

# นาย กิตติภณ สรุ่งเรืองสกุล 61070278

## Assignment 1

```
In [14]: #!pip install urllib requests  
#ລັງ urllib ແລະ requests ກ່ອນດ້າວື່ນໃນນີ້  
import json, urllib  
import pandas as pd  
import numpy as np
```

### ดัชนีราคาผู้บริโภคทั่วไป (Consumer Price Index)

ວິເຄາະທີ່ດັชนີ່ราคาຜູ້ບໍ່ຮົມທຳກຳທີ່ມີສະແດງໃນກຸງເຖິງແຫຼ່ງທີ່ມີສະແດງໃນກຸງເຖິງພິບປະເທດ 2017-2020 ແລະ ໃນຍຸດໂຄວິດ-19

ແທລ່ງຂ່ອງນຸລ: <https://data.moc.go.th/OpenData/CPIIndexes>

#### Query Parameters

Name	Data type	Description	Default
region_id	Integer	ລັດສຸກມີກັດ (0 = ກຽງເທັກແລະປ່ຽນແຫຼ່ງ, 1 = ກາດກລາງ, 2 = ກາດຕະວັນອອກເຈີຍເໜືອ, 3 = ກາດເທົ່ານີ້, 4 = ກາດໄດ້, 5 = ປະເທດ)	-
index_id	String	ຮັດສົມວັດ / ກລຸມດັບນີ້	000000000000000000
from_year	Integer	ປີທີ່ຈົດທ່າດ້ານນີ້ເລີ່ມຕົ້ນ (ຄ.ສ.)	-
to_year	Integer	ປີທີ່ຈົດທ່າດ້ານນີ້ສິ້ນສຸດ (ຄ.ສ.)	-

Link API: <https://dataapi.moc.go.th/cpig-indexes>

Example: [https://dataapi.moc.go.th/cpig-indexes?region\\_id=1&index\\_id=10000000000000000000&from\\_year=2017&to\\_year=2020](https://dataapi.moc.go.th/cpig-indexes?region_id=1&index_id=10000000000000000000&from_year=2017&to_year=2020)

- គັນທາງຂ່ອງນຸລໃນ ກາດກລາງ
- ຮັດສົມວັດທີ່ 10000000000000000000
- ມີທີ່ຈົດທ່າດ້ານນີ້ເລີ່ມຕົ້ນທີ່ປີ 2017
- ມີທີ່ຈົດທ່າດ້ານນີ້ສິ້ນສຸດທີ່ປີ 2020

#### Responses

Type: Array

- ຜລລັບຮັບຈະມີກາຣເຮັງລໍາດັບໂດຍປີທີ່ຈົດທ່າດ້ານນີ້ ດາມຕ່າຍເຕືອນທີ່ຈົດທ່າດ້ານນີ້ ດາມຕ່າຍຮ້າສຸກມີກັດ

Name	Data type	Description
index_id	String	ຮັດສົມວັດ / ກລຸມດັບນີ້
index_description	String	ຊື່ອໝາວດ / ກລຸມດັບນີ້
region_id	Integer	ລັດສຸກມີກັດ
region_name	String	ຊື່ອຸກມີກັດ
base_year	Integer	ປີສ້າງຕົ້ນນີ້
year	Integer	ປີທີ່ຈົດທ່າດ້ານນີ້
month	Integer	ເດືອນທີ່ຈົດທ່າດ້ານນີ້
price_index	Float	ດັບນີ້ຄາ
mon	Float	ອັດຮາກເປົ້າຍັນແປລັງເທືນກົນຕົວກອນທຳນາ
yoy	Float	ອັດຮາກເປົ້າຍັນແປລັງເທືນກົນເຕືອນເຕີຍກົນນີ້ກອນທຳນາ
aoa	Float	ອັດຮາກເປົ້າຍັນແປລັງເທືນກົນນ່ວງເຕີຍກົນນີ້ກອນທຳນາ

```
[  
  {  
    "index_id": "10000000000000000000",  
    "index_description": "ໜມວັດອາຫານແລະເຄື່ອງດືມໄມ້ມີແອລກອອສອລ",  
    "region_id": 1,  
    "region_name": "ກາດກລາງ",  
    "base_year": 2015,  
    "year": 2017,  
    "month": 1,  
    "price_index": 101.53,  
    "mon": 0.4,  
    "yoy": 1.61,  
    "aoa": 1.61  
  },  
]
```

## รหัสหมวด/กลุ่มตัวชี้ (Index\_id)

แสดงรหัสหมวด/กลุ่มตัวชี้ (Index\_id) ที่ใช้ในลิงก์ API ค่าไม่ใช่เป็นค่าเริ่มต้นคือ 0000000000000000

```
In [101]: index_id_df = pd.read_csv('data/รหัสหมวดตัวชี้ CPI_INDEX 31-8-2020.csv', dtype={'รหัสตัวชี้': np.int64})  
index_id_df
```

Out[101]:

รหัสตัวชี้	ชื่อหมวด/กลุ่มตัวชี้
0	รวมทุกรายการ
1	หมวดอาหารและเครื่องดื่มน้ำมันและก่ออํส
2	อาหารปรุงสำปัน
3	ข้าว แป้งและผลิตภัณฑ์จากแป้ง
4	ข้าว
...	...
185	น้ำมัน 91 95 ตีเซล
186	CPI ในรวมน้ำมัน 91 95 ตีเซล
187	CPI ในรวมน้ำมัน 91 95 ตีเซล และรายการของ คน.
188	CPI ในรวมอาหารสด
189	CPI ในรวมพลังงาน

190 rows × 2 columns

## Data collection and storage

### ดึงข้อมูลจากลิงก์ API

อ่านข้อมูลรหัสหมวด/กลุ่มตัวชี้ (Index\_id) จากไฟล์ รหัสหมวดตัวชี้ CPI\_INDEX 31-8-2020.csv ที่เก็บไว้ในโฟลเดอร์ data

ข้อมูลที่อ่านมาจะเก็บไว้ในตัวแปร **CPI\_Index** แต่ค่าที่ต้องการใช้มีแค่รหัสหมวด/กลุ่มตัวชี้ (Index\_id) จึงทำการเลือกเฉพาะคอลัมน์รหัสตัวชี้แล้วเก็บไว้ในตัวแปร **CPI\_index\_id**

ข้อมูลที่เก็บไว้ในตัวแปร **CPI\_index\_id** อยู่ในรูปแบบของ pandas.Series ซึ่งไม่สะดวกต่อการใช้งานจึงทำการเปลี่ยนให้อยู่ในรูปแบบของ list โดยเก็บข้อมูลไว้ในตัวแปร **CPI\_index\_id\_list**

```
In [3]: CPI_Index = pd.read_csv("data/รหัสหมวดตัวชี้ CPI_INDEX 31-8-2020.csv", dtype={'รหัสตัวชี้': np.int64})  
CPI_index_id = CPI_Index['รหัสตัวชี้']  
CPI_index_id_list = CPI_index_id.tolist()  
print('Sample index_id:', CPI_index_id_list[:3])
```

Sample index\_id: [0, 1000000000000000, 1100000000000000]

สร้างตัวแปร data เพื่อรอเก็บข้อมูลที่อ่านเข้ามา

```
In [4]: data = []  
  
def addData(lists): #รับข้อมูลที่อยู่ในรูปแบบของ List  
    for i in range(len(lists)): #อ่านข้อมูลตามความยาวของ List นั้น  
        data.append(lists[i]) #เอาข้อมูลใน List แต่ละตัวซึ่งอยู่ในรูปของ dict ไปเก็บไว้ในตัวแปร data
```

ในงานนี้จะรีเคราะห์เฉพาะตัวชี้นิรภัยริก็ท์ว่าไปในกรุงเทพและปริมณฑล ตั้งแต่ปีค.ศ. 2017-2020 เลยทำการสร้างและกำหนดค่าตัวแปร

- **region\_id** เท่ากับ 0 (กรุงเทพและปริมณฑล)
- **from\_year** เท่ากับ 2017 (เริ่มอ่านข้อมูลตั้งแต่ปีค.ศ. 2017)
- **to\_year** เท่ากับ 2020 (อ่านข้อมูลถึงปีค.ศ. 2020)

ตัวต้องการดึงข้อมูลในกฎีภาค มีเรื่องคันและปีสุดท้ายอัน ๆ สามารถแก้ไขค่าที่ตัวแปรได้เลย

เช่นจากข้อมูลที่ต้องการดึงมีรายละเอียดต้องทำการสร้างสิ่ง API แล้วค่อย ๆ ตั้งที่ล่ะรอบ

โดยเริ่มจากลูปในตัวแปร **CPI\_index\_id\_list** ที่เก็บรหัสหมวด/กลุ่มตัวชี้ (Index\_id) โดย i ในแต่ละรอบคือรหัสหมวดที่ต้องใส่ไปในสิ่งที่ต้องการสร้างตัวแปร url เพื่อใช้เก็บลิงก์ที่จะขอข้อมูลในแต่ละรอบ จากนั้นก็เอาระบบ parameters ต่าง ๆ ใส่เข้าไป

โดยต้องใส่โคดส่วนนี้ str(i) + ('0' \* (16 - len(str(i)))) เพราะว่า index\_id ที่ใช้ในสิ่งที่ต้องมีทั้งหมด 16 หลักและเลข 0 ต้องหายากในคราว 16 หลัก (0000000000000000) แต่ข้อมูลใน **CPI\_index\_id\_list** บางตัวมีไม่ถึง 16 หลัก เช่น 0

- โคดในส่วนนี้ str(i) จึงทำการแปลงข้อมูลเป็น string เพื่อร่วมกับส่วนด้านหลัง
- โคดในส่วนนี้ ('0' \* (16 - len(str(i)))) จะทำการพับความยาวของ i ซึ่งต้อง index\_id ที่วน回来ในแต่ละรอบ จึงทำการแปลงข้อมูล string แล้วค่อยทำการนับความยาวของ i และป่าว 16 - ความยาวของ i เพื่อนำไปปะกับ '0' เพื่อสร้าง 0 ที่เหลือมาต่อท้ายกับโคดในส่วนด้านหน้าเพื่อให้ index\_id ครบ 16 หลัก

Example รอบที่ 1-3:

1. [https://dataapi.moc.go.th/cpig-indexes?region\\_id=0&index\\_id=0000000000000000&from\\_year=2017&to\\_year=2020](https://dataapi.moc.go.th/cpig-indexes?region_id=0&index_id=0000000000000000&from_year=2017&to_year=2020)
2. [https://dataapi.moc.go.th/cpig-indexes?region\\_id=0&index\\_id=1000000000000000&from\\_year=2017&to\\_year=2020](https://dataapi.moc.go.th/cpig-indexes?region_id=0&index_id=1000000000000000&from_year=2017&to_year=2020)
3. [https://dataapi.moc.go.th/cpig-indexes?region\\_id=0&index\\_id=1100000000000000&from\\_year=2017&to\\_year=2020](https://dataapi.moc.go.th/cpig-indexes?region_id=0&index_id=1100000000000000&from_year=2017&to_year=2020)

```
In [7]: region_id = "0"  
from_year="2017"
```

```

to_year="2020"

# url = 'https://dataapi.moc.go.th/cpig-indexes?region_id=0&index_id=10000000000000000000&from_year=2017&t_o_year=2020'

for i in CPI_index_id_list:
    url = "https://dataapi.moc.go.th/cpig-indexes?region_id=" + region_id + "&index_id=" + str(i) + ('0' * (16 - len(str(i)))) + "&from_year=" + from_year + "&to_year=" + to_year

    response = urllib.request.urlopen(url) #ส่ง request ขอข้อมูล
    raw_json = response.read().decode("utf-8") #อ่านข้อมูลที่รับกลับมา ข้อมูลมีลักษณะเป็นภาษาไทยเลยต้อง decode ด้วย utf-8
    addData(json.loads(raw_json)) #Loads json ที่อ่านค่าได้ซึ่งมีอ่านแล้วจะอยู่ในรูป [{key:value}] ส่งค่าที่อ่านได้ที่อยู่ในรูปของ list ไปที่ function addData() เพื่อเอาข้อมูลมาเก็บไว้

print(len(data))

```

7748

แปลงจากรูปแบบ json เป็น dataframe

In [8]: df = pd.json\_normalize(data) #นำตัวแปร data ที่เก็บข้อมูลทั้งหมดไว้มาแปลงให้อยู่ในรูปของ dataframe  
df

Out[8]:

	index_id	index_description	region_id	region_name	base_year	year	month	price_index	mon	yoy	aoa
0	0000000000000000	รวมทุกรายการ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	1	100.53	0.07	1.18	1.18
1	0000000000000000	รวมทุกรายการ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	2	100.53	0.00	1.00	1.09
2	0000000000000000	รวมทุกรายการ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	3	100.18	-0.35	0.49	0.88
3	0000000000000000	รวมทุกรายการ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	4	100.34	0.16	0.18	0.71
4	0000000000000000	รวมทุกรายการ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	5	100.48	0.14	-0.04	0.56
...	...	...	...	...	...	...	...	...	...	...	...
7743	9520000000000000	CPI ไม่รวมน้ำมัน 91, 95, ดีเซล	0	กรุงเทพมหานคร และปริมณฑล	2015	2020	8	102.67	0.23	-0.21	-0.69
7744	9700000000000000	CPI ไม่รวมอาหารสด	0	กรุงเทพมหานคร และปริมณฑล	2015	2019	11	102.45	0.04	0.19	0.39
7745	9700000000000000	CPI ไม่รวมอาหารสด	0	กรุงเทพมหานคร และปริมณฑล	2015	2019	12	102.48	0.03	0.67	0.41
7746	9800000000000000	CPI ไม่รวมพลังงาน	0	กรุงเทพมหานคร และปริมณฑล	2015	2019	11	103.05	-0.13	0.84	1.27
7747	9800000000000000	CPI ไม่รวมพลังงาน	0	กรุงเทพมหานคร และปริมณฑล	2015	2019	12	103.04	-0.01	0.99	1.25

7748 rows × 11 columns

## แปลงข้อมูลจาก DataFrame ไปเป็น .csv

เก็บข้อมูลที่ตั้งมาไว้ในชื่อ data.csv

In [ ]: #เก็บข้อมูลในรูปแบบไฟล์ csv โดยเก็บไว้ในชื่อว่า data.csv และต้องศั้งค่า encoding เป็น utf-8-sig เพราะมีข้อมูลที่เป็นภาษาไทย  
df.to\_csv('data/data.csv', header=True, index=False, encoding='utf-8-sig')

## Data pre-processing and analysis

อ่านข้อมูลที่เก็บไว้ในชื่อ data.csv เพื่อนำมาทำความสะอาดก่อนนำไปใช้ในปัจจุบัน

In [102]: #อ่านข้อมูล data.csv แล้วเก็บไว้ในตัวแปร df เพื่อใช้ในการทำ pre-processing  
df = pd.read\_csv('data/data.csv')  
df

Out[102]:

	index_id	index_description	region_id	region_name	base_year	year	month	price_index	mon	yoy	aoa
0	0	รวมทุกรายการ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	1	100.53	0.07	1.18	1.18
1	0	รวมทุกรายการ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	2	100.53	0.00	1.00	1.09
2	0	รวมทุกรายการ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	3	100.18	-0.35	0.49	0.88
3	0	รวมทุกรายการ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	4	100.34	0.16	0.18	0.71
4	0	รวมทุกรายการ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	5	100.48	0.14	-0.04	0.56
...	...	...	...	...	...	...	...	...	...	...	...
7743	9520000000000000	CPI ไม่รวมน้ำมัน 91, 95, ดีเซล	0	กรุงเทพมหานคร และปริมณฑล	2015	2020	8	102.67	0.23	-0.21	-0.69
7744	9700000000000000	CPI ไม่รวมอาหารสด	0	กรุงเทพมหานคร และปริมณฑล	2015	2019	11	102.45	0.04	0.19	0.39

				และปริมาณเทล								
7745	970000000000000000000000	CPI ไม่รวมอาหารสด	0	กรุงเทพมหานคร และปริมณฑล	2015	2019	12	102.48	0.03	0.67	0.41	
7746	980000000000000000000000	CPI ไม่รวมพลังงาน	0	กรุงเทพมหานคร และปริมณฑล	2015	2019	11	103.05	-0.13	0.84	1.27	
7747	980000000000000000000000	CPI ไม่รวมพลังงาน	0	กรุงเทพมหานคร และปริมณฑล	2015	2019	12	103.04	-0.01	0.99	1.25	

7748 rows × 11 columns

หากค่า missing value โดยใช้คำสั่ง `isna()` ในการดูว่าเป็นค่า missing value หรือไม่ ถ้าเป็นจะชี้ว่า True แต่ถ้าไม่เป็นจะชี้ว่า False และใช้คำสั่ง `sum()` ในการนับจำนวน missing value ทั้งหมด

In [103]: `df.isna().sum()`

```
Out[103]: index_id      0
index_description  0
region_id        0
region_name       0
base_year         0
year              0
month             0
price_index      0
mon               0
yoy                0
aoa               0
dtype: int64
```

หลังจากตรวจสอบแล้วข้อมูลที่ดึงมาไม่มีค่า missing value

ทำการ `groupby` และที่สำคัญ `index_description` เพื่อกันแสวงหาการนับจำนวนแคล ข้อมูลที่ได้จะอยู่ในรูปของ pandas.Series และเก็บไว้ในตัวแปร `countGroup`

เนื่องจากตัวแปร `countGroup` ค่าน้ำหนักมาก เลยแปลงให้อยู่ในรูปของ dict เพื่อที่จะได้เห็นชัดเจนว่า `index_description` แต่ละตัวมีจำนวนเท่าไรและเก็บไว้ในตัวแปร `dictCountGroup`

เข้าด้วยตัวแปร `dictCountGroup` มาเรียงลำดับจากน้อยไปมากโดยเรียงจาก value ซึ่งคือจำนวนแคลทั้งหมด เพื่อให้ง่ายต่อการวิเคราะห์

In [104]: `countGroup = df['index_description'].groupby(df['index_description']).count()
dictCountGroup = countGroup.to_dict()
{k: v for k, v in sorted(dictCountGroup.items(), key=lambda item: item[1])}`

```
Out[104]: {'CPI ไม่รวมพลังงาน': 2,
'CPI ไม่รวมอาหารสด': 2,
'CPI ไม่วน้ำมัน 91 , 95 , ตีเซล': 44,
'กลุ่มอาหารสดและพลังงาน': 44,
'กาแฟบุหรี่': 44,
'กาแฟสดและกระป๋องสด': 44,
'กาแฟสด': 44,
'กาแฟบด': 44,
'กาแฟบดและกาแฟอ่อน': 44,
'กาแฟ': 44,
'การศึกษา': 44,
'การศึกษาระดับมัธยมศึกษา ดำเนินการนิดและอาชีวะ': 44,
'การศึกษาระดับมัธยมศึกษาสามัญ': 44,
'การศึกษาระดับอุดมศึกษา': 44,
'การสื่อสาร': 44,
'ของเล่น': 44,
'ข้าว': 44,
'ข้าว แบ่งและผลิตภัณฑ์จากเมือง': 44,
'คนรับใช้ / คนงาน': 44,
'คนใช้ใน': 44,
'คนใช้': 44,
'ค้าการบันเทิง': 44,
'ค้าของใช้จำพวกคุกคัก': 44,
'ค้าจ้าง': 44,
'ค้าจ้างตัดเย็บและอุปกรณ์': 44,
'ค้าเชื้อราพานา': 44,
'ค้าซ้อมแขวน': 44,
'ค้าข้อมูลและอุปกรณ์': 44,
'ค้าจราจรรักษารถทางและค่าใช้': 44,
'ค้าจราจรที่จราจรทางท้องถนนปัจจัยภัยมีการ': 44,
'ค้าจราจรสายตา': 44,
'ค้าจราจรโทรรักษารถทางและค่าบริการ แหล่งรายได้': 44,
'ค้าทำฟัน': 44,
'ค้าที่พักอาศัย': 44,
'ค้าธรรมเนียมและค่าสมนาคุณ': 44,
'ค้านิการการสื่อสาร': 44,
'ค้านิการดำเนินรัฐสิทธิ์': 44,
'ค้านิการรัฐบาลส่วนบุคคล': 44,
'ค้านิการรัฐบาลส่วนบุคคลเพื่อความผ่านคลาย': 44,
'ค้านิการรัฐบาลส่วนบุคคลเพื่อความสวยงาม': 44,
'ค้านิการรัฐบาลสุขภาพและประภากันชีวิต': 44,
'ค้าประทับน้ำ': 44,
'ค้าภาษี': 44,
'ค้าภาษีและประภากันภัยภัยด้วย': 44,
'ค้ายา': 44,
'ค้ายาและเวชภัณฑ์': 44,
'ค้าอักษรพยานนาชาติ': 44,
'ค้าอักษรพยานนาชาติ': 44,
'ค้าอุปกรณ์การบันเทิง': 44,
```



```
'เครื่องแบบเด็ก': 44,
'เครื่องใช้บ้านเดลล์ในปีน': 44,
'เชื้อเพลิงในปีน': 44,
'เด็ก': 44,
'เบี้ยสัตว์': 44,
'เบี้ยสัตว์ เปิดໄก และสัตว์น้ำ': 44,
'เบี้ยสัตว์ดู': 44,
'เบี้ยสัตว์แม่รูป': 44,
'เป็น ใจ': 44,
'เปิด ใจ สต': 44,
'เปิด ใจ สำเร็จรูปและประรูป': 44,
'เวชภัณฑ์': 44,
'เลือบธุรกษา': 44,
'เลือบผ้าผูกอัน': 44,
'เลือบผ้าสารอิฐ': 44,
'เลือบผ้าตีกอิฐ': 44,
'เลี้ยงสัตว์': 44,
'เลือบเด็ก': 44,
'แนะนำและผลิตภัณฑ์จากเมือง': 44,
'แสงสว่าง': 44,
'ไข': 44,
'ไขและผลิตภัณฑ์': 44,
'ไฟฟ้า': 44,
'ไฟฟ้า เชื้อเพลิง น้ำประปาและแสงสว่าง': 44,
'ไม่รวมอาหาร เครื่องดื่ม และพลังงาน': 44,
'ค่าบริการอื่น ๆ': 88}
```

หลังจากทำการรับแล้วจะเห็นได้ว่าหมวดส่วนใหญ่จะมีแค่หัวหนอด 44 แต่ โดย 44 แ苦笑มาจากปีค.ศ. 2017-2020 โดยปีค.ศ. 2017-2019 แต่ละปีจะมีหัวหนอด 12 แต่ แต่ปีค.ศ. 2020 ยังไม่จบมีข้อมูลถึงแค่เดือนสิงหาคมจึงทำให้มีหัวหนอด 8 แต่ เมื่อนำมารวมกันแล้ว ( $12+12+12+8$ ) ซึ่งมีหัวหนอด 44 แต่ ( $1 \text{ แต่ } = 1 \text{ เม็ด }$ )

แต่จะมี 3 หมวดที่แตกต่างจากหมวดอื่น ๆ ได้แก่ **CPI ไม่รวมพลังงาน**, **CPI ไม่รวมอาหารสด** มีหัวหนอด 2 แต่และ **ค่าบริการอื่น ๆ** มีหัวหนอด 88 แต่

ลองแค่ที่มี `index_description` เท่ากับ **CPI ไม่รวมพลังงาน** และ **CPI ไม่รวมอาหารสด** เพราะมีข้อมูลมอยู่ไป เป็นองจากการนี้จะเคราะห์ดูมีรายการสุนทรีย์ที่นำไปในกรุงเทพและปริมทุล ตั้งแต่ปีค.ศ. 2017-2020 ซึ่งต้องมีข้อมูลหัวหนอดทั้งหมด 44 แต่ แต่ 2 หมวดดังกล่าวมีข้อมูลในห้องต่อลงลับทั้ง

```
In [105]: df = df[df.index_description != 'CPI ไม่รวมพลังงาน']
df = df[df.index_description != 'CPI ไม่รวมอาหารสด']
```

ตรวจสอบแค่ที่มี `index_description` เท่ากับ **ค่าบริการอื่น ๆ** เพราะมีจำนวนแค่ที่แตกต่างจากข้อมูลต่ออื่น ๆ

```
In [106]: df[df.index_description == 'ค่าบริการอื่น ๆ']
```

Out[106]:

index_id	index_description	region_id	region_name	base_year	year	month	price_index	mon	yoy	aoa
4576	4121500000000000	ค่าบริการอื่น ๆ	0 กรุงเทพมหานคร และปริมทุล	2015	2017	1	102.08	0.0	1.46	1.46
4577	4121500000000000	ค่าบริการอื่น ๆ	0 กรุงเทพมหานคร และปริมทุล	2015	2017	2	102.08	0.0	1.46	1.46
4578	4121500000000000	ค่าบริการอื่น ๆ	0 กรุงเทพมหานคร และปริมทุล	2015	2017	3	102.08	0.0	1.46	1.46
4579	4121500000000000	ค่าบริการอื่น ๆ	0 กรุงเทพมหานคร และปริมทุล	2015	2017	4	102.08	0.0	1.46	1.46
4580	4121500000000000	ค่าบริการอื่น ๆ	0 กรุงเทพมหานคร และปริมทุล	2015	2017	5	102.08	0.0	1.46	1.46
...	...	...	...	...	...	...	...	...	...	...
4835	4122400000000000	ค่าบริการอื่น ๆ	0 กรุงเทพมหานคร และปริมทุล	2015	2020	4	103.82	0.0	0.00	0.00
4836	4122400000000000	ค่าบริการอื่น ๆ	0 กรุงเทพมหานคร และปริมทุล	2015	2020	5	103.82	0.0	0.00	0.00
4837	4122400000000000	ค่าบริการอื่น ๆ	0 กรุงเทพมหานคร และปริมทุล	2015	2020	6	103.82	0.0	0.00	0.00
4838	4122400000000000	ค่าบริการอื่น ๆ	0 กรุงเทพมหานคร และปริมทุล	2015	2020	7	103.82	0.0	0.00	0.00
4839	4122400000000000	ค่าบริการอื่น ๆ	0 กรุงเทพมหานคร และปริมทุล	2015	2020	8	103.82	0.0	0.00	0.00

88 rows × 11 columns

หลังจากตรวจสอบจะเห็นว่าทั้ง 2 ตัวมีข้อมูล `index_id`, `price_index` ที่แตกต่างกัน แต่มี `index_description` ที่เหมือนกันเลยทำให้มีจำนวนแคนมากกว่าหมวดอื่น ๆ

จึงทำการแยกโดยนำ `price_index` ของทั้ง 2 ตัวมาเฉลี่ยกัน และนำค่าเฉลี่ยที่ได้แทนลงไว้ใน `price_index` ของ `index_id` ที่เท่ากับ `4121500000000000` และทำการลบแค่ที่ `index_id` เท่ากับ `4122400000000000`

```
In [107]: print('index_id = 4121500000000000\n', 'price_index =', df[df.index_id == 4121500000000000]['price_index'].drop_duplicates().iloc[0],
          '\n\n',
          'index_id = 4122400000000000\n', 'price_index =', df[df.index_id == 4122400000000000]['price_index'].drop_duplicates().iloc[0])
```

```
index_id = 4121500000000000
price_index = 102.08
```

```
index_id = 4122400000000000
price_index = 103.82
```

aoa = 102.92

เก็บ price\_index ของ index\_id ที่เท่ากับ 4121500000000000 ไว้ในตัวแปร priceIndexFirst

เก็บ price\_index ของ index\_id ที่เท่ากับ 4122400000000000 ไว้ในตัวแปร priceIndexSecond

```
In [108]: #กู้หันดูที่ priceIndexFirst เก็บค่า price_index ของ index_id ที่เท่ากับ 4121500000000000  
priceIndexFirst = df[df.index_id == 4121500000000000]['price_index'].iloc[0]  
  
#กู้หันดูที่ priceIndexSecond นึ่งค่า price_index ของ index_id ที่เท่ากับ 4122400000000000  
priceIndexSecond = df[df.index_id == 4122400000000000]['price_index'].iloc[0]
```

นำ price\_index ของทั้งสองมาเฉลี่ยกันและเก็บไว้ในตัวแปร newPriceIndex เพื่อนำไปใช้ในการแทนค่าใหม่

นำตัวแปร newPriceIndex แทนค่าลงไปที่ index\_id เท่ากับ 4121500000000000

```
In [109]: #นำ price_index ของทั้งสองค่าวางกันแล้วหาร 2 และเก็บไว้ในตัวแปร newPriceIndex  
newPriceIndex = ((df[df.index_id == 4121500000000000]['price_index'].add(priceIndexSecond, fill_value=0))/2).iloc[0]  
print(newPriceIndex)  
  
#นำ newPriceIndex ที่เป็นค่าเฉลี่ยของทั้งสองค่าว มาแทนค่าลงไปใน index_id ที่เท่ากับ 4121500000000000  
df[df.index_id == 4121500000000000] = df[df.index_id == 4121500000000000].replace(priceIndexFirst, newPriceIndex)
```

102.94999999999999

หลังจากแทนค่าใหม่ลงไปแล้ว ทำการลบ index\_id ที่เท่ากับ 4122400000000000 เพื่อให้เหลือหมวดค่าบริการอีก ๆ อันเดียว

```
In [110]: #หลังจากแทนค่าใหม่ลงไปแล้ว ซึ่งทำการลบ index_id ที่เท่ากับ 4122400000000000 ออกจาก df  
df = df[df.index_id != 4122400000000000]  
df[df.index_description == 'ค่าบริการอื่น ๆ']
```

Out[110]:

	index_id	index_description	region_id	region_name	base_year	year	month	price_index	mon	yoy	aoa
4576	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	1	102.95	0.0	1.46	1.46
4577	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	2	102.95	0.0	1.46	1.46
4578	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	3	102.95	0.0	1.46	1.46
4579	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	4	102.95	0.0	1.46	1.46
4580	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	5	102.95	0.0	1.46	1.46
4581	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	6	102.95	0.0	1.46	1.46
4582	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	7	102.95	0.0	1.46	1.46
4583	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	8	102.95	0.0	1.46	1.46
4584	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	9	102.95	0.0	1.46	1.46
4585	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	10	102.95	0.0	1.46	1.46
4586	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	11	102.95	0.0	0.00	1.33
4587	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	12	102.95	0.0	0.00	1.21
4588	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2018	1	102.95	0.0	0.00	0.00
4589	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2018	2	102.95	0.0	0.00	0.00
4590	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2018	3	102.95	0.0	0.00	0.00
4591	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2018	4	102.95	0.0	0.00	0.00
4592	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2018	5	102.95	0.0	0.00	0.00
4593	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2018	6	102.95	0.0	0.00	0.00
4594	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2018	7	102.95	0.0	0.00	0.00
4595	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2018	8	102.95	0.0	0.00	0.00
4596	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2018	9	102.95	0.0	0.00	0.00
4597	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2018	10	102.95	0.0	0.00	0.00
4598	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2018	11	102.95	0.0	0.00	0.00
4599	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2018	12	102.95	0.0	0.00	0.00

ผลบวกมณฑล											
4600	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2019	1	102.95	0.0	0.00	0.00
4601	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2019	2	102.95	0.0	0.00	0.00
4602	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2019	3	102.95	0.0	0.00	0.00
4603	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2019	4	102.95	0.0	0.00	0.00
4604	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2019	5	102.95	0.0	0.00	0.00
4605	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2019	6	102.95	0.0	0.00	0.00
4606	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2019	7	102.95	0.0	0.00	0.00
4607	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2019	8	102.95	0.0	0.00	0.00
4608	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2019	9	102.95	0.0	0.00	0.00
4609	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2019	10	102.95	0.0	0.00	0.00
4610	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2019	11	102.95	0.0	0.00	0.00
4611	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2019	12	102.95	0.0	0.00	0.00
4612	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2020	1	102.95	0.0	0.00	0.00
4613	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2020	2	102.95	0.0	0.00	0.00
4614	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2020	3	102.95	0.0	0.00	0.00
4615	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2020	4	102.95	0.0	0.00	0.00
4616	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2020	5	102.95	0.0	0.00	0.00
4617	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2020	6	102.95	0.0	0.00	0.00
4618	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2020	7	102.95	0.0	0.00	0.00
4619	4121500000000000	ค่าบริการอื่น ๆ	0	กรุงเทพมหานคร และปริมณฑล	2015	2020	8	102.95	0.0	0.00	0.00

หลังจากทำการลọcแล้วที่จำนวนมอยเกินไปสำหรับเคราะห์และแควที่มีความซ้ำซ้อนกันออกไป จึงทำการเช็คจำนวนแควทั้งหมดของแต่ละหมวดร่อง จะเห็นได้ว่าแต่ละหมวดมีจำนวน 44 แควเท่ากัน

```
In [111]: #ทำการเช็คจำนวน หลังจากที่ขูนหมวดที่มีจำนวนแควแตกต่างจากหมวดอื่นแล้วจึงเรียบร้อย
#จะเห็นได้ว่าทุกหมวดมีจำนวนแควเท่ากันทั้งหมด
countGroup = df['index_description'].groupby(df['index_description']).count()
dictCountGroup = countGroup.to_dict()
{k: v for k, v in sorted(dictCountGroup.items(), key=lambda item: item[1])}
```

```
Out[111]: {'CPI ในรุ่มน้ำมัน': 91, '95 , ดีเซล': 44, 'กลุ่มอาหารและพัฒนา': 44, 'กางเกงบุรุษ': 44, 'กางเกงสตรีและรองสตรี': 44, 'กางเกงเด็ก': 44, 'การบันทึกและการอ่าน': 44, 'การศึกษา': 44, 'การศึกษา': 44, 'การศึกษาระดับประถมศึกษา': 44, 'การศึกษาระดับมัธยมศึกษา ดำเนินเทคโนโลยีดิจิตอลและอาชีวะ': 44, 'การศึกษาระดับมัธยมศึกษาสายสามัญ': 44, 'การศึกษาระดับอนุบาลศึกษา': 44, 'การสื่อสาร': 44, 'ของเล่น': 44, 'ข้าว': 44, 'ข้าว': 44, 'แป้งและผลิตภัณฑ์จากแป้ง': 44, 'คนรับใช้ / คนงาน': 44, 'คนไข้': 44, 'คนไข้ใน': 44, 'ค่าการบันทึก': 44, 'ค่าของใช้ในส่วนบุคคล': 44, 'ค่าจ้าง': 44, 'ค่าวัสดุเบี้ยนและอุปกรณ์': 44, 'ค่าเชื้อราพานพาหนะ': 44, 'ค่าเชื้อราเชื้อ': 44, 'ค่าเชื้อราเชื้อ': 44, 'ค่าเชื้อราเชื้อและดูแลรักษา': 44, 'ค่าครัวครัวกษา ค่าบริการ และค่ายาของคนไข้ใน': 44, 'ค่าครัวครัวกษาและค่ายา': 44, 'ค่าครัวครัวที่จัดซื้อทางบองปฏิบัติการ': 44, 'ค่าครัวครัว': 44, 'ค่าครัวครัว ค่าบริการ และค่ายาของคนไข้ใน': 44, 'ค่าท่าไฟน์': 44, 'ค่าที่พักอาศัย': 44, 'ค่าธรรมเนียมและค่าสมาชิก': 44, 'ค่าบริการการสื่อสาร': 44, 'ล่าเริ่มการล่าเริ่มเสื้อ': 44}
```

'ค่าบริการบำรุงรักษาภายนอกนั่นด้': 44,  
'ค่าบริการส่วนบุคคล': 44,  
'ค่าบริการส่วนบุคคลเพื่อความผ่านคลาย': 44,  
'ค่าบริการส่วนบุคคลเพื่อความสวยงาม': 44,  
'ค่าบริการอื่น ๆ ': 44,  
'ค่าเบ็ดประกันสูนภาพและประกันชีวิต': 44,  
'ค่าประกันภัยรถ': 44,  
'ค่าภาษี': 44,  
'ค่ายาและเวชภัณฑ์': 44,  
'ค่ารักษาพยาบาล': 44,  
'ค่าห้องพักในชีวี': 44,  
'ค่าอาหาร ของว่ายพระ ให้ว่าเจ้าและอื่น ๆ ': 44,  
'ค่าอุปกรณ์การบันทึก': 44,  
'ค่าอุปกรณ์ที่ฟานแลเครื่องเล่น': 44,  
'ค่าเช่า': 44,  
'ค่าเรียนที่ศษ': 44,  
'ค่าเรียนและค่าธรรมเนียมการศึกษา': 44,  
'ค่าแรง': 44,  
'ค่าโดยสารออก กองถัง': 44,  
'ค่าโดยสารสาธารณะ': 44,  
'ค่าโดยสารใน กองถัง': 44,  
'ค่าใช้จ่ายที่เกี่ยวกับรถ': 44,  
'ค่าใช้จ่ายของวัสดุคอล': 44,  
'ค่าใช้จ่ายอื่น ๆ เกี่ยวกับการศึกษา': 44,  
'ค่าใช้จ่ายเกี่ยวกับการท่องเที่ยวบ้าน': 44,  
'ค่าใช้จ่ายที่เกี่ยวกับการอ่าน': 44,  
'ค่าใช้จ่ายเก็บบันเด็งเสี้ียง': 44,  
'ค่าใช้จ่ายในการเดินทางไปօISTERพิเศษและห้องเที่ยว': 44,  
'ตัวเป้าหมายที่ต้องพิสูจน์': 44,  
'ตัวน้ำมัน 91 , 95 , ตีเซล': 44,  
'น้ำมันเชื้อเพลิง': 44,  
'น้ำมันและไข่มัน': 44,  
'บริการที่อื่น ๆ ': 44,  
'บุรุษ': 44,  
'ปลาบ้าสีดสด': 44,  
'ปลาบ้าทะเลสด': 44,  
'ปลาเบอร์บี': 44,  
'ปลาและสัตว์น้ำ': 44,  
'ผลิตภัณฑ์จากแป้ง': 44,  
'ผลิตภัณฑ์น้ำชา': 44,  
'ผลิตภัณฑ์ยาสูบ': 44,  
'ผลิตภัณฑ์รองเท้า': 44,  
'ผลิตภัณฑ์สุรา': 44,  
'ผลิตภัณฑ์เมียร์': 44,  
'ผลิตภัณฑ์ไม้': 44,  
'ผลไม้สด': 44,  
'ผลไม้สดแปรรูปและอื่น ๆ ': 44,  
'ผลไม้แปรรูปและอื่น ๆ ': 44,  
'ผักสด': 44,  
'ผักสดเบรร์และอื่น ๆ ': 44,  
'ผักแปรรูปและอื่น ๆ ': 44,  
'ผักและผลไม้': 44,  
'ผ้าและเสื้อผ้า': 44,  
'พัฒนาน': 44,  
'ยานพาหนะ': 44,  
'ยานพาหนะและน้ำมันเชื้อเพลิง': 44,  
'รวมทุกรายการ': 44,  
'รองเท้ากีฬาและผ้าใบบุรุษ': 44,  
'รองเท้ากีฬาและผ้าใบสตรี': 44,  
'รองเท้ากีฬาและผ้าใบเด็ก': 44,  
'รองเท้านุรุษ': 44,  
'รองเท้าสตั๊ด': 44,  
'รองเท้าหุ้มส้นบุรุษ': 44,  
'รองเท้าหุ้มส้นสตรี': 44,  
'รองเท้าเด็ก': 44,  
'รองเท้านะจะบุรุษ': 44,  
'รองเท้าแฟชั่น': 44,  
'รองเท้าแฟชั่นสตรี': 44,  
'รองเท้านะจะเด็ก': 44,  
'รีสุดก่อสร้าง': 44,  
'สตั๊ด': 44,  
'สตั๊ดวัน': 44,  
'สีตัวน้ำแปรรูป': 44,  
'สิงห์ทองคำหรูนิ่งม่าน': 44,  
'สีสันที่เกี่ยวกับทำความสะอาด': 44,  
'หนังสือ': 44,  
'หนังสือแบบเรียบเอกชน': 44,  
'หนังสือและอุปกรณ์การศึกษา': 44,  
'หมวดการบันทึกการส่วนบุคคล': 44,  
'หมวดการบันทึกการอ่าน การศึกษา และการศาสนา': 44,  
'หมวดพากหนาน การแขวนสี และการสื่อสาร': 44,  
'หมวดยาสูบและเครื่องดื่มน้ำมันแอกลกอซอส': 44,  
'หมวดอาหารและเครื่องดื่มน้ำมันแอกลกอซอส': 44,  
'หมวดอื่น ๆ ': 44,  
'หมวดเครื่องบุหรี่และรองเท้า': 44,  
'หมวดเครื่องเคลือบ': 44,  
'อาหารบรู๊ฟ-โนกบัน': 44,  
'อาหารบรู๊ฟ-โนก ในม่าน': 44,  
'อาหารบูรงที่บัน': 44,  
'อาหารสด': 44,  
'อาหารสำเร็จรูป': 44,  
'อาหารและเครื่องดื่ม (ในรวมอาหารสด)': 44,  
'อุปกรณ์ท่องเที่ยว': 44,  
'อุปกรณ์ท่องเที่ยว': 44

'อุปกรณ์เยานพาหนะ': 44,  
 'อุปกรณ์เกี่ยวกับการศึกษา': 44,  
 'เครื่องดื่มน้ำและกอฮอลล์': 44,  
 'เครื่องต้มไม่มีแอลกอฮอลล์': 44,  
 'เครื่องพยายาม': 44,  
 'เครื่องประดับน้ำชา': 44,  
 'เครื่องประดับบุรุษ': 44,  
 'เครื่องประดับ': 44,  
 'เครื่องประดับอาหาร': 44,  
 'เครื่องรับอุปกรณ์สือสาร': 44,  
 'เครื่องซักผ้า': 44,  
 'เครื่องแต่งม่าน': 44,  
 'เครื่องแต่งบ้านและรีดผ้าอ่อนๆ': 44,  
 'เครื่องแบบชาย': 44,  
 'เครื่องแบบหญิง': 44,  
 'เครื่องแบบเด็ก': 44,  
 'เครื่องใช้เบ็ดเตล็ดในบ้าน': 44,  
 'เชื้อเพลิงในบ้าน': 44,  
 'เด็ก': 44,  
 'เด็กลูก': 44,  
 'เด็กลูก เปิดໄກ และลูกน้ำ': 44,  
 'เด็กลูกเด็ก': 44,  
 'เด็กลูกแม่รูป': 44,  
 'เป้า': 44,  
 'เปิด ໄກ สัก': 44,  
 'เปิด ໄກ สำเร็จรูปและปรุง': 44,  
 'เข็มกีด': 44,  
 'เลื่อนบุรุษ': 44,  
 'เลื่อนผ้าอุ้ยอัน': 44,  
 'เลื่อนผ้าอุ้ยอี้นา': 44,  
 'เลื่อนผ้าเด็กอี้นา': 44,  
 'เลื่อนสต๊รี': 44,  
 'เลื่อนเด็ก': 44,  
 'แนะนำและจัดกิจกรรม': 44,  
 'แสงสว่าง': 44,  
 'ไฟ': 44,  
 'ไฟและผลิตภัณฑ์น้ำ': 44,  
 'ไฟฟ้า': 44,  
 'ไฟฟ้า เชื้อเพลิง น้ำประปาและแสงสว่าง': 44,  
 'ไม้รวมอาหาร เครื่องต้ม และพลังงาน': 44}

## Save ข้อมูลที่ pre-processing เรียบร้อยแล้ว จาก DataFrame ไปเป็น .csv

เก็บข้อมูลที่ทำความสะอาดแล้วไว้ในชื่อ clean\_data.csv

```
In [112]: #เก็บข้อมูลในรูปแบบไฟล์ csv โดยเก็บไว้ในชื่อว่า clean_data.csv และต้องตั้งค่า encoding เป็น utf-8-sig เพราะมีข้อมูลที่เป็นภาษาไทย
df.to_csv('data/clean_data.csv', header=True, index=False, encoding='utf-8-sig')
```

## Visualization

```
In [15]: # -*- coding: utf-8 -*-
import matplotlib.pyplot as plt
%matplotlib inline
```

```
In [16]: #อ่านข้อมูล clean_data.csv และเก็บไว้ในตัวแปร df เพื่อใช้ในการทำ Visualization
df = pd.read_csv('data/clean_data.csv')
df
```

Out[16]:

	index_id	index_description	region_id	region_name	base_year	year	month	price_index	mon	yoy	aoa
0	0	รวมทุกรายการ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	1	100.53	0.07	1.18	1.18
1	0	รวมทุกรายการ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	2	100.53	0.00	1.00	1.09
2	0	รวมทุกรายการ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	3	100.18	-0.35	0.49	0.88
3	0	รวมทุกรายการ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	4	100.34	0.16	0.18	0.71
4	0	รวมทุกรายการ	0	กรุงเทพมหานคร และปริมณฑล	2015	2017	5	100.48	0.14	-0.04	0.56
...	...	...	...	...	...	...	...	...	...	...	...
7695	9520000000000000	CPI ในรวมน้ำมัน 91 , 95 , ดีเซล	0	กรุงเทพมหานคร และปริมณฑล	2015	2020	4	100.43	-1.72	-2.14	-0.14
7696	9520000000000000	CPI ในรวมน้ำมัน 91 , 95 , ดีเซล	0	กรุงเทพมหานคร และปริมณฑล	2015	2020	5	100.29	-0.14	-2.78	-0.66
7697	9520000000000000	CPI ในรวมน้ำมัน 91 , 95 , ดีเซล	0	กรุงเทพมหานคร และปริมณฑล	2015	2020	6	101.70	1.41	-1.31	-0.78
7698	9520000000000000	CPI ในรวมน้ำมัน 91 , 95 , ดีเซล	0	กรุงเทพมหานคร และปริมณฑล	2015	2020	7	102.43	0.72	-0.64	-0.75
7699	9520000000000000	CPI ในรวมน้ำมัน 91 , 95 , ดีเซล	0	กรุงเทพมหานคร และปริมณฑล	2015	2020	8	102.67	0.23	-0.21	-0.69

7700 rows × 11 columns

แสดงดัชนีราคาผู้บริโภคทั่วไปในกรุงเทพและปริมณฑล ตั้งแต่ปีค.ศ. 2017-

**Line plot**

อ่านข้อมูลรหัสหมวด/กลุ่มตัวชี้นิ (Index\_id) จากไฟล์ รหัสหมวดตัวชี้นิ CPI\_INDEX 31-8-2020.csv ที่เก็บไว้ในโฟลเดอร์ data

ข้อมูลที่อ่านมาจะเก็บไว้ในตัวแปร CPI\_Index แต่ค่าที่ต้องการใช้มีแค่รหัสหมวด/กลุ่มตัวชี้นิ (Index\_id) จึงทำการเลือกเฉพาะคอลัมน์รหัสตัวชี้นิแล้วเก็บไว้ในตัวแปร CPI\_index\_id

ข้อมูลที่เก็บไว้ในตัวแปร CPI\_index\_id ออยู่ในรูปแบบของ pandas.Series ซึ่งไม่สะดวกต่อการใช้งานจึงทำการเปลี่ยนให้อยู่ในรูปแบบของ list โดยเก็บข้อมูลไว้ในตัวแปร CPI\_index\_id\_list

```
In [17]: #แปลงข้อมูลในคอลัมน์ 'รหัสตัวชี้นิ' ให้มีเป็นประเภท int64 เพื่อกำหนดจุดทดสอบ
CPI_Index = pd.read_csv("data/รหัสหมวดตัวชี้นิ CPI_INDEX 31-8-2020.csv", dtype={'รหัสตัวชี้นิ': np.int64})
CPI_index_id = CPI_Index['รหัสตัวชี้นิ']
CPI_index_id_list = CPI_index_id.tolist()
# CPI_index_id_list
```

ทำการพล็อต line chart ของหมวดหมู่รายการเพื่อดูภาพรวม

```
In [18]: index_ = df[df.index_id == 0] #เลือก index_id ที่เท่ากับ 0 เพราะเป็นหมวดหมู่ทุกรายการ แล้วเก็บไว้ในตัวแปร index_
month = index_[‘month’].tolist() #เลือกคอลัมน์ month จากตัวแปร index_ และแปลงให้มีเป็น List จากนั้นเก็บไว้ในตัวแปร month
year = index_[‘year’].tolist() #เลือกคอลัมน์ year จากตัวแปร index_ และแปลงให้มีเป็น List จากนั้นเก็บไว้ในตัวแปร year
price_index = index_[‘price_index’].tolist() #เลือกคอลัมน์ price_index จากตัวแปร index_ และแปลงให้มีเป็น List จากนั้นเก็บไว้ในตัวแปร price_index
month_year = [] #สร้างตัวแปร month_year เพื่อรับเก็บค่าห้ามจาก เอา month กับ year มาต่อกันเพื่อนำไปใช้ในการพล็อตกราฟ
for i in range(len(month)): #นำอุปกรณ์มาต่อๆกันเพื่อให้month/year สามารถต่อตัวกันได้
    #นำตัวแปร month index ที่ i มาต่อ กับตัวแปร year 2 ตัวท้าย index ที่ i โดยขั้นตอนกลางด้วย /
    #หลังจากนั้นนำค่าที่ได้เพิ่มเข้าไปใน month_year
    month_year.append(str(month[i]) + ‘/’ + str(year[i])[2:] )
#example i = 0
# 1/17

plt.figure(figsize=(25, 3))
#ทำการแปลงฟอนต์ที่รองรับภาษาไทย
plt.title(‘รวมทุกรายการ’, fontname=‘Tahoma’, fontsize=‘20’)
plt.xlabel(‘เดือน/ปี’, fontname=‘Tahoma’, fontsize=‘18’)
plt.ylabel(‘ตัวชี้ราคา’, fontname=‘Tahoma’, fontsize=‘18’)
#ผลลัพธ์กราฟโดยแกน x คือเดือน/ปี (จากตัวแปร month_year) และ y คือตัวชี้ราคา (จากตัวแปร price_index) และตั้งค่าให้เส้นเป็นสีน้ำเงินและมีจุดวงกลม
plt.plot(month_year, price_index, ‘-bo’);
```

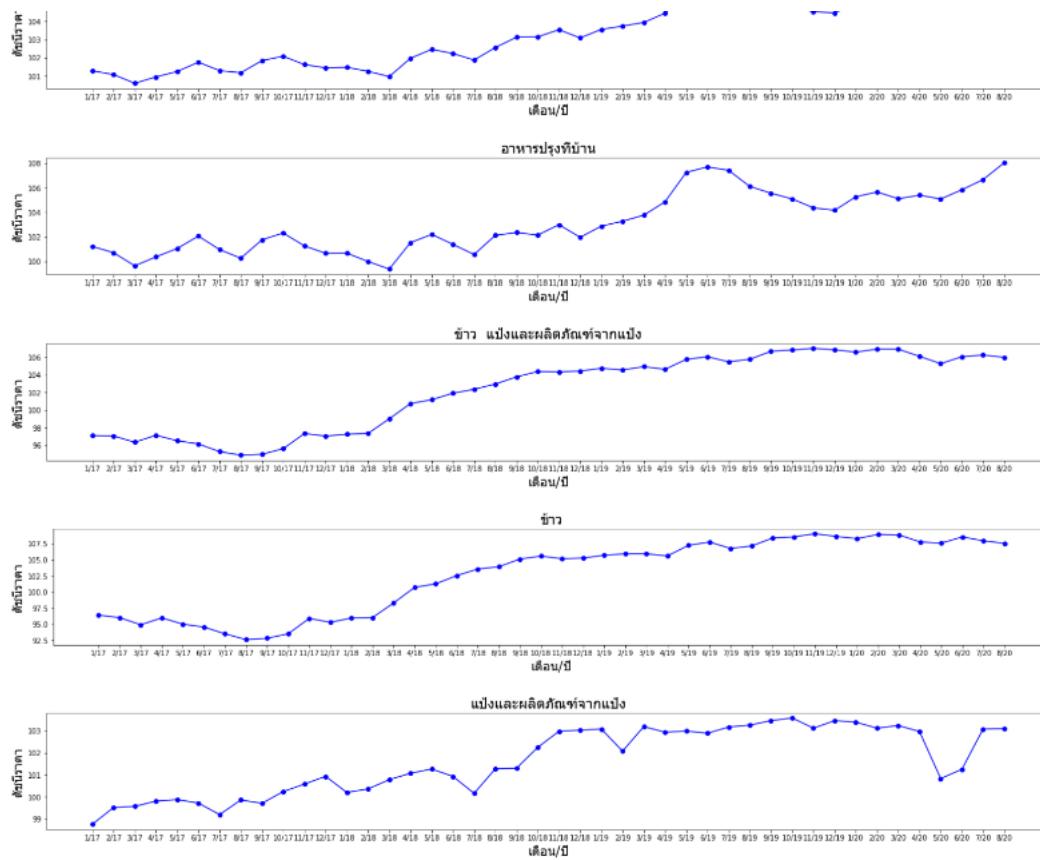


จากราฟเส้น รวมทุกรายการ จะเห็นได้ว่าเพราะข่าวไวรัสโควิด-19 ในช่วงปี2020 สงผลให้คืนเริ่มไม่กล้าใช้เงิน ทำให้ต้องปรับราคาสินค้าลง (ลดราคาลงเพื่อให้ห้องขายออก) และเมื่อเมียประกาศพรก.ฉุกเฉินจารีตรูบานเลือวันที่ 25/03/2020 (<https://www.bangkokbiznews.com/news/detail/872653>) จะเห็นได้ว่าราคาสินค้าต่างๆลดลงอย่างชัดเจน ซึ่งเมื่อข่าวไวรัสโควิด-19 เริ่มต้นขึ้นราคาสินค้าต่างๆก็เริ่มกลับขึ้นมา ซึ่งในช่วงโควิดก็มีทั้งสินค้าที่ราคาเพิ่มขึ้นและลดลง

```
In [19]: #ถ้าห้องนัด 189 หมวด (1-189) ในบันทุมหมวดหมู่ทุกรายการ ดำเนินเรียบไม่คล้ายจะใช้เวลาอ่านนาน
show = 5 #ตัวแปร show จะแสดงจำนวนกราฟตามที่กำหนด
for i in CPI_index_id_list[1:show+1]:
    #วนลูปในตัวแปร CPI_index_id_list ซึ่งเก็บ index_id ห้องนัด แต่จะเริ่นพล็อตกราฟตั้งแต่ index ที่ 1 เพราะได้ทำการพล็อตกราฟของหมวดหมู่ทุกรายการไปแล้ว
    #i แต่ละรอบต้อง index_id
    index_ = df[df.index_id == i] #เลือก index_id ที่เท่ากับ i และเก็บไว้ในตัวแปร index_
    month = index_[‘month’].tolist() #เลือกคอลัมน์ month จากตัวแปร index_ และแปลงให้มีเป็น List จากนั้นเก็บไว้ในตัวแปร month
    year = index_[‘year’].tolist() #เลือกคอลัมน์ year จากตัวแปร index_ และแปลงให้มีเป็น List จากนั้นเก็บไว้ในตัวแปร year
    price_index = index_[‘price_index’].tolist() #เลือกคอลัมน์ price_index จากตัวแปร index_ และแปลงให้มีเป็น List จากนั้นเก็บไว้ในตัวแปร price_index
    month_year = [] #สร้างตัวแปร month_year เพื่อรับเก็บค่าห้ามจาก เอา month กับ year มาต่อ กับ year โดยขั้นตอนกลางด้วย /
    #หลังจากนั้นนำค่าที่ได้เพิ่มเข้าไปใน month_year
    month_year.append(str(month[i]) + ‘/’ + str(year[i])[2:] )

    #เช็คจำนวนรายการของ index_ ต้องมากกว่า 0 ถึงจะทำการพล็อตกราฟเพราะว่า index_id บางตัวนั้นไม่มีข้อมูล
    if len(index_[‘index_description’].tolist()) > 0:
        plt.figure(figsize=(25, 3))
        #กำหนดให้ title และเส้นเป็น index_description ของแต่ละรอบและเปลี่ยนฟอนต์ที่รองรับภาษาไทย
        plt.title(‘+’+index_[‘index_description’].tolist()[0], fontname=‘Tahoma’, fontsize=‘20’)
        plt.xlabel(‘เดือน/ปี’, fontname=‘Tahoma’, fontsize=‘18’)
        plt.ylabel(‘ตัวชี้ราคา’, fontname=‘Tahoma’, fontsize=‘18’)
        #ผลลัพธ์กราฟโดยแกน x คือเดือน/ปี (จากตัวแปร month_year) และ y คือตัวชี้ราคา (จากตัวแปร price_index) และตั้งค่าให้เส้นเป็นสีน้ำเงินและมีจุดวงกลม
        plt.plot(month_year, price_index, ‘-bo’);
    else:
        continue
```





จากที่กล่าวข้างต้นว่าในช่วงโควิดก็มีทั้งสินค้าที่ราคาเพิ่มขึ้นและลดลง เมื่อตู้จากกราฟเส้นที่แสดงจะเห็นได้ว่าหมวดอาหารและเครื่องดื่มไม่มีผลก่อให้สูญเสียในช่วงโควิด-19 ส่วนหมวดที่แนวโน้มดีขึ้นคือราคากล่องในช่วงโควิด-19 เช่น หมวดแป้งและผลิตภัณฑ์จากแป้ง

## หมวดที่ราคาลดลง/เพิ่มขึ้น มากที่สุด 10 อันดับแรก

### Bar plot

พล็อกเดื่อคู่หมวดไหนมีราคาลดลง/เพิ่มขึ้นจากปีก่อน 10 อันดับแรก

```
In [20]: #เดือนที่จะเปลี่ยนคือเดือน 8 และปี 2020 กับ 2019
#หมวดที่ราคาลดลงจากปีก่อนมากที่สุด 10 อันดับแรก เก็บไว้ในตัวแปร decreaseFiveYOY
decreaseTenYOY = df[(df.year == 2020) & (df.month == 8)].sort_values(by=['yoy']).head(10)[::-1]

#หมวดที่ราคานั้นเพิ่มขึ้นมากที่สุด 10 อันดับแรก เก็บไว้ในตัวแปร increaseFiveYOY
increaseTenYOY = df[(df.year == 2020) & (df.month == 8)].sort_values(by=['yoy']).tail(10)

print(decreaseTenYOY[['index_description', 'yoy']], '\n\n', increaseTenYOY[['index_description', 'yoy']])
```

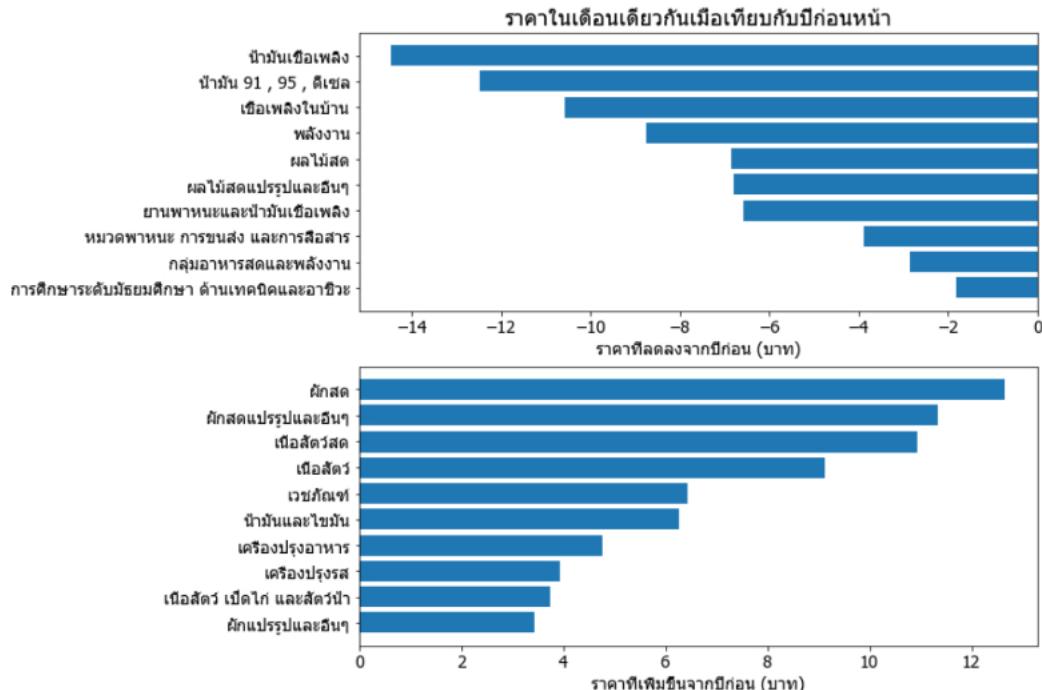
	index_description	yoy
6555	การศึกษาระดับมัธยมศึกษา ด้านเทคโนโลยีคอมพิวเตอร์ฯลฯ	-1.83
7435	กลุ่มอาหารสดและผักจีน	-2.86
5103	หมวดพาหนะ การขนส่งสินค้าและสารเคมี	-3.89
5279	ยานพาหนะและน้ำมันเชื้อเพลิง	-6.58
1231	ผลไม้สดและรูปแบบอื่นๆ	-6.79
1275	ผลไม้สด	-6.86
7523	พวงงาน	-8.77
3695	เชื้อเพลิงในบ้าน	-10.59
7655	น้ำมัน 91 , 95 , ดีเซล	-12.49
5455	น้ำมันเชื้อเพลิง	-14.46
	index_description	yoy
1187	ผักกาดขาวและอื่นๆ	3.43
351	ผักกาดเขียว เป็ดไก่ และสัตว์น้ำ	3.75
1583	เครื่องปรุงรส	3.94
1495	เครื่องปรุงอาหาร	4.78
1539	น้ำมันและไขมัน	6.28
4311	เบซเก็ตซ์	6.43
395	เนื้อสัตว์สด	9.13
439	เนื้อสัตว์สด	10.95
1099	ผักสดและอื่นๆ	11.35
1143	ผักสด	12.66

```
In [21]: #แปลงเป็นฟอนต์ที่รองรับภาษาไทย ที่ต้องกำหนดตรงๆ เพราะจะได้เปลี่ยนในส่วนของ index_description ที่แสดงในแกน y
plt.rcParams['font'] = {'family': 'Tahoma', 'size': '14'}

plt.figure(figsize=(10, 7))
plt.subplots_adjust(top = 10, bottom=9)

plt.subplot(2,1,1)
plt.title("ราคายอดขายที่เปลี่ยนแปลงมากที่สุด 10 อันดับแรก", fontname='Tahoma', fontsize='18')
```

```
#แกน y ต้องคือลิม 'index_description' และแกน x ต้องคือลิม 'yoys' ของ decreaseFiveYOY
plt.barh(decreaseTenYOY['index_description'], decreaseTenYOY['yoys'])
plt.xlabel('ราคาที่ลดลงจากปีก่อน (บาท)')
plt.subplot(2,1,2)
#แกน y ต้องคือลิม 'index_description' และแกน x ต้องคือลิม 'yoys' ของ increaseFiveYOY
plt.barh(increaseTenYOY['index_description'], increaseTenYOY['yoys'])
plt.xlabel('ราคาที่เพิ่มขึ้นจากปีก่อน (บาท)');
```



จาก Bar plot ข้างบนจะเห็นได้ว่าหมวดที่ราคาลดลงจากปีก่อนหน้ามากที่สุดคือ น้ำมันเชื้อเพลิง โดยราคาลดลงไป 14.46 บาทและหมวดที่ราคาเพิ่มขึ้นจากปีก่อนหน้ามากที่สุดคือ ผ้าสัก โดยราคาเพิ่มขึ้น 12.66 บาท

## กราฟเส้นหมวดที่ราคาลดลงจากปีก่อนมากที่สุด 10 อันดับแรก

### Line plot

แสดงแนวโน้มของหมวดที่ราคาลดลงจากปีก่อนมากที่สุด 10 อันดับแรก เพื่อดูว่ามีหมวดไหนบ้างที่กราฟไปในทางเดียวกัน

```
In [22]: #ดึงมาปรับ relatedCategoriesDecrease เก็บค่า index_description ของหมวดที่ราคาลดลงจากปีก่อนมากที่สุด 10 อันดับแรก
#ค่าตัววันนี้ในตัวปรับ relatedCategoriesDecrease จะมีตัวที่เกี่ยวข้องกับ
relatedCategoriesDecrease = decreaseTenYOY['index_description'].tolist()
for i in relatedCategoriesDecrease:
    print(i)
```

การศึกษาระดับมัธยมศึกษา ด้านเทคโนโลยีและอาชีวะ  
กลุ่มอาหารสดและพลังงาน  
หมวดอาหาร การชราสัง และการสื่อสาร  
ยานพาหนะและน้ำมันเชื้อเพลิง  
ผลไม้สดแปรรูปและอื่นๆ  
ผลไม้สด  
พลังงาน  
เชื้อเพลิงในบ้าน  
น้ำมัน 91 , 95 , ดีเซล  
น้ำมันเชื้อเพลิง

```
In [23]: #function relatedCategoriesPriceIndexDecrease จะรับค่าตัวเลขเข้ามา
#ตัวเลขตั้งแต่ 0 ถึง ล่าสุด index ที่เอามาใช้ในชื่อคอลัมน์จากตัวปรับ relatedCategoriesDecrease
#ค่าที่ function นี้จะส่งกลับไปเก็บใน price_index
def relatedCategoriesPriceIndexDecrease(index):
    return df[df.index_description == relatedCategoriesDecrease[index]]['price_index'].tolist()
```

```
In [24]: #เปลี่ยนเป็นฟอนต์ที่รองรับภาษาไทย ที่ดองกานดครงที่เพรจะได้เปลี่ยนในส่วนของ index_description ที่แสดงใน Legend
plt.rcParams['font'] = {'family': 'Tahoma',
                       'size': '14'}

#สร้างตัวปรับ month_year เพื่อให้เก็บค่าเดือน/ปี เพื่อนำไปกำหนดค่าในแกน x
index_ = df[df.index_id == 0]
month = index_[['month']].tolist()
year = index_[['year']].tolist()
month_year = []
for i in range(len(month)):
    month_year.append(str(month[i]) + '/' + str(year[i])[2:])

plt.figure(figsize=(30, 5))

#นำลูกปัดความยาวของ relatedCategoriesDecrease
for i in range(len(relatedCategoriesDecrease)):
    #สร้างตัวปรับ relatedCategories_i เพื่อเก็บ price_index ที่ function relatedCategoriesPriceIndexDecrease ลงลิบ
    relatedCategories_i = relatedCategoriesPriceIndexDecrease(i)
    #พล็อตกราฟเส้นข้อมูลไปเรื่อยๆ จนกว่าจะจบลูป
```

```

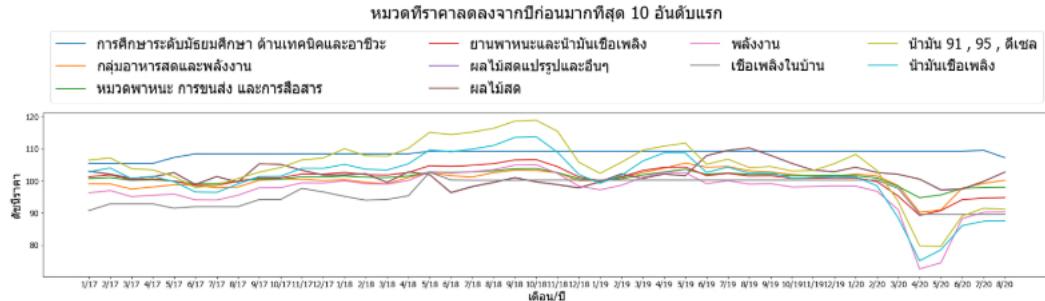
#แกน x คือเดือน/บันทึกเวนดูมานร์ month_year
#แกน y คือตัวบัญชีราคาที่เก็บไว้ในตัวบันร์ relatedCategoriesDecrease
#label คือ index_description และกำหนดให้ความหมายของเส้นเท่ากับ 2
plt.plot(month_year, relatedCategories_i, label=relatedCategoriesDecrease[i], linewidth=2)

plt.title(บ'หมวดที่ราคาลดลงจากปีก่อนมากที่สุด 10 อันดับแรก', y=1.55, fontsize='28')
plt.xlabel(บ'เดือน/ปี', fontsize='20')
plt.ylabel(บ'ตัวบัญชีราคา', fontsize='20')

plt.legend(bbox_to_anchor=(0., 1.02, 1., .102), loc='lower left',
           ncol=4, mode="expand", fontsize='24')

plt.show();

```



จากราฟเส้นขึ้นบนจะเห็นว่า ยานพาหนะและน้ำมันเชื้อเพลิง, พลังงาน, น้ำมัน 91 95 ดีเซล, ก่อสร้างและห้องน้ำ, การขนส่ง และการสื่อสาร แนวโน้มของรา飚ไปในทิศทางเดียวกัน โดยสินค้าดังกล่าวจะติดอยู่ในหมวดของเชื้อเพลิง พลังงานและยานพาหนะ เมื่อดูกราฟในช่วงโควิดจะเห็นว่าตัวบัญชีราคาในช่วงเวลาดังกล่าวมีค่าลดลง เพราะประชาชัชน์ส่วนใหญ่ต้องกักตัวอยู่บ้าน เดินทางน้อยลง ทำงานอยู่บ้าน เนื่องจากสถานการณ์โควิดที่ต้องการน้อยลงจึงทำให้ราคาน้ำมันลดลงตามไปด้วย (<https://www.posttoday.com/economy/news/621491>)

## กราฟเส้นหมวดที่ราคาเพิ่มขึ้นจากปีก่อนมากที่สุด 10 อันดับแรก

### Line plot

แสดงแนวโน้มของหมวดที่ราคาเพิ่มจากปีก่อนมากที่สุด 10 อันดับแรก เพื่อดูว่ามีหมวดไหนบ้างที่กราฟไปในทางเดียวกัน

```

In [25]: #ดูว่ามี relatedCategoriesIncrease เท่าไหร่ index_description ของหมวดที่ราคาเพิ่มขึ้นจากปีก่อนมากที่สุด 10 อันดับแรก
#ค่าตัวบัญชีในตัวบันร์ relatedCategoriesIncrease จะมีตัวที่เก็บไว้ของกัน
relatedCategoriesIncrease = increaseTenYOY['index_description'].tolist()
for i in relatedCategoriesIncrease:
    print(i)

```

ผักแพรรูปและอื่นๆ  
เต็อกซิว เป็ดไก่ และสัตว์น้ำ  
เครื่องปูรงรส  
เครื่องปูรงอาหาร  
น้ำมันและไขมน้ำ  
เวชภัณฑ์  
เนื้อสัตว์สด  
ผักสดแปรรูปและอื่นๆ  
ผักสด

```

In [26]: #function relatedCategoriesPriceIndexIncrease จะรับค่าตัวบัญชีเข้ามา
#ตัวเลขตั้งแต่วันที่ index ท่านไปใช้ตั้งข้อมูลจากตัวบันร์ relatedCategoriesIncrease
#ค่าตัว function นี้จะลงกลับไปที่คือ price_index
def relatedCategoriesPriceIndexIncrease(index):
    return df[df.index_description == relatedCategoriesIncrease[index]]['price_index'].tolist()

```

```

In [27]: #เปลี่ยนเป็นฟอนต์ที่รองรับภาษาไทย ที่ต้องกำหนดตรงนี้เพราะจะได้เปลี่ยนในส่วนของ index_description ที่แสดงใน Legend
plt.rc('font', **{'family' : 'Tahoma',
                  'size' : '14'})

#สร้างตัวบันร์ month_year เพื่อใช้เก็บค่าเดือน/ปี เพื่อนำไปกำหนดค่าในแกน x
index_ = df[df.index_id == 0]
month = index_[‘month’].tolist()
year = index_[‘year’].tolist()
month_year = []
for i in range(len(month)):
    month_year.append(str(month[i]) + ‘/’ + str(year[i])[2:])

plt.figure(figsize=(30, 5))

#วนลูปคำนวณของ relatedCategoriesIncrease
for i in range(len(relatedCategoriesIncrease)):
    #สร้างตัวบันร์ relatedCategories_i เพื่อเก็บ price_index ที่ function relatedCategoriesPriceIndexIncrease แล้วนำมา
    relatedCategories_j = relatedCategoriesPriceIndexIncrease(i)
    #พล็อตกราฟเส้นที่อ่อนก้นไปเรื่อยๆ จนกว่าจะจบสูง
    #แกน x คือตัวบัญชีราคาที่เก็บไว้ในตัวบันร์ month_year
    #แกน y คือตัวบัญชีราคาที่เก็บไว้ในตัวบันร์ relatedCategoriesDecrease
    #label คือ index_description และกำหนดให้ความหมายของเส้นเท่ากับ 2
    plt.plot(month_year, relatedCategories_j, label=relatedCategoriesIncrease[i], linewidth=2)

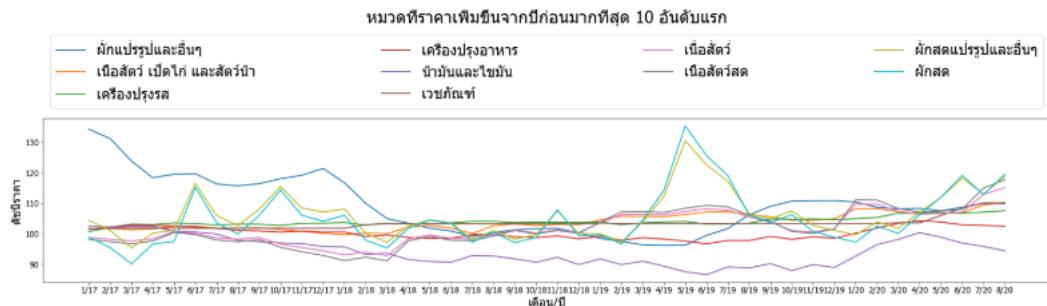
plt.title(บ'หมวดที่ราคาเพิ่มขึ้นจากปีก่อนมากที่สุด 10 อันดับแรก', y=1.55, fontsize='28')
plt.xlabel(บ'เดือน/ปี', fontsize='20')
plt.ylabel(บ'ตัวบัญชีราคา', fontsize='20')

plt.legend(bbox_to_anchor=(0., 1.02, 1., .102), loc='lower left',
           ncol=4, mode="expand", fontsize='24')

```

```
ncol=4, mode="expand", fontsize='24')
```

```
plt.show();
```



จากการไฟเข็นด่วนจนกระทั่งเดินทางกลับไปรับประทานอาหาร เนื้อสันดู ผักสดแปรรูปและอื่นๆ เนื้อสันดู เปิดໄก และสังฆภู่ น้ำมันและไขมัน เนื้อสันดูสด ผักสด เครื่องปูร์เช แนวโน้มของกระเพาะจะไปในทิศทางเดียวกัน โดยสันดูตั้งกล่าวจะจัดอยู่ในหมวดของอาหาร เมื่อถูกกราฟในช่วงโควิดเดินทางไปร้านอาหารในช่วงเวลาดังกล่าวมีค่าที่เพิ่มสูงขึ้นเพราบประมาณส่วนใหญ่ต้องก้าวเดินทางไกล ลง ท่านอนอยู่บ้าน เหตุผลเหล่านี้ส่งผลให้สินค้าในหมวดชิ้นเป็นที่ต้องการมากขึ้นเพราบการที่ต้องออกเดินทางไกล การจับจ่ายซื้อของบ้านแต่ละที่จึงต้องมีการซื้อต้นไว้จึงทำให้ราคาสินค้าเพิ่มขึ้นตามไปด้วย

แต่จะมีหมายเหตุที่ถูกต้องจากแพทย์ เช่นค่าเดินทาง (อุปกรณ์ทางการแพทย์) เนื่องจากต้องเดินทางไปรักษาตัวที่โรงพยาบาลที่อยู่ห่างไกล ค่าเดินทางจะสูงกว่าค่าห้องพักและอาหาร

### หาสัดส่วนราคาสินค้าลดลง/เพิ่มขึ้น/เท่าเดิม

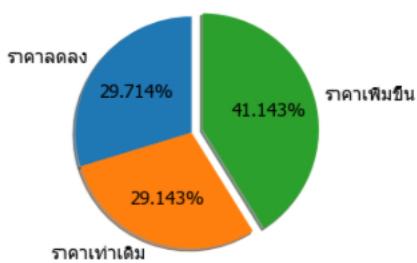
## Pie chart

พอดีด้วย pie chart เพื่อหาสัดส่วนว่าในเดือนธันวาคมของปีที่ผ่านมา (nov) เป็นอย่างไรซึ่งมีราคาน้ำในกล่อง

```
In [28]: #ดูว่ามี LenDfNow เก็บค่าจำนวนของ Dataframe ในปีจุบัน (เดือน 8 ปี 2020) เพื่อใช้ในการหาราสัดส่วน
lenDfNow = len(df[(df.year == 2020) & (df.month == 8)])  
  
#ดูว่ามี percentDecrease เก็บเปอร์เซ็นต์หมวดสินค้าที่มีตัวบิราราคาลดลง
percentDecrease = round(len(df[(df.yoy < 0) & (df.year == 2020) & (df.month == 8)]) / lenDfNow *100, 3)  
  
#ดูว่ามี percentSame เก็บเปอร์เซ็นต์หมวดสินค้าที่มีตัวบิราราคาเท่าเดิม
percentSame = round(len(df[(df.yoy == 0) & (df.year == 2020) & (df.month == 8)]) / lenDfNow*100, 3)  
  
#ดูว่ามี percentIncrease เก็บเปอร์เซ็นต์หมวดสินค้าที่มีตัวบิราราคาเพิ่มขึ้น
percentIncrease = round(len(df[(df.yoy > 0) & (df.year == 2020) & (df.month == 8)]) / lenDfNow*100, 3)  
  
#รวม 100 เพื่อแปลงเป็นเปอร์เซ็นต์ แล้วใช้ค่าล้าง round เพื่อบิดจุดทศนิยมให้เหลือ 3 หลัก
print('ราคารลดลง:\t',percentDecrease, '%\nราคากำไรเดิม:\t',percentSame, '%\nราคากำไรเพิ่มขึ้น:\t', percentIncrease, '%')
print('รวมทั้งหมด:\t', percentDecrease + percentSame + percentIncrease, '%')  
  
ราคารลดลง: 29.714 %
ราคากำไรเดิม: 29.143 %
ราคากำไรเพิ่มขึ้น: 41.143 %
รวมทั้งหมด: 100.0 %
```

```
In [29]: values = [percentDecrease, percentSame, percentIncrease] #เก็บสัดส่วนของราคอลดลง ราคาเท่าเดิมและราคายืนยัน
labels = ['ราคอลดลง', 'ราคาเท่าเดิม', 'ราคายืนยัน'] #เก็บ Label ของราคอลดลง ราคาเท่าเดิมและราคายืนยัน ตามลำดับ
explode = (0, 0, 0.1) #ทำให้ pie ส่วนที่ 3 เสือโก开
plt.title(u'สัดส่วนถึงการเปลี่ยนแปลงของกินเดือนต่อปีก่อนหน้า (YOY)', fontname='Tahoma', fontsize=18)
plt.pie(values, explode=explode, labels=labels, shadow=True, autopct='%1.3f%%', startangle=90);
```

สัดส่วนคัดกรองการเบี่ยงเบนกันเดือนเดียวกันปีก่อนหน้า (YOY)



จาก Pie chart ข้างบนจะเห็นว่าตัวชี้ราคานิค้าส่วนใหญ่เพิ่มขึ้น ต่อเป็น 41.143% จากทั้งหมด

อีกสองส่วนที่เหลือคือ ราคากล่องและราคาน้ำหน้าเดิม มีสัดส่วนเป็น 29.714% และ 29.143% ตามลำดับ ก็อว่ามีสัดส่วนพอๆ กัน

สรุปได้ว่าในช่วง 1 ปีที่ผ่านมาดัชนีราคาผู้ผลิตทั่วไปมีแนวโน้มที่เพิ่มสูงขึ้น

วิเคราะห์ดัชนีราคาที่เพิ่ม/ลดส่วนใหญ่เป็นจำนวนกี่บาท

## Histogram

วิเคราะห์ว่าตัวนี่ราคาส่วนใหญ่ที่เพิ่มขึ้นหรือลดลงเป็นจำนวนกี่นาท โดยวิธีกำหนดจำนวน bin จะใช้ Rice rule

```
In [152]: #ดูว่าปร priceIndexYOY เก็บค่าอัตราการเปลี่ยนแปลงที่ยืนกับเดือนเดียวกันปีก่อนหน้า (เดือน 8 ปี 2020 ที่เทียบเดือนเดียวกันในปี 2019)
priceIndexYOY = df[(df.year == 2020) & (df.month == 8)][['yoy']]

#ดูว่าปร priceIndexMON เก็บค่าอัตราการเปลี่ยนแปลงที่ยืนกับเดือนก่อนหน้า (เดือน 8 ปี 2020 ที่เทียบกับปีเดียวกันในเดือน 7)
priceIndexMON = df[(df.year == 2020) & (df.month == 8)][['mon']]

#ดูว่าปร priceIndexAOA เก็บค่าอัตราการเปลี่ยนแปลงที่ยืนกับช่วงเดียวกันปีก่อนหน้า (AOA)
priceIndexAOA = df[(df.year == 2020) & (df.month == 8)][['aoa']]

priceIndexList = [priceIndexYOY, priceIndexMON, priceIndexAOA]
priceIndexNameList = ['อัตราการเปลี่ยนแปลงราคายืนเดือนเดียวกันปีก่อนหน้า (YOY)', 'อัตราการเปลี่ยนแปลงราคายืนเดือนก่อนหน้า (MON)', 'อัตราการเปลี่ยนแปลงราคายืนช่วงเดียวกันปีก่อนหน้า (AOA)']
```

ใช้ Rice Rule ในการหาว่าควรจะกำหนดจำนวน bin เท่าไร

ก คือจำนวนข้อมูล

$$binSize = 2\sqrt{n}^{(1/3)}$$

$$binSize = 2\sqrt{175}^{(1/3)}$$

$$binSize \approx 11$$

```
In [153]: binSize = int(2 * (len(priceIndexYOY)**(1/3)))
binSize
```

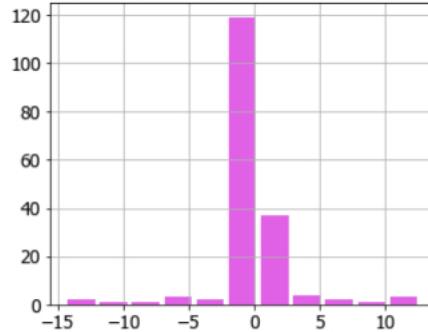
Out[153]: 11

```
In [154]: #แสดงความถี่ของราคายืนเดือน/ลดลงในทุกปีเป็นจำนวนกี่นาท
fig, axes = plt.subplots(3,1, figsize = (6,12))
colorList = ['#D31EDB', '#DC941F', '#10B788']

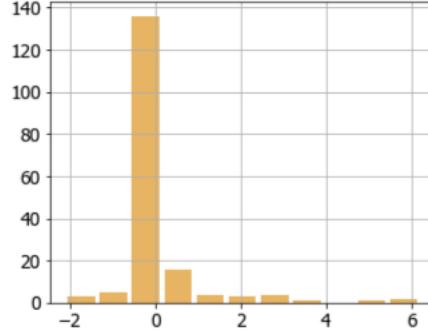
for i in range(len(priceIndexList)):
    axes[i].hist(priceIndexList[i], binSize, color=colorList[i], alpha=0.7, rwidth=0.85)
    axes[i].grid(True)
    axes[i].set_title(priceIndexNameList[i])

fig.tight_layout()
plt.show();
```

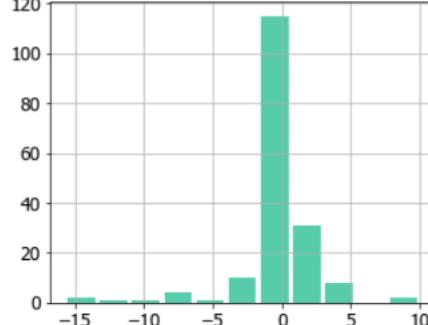
อัตราการเปลี่ยนแปลงราคายืนเดือนเดียวกันปีก่อนหน้า (YOY)



อัตราการเปลี่ยนแปลงราคายืนเดือนก่อนหน้า (MON)



อัตราการเปลี่ยนแปลงราคายืนช่วงเดียวกันปีก่อนหน้า (AOA)



จะเพิ่มหรือลด ราคางบ ± ไม่เกิน 5 บาท

จาก Histogram ของอัตราการเปลี่ยนแปลงราคาเทียบกับเดือนก่อนหน้า (MON) จะเห็นว่าสิ่งแม่ร้าจะมีการเปลี่ยนแปลงไม่ว่าจะเพิ่ม หรือลด ราคางบ ± ไม่เกิน 2 บาท

จาก Histogram ของอัตราการเปลี่ยนแปลงราคาเทียบกับช่วงเดียวกันปีก่อนหน้า (AOA) จะเห็นว่าสิ่งแม่ร้าจะมีการเปลี่ยนแปลงไม่ว่าจะเพิ่ม หรือลด ราคางบ ± ไม่เกิน 5 บาท

เมื่อนำผลวิเคราะห์ที่ได้จาก Pie chart ข้างบนมาดูด้วยกัน จะเห็นว่าในช่วง 1 ปีที่ผ่านมาตัวชี้ว่าค่าผู้บริโภคทั่วไปส่วนใหญ่มีราคาที่เพิ่มสูงขึ้นใน ช่วง 0.1 - 5 บาท

## สรุปผล

จากการเฝ้าระวังทุกรายการ จะเห็นได้ว่า เพราะข่าวไวรัสโคโรนา-19 ในช่วงต้นปี 2020 ส่งผลให้ตัวชี้ว่าค่าผู้บริโภคทั่วไปมีแนวโน้มลดลงในช่วงครึ่งปีแรกแล้วคืออยู่ที่ 0.1-5 บาท ในการเฝ้าระวังจะเห็นว่าเศรษฐกิจในครึ่งปีหลังเริ่มตื้นชื้นเรื่อยๆ ถึงขั้นที่ว่า เมื่อทำการหาสัดส่วนของราคสินค้าเพิ่ม/ลด/เท่าเดิม จะพบว่า เมื่อเทียบในเดือนเดียวกันกับปีก่อนหน้า สัดส่วนของราคสินค้าเพิ่มขึ้นมีสัดส่วนที่มากที่สุด ซึ่งสัดส่วนดังกล่าวแสดงให้เห็นว่าเศรษฐกิจกำลังดีขึ้นเรื่อยๆ

ในช่วงโควิดก็มีทั้งหมวดที่ราคาเพิ่มขึ้นและลดลง หมวดที่ราคาสินค้าเพิ่มขึ้นคือหมวดอาหาร ได้แก่ ผักแพรrop และอื่นๆ, เครื่องปรุงอาหาร, เนื้อสัตว์, ผักสดและร้อน เช่นฯ, เนื้อสัตว์ เป็นต้น และสัตว์น้ำ, น้ำมัน植物油 ไขมัน, เนื้อสัตว์สด, ผักสด, เครื่องปรุงรส อีกหมวดที่ราคาเพิ่มขึ้นเป็นกันก็คือหมวดเจ้ากินๆ (อุปกรณ์ทางการแพทย์) แต่ก็มีหมวดที่ราคาสินค้าลดลงเหมือนกันคือหมวดของเชื้อเพลิง พลังงานและยานพาหนะ ได้แก่ ยานพาหนะและน้ำมันเชื้อเพลิง, พลังงาน, น้ำมัน 91-95 ลิตร, กลุ่มอาหารสดและพลังงาน, เชื้อเพลิงในบ้าน, น้ำมันเชื้อเพลิง และหมวดพาหนะ การขนส่ง และการสื่อสาร

สิ่งที่มีผลต่อราคาก็มีทั้งหมวดที่ราคาเพิ่มขึ้นและลดลง แต่เมื่อเทียบตัวชี้ว่าค่าเดือนเดียวกันในปีก่อนหน้า สัดส่วนใหญ่ราคางบ ± ไม่เกิน 5 บาท

แม้ว่าสินค้าส่วนใหญ่ราคางบ ± ไม่เกิน 5 บาทแต่ก็มีสินค้าที่ราคาเพิ่มขึ้นและลดลงมากกว่าที่นั้น เช่น

- หมวดผักสด ราคาเพิ่มขึ้น 12.66 บาท
- หมวดน้ำมันเชื้อเพลิง ราคาลดลงไป 14.46 บาท

เศรษฐกิจเริ่มฟื้นฟูกลับมานานหลังจากสถานการณ์โควิด-19 เริ่มตื้นชื้นในช่วงครึ่งปีหลังของปี 2020

## ปัญหาที่พบ

- ตัวอักษรภาษาไทยยังมีปัญหาเรื่องสระหาย
- พบปัญหานี้ในเรื่องของการพิสูจน์ไฟไหม้ ตัวอักษรเป็นตัวเดิมกันเกินไป
- โคดในบางส่วนยังมีการซ้ำกัน อาจจะส่งผลให้รันนานขึ้น
- ใช้ชื่อโมดูลจากแหล่งเดียว ทำให้เก็บเสนอตได้ไม่หลากหลาย
- ไม่เข้าใจโคดลิ้งaoa เพราะทางผู้ให้บริการให้ค่าอธิบายที่ไม่ชัดเจน จึงอาจทำให้เก็บเสนอชื่อมูลในส่วนนี้ได้น้อยลง