employees")
    If EmployeeArrayHandle NE 0 then

        // Loop through each employee, extracting first and last name
        NumEmployees = SRP_Json(EmployeeArrayHandle, "GetCount")
        For iEmployee = 1 to NumEmployees
            FirstName = SRP_Json(EmployeeArrayHandle, "GetValue", "[" & iEmployee & "].firstName")
            LastName = SRP_Json(EmployeeArrayHandle, "GetValue", "[" & iEmployee & "].lastName")
        Next iEmployee

    end

end else

    // There was an error in parsing, so show it to the user
    Msg(@Window, ParseResult:@FM:@FM:@FM:"!")

end

// We're all done with the JSON entity that was created when parsing
SRP_Json(EmployeesHandle, "Release")

```

Now, let's create the sample JSON from scratch:

```

// Create the root object, which we'll call employees
If SRP_Json(EmployeesHandle, "New") then

    // Create a new array
    If SRP_Json(EmployeeArrayHandle, "New", "Array") then

        // Add the first employee
        If SRP_Json(SingleEmployeeHandle, "New") then
            SRP_Json(SingleEmployeeHandle, "SetValue", "firstName", "John")
            SRP_Json(SingleEmployeeHandle, "SetValue", "lastName", "Doe")
            SRP_Json(EmployeeArrayHandle, "Add", SingleEmployeeHandle)
            SRP_Json(SingleEmployeeHandle, "Release")
        end

        // Add the second employee
        If SRP_Json(SingleEmployeeHandle, "New") then
            SRP_Json(SingleEmployeeHandle, "SetValue", "firstName", "Anna")
            SRP_Json(SingleEmployeeHandle, "SetValue", "lastName", "Smith")
            SRP_Json(EmployeeArrayHandle, "Add", SingleEmployeeHandle)
            SRP_Json(SingleEmployeeHandle, "Release")
        end

        // Add the third employee
        If SRP_Json(SingleEmployeeHandle, "New") then
            SRP_Json(SingleEmployeeHandle, "SetValue", "firstName", "Peter")
            SRP_Json(SingleEmployeeHandle, "SetValue", "lastName", "Jones")
            SRP_Json(EmployeeArrayHandle, "Add", SingleEmployeeHandle)
            SRP_Json(SingleEmployeeHandle, "Release")
        end

        // Now add the array as a member of the root object
        SRP_Json(EmployeesHandle, "Set", "employees", EmployeeArrayHandle)

        // All done with the array object
        SRP_Json(EmployeeArrayHandle, "Release")

    end

    // Now get the actual JSON
    SampleJSON = SRP_Json(EmployeesHandle, "Stringify", "STYLED")

    // All done with the root object
    SRP_Json(EmployeesHandle, "Release")

end

```

Services

SRP_Json provides the following services. The first parameter is always a handle to an entity. Sometimes this handle is just an empty or unassigned variable, such as when you want to create a new JSON Entity. All other times the handle references a JSON Entity already in memory. The rest of the parameters depend upon the service in question, and if you are using the latest SRP Editor you'll see the tooltips change once you select a service--making it easy to know how to use it.

Service	Description
Add	Adds a JSON Entity to a JSON array.
AddValue	Adds a value to a JSON array.
AddValueArray	Adds an array of values to a JSON array. New in 2.1.1
Contains	Determines if a JSON object contains the given member or if a JSON array contains the given index.
Get	Returns a descendant JSON Entity within a given JSON Entity.
GetCount	Returns the number of members in a JSON object or elements in a JSON array.
GetMembers	Returns a list of a JSON object's members.
GetValue	Returns a descendant value within a given JSON Entity.

New	Creates a new JSON Entity.
Parse	Parses a standardized JSON string into a new JSON Entity.
Release	Deallocates a JSON Entity from memory.
Remove	Removes a member from a JSON object or an element from a JSON array.
RemoveAll	Removes all members from a JSON object or all elements from a JSON array.
Set	Sets a member within a JSON object or element within a JSON array to the given JSON Entity.
SetValue	Sets a member within a JSON object or element within a JSON array to the given value.
SetValueArray	Sets a member within a JSON object or element within a JSON array to the given array of values. New in 2.1.1
Stringify	Formats the given JSON Entity into a standardized JSON string.
Type	Returns a JSON Entity's type.
Validate	Validates JSON against a given schema. New in 2.2.1

Click on the service above to read more details.