Data Transform

The goal of this project is to read in and transform json data. The data needs to be read in and certain fields need to be extracted from each json object. The fields need to then be written to a csv file.

Data Content: The General Service Administration's data.json harvest source. This file contains the metadata for the GSA's public data listing shown on data.gov as defined by the Project Open Data Source Data

Sample:


```
{
            "@type": "dcat:Dataset",
            "title": "2015 GSA Common Baseline Implementation Plan and CIO
Assignment Plan",
            "description": "This is GSA's 2015 Common Baseline
Implementation Plan and its CIO Assignment Plan per the requirements set
forth in FITARA legislation.",
            "modified": "2017-05-15",
            "accessLevel": "public",
            "identifier": "GSA-2016-01-22-01",
            "dataQuality": true,
            "license":
"https://creativecommons.org/publicdomain/zero/1.0/",
            "publisher": {
                "@type": "org:Organization",
                "name": "General Services Administration"
            },
            "accrualPeriodicity": "R/P1Y",
            "contactPoint": {
                "@type": "vcard:Contact",
                "fn": "Mick Harris",
                "hasEmail": "mailto:michael.harris@gsa.gov"
            },
            "distribution": [{
                    "@type": "dcat:Distribution",
                    "mediaType": "application/pdf",
                    "format": "pdf",
                    "title": "2015 GSA Common Baseline Implementation Plan
and CIO Assignment Plan",
                    "description": "This is GSA's 2015 Common Baseline
Implementation Plan and its CIO Assignment Plan per the requirements set
forth in FITARA legislation. Updated April 2017. Last Major Change to
version updated on March 4, 2019. Last Major change to version update on
8/5/2020. Last Major change to version update on 03/24/2022.",
                    "downloadURL":
"https://inventory.data.gov/dataset/64c56cec-4b8f-44c7-ba69-
090517f9f32e/resource/87e53999-aff1-4560-8bf0-
```

```
42d9dc8e4a69/download/2015gsafitaraimplementationandcioassignmentplan.pdf"
                }
            ],
            "keyword": ["Assignment Plan", "CIO", "Common Baseline",
"FITARA", "GSA IT", "Implementation Plan"],
            "bureauCode": ["023:00"],
            "programCode": ["023:000"],
            "theme": ["IT Initiatives"]
        }, {
            "@type": "dcat:Dataset",
            "title": "Award Exploration Tool",
            "description": "Interactive query tool designed to support in-
depth data exploration and exports; users are able to search for specific
award records, query expiring contracts, and export line item data with
added Category Management enrichments such as Level 1/2 categories, SUM
Tier, Addressable BIC / Tier 2 Contract, Contract Name (if applicable).",
            "modified": "2021-03-30T15:14:53.668Z",
            "accessLevel": "public",
            "identifier": "GSA-2021-03-30-03",
            "license":
"https://creativecommons.org/publicdomain/zero/1.0/",
            "rights": "true",
            "publisher": {
                "@type": "org:Organization",
                "name": "Federal Acquisition Service",
                "subOrganizationOf": {
                    "@type": "org:Organization",
                    "name": "General Services Administration"
                }
            },
            "contactPoint": {
                "@type": "vcard:Contact",
                "fn": "Kristen Wilson",
                "hasEmail": "mailto:govtwidecmdashboards@gsa.gov"
            },
            "distribution": [{
                    "@type": "dcat:Distribution",
                    "mediaType": "text/html",
                    "format": "html",
                    "title": "Award Exploration Tool",
                    "downloadURL": "https://d2d.gsa.gov/report/government-
wide-category-management-contract-management-and-operational-reporting-
tools"
                }
            ],
            "keyword": ["award", "category management", "contract",
"exploration", "obligation", "vendor"],
            "bureauCode": ["015:11"],
            "programCode": ["015:001"],
            "language": ["en-US"]
        }
...
```

Extracting the fields can be tricky because some of the data lives in arrays. Every unique combination of the values in each extractable field needs its own row in the csv file. For example: If we want the fields: modified, contactPoint.fn, keyword, The output would be:

```
modified, contactPoint.fn, keyword
"2017-05-15", Mick Harris", "Assignment Plan"
"2017-05-15", Mick Harris", "CIO"
"2017-05-15", Mick Harris", "Common Baseline"
"2017-05-15", Mick Harris", "FITARA"
"2017-05-15", Mick Harris", "GSA IT"
"2017-05-15", Mick Harris", "Implementation Plan"
"2021-03-30T15:14:53.668Z", "Kristen Wilson","award"
"2021-03-30T15:14:53.668Z", "Kristen Wilson", "category management"
"2021-03-30T15:14:53.668Z", "Kristen Wilson", "contract"
"2021-03-30T15:14:53.668Z", "Kristen Wilson", "exploration"
"2021-03-30T15:14:53.668Z", "Kristen Wilson", "obligation"
"2021-03-30T15:14:53.668Z", "Kristen Wilson", "vendor"
```

Solve the solution with the fields: modified, publisher.name, publisher.subOrganizationOf.name, contactPoint.fn, keyword

It is possible that a field doesn't exist for certain json objects (see publisher.subOrganizationOf.name in the example data). In the case where a field for a particular object doesn't exist, just put a blank value("") and publish the data that does exist for that row.

When working through this problem keep in mind the data files we may get can be multiple GB large. We want the solution to read in portions of the file at a time and stream the data out.

Grading Criteria

You code should be written in Go.

Our review team will consider the following aspects when evaluating your submission:

1. **Meeting Requirements**
    - Does your solution fulfill all the needs outlined in this document?
2. **Code Testing**
    - Have you written adequate tests for your code?
    - Do your tests cover critical functionality and edge cases?
3. **Code Documentation and Ease of Use**
    - Is your code well-documented and easy to understand?
    - Is it clear how to run your code?
    - **NOTE:** If it's not immediately obvious how to run your code as submitted, your submission will not make a good first impression.
4. **Idiomatic Go Code**

    - Would a developer familiar with other Go projects recognize the patterns and style of your code?
    - Is this reflected in both your code and project structure?

5. **Future-Proofing**

    - Does your solution consider potential future requirements?
    - Is your code flexible enough to easily accommodate future adjustments or expansions?

We encourage you to ask questions about design, requirements clarification, features, or any other aspects of the project. It's always better to ask questions than to make incorrect assumptions.

Remember, this is your chance to showcase your skills! Your expertise should be evident throughout your submission. This project provides us with concrete evidence to determine if you would be a good fit for our team. We're excited to see what you can do!

Zip up your solution and email it to: twendell@centripetal.ai when you are finished.