

1. Read multiple json files into a working directory for further converting into a dataset.

I have files text1, text2, text3 in the directory json.

```
install.packages("rjson")
```

```
library(rjson)
```

```
setwd("C://Users//aruna//Desktop//Assignments//jsonf")
```

```
x <-list.files(pattern="*.json")
```

```
l<-lapply(x,function(x) fromJSON(file=x))
```

```
|
```

2. Parse the following JSON into a data frame

```
js<-'{  
  "name": null, "release_date_local": null, "title": "3 (2011)",  
  "opening_weekend_take": 1234, "year": 2011,  
  "release_date_wide": "2011-09-16", "gross": 59954  
}'
```

A data frame is a table or a two-dimensional array-like structure in which each column contains values of one variable and each row contains one set of values from each column.

Following are the characteristics of a data frame.

- The column names should be non-empty.
- The row names should be unique.

- The data stored in a data frame can be of numeric, factor or character type.
- Each column should contain same number of data items.
- -----

Data frame cant accept null values(it gives object not found as error) hence we use simplification function from jsonlite package.

Hence we have to restructure the above data to run successfully into acceptable format for data frames.

Null values will be converted into NA in data frame when we use jsonlite package simplifyvector function.

Simplification is the process where JSON arrays automatically get converted from a list into a more specific R class.

When `simplifyVector` is enabled, JSON arrays containing **primitives** (strings, numbers, booleans or null) simplify into an atomic vector

When `simplifyDataFrame` is enabled, JSON arrays containing **objects** (key-value pairs) simplify into a data frame.

The data frame gets converted back into the original JSON structure by `toJSON` (whitespace and line breaks are ignorable in JSON).

Hence you can go back and forth between dataframes and JSON, without any manual data restructuring.

3. Write a script for variable binning using R.

Using cut and range