

# Represent Nested Data in JSON Format Sample

Last Modified on 05/24/2022 11:14 am EDT

In some scenarios you may need to convert the nested data from an activity into JSON format, for example in an HTTP Consumer activity.

This sample demonstrates one such scenario of nested data of an organization structure where each organization group has  $n$  employees, and you want to represent it in JSON format:

## SQL data source

```
SELECT ETO.fldOrgId, ETO.fldEmpId, ORG.fldName AS OrgName, ORG.fldPath AS OrgPath, EMP.Employee_Name
FROM tblEmpToOrg ETO
INNER JOIN tblOrganization ORG ON ETO.fldOrgId = ORG.fldId
INNER JOIN ViewEmp EMP ON ETO.fldEmpId = EMP.fldID
```

This is mapped to a Data Model view called ViewOrgsAndEmps.

## JSON body expression

```
{
  "Groups" : [
    { {#for ( {Data Model}.Query("ViewOrgsAndEmps").GroupBy(it.fldOrgId).OrderBy(it.Key).Select(it) ) } }
    {
      "Group Id" : { {: it } }
      "Employees" : [
        { {#for (it) } }
        {
          "Employee Id" : { {: it.fldEmpId } },
          "Employee Name" : { {: it.Employee_Name } }
        } { {#if ( #t.Scope.Index < #t.Scope.Count-1 ) } }, { {#endif} }
        { {#endfor} }
      ]
    } { {#if ( #t.Scope.Index < #t.Scope.Count-1 ) } }, { {#endif} }
    { {#endfor} }
  ]
}
```

## Result

```
{
  "Groups": [
    {
      "GroupId": 8,
      "Employees": [
        {
          "Employee": 724
        },
        {
          "Employee": 1130
        }
      ]
    },
    {
      "GroupId": 12,
      "Employees": [
        {
          "Employee": 1234
        },
        {
          "Employee": 1252
        },
        {
          "Employee": 1253
        }
      ]
    },
    {
      "GroupId": 13,
      "Employees": [
        {
          "Employee": 1301
        },
        {
          "Employee": 1208
        },
        {
          "Employee": 1168
        }
      ]
    }
  ]
}
```