Code - DART.R

LearningSpoonsR

2018-05-15

KMA vs DART

KMA

```r
# 0. setup environment
source("LSR.R")
activate(c("dplyr", "tidyverse", "jsonlite"))

# 1. setup API
svc_key <- paste0(
  "uEID6no5WOeFLEu%2FYZpdjKHQVrE2HtFEig4lJ7iHWiE5wGS1L3RvmusPMDkumoj",
  "8f%2BSffvPYWO%2B5xXu%2FrQ%2Bvzg%3D%3D")
today   <- gsub("-", "", Sys.Date())
url     <- paste0(
  "http://newsky2.kma.go.kr/service/SecndSrtpdFrcstInfoService2",
  "/ForecastSpaceData")

# 2. get maxPage
fields  <- c("ServiceKey", "base_date", "base_time", "nx", "ny" , "_type")
values  <- c(svc_key     , today      , "0800"     , "60", "127", "json")
request <- paste(fields, values, sep = "=") %>% paste(collapse="&")
query   <- paste0(url, "?", request)
raw     <- readLines(query, warn = "F", encoding = "UTF-8") %>% fromJSON()
maxPage <- ceiling(raw$response$body$totalCount/raw$response$body$numOfRows)
```

```r
# 3. collect all pages
fields  <- c("ServiceKey", "base_date", "base_time", "nx", "ny" , "_type", "pageNo")
dataset <- data.frame()
for (i in 1:maxPage) {
  values_i  <- c(svc_key  , today      , "0800"    , "60", "127", "json" , i)
  request_i <- paste(fields, values_i, sep = "=") %>% paste(collapse="&")
  query_i   <- paste0(url, "?", request_i)
  raw_i     <- readLines(query_i, warn = "F", encoding = "UTF-8") %>% fromJSON()
  dataset_i <- raw_i$response$body$items$item
  dataset   <- rbind(dataset, dataset_i)
}

# 4. deliver output
# print(dataset)
```

DART

```
loadDART <- function(start_dt = Sys.Date()-3, end_dt = Sys.Date()-1) {
  # loadDART()
  # 0. setup environment
  source("LSR.R")
  activate("dplyr", "jsonlite")

  # 1. setup API
  svc_key <- "f96deaa9b5f380145bc2d57eddc9b8b0a8c358b4"
  url     <- "http://dart.fss.or.kr/api/search.json"

  # 2. get maxPage
  fields <- c("auth", "start_dt", "end_dt", "page_set")
  values <- c(svc_key,
              format(start_dt,"%Y%m%d"),
              format(end_dt,"%Y%m%d"),
              100)
  request <- paste(fields, values, sep = "=") %>% paste(collapse="&")
  query   <- paste0(url, "?", request)
  raw     <- readLines(query, warn = "F", encoding = "UTF-8") %>% fromJSON()
  maxPage <- raw$total_page
```

```r
  # 3. collect all pages
  fields <- c(fields, "page_no")
  dataset <- data.frame()
  for (i in 1:maxPage) {
    values_i  <- c(values, i)
    request_i <- paste(fields, values_i, sep = "=") %>% paste(collapse="&")
    query_i   <- paste0(url, "?", request_i)
    raw_i     <- readLines(query_i, warn = "F", encoding = "UTF-8") %>% fromJSON()
    dataset_i <- raw_i$list
    dataset   <- rbind(dataset, dataset_i)
  }

  # 4. deliver output
  return(dataset)
}

dartLookup <- function(dartObj, keyword) {
  # loadDART() %>% dartLookup("합병")
  selects <- which(grepl(keyword, dartObj[,4]))
  deliver <- dartObj[selects, c(7,2,4,3,6,5)]
  return(data.frame(deliver, stringsAsFactors = FALSE))
}
```

“Try n=1,2,3, then generalize it” - Emanuel Derman

“Try n=1,2,3, then generalize it” - Emanuel Derman

blank

KMA vs molit

KMA

```r
# 0. setup environment
source("LSR.R")
activate(c("dplyr", "tidyverse", "jsonlite"))

# 1. setup API
svc_key <- paste0(
  "uEID6no5WOeFLEu%2FYZpdjKHQVrE2HtFEig4lJ7iHWiE5wGS1L3RvmusPMDkumoj",
  "8f%2BSffvPYWO%2B5xXu%2FrQ%2Bvzg%3D%3D")
today   <- gsub("-", "", Sys.Date())
url     <- paste0(
  "http://newsky2.kma.go.kr/service/SecndSrtpdFrcstInfoService2",
  "/ForecastSpaceData")

# 2. get maxPage
fields  <- c("ServiceKey", "base_date", "base_time", "nx", "ny" , "_type")
values  <- c(svc_key     , today      , "0800"     , "60", "127", "json")
request <- paste(fields, values, sep = "=") %>% paste(collapse="&")
query   <- paste0(url, "?", request)
raw     <- readLines(query, warn = "F", encoding = "UTF-8") %>% fromJSON()
maxPage <- ceiling(raw$response$body$totalCount/raw$response$body$numOfRows)
```

```r
# 3. collect all pages
fields  <- c("ServiceKey", "base_date", "base_time", "nx", "ny" , "_type", "pageNo")
dataset <- data.frame()
for (i in 1:maxPage) {
  values_i  <- c(svc_key  , today      , "0800"    , "60", "127", "json" , i)
  request_i <- paste(fields, values_i, sep = "=") %>% paste(collapse="&")
  query_i   <- paste0(url, "?", request_i)
  raw_i     <- readLines(query_i, warn = "F", encoding = "UTF-8") %>% fromJSON()
  dataset_i <- raw_i$response$body$items$item
  dataset   <- rbind(dataset, dataset_i)
}

# 4. deliver output
# print(dataset)
```

Molit

```
# 0. setup environment
options(stringsAsFactors = FALSE)
source("LSR.R")
activate("XML", "stringr", "tidyverse")

# 1. setup API
svc_key <- paste0("EeBjN2xdCzzcqHvefO0rZXaycAim0uGpKxnOX72PY1UpkSZnifzIK1kx",
                  "Lm61XXaQ4pFxhbW%2F%2FZbmQDKFiAFNVA%3D%3D")
url     <- paste0("http://openapi.molit.go.kr:8081/OpenAPI_ToolInstallPackage/",
                  "service/rest/RTMSOBJSvc/getRTMSDataSvcAptTrade")

# 2. get maxPage - One page is enough!

# 3. collect all data
fields  <- c("LAWD_CD", "DEAL_YMD", "serviceKey")
values  <- c("11110", "201712", svc_key) # Jongro-Gu
request <- paste(fields, values, sep = "=") %>% paste(collapse="&")
query   <- paste0(url, "?", request)
raw     <- xmlTreeParse(query, useInternalNodes = TRUE, encoding = "utf-8") %>%
  xmlRoot()
items   <- raw[['body']][['items']]
```

```r
# XML is trickier than JSON! (Needs to collect from each item)
# No i-loop for page, because 1 page is enough.
# This j-loop is iterating item.
dataset <- data.frame()
for (j in 1:xmlSize(items)) {
  item    <- items[[j]] %>% xmlSApply(xmlValue)
  dataset <- rbind(dataset, as.vector(item))
}

# A little clean-up
colnames(dataset) <-
  c("price", "builtAt", "year", "dong", "aptName",
    "month", "day", "size", "address", "guCode", "floor")
dataset$price <- as.numeric(sub(",", "", dataset$price))
dataset[,c(2,3,6,8,11)] <- sapply(dataset[,c(2,3,6,8,11)], as.numeric)

# 4. deliver output
# print(head(dataset))
```

```r
# 4. deliver output
head(dataset)
```

```
##    price builtAt year    dong                    aptName month   day   size
## 1  91000    2000 2017  청운동                  청운현대    12 11~20 129.76
## 2  54000    1973 2017  필운동              사직파크맨션    12  1~10  99.17
## 3 140000    2008 2017  사직동 광화문풍림스페이스본(9-0)    12  1~10 158.99
## 4  89900    2008 2017  사직동 광화문풍림스페이스본(9-0)    12  1~10 121.37
## 5 111000    2008 2017  사직동 광화문풍림스페이스본(9-0)    12  1~10 158.99
## 6  95000    2008 2017  사직동 광화문풍림스페이스본(9-0)    12 11~20 108.07
##   address guCode floor
## 1   56-45  11110     5
## 2     142  11110     5
## 3       9  11110    12
## 4       9  11110     2
## 5       9  11110     1
## 6       9  11110     6
```

- 메뉴얼
- https://www.data.go.kr/commonUser/fileDownload.do?atchFileId=FILE_000000001429779&fileDetailSn=0
- 구별 코드 조회
- http://code.mogaha.go.kr/jsp/stdcode/regCodeL.jsp