

Article

[Kenneth Poindexter](#) · May 26, 2016 2m read

## Using SQL to Create JSON From an Existing Persistent Object

With the release of Cache 2016.1, JSON support was re-architected and made part of the core object model with the creation of %Object and %Array classes, which allow you to create dynamic JSON enabled objects and arrays.

On a recent demonstration I was working on, I had the need to create a REST web service that returned a JSON representation of a persistent object. After searching for methods that would allow me to accomplish this, ultimately I found none, until now.

With the release of 2016.2 (currently in beta) we introduce the Document Data Model along with SQL enhancements that allow you to query the documents and intermix the information within the document model with data contained within existing relational tables. Those same SQL enhancements can be used to create a JSON representation of a persistent object.

You can use SQL to query the Sample.Person table and return a JSON representation for each record. Take the following example Cache ObjectScript:

```
&sql(DECLARE PERSONS CURSOR FOR
      SELECT JSON_OBJECT('Name':NAME, 'DOB':DOB, 'Age':AGE) into :myJSONString FROM SAM
PLE.PERSON)

&sql(open PERSONS)

&sql(fetch PERSONS)

while (SQLCODE=0)
  {
    write !,myJSONString
    &sql(fetch PERSONS)
  }

quit
```

Looking at the code above, I declare a cursor called PERSONS allowing the script to traverse each record returned from the Sample.Person table. The query itself uses a new function that allows it to build a JSON object from the columns of the table, selecting, NAME, DOB, and AGE. For each record it returns a JSON string into the local variable named myJSONString.

It then writes out a JSON object for each record of Sample.Person. Each JSON object will look something like this:

```
{ "Name" : "Garcia,Julie F." , "DOB" : 30778 , "Age" : 91 }
```

Once I have my data in JSON format, I can do different things with it:

If I were creating a REST web service that returned a persistent object in JSON form I could simply write out

myJSONString as the response to the REST service call.

If I wanted to convert this to a dynamic %Object, I could use the \$fromJSON system method to convert this to a dynamic object allowing me to work with it using regular object properties, such as:

```
Set myObject=##class(%Object).$fromJSON(myJSONString)
```

Once I ' ve converted the JSON string to a dynamic object I am then able to access the methods using standard object syntax, such as:

```
Write myObject.Name
```

Yes, it would still be nice to be able to have a system method that would allow me to just convert an instance of a persistent object into JSON, but until then, this serves as an option to accomplish the goal.

[#Code Snippet](#) [#JSON](#) [#Caché](#)

---

Source URL: <https://community.intersystems.com/post/using-sql-create-json-existing-persistent-object>