DATA SCIENCE &
INFORMATION TECHNOLOGY

# KNOWLEDGE MANAGEMENT

KM-DS2565/01

องค์ความรู้

# WEB SCRAPING

## การดึงข้อมูลจากเว็บเพจ


ผศ.วีรวรรณ จันทนะทรัพย์
อาจารย์ผู้รับผิดชอบหลักสูตร
วิทยาศาสตรบัณฑิตสาขาวิชาวิทยาการข้อมูลฯ
คณะวิทยาศาสตร์และเทคโนโลยี
มหาวิทยาลัยเทคโนโลยีราชมงคลพระนคร

veerawan.j@rmutp.ac.th
http://sci.rmutp.ac.th

# การดึงข้อมูลจากเว็บเพจ WEB SCRAPING

ผศ.วีรวรรณ จันทนะทรัพย์
อาจารย์ผู้รับผิดชอบหลักสูตร วิทยาศาสตรบัณฑิตสาขาวิชาวิทยาการข้อมูลฯ
คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยเทคโนโลยีราชมงคลพระนคร

## นิยาม การดึงข้อมูล

เป็นเทคโนโลยีสำหรับการดึงข้อมูล หรือค้นคืนข้อมูลจากกองข้อมูล (Collection Data) ที่ตอบโจทย์ความต้องการทางข้อมูล (Data Need) โดยแหล่งของข้อมูลมาจากไฟล์แฟ้มข้อมูล (File Data), เว็บไซต์ (Web Site) หรือแหล่งอื่นๆ โดยข้อมูลที่ดึงได้จะถูกนำมาใช้งานตามวัตถุประสงค์ที่แตกต่างกัน เช่น งานวิเคราะห์ข้อมูล (Data Analytics) งานนำเสนอข้อมูล (Data Visualization) เป็นต้น

สำหรับเทคนิคการดึงข้อมูล นั้นขึ้นอยู่กับแหล่งข้อมูลที่ต้องการดึง โดยทั่วไปการดึงข้อมูลแบ่งออกได้ 2 เทคนิคหลักคือ ดึงแบบใช้มนุษย์ (Manual Method) และแบบอัตโนมัติ (Automatic Method)

**1 MANUAL METHOD** เป็นเทคนิคที่ผู้ดึงต้องเข้าไปดึงข้อมูลที่ต้องการด้วยการคัดลอกด้วยตนเอง แล้วนำมาจัดรูปแบบให้พร้อมต่อการนำไปประยุกต์ใช้งานตามวัตถุประสงค์ วิธีการนี้ค่อนข้างใช้เวลามาก

**2 AUTOMATIC METHOD** เป็นวิธีการที่ใช้หลักการทางภาษาคอมพิวเตอร์เขียนคำสั่ง ร่วมกับเทคนิคต่าง ๆ เพื่อดึงข้อมูลที่ต้องการ วิธีการนี้เป็นที่นิยมมาก เพราะสามารถดึงข้อมูลได้แบบอัตโนมัติ รวดเร็ว และข้อมูลมีความเป็นปัจจุบัน

**ปัจจุบัน**เทคนิคการดึงข้อมูลแบบ AUTOMATIC-METHOD ได้รับความนิยมมากในขั้นตอนการเก็บรวบรวมข้อมูล (Collection Data) ของศาสตร์ด้านวิทยาการข้อมูล (Data Science) โดยเทคนิคการเก็บรวบรวมข้อมูลแบบอัตโนมัติสามารถทำได้หลายวิธี พอสรุปได้ดังนี้

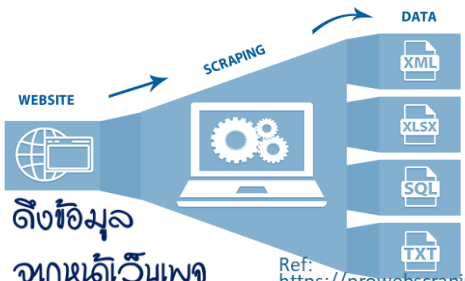

### การดึงข้อมูลด้วย LINK FILE

ใช้ในกรณีเจ้าของเว็บไซต์ผู้ให้บริการข้อมูลได้จัดเตรียมไฟล์ข้อมูลในรูปแบบต่างๆ เช่น .xlsx, csv, xml และ .json เป็นต้น แล้วแชร์ลิงก์ไฟล์ข้อมูลเพื่อให้ผู้ดึงข้อมูลหรือนักพัฒนาโปรแกรมได้ดาวน์โหลดไฟล์ใช้งาน


### การดึงข้อมูลด้วย api

Application Programming Interface

Application Programming Interface หรือ API คือ ชุดคำสั่ง (Script Code) ที่อนุญาตให้ Software Program สามารถสื่อสารระหว่างกันได้เพื่อขอใช้บริการจาก OS/ Application อื่นๆ หลักการทำงานพื้นฐาน คือ ติดตั้ง Function และเรียกใช้งานตามข้อกำหนดที่ได้เขียนไว้




### ดึงข้อมูลจากหน้าเว็บเพจ WEB Scraping

Ref: https://prowebscraping.com/what-is-web-scraping/

เป็นวิธีดึงข้อมูลต่าง ๆ จากหน้าเว็บที่เปิดเผยต่อสาธารณะ โดยข้อมูลที่ดึงประกอบด้วย เช่น ข้อมูลราคา ข้อความ รูปภาพ และอื่นๆ Web Scraping มีประโยชน์อย่างมากสำหรับการรวบรวมข้อมูล ช่วยประหยัดเวลาในการจัดเก็บข้อมูล

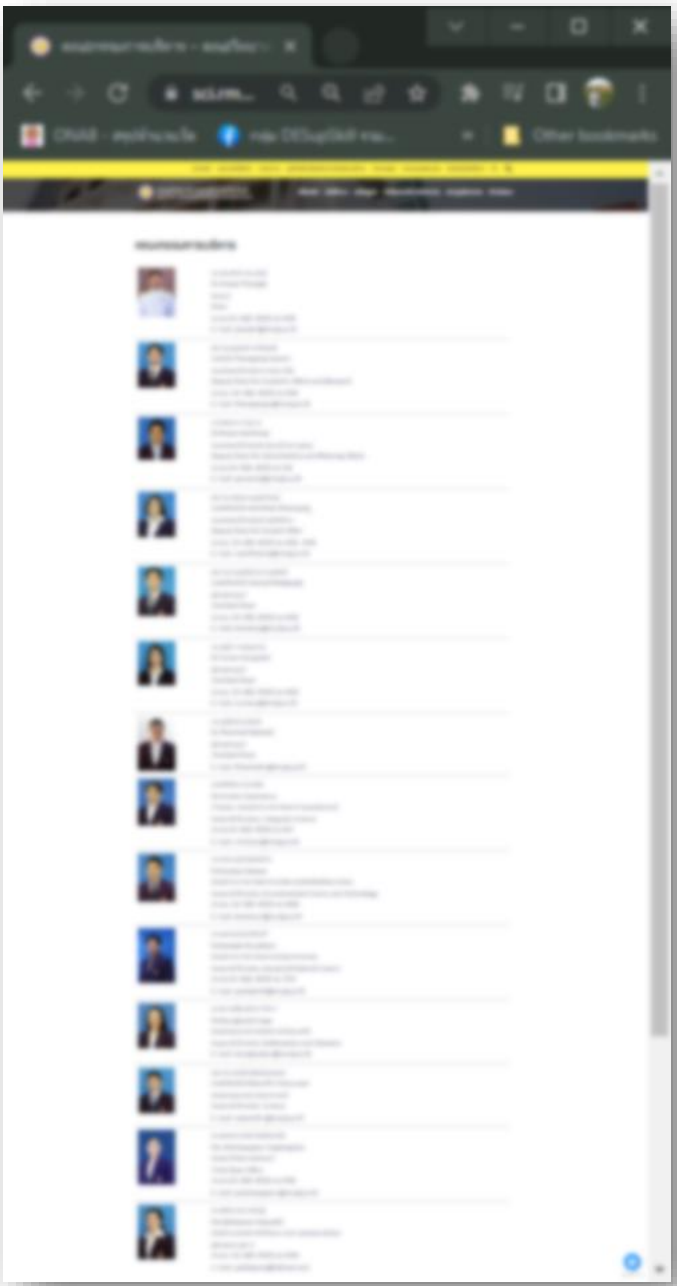
การดึงข้อมูลจากเว็บเพจ WEB SCRAPING

ผศ.วีรวรรณ จันทนะทรัพย์
อาจารย์ผู้รับผิดชอบหลักสูตร วิทยาศาสตรบัณฑิตสาขาวิชาวิทยาการข้อมูลฯ
คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยเทคโนโลยีราชมงคลพระนคร

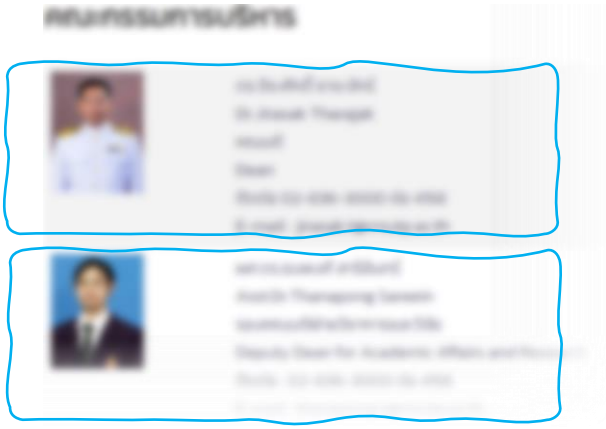
สำหรับองค์ความรู้นี้ ผู้เขียนขออธิบายพร้อมยกตัวอย่างกรณีศึกษาการดึงข้อมูลจากเว็บเพจ โดยความรู้ที่นำมาใช้เขียน KM ฉบับนี้ได้มาจากการได้รับฝึกอบรมในหลักสูตร Machine Learning investing 101

KM ฉบับนี้นำเสนอเทคนิคการดึงข้อมูลจากหน้าเว็บกรณีศึกษา โดยกรณีศึกษาใช้หน้าเว็บเพจของหน่วยงานต้นสังกัดของผู้เขียน คือ หน้าเว็บเพจ คณะกรรมการบริหารคณะวิทยาศาสตร์และเทคโนโลยี โดยมีลิงก์ คือ

https://sci.rmutp.ac.th/%e0%b8%9c%e0%b8%b9%e0%b9%89%e0%b8%9a%e0%b8%a3%e0%b8%b4%e0%b8%ab%e0%b8%b2%e0%b8%a3/



ข้อมูลคณะกรรมการฯมีจำนวน 14 ท่าน โดยรายการข้อมูลประกอบด้วย 7 รายการ คือ ภาพ ชื่อภาษาไทย ชื่อภาษาอังกฤษ ตำแหน่งภาษาไทย ตำแหน่งภาษาอังกฤษ ส่วนการติดต่อ และที่อยู่จดหมายอิเล็กทรอนิกส์



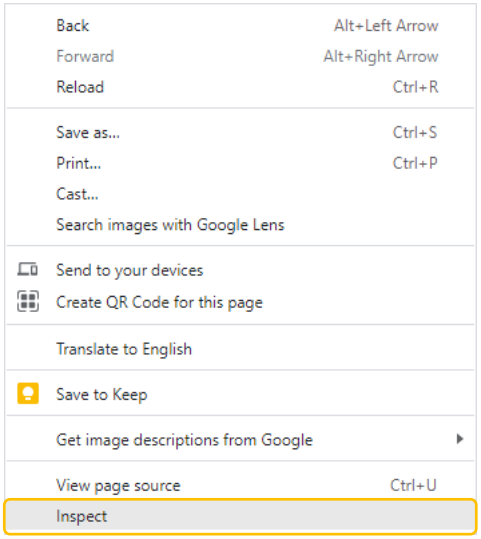
กำหนดโจทย์ปัญหาสำหรับการดึงข้อมูลในครั้งนี้ เป็นการกำหนดเป็นรายข้อดังนี้

1. ดำเนินการดึงข้อมูลจำนวน 7 รายการข้างต้นจากหน้าเว็บเพจกรณีศึกษา
2. บันทึกข้อมูลที่ดึงลงในไฟล์ข้อมูลในรูปแบบ .csv
3. บันทึกและจัดเก็บข้อมูลภาพคณะกรรมการฯ

จากโจทย์ปัญหาข้างต้น ผู้เขียนขอนำเสนอวิธีการดึงข้อมูลจากหน้าเว็บเพจด้วยเครื่องมือภาษาคอมพิวเตอร์ คือ ภาษาไพธอน (Python Language Programming) ร่วมกับไลบรารีสำหรับดึงข้อมูล ดำเนินการดึงข้อมูลทั้ง 7 รายการของคณะกรรมการฯ ในแต่ละราย ประกอบด้วย Requests, Beautifulfoup และ Pandas โดยสาธิตการดึงข้อมูลบนเว็บบริการ Software as a Service (Saas) โฮสต์โปรแกรม Jupyter Notebook บน Cloud ของ Google


ก่อนเริ่มขั้นตอนการดึงข้อมูล มาทำความเข้าใจชุดคำสั่ง HTML หรือ HTML Tag ของหน้าเว็บเพจกรณีศึกษาก่อนว่า ข้อมูลที่ต้องการดึงอยู่ Tag ใด วิธีการตรวจสอบ HTML Tag ดำเนินการได้ดังนี้

ให้คลิกเมาส์ขวาบนหน้าเว็บเพจคณะกรรมการฯ เลือกรายเมนู Inspect




จะปรากฏหน้าจอส่วนชุดคำสั่ง HTML ของหน้าเว็บเพจคณะกรรมการฯ ดังภาพ

```
▼<div class="row">
  ::before
  ▼<div id="primary" class="content-area"> == $0
    ▼<main id="main" class="post-wrap" role="main">
      ▼<article id="post-72" class="post-72 page type-page status-publish hentry">
        ▶<header class="entry-header">…</header>
        <!-- .entry-header -->
        ▼<div class="entry-content">
          ▼<table id="tablepress-4" class="tablepress tablepress-id-4">
            ▼<tbody class="row-hover">
              ▼<tr class="row-1">
                ▼<td class="column-1">
                  <img src="http://sci.rmutp.ac.th/web2558/wp-content/uploads/2020/09/jirasak5.jpg"
                  alt="Jirasak" width="110" height="150" class="alignnone size-large wp-image-1465">
                </td>
                ▼<td class="column-2">
                  "ดร.จิระศักดิ์ ธาระจักร์"
                  <br>
                  " Dr.Jirasak Tharajak"
                  <br>
                  " คณบดี"
                  <br>
                  " Dean "
                  <br>
                  " ติดต่อ 02-836-3000 ต่อ 4158"
                  <br>
                  " E-mail : jirasak.t@rmutp.ac.th"
                </td>
              </tr>
              ▶<tr class="row-2">…</tr>
              ▶<tr class="row-3">…</tr>
              ▶<tr class="row-4">…</tr>
              ▶<tr class="row-5">…</tr>
              ▶<tr class="row-6">…</tr>
              ▶<tr class="row-7">…</tr>
              ▶<tr class="row-8">…</tr>
              ▶<tr class="row-9">…</tr>
              ▶<tr class="row-10">…</tr>
              ▶<tr class="row-11">…</tr>
              ▶<tr class="row-12">…</tr>
              ▶<tr class="row-13">…</tr>
              ▶<tr class="row-14">…</tr>
            </tbody>
          </table>
```

ข้อมูลอยู่ภายใต้ Tag 

ลิงก์ภาพอยู่ภายใน Tag <img src>

ข้อมูลชื่อภาษาไทย ชื่อภาษาอังกฤษ ตำแหน่งภาษาไทย ตำแหน่งภาษาอังกฤษ ส่วนติดต่อโทรศัพท์ และที่อยู่จดหมายอิเล็กทรอนิกส์ อยู่ภายใน Tag <td class=column-2>

ข้อมูลรายละเอียดของคณะกรรมการฯ แต่ละท่าน จะถูกจัดวางอยู่ใน <tr class=row-i> เมื่อ i คือ 1-14

การดึงข้อมูลจากเว็บเพจ
WEB SCRAPING

ผศ.วิรวรรณ จันทนะทรัพย์
อาจารย์ผู้รับผิดชอบหลักสูตร วิทยาศาสตรบัณฑิตสาขาวิชาวิทยาการข้อมูลฯ
คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยเทคโนโลยีราชมงคลพระนคร

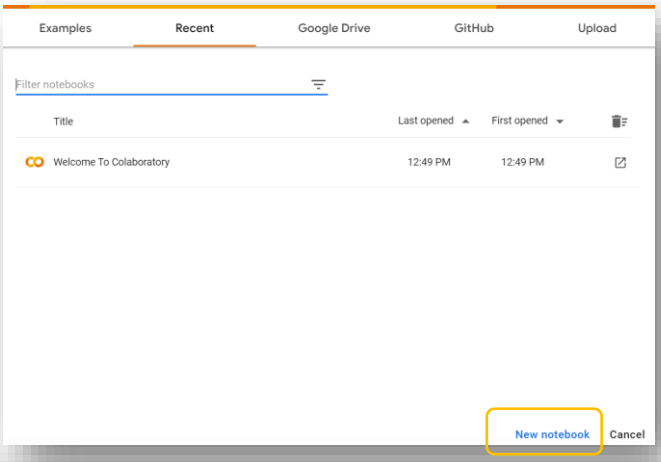
# ขั้นตอนการดึงข้อมูล

สำหรับขั้นตอนการดึงข้อมูลมีรายละเอียดดังนี้

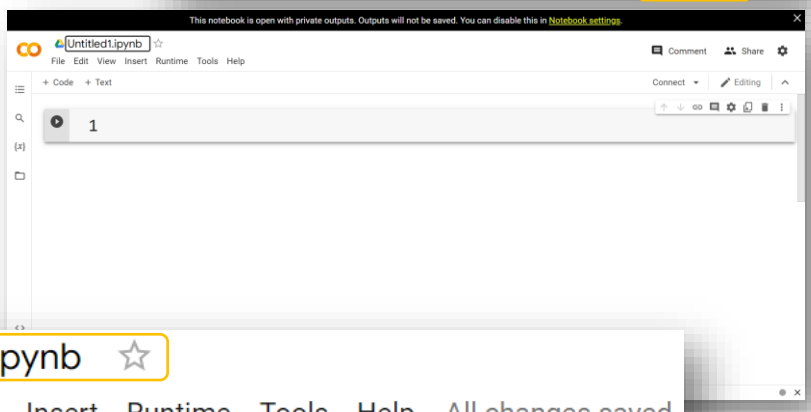
ดำเนินการร้องขอใช้บริการ Saas โฮสต์โปรแกรม Jupyter Notebook บน Cloud ของ Google ด้วยการ Log in บัญชีผู้ใช้งานของ Google จากนั้นดำเนินการเปิดลิงก์เว็บของ Colab ดังนี้

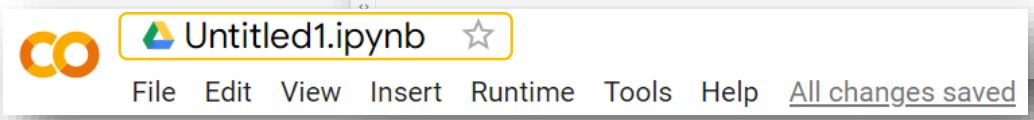
https://colab.research.google.com/

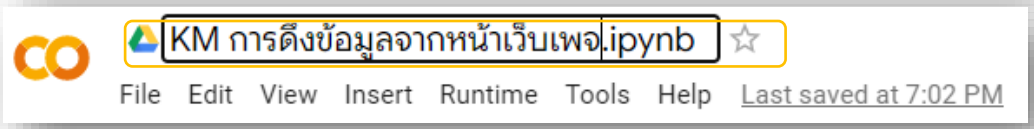
เมื่อปรากฏหน้าจอดังภาพนี้ ให้คลิกปุ่ม New notebook เพื่อสร้างสมุดงานใหม่ในการเขียนชุดคำสั่งดึงข้อมูล




เมื่อปรากฏหน้าจอดังภาพ ให้คลิกส่วนชื่อสมุดงาน แล้วดำเนินการแก้ไขชื่อสมุดงานเป็น KM การดึงข้อมูลจากหน้าเว็บเพจ.ipynb

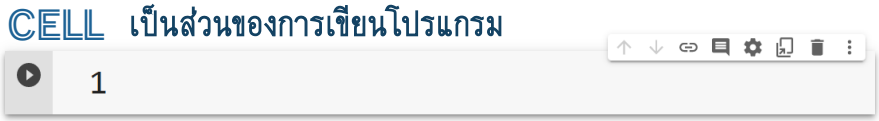


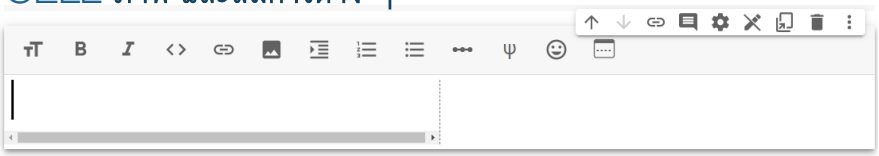




Colab มีส่วนสำคัญ 2 ส่วน ในการเขียนสมุดงาน คือ ส่วนของ Code cell และส่วนของ Text cell

CODE CELL เป็นส่วนของการเขียนโปรแกรม




TEXT CELL เป็นส่วนเขียนคำอธิบายสามารถแทรกข้อความ ภาพ และสมการต่าง ๆ



การดึงข้อมูลจากเว็บเพจ WEB SCRAPING

ผศ.วิวรรณ จันทนะทรัพย์
อาจารย์ผู้รับผิดชอบหลักสูตร วิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการข้อมูล
คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยเทคโนโลยีราชมงคลพระนคร

## 2 เขียนชุดคำสั่งดังต่อไปนี้ลงใน Code Cell โดยคำสั่งเป็นชุดคำสั่งภาษาไพธอน

```
import pandas as pd
import requests
from  bs4 import BeautifulSoup
```

เรียกใช้งานไลบรารีที่เกี่ยวข้อง จำนวน 3 ไลบรารี

---

```
weburl = 'https://sci.rmutp.ac.th/%e0%b8%9c%e0%b8%b9%e0%b9%89%e0%b8%9a%e0%b8%a3%e0%b8%b4%e0%b8%ab%e0%b8%b2%e0%b8%a3/'
```

กำหนดตัวแปรสำหรับลิงค์เว็บไซต์คณะ กรรมการบริหารคณะ วิทยาศาสตร์และเทคโนโลยี ผู้อ่านสามารถไปที่หน้าเว็บเพจคณะ กรรมการบริหารคณะฯ แล้วคัดลอกลิงค์มาวาง

---

```
r  = requests.get(weburl)
```

เชื่อมต่อเว็บลิงค์คณะ กรรมการบริหารคณะ วิทย์ฯ ด้วยคำสั่ง get ของไลบรารี requests

---

```
s = BeautifulSoup(r.text,'lxml')
```

อ่านข้อมูล html tag ด้วยไลบรารี BeautifulSoup จัดรูปแบบเป็น lxml

---

```
d = s.find('div',{'id': 'primary'})
```

ค้นหา HTML tag  เนื่องจากเป็นส่วนที่อยู่ของข้อมูลที่ต้องการดึง

---

```
tr_tags = d.find_all('tr')
```

ค้นหา HTML tag <tr> รายใน tag  เนื่องจากเป็นส่วนของที่อยู่ของข้อมูลที่ต้องการดึง

---

```
imglink  = []
htmllink = []
```

กำหนดลิสต์ว่างจำนวน 2 ตัวเพื่อเก็บตำแหน่งที่อยู่รูปภาพ และสร้างลิงค์ html สำหรับรูปภาพคณะ กรรมการฯ แต่ละราย

---

```
for e in tr_tags:
  img = e.find('img')
  src  = img['src']
  link = f'<img src="{src}" width="110" height="150">'
  imglink.append(src)
  htmllink.append(link)
```

ดำเนินการวนลูป เพื่อทำซ้ำเพื่อเก็บค่าลิงค์เว็บรูปภาพคณะ กรรมการฯ ทั้งหมด 14 ราย โดยเก็บไว้ในลิสต์ชื่อ imglink และ htmllink

---

# 2 เขียน Code กันต่อ

```
td_tags = d.find_all('td','column-2')
```

คำสั่งนี้ใช้ค้นหา HTML tag <td class="column-2"> เนื่องจากเป็นส่วนของข้อมูล 6 รายการที่ต้องการจัดเก็บ คือ ชื่อภาษาไทย ชื่อภาษาอังกฤษ ตำแหน่งภาษาไทย ตำแหน่งภาษาอังกฤษ ส่วนติดต่อหมายเลขโทรศัพท์ และที่อยู่จดหมายอิเล็กทรอนิกส์

```
nameTH = []
nameEN = []
posTH  = []
posEN  = []
tel    = []
email  = []
```

สร้างลิสต์ว่างจำนวน 6 ตัวเพื่อใช้เก็บข้อมูล
ชื่อภาษาไทย nameTH
ชื่อภาษาอังกฤษ nameEN
ตำแหน่งภาษาไทย posTH
ตำแหน่งภาษาอังกฤษ posEN
หมายเลขโทรศัพท์ติดต่อ tel และที่อยู่จดหมายอิเล็กทรอนิกส์ email

```
for i in td_tags:
  a = i.text
  data = a.split('\n')
  nameTH.append(data[0])
  nameEN.append(data[1])
  posTH.append(data[2])
  posEN.append(data[3])
  str = data[4]
  if (str[0:6])=="ติดต่อ":
    tel.append(data[4])
    email.append(data[5])
  else:
    tel.append('-')
    email.append(data[4])
```

เขียนคำสั่งวนลูป หรือทำซ้ำเพื่อเพิ่มข้อมูลที่ได้อ่านจาก HTML Tag จากหน้าเว็บคณะกรรมการบริหารคณะฯ เพิ่มเข้าลิสต์ข้อมูลครั้งละรายการ

อย่างไรก็ตามพบว่า ข้อมูลใน HTML Tag มีจำนวนรายการข้อมูลที่มีขนาดไม่เท่ากันดังภาพ

ข้อมูลรายละเอียดมี 6 รายการครบ

ข้อมูลรายละเอียดมี 5 รายการ ขาดรายการส่วนติดต่อ

จากปัญหาดังกล่าวผู้อ่านข้อมูลจะต้องใช้หัตถะการเขียนโปรแกรมเพื่อตรวจสอบเงื่อนไขว่าคณะกรรมการฯ รายใดไม่มีรายการข้อมูลส่วนติดต่อต้องข้าม โดยใส่ค่า Default ใด ๆ แทนลงไป

**ซึ่งประเด็นอย่างๆ เหล่านี้จะเรียงขั้นตอนปัญหาเฉพาะหน้าในแต่ละงานไป

## เขียน Code กันต่อ

```
rows=[]
N = 14
for i in range(N):
  rows.append((nameTH[i],
               nameEN[i],
               posTH[i],
               posEN[i],
               tel[i],
               email[i],
               imglink[i],
               htmllink[i]))
```

สิ่งขั้นตอนการสร้างข้อมูลที่ได้ดึงมาแล้วเก็บในตัวแปร
มาสร้างเป็นรายการข้อมูลจำนวน 8 คอลัมน์
แล้วเพิ่มลงในแปร rows คัน 8 คอลัมน์ประกอบด้วย

1. ชื่อภาษาไทย
2. ชื่อภาษาอังกฤษ
3. ตำแหน่งภาษาไทย
4. ตำแหน่งภาษาอังกฤษ
5. ส่วนติดต่อโทรศัพท์
6. ที่อยู่อีเมลอิเล็กทรอนิกส์
7. ลิงรูปภาพคณะกรรมการฯ
8. ลิงค์ Html รูปคณะกรรมการฯ

โดยดำเนินการเพิ่มจำนวน 14 ครั้งนั่นคือ
ข้อมูลคณะกรรมการบริหารคณะจำนวน 14 ราย

---

```
#สร้าง DataFrame ชื่อ df
df = pd.DataFrame(rows,
                  columns=['NameTH',
                           'NameEN',
                           'PosTH',
                           'PosEN',
                           'Tel',
                           'Email',
                           'ImageLink',
                           'HTMLlink'])
```

ดำเนินการสร้าง DataFrame ด้วยไลบรารี Pandas ชื่อ df โดยกำหนดคอลัมน์ข้อมูลเท่ากับจำนวนคอลัมน์ที่เก็บอยู่ในตัวแปร rows ข้างต้น เพื่อแทนค่าข้อมูลในตัวแปร rows มายังตัวแปร df

```
#จัดเก็บไฟล์
df.to_csv('MGteamSciRMUTP.csv',
          encoding='utf-8',index = True)
```

ดำเนินการนำข้อมูลจากตัวแปร df มาทำการบันทึกข้อมูลในรูปแบบ csv File ด้วยคำสั่ง df.to_csv กำหนดชื่อไฟล์ 'MGteamSciRMUTP.csv' เข้ารหัสข้อความภาษาไทยด้วยรหัส 'utf-8' และกำหนดให้สร้างดัชนีแนวแลวและคอลัมน์

---

**สำหรับผู้อ่านที่ต้องการ**
**สมุดงานนี้สามารถดาวน์โหลด**
**ได้จากลิงก์นี้ หรือ สแกน QRcode**

KM_WEB SCRAPING

https://colab.research.google.com/drive/1QRnLhOyeKOxtlf3Zog6YW9lToKEUQtfa?usp=sharing

## 3 ดำเนินการตรวจสอบไฟล์ผลลัพธ์ว่าได้ไฟล์ 'MGteamSciRMUTP.csv หรือไม่ มาลองอ่านไฟล์ แล้วแสดงผลกัน

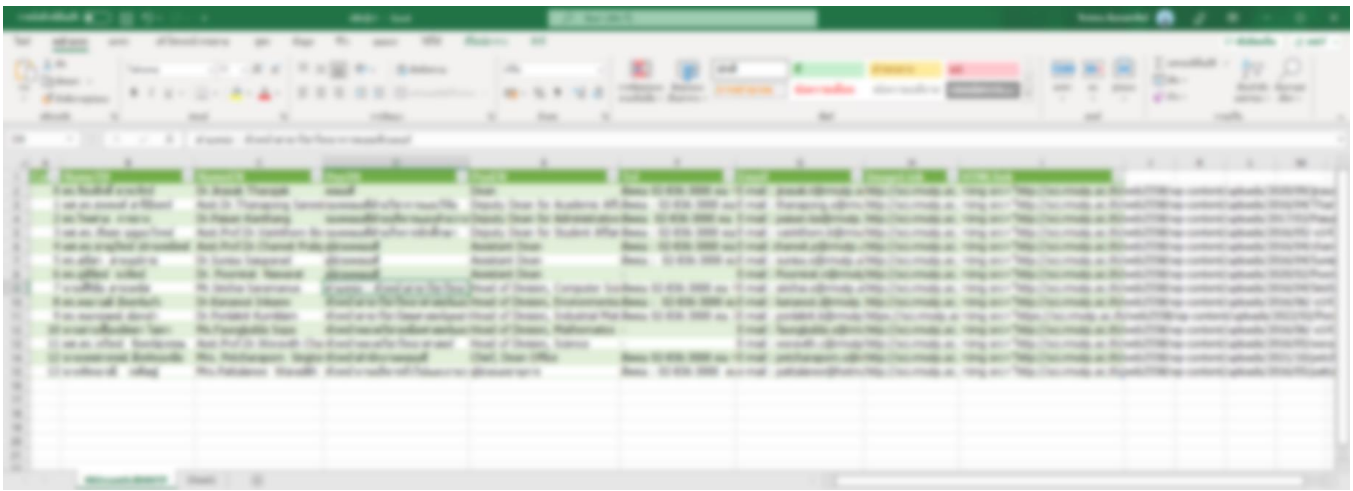
```
MGscidf = pd.read_csv('MGteamSciRMUTP.csv')
from IPython.display import HTML
HTML(MGscidf.to_html(escape=False))
```

เขียนคำสั่งเพื่ออ่านข้อมูลจาก CSV File จากนั้นดำเนินการแสดงผลข้อมูลด้วยเครื่องมือ IPython ผลลัพธ์ประกฏผลลัพธ์ดังรูปภาพนี้

หรืออาจเขียนคำสั่งเพื่อดาวน์โหลดไฟล์เพื่อนำไปใช้งานต่อตามวัตถุประสงค์ต่าง ๆ เขียนคำสั่งได้ดังนี้

```
from google.colab import files
files.download('/content/MGteamSciRMUTP.csv')
print('Finish Download')
```

เมื่อดาวน์โหลดเสร็จสิ้น ลองทดสอบด้วยการเปิดด้วยซอฟต์แวร์ Excel ก็จะแสดงข้อมูลดังรูปภาพ



4 สำหรับในส่วนงานการเขียนคำสั่งเพื่อบันทึกไฟล์ภาพคณะกรรมการฯ สามารถเขียนคำสั่งได้ดังนี้

```
from skimage import io
from google.colab.patches import cv2_imshow
import cv2 as cv2

linkimg = MGscidf['ImageLink']

i=1
for urlimg in linkimg :
  img    = io.imread(urlimg)
  imgnew = cv2.resize(img,[180,250])
  imgRGB = cv2.cvtColor(imgnew, cv2.COLOR_BGR2RGB)
  cv2_imshow(imgRGB)
  cv2.imwrite('MGsci_'+str(i)+'.jpg',imgRGB)
  print('Save File :'+'MGsci_'+str(i)+'.jpg')
  i+=1
```

ในส่วนคำสั่งอ่านไฟล์รูปภาพและบันทึกไฟล์รูปภาพคณะกรรมการบริหารคณะฯ ทั้ง 14 ราย เราจะใช้ไลบรารีสำหรับอ่านรูปภาพ และประมวลผลรูปภาพดิจิทัลเข้ามาช่วยอีก 2 ไลบรารี คือ

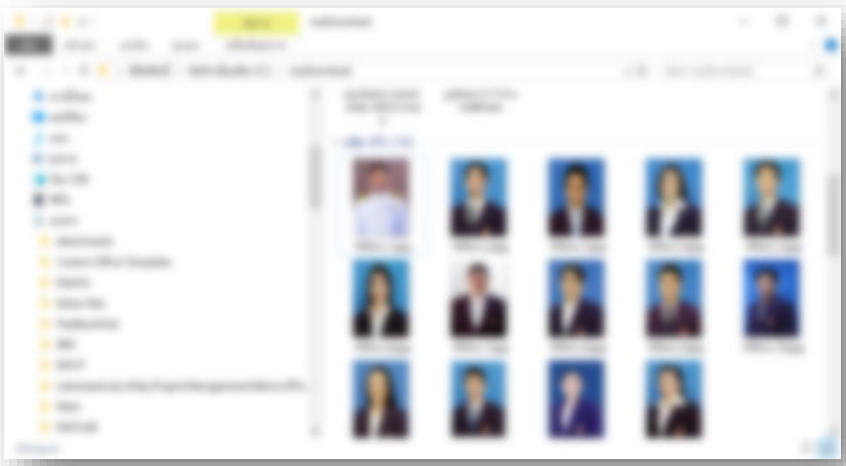
1. Skimage Library
2. Open Cv Library

ดำเนินการอ่านรูปภาพจากข้อมูลลิงค์รูปภาพในคอลัมน์ ImageLink จากตัวแปร Data frame ชื่อ MGscidf ข้างต้น โดยดำเนินการอ่าน และเขียนไฟล์ซ้ำจำนวน 14 ครั้ง โดยแต่ละครั้งเมื่ออ่านรูปภาพจากลิงค์ด้วยคำสั่ง `io.imread(urlimg)` เราจะดำเนินการปรับปรุงขนาดรูปภาพให้มีขนาดที่เท่ากันคือ 180x250px ด้วยคำสั่ง `cv2.resize` จากนั้นดำเนินการแปลงสีรูปภาพด้วยคำสั่ง `cv2.COLOR_BGR2RGB` และบันทึกไฟล์รูปภาพด้วยคำสั่ง `cv2.imwrite` โดยกำหนดชื่อรูปภาพ MGsci_1.jpg, MGsci_2.jpg, MGsci_3.jpg, ... จนครบ MGsci_14.jpg

ต่อจากนั้นให้เขียนคำสั่งเพื่อดาวน์โหลดภาพทั้ง 14 เก็บลงเครื่องคอมพิวเตอร์

```
from google.colab import files
for i in range(1,14+1):
  files.download('/content/MGsci_'+str(i)+".jpg")
  print('dowload file MGsci_'+str(i)+'.jpg Finish')
print('Finish Download')
```

ผลลัพธ์จากการประมวลผลคำสั่งข้างต้น จะได้ผลลัพธ์คือไฟล์รูปภาพคณะกรรมการบริหารคณะทั้ง 14 ราย ถูกดาวน์โหลดเข้าสู่โฟลเดอร์ที่กำหนดบนเครื่องคอมพิวเตอร์ของผู้อ่าน จำนวน 14 ไฟล์ ดังรูปภาพ



**สรุปเนื้อหาจากผู้เขียน:**

จากเนื้อหาทั้งหมดของ KM การดึงข้อมูลจากเว็บเพจ (WEB SCRAPING) โดยใช้กรณีศึกษาหน้าเว็บเพจคณะกรรมการบริหารคณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชมงคลพระนคร ได้อธิบายถึงขั้นตอนวิธีการดึงข้อมูลจาก HTML Tag โดยใช้เครื่องมือภาษาคอมพิวเตอร์ คือ ภาษาไพธอนที่ทำงานร่วมกับไลบรารีที่เกี่ยวข้องประกอบด้วย Request Library สำหรับอ่านข้อมูลจากเว็บลิงก์ Beautifulfoup Library สำหรับจัดรูปแบบ HTML Tag เพื่อให้ง่ายต่อการดึงข้อมูลรวมถึง Pandas Library สำหรับจัดรูปแบบข้อมูลที่ดึงมาให้อยู่ในรูปแบบ Data Frame และง่ายต่อการจัดเก็บในรูปแบบไฟล์ข้อมูล CSV หรือ XLSX นอกจากนั้นยังมี OpenCV Library สำหรับประมวลผลภาพของคณะกรรมการบริหารของเนื้อหาในส่วนสุดท้าย

อย่างไรก็ตามเนื้อหาขององค์ความรู้นี้ เป็นเพียงพื้นฐานของเทคนิคการดึงข้อมูลจากหน้าเว็บประเภท Static Website กล่าวคือ ข้อมูลไม่เปลี่ยนแปลงไปตามเงื่อนไข หรือพารามิเตอร์ที่ผู้ใช้งานหรือโปรแกรมกำหนด แต่สำหรับเว็บประเภท เว็บเพจพลวัต หรือ ไดนามิกเว็บเพจ (dynamic web page) นั้นความสามารถของ Beautifulfoup Library จะไม่สามารถดึงข้อมูลได้ ซึ่งกรณีเป็นการดึงข้อมูลจากไดนามิกเว็บอาจใช้ความสามารถของฟังก์ชัน หรือคำสั่งจาก Selenium ซึ่งใน KM เรื่องต่อไปผู้เขียนจะมาถ่ายทอดความรู้เกี่ยวกับการดึงข้อมูลจากไดนามิกเว็บเพจ ด้วย Selenium กันคะ

:: ผู้เขียนหวังเป็นอย่างยิ่งว่า ผู้อ่านองค์ความรู้ฉบับนี้จะได้รับความรู้ และนำความรู้ไปใช้ประโยชน์ต่อการศึกษาเล่าเรียน หรือใช้ในการทำงานต่อไป และขอขอบคุณข้อมูลจากเว็บเพจคณะกรรมการบริหารคณะฯ และท้ายนี้ผู้เขียนขอขอบคุณคณะกรรมการบริหารคณะฯ ทุกท่านที่ผู้เขียนได้นำข้อมูลมาใช้เป็นกรณีศึกษา อันเป็นการสนับสนุนการเรียนรู้ของผู้อ่าน โดยคุณประโยชน์ที่เกิดจากองค์ความรู้นี้ ผู้เขียนขอมอบแด่ครูอาจารย์ หน่วยงานกรณีศึกษา และบุคคลที่เป็นเจ้าของข้อมูลทุกท่าน