Rough schedule for week #5 and week #6

(from 20201019 to 20201026)

[**政府資料開放平台**](https://data.gov.tw/)**-資料下載範例**

1 [高雄市重要環保統計資料 \_ 政府資料開放平臺](http://data.gov.tw/node/9132)

※ CSV (線上.csv檔案直接下載，更改檔名後使用)

test = read.csv("高雄市重要環保統計資料.csv")

str(test)

View(test)

## 有時線上.csv檔案下載後，需將檔案編碼改為ANSI，則可在R離線使用。

## 線上.csv檔案下載後，以「記事本」開啟，以存檔類型為所有檔案、編碼為ANSI，再回存該檔。最好另取檔名(本例為：高雄市重要環保統計資料.csv)，在R離線使用則可執行。

* JSON (亦可線上開啟連結，再另存新檔在R離線使用)

library(jsonlite)

url <- 'https://data.epa.gov.tw/api/v1/stat\_p\_07?limit=1000&api\_key=9be7b239-557b-4c10-9775-78cadfc555e9&format=json'

高雄市重要環保統計資料 <- fromJSON(url)

class(高雄市重要環保統計資料)

head(高雄市重要環保統計資料)

View(高雄市重要環保統計資料$records)

df <- 高雄市重要環保統計資料$records

str(df)

is.na(df)

#na.omit(df)

df$ItemValue <- as.numeric(df$ItemValue)

str(df)

## 該json檔的結構有問題:有fields與records

library(dplyr)

fromJSON(url) %>% as.data.frame #就不能用

※ XML

## 離線(offline)使用資料可執行(executable )，線上(online)則否

library("XML")

library("methods")

# Convert the input xml file to a data frame.

xmldataframe <- xmlToDataFrame("stat\_p\_07.xml")

# NOT work

xmldataframe <- xmlToDataFrame("https://data.epa.gov.tw/api/v1/stat\_p\_07?limit=1000&api\_key=9be7b239-557b-4c10-9775-78cadfc555e9&format=xml")

print(xmldataframe)

View(xmldataframe)

**但，亂碼! (目前無解)**

2 [臺灣地區蔬菜生產概況 \_ 政府資料開放平臺](http://data.gov.tw/node/9539)

* **JSON**

View(jsonlite::fromJSON("https://data.coa.gov.tw/Service/OpenData/DataFileService.aspx?UnitId=113"))

View(jsonlite::fromJSON("臺灣地區蔬菜生產概況.json"))

**※ CSV**

### 線上.csv檔案下載後，以「記事本」開啟，以存檔類型為所有檔案、編碼為ANSI，再回存該檔。最好另取檔名(本例為：臺灣地區蔬菜生產概況.csv)，在R離線使用則可執行。

df <- read.csv("臺灣地區蔬菜生產概況.csv")

View(df)

**※** XML (無法順利執行)

library("XML")

library("methods")

# Convert the input xml file to a data frame.

xmldataframe <- xmlToDataFrame("本機或網路上的檔案名稱")

3[自來水全球資訊網本公司開放資料專區 - 台灣自來水公司](https://www.water.gov.tw/ch/OpenData?nodeId=3395) (項次:1與2為例)

## CSV

df <- read.csv("停水資訊-CSV.csv") #UTF-8編碼，會出問題

head(df)

df01 <- read.csv("停水資訊.csv") #ANSI編碼，可讀

head(df01)

## JSON (要先改為ANSI)

View(jsonlite::fromJSON("停水資訊-JSON.json"))

亦可參考簡易版的「[政府資料開放平臺備份](http://data.g0v.ronny.tw/index/list)」

小心避開一些檔案整理不好的機關網站，例如：

[鄉鎮天氣預報-鄉鎮沿海未來2天逐3小時天氣預報](https://data.gov.tw/dataset/130243)

###################################################################

取得資料後，如何應用與分析？舉一範例如下：

以「[教育部大專校院校務資訊公開平臺](https://udb.moe.edu.tw/ReportCategories)」的資料為例

先下載「學12-1.新生註冊率-以「系(所)」統計」資料檔，接著

* 下載後更名為「新生註冊率.csv」；
* 以「記事本」開啟「新生註冊率.csv」；
* 以存檔類型為所有檔案、編碼為ANSI，再另存該檔(以利R正確地去讀取「新生註冊率.csv」)。

1. 先資料調處(data manipulation)
2. 用 Shiny展示新生註冊率查詢 (參考shiny cheat sheet)

library(markdown)

library(shiny)

library(ggplot2)

df<- read.csv("新生註冊率.csv")

str(df)

df <- df[,1:13]

colnames(df)[4] <- "學校代碼"

colnames(df)[10] <- "核定新生招生名額"

colnames(df)[11] <- "保留入學資格人數"

colnames(df)[12] <- "實際註冊人數"

colnames(df)[13] <- "新生註冊率"

df$學校代碼 <- as.factor(df$學校代碼)

options(width=120)

library(dplyr)

學年度 <- distinct(df, 學年度)

學校名稱 <- distinct(df, 學校名稱)

學制班別 <- distinct(df, 學制班別)

系所名稱 <- distinct(df, 系所名稱)

ui <- navbarPage("大專校院新生註冊率查詢",

tabPanel("系所新生註冊率查詢",

fluidPage(

br(),

fluidRow(

column(4,

hr(),

selectInput('in4', '學年度', c(Choose='', 學年度), selectize=TRUE)

),

column(4,

hr(),

selectInput('in5', '學校名稱', c(Choose='', 學校名稱), selectize=TRUE)

),

column(4,

hr(),

selectInput('in6', '學制班別', c(Choose='', 學制班別), selectize=TRUE)

),

mainPanel(

width = 100,

tableOutput("out4")

)

)

)

)

)

server <- function(input, output, session) {

output$out4 <- renderTable({

(a=filter(df, 學年度==input$in4, 學校名稱==input$in5, 學制班別==input$in6))

})

}

shinyApp(ui = ui, server = server)