



zmwJSON Version 1.1

Used for stringify JSON Objects and parse JSON Text.

As we know JSON is getting more popular day by day. Many applications are using JSON for response like Google Api , Microsoft Api. Currently VisualNeo Win does not support JSON Parser . so basically we use tricks, Regex and other hacks in order to get JSON Values but sometimes it fails getting some JSON data because its not a real JSON Parser. So zmwJSON Plugin will help us in parsing JSON Text and stringify the JSON Objects.

Let's install zmwJSON Plugin in VisualNeo Win.

- Step 1- Open VisualNEO Win.
- Step 2- Click on Options Menu and select Install-Plugins...
- Step 3- In Plug-Ins Installer window. Click on install button and select zmwJSON.nbp from the location where you have downloaded.
- Step 4- If zmwJSON can be seen in Installed Plug-Ins list it means zmwJSON Plugin has been installed

Registering a zmwJSON Plugin.

- Step 1- In Plug-Ins Installer window. Select zmwJSON from Installed Plug-Ins list.
- Step 2- Click on Register this Plug-In button. After couple of seconds Registration Dialog will appear which contains Registration ID.
- Step 3- Send us this Registration ID with Your Name and Version No on **contact@zmware.rcmediaservices.net**.
- Step 4- You will receive an email containing your Unlock Code within two to three business days.
- Step 5- Just type that Unlock Code into the Registration Dialog's Unlock Code field and click Register button.

 < == Sign will be shown on the registered version of zmwJSON

How to use zmwCreateJSONVariables

This action creates JSON Variables. It has two parameters the first one is JSON Variable Names and second is Type {Object,List,Base}. It is better to use setVar first to define JSON Variable Names and use suffix, prefix for different types as shown below. (You can even use it without setVar).

```
SetVar "[JSO_yourvarname]" "JSO_yourvarname" {Type is Object}
SetVar "[JSL_yourvarname]" "JSL_yourvarname" {Type is List}
SetVar "[JSB_yourvarname]" "JSB_yourvarname" {Type is Base}
```

Now call zmwCreateJSONVariables action for each setVar as shown below.

```
zmwCreateJSONVariables "[JSO_yourvarname]" "0"
zmwCreateJSONVariables "[JSL_yourvarname]" "1"
zmwCreateJSONVariables "[JSB_yourvarname]" "2"
```

Now these variables considered as JSON Variables and can be used at the place of JSON Variables.
[JSO_yourvarname] [JSL_yourvarname] [JSB_yourvarname]

Parsing JSON Text Through zmwJSON Plugin

This is a JSON Text containing **JSON Objects** and **JSON Lists**.
Red Color indicates JSON Objects and Green Color indicates JSON Lists.

```
{
  "success": true,
  "errors": null,
  "data": {
    "virtual_hosts": {
      "example.org": {
        "virtual_host": "example.org",
        "dns_identifiers": [
          "*.example.org",
          "example.org"
        ],
        "challenge_method": "dns-01",
        "cpanel_cert_id": "example_org_c17ac_e7973_1562912741_feadfa305d9ae4762638c89ce15",
        "cpanel_key_id": "c17ac_e7973_e2b23c1a4c9b7d3355e1a7a1acc8ceba",
        "cert_pem": "-----BEGIN CERTIFICATE-----...",
        "issuer_pem": "-----BEGIN CERTIFICATE-----...",
        "privkey_pem": "-----BEGIN RSA PRIVATE KEY-----...",
        "order_url": "https://acme-v02.api.letsencrypt.org/acme/order/000/001"
      }
    }
  }
}
```

So let’s parse the yellow highlighted text step by step.

Step 1- Create JSON Variable name it JSB_Parser. Type must be Base.

```
SetVar "[JSB_Parser]" "JSB_Parser"
zmwCreateJSONVariables "[JSB_Parser]" "2"
```

Step 2- Now parse the JSON Text into JSON Base.

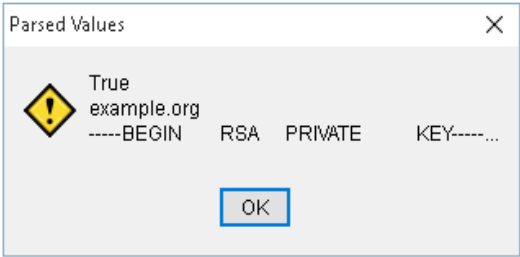
```
zmwParseJSONText "the JSON Text in the above grey box" "[JSB_Parser]" "True"
If your using VisualNeo Win Variable at the place of JSON Text Encode option must be false.
zmwParseJSONText "[VisualNeo Variable]" "[JSB_Parser]" "False"
```

Step 3- Now to retrieve yellow highlighted text values.

```
zmwGetJSONVarProperties "[JSB_Parser]" "[#34]success[#34]" "0" "[sucess]"
zmwGetJSONVarProperties "[JSB_Parser]" "[#34]data[#34]][#34]virtual_hosts[#34]][#34]example.org[#34]][#34]dns_identifiers[#34]|1" "0" "[dns]"
zmwGetJSONVarProperties "[JSB_Parser]" "[#34]data[#34]][#34]virtual_hosts[#34]][#34]example.org[#34]][#34]privkey_pem[#34]" "0" "[privkey]"
```

Step 4- You can use alertbox to check the result.

```
AlertBox "Parsed Values" "[sucess]][[dns]][[privkey]"
```



Convert JSON Variables into JSON Text(Stringify)

Let's create a simple JSON object containing three values and one JSON List.

Step 1- Create two JSON Variables both should be JSON Object.

```
SetVar "[JSO_Main]" "JSO_Main"
```

```
SetVar "[JSO_PersonDetail]" "JSO_PersonDetail"
```

```
zmwCreateJSONVariables "[JSO_Main]" "0"
```

```
zmwCreateJSONVariables "[JSO_PersonDetail]" "0"
```

Step 2- Create a JSON Variable which should be JSON List.

```
SetVar "[JSL_Childs]" "JSO_Childs"
```

```
zmwCreateJSONVariables "[JSL_Childs]" "1"
```

Step 3- Add these values into [JSO_PersonDetail] JSON Variable using zmwAddInJSONObject Action.

```
Key=Name; Value=Aslam; Value Type= String;
```

```
Key=Phone No; Value=034356656 Value Type= Integer;
```

```
Key=City; Value=Lahore; Value Type= String;
```

```
zmwAddInJSONObject "[JSO_PersonDetail]" "Name" "Aslam" "0"
```

```
zmwAddInJSONObject "[JSO_PersonDetail]" "Phone Number" "034356656" "1"
```

```
zmwAddInJSONObject "[JSO_PersonDetail]" "City" "Lahore" "0"
```

Step 4- Now add this [JSO_PersonDetail] JSON Object into another JSON Object

[JSO_Main] to separate each person's detail by its name. Select the value type JSON Variable.

```
zmwAddInJSONObject "[JSO_Main]" "Aslam" "[JSO_PersonDetail]" "4"
```

Step 5- Add these two values in JSON List [JSL_Childs] using zmwAddInJSONList Action.

```
Value=Amir; Value Type=String;
```

```
Value=Haris Value Type=String;
```

```
zmwAddInJSONList "[JSL_Childs]" "Amir" "0"
```

```
zmwAddInJSONList "[JSL_Childs]" "Haris" "0"
```

Step 6- Now add JSON List [JSL_Childs] into JSON Object [JSO_PersonDetail] using zmwAddInJSONObject Action.

```
Key=Childs; Value=[JSL_Childs]; Value Type=JSON Variable.
```

```
zmwAddInJSONObject "[JSO_PersonDetail]" "Childs" "[JSL_Childs]" "4"
```

Step 7- Use zmwStringify action to convert JSON Object [JSO_Main] into JSON Text.

```
zmwStringify "[JSO_Main]" "0" "[myJSONText]"
```

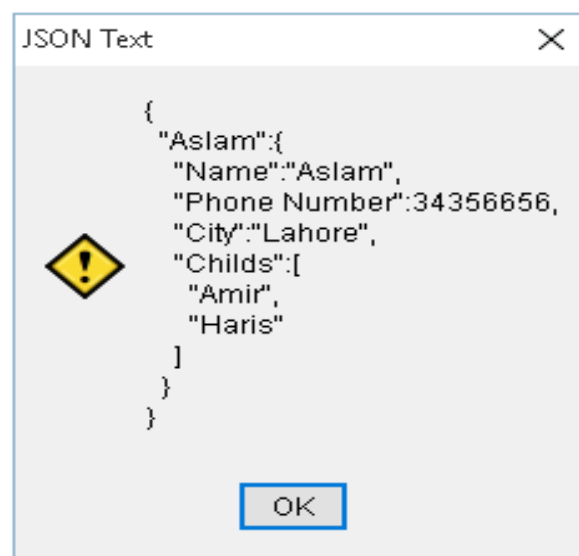
Step 8- Delete these two JSON Variables Values. [JSO_PersonDetail] and [JSL_Childs].

```
zmwDeleteJSONVariables "[JSO_PersonDetail]" "True"
```

```
zmwDeleteJSONVariables "[JSL_Childs]" "True"
```

Step 9- Use alertbox to see the result.

```
AlertBox "JSON Text" "[myJSONText]"
```



Technical Support

This help file contains just about everything you will need to know in order to use zmwJSON Plugin effectively. If you should encounter a technical problem or question not covered here, you may send us email at support@zmware.rcmediaservices.net

When contacting technical support, please include your VisualNeo Win version number, Windows version, zmwJSON Plugin version number and a detailed description of the problem and how to reproduce it.

Thanks.