

φυβλαςのblog

บล็อกของ phyblas (https://phyblas.hinaboshi.com)

ไทย
日本語 (https://phyblas.hinaboshi.com/nh)
中文 (https://phyblas.hinaboshi.com/zh)

หวัด

ถูกใจ

แชร์

ถูกใจ

แชร์

มีหนึ่งคนถูกใจสิ่งนี้

ถูกใจเป็นคนแรกในกลุ่มเพื่อนคุณเลิ

จัดการข้อมูลด้วย pandas เบื้องต้น บทที่ ๑๒: การรวมข้อมูลเป็นตารางเดียว

เขียนเมื่อ 2016/09/25 15:30

แก้ไขล่าสุด 2020/06/02 19:58

มีหลากหลายวิธีในการนำข้อมูลที่มีอยู่ในซีรีส์หรือเดตาเฟรมสองอันขึ้นไปมารวมเข้าด้วยกัน ในบทนี้จะพูดถึงวิธีการรวมโดยใช้ฟังก์ชัน pd.concat และเมธอด append

การเอาซีรีส์มาต่อกัน

เมื่อมีซีรีส์สองอันขึ้นไปแล้วจะเอามารวมกันอาจทำได้โดยใช้ฟังก์ชัน pd.concat โดยใส่ลิสต์ของซีรีส์ที่ต้องการรวมลงไป

ตัวอย่าง สร้างซีรีส์ขึ้นมา ๒ อัน

```
import pandas as pd
pokemon1 = pd.Series(['โป๊ปปโป', 'พีเจียน', 'พีจีอด'])
pokemon2 = pd.Series(['โอนิสึชิมะ', 'โอนิดริล'])
print(pokemon1)
print(pokemon2)
```



ได้

```
0    โป๊ปปโป
1    พีเจียน
2    พีจีอด
dtype: object
0    โอนิสึชิมะ
1    โอนิดริล
dtype: object
```

จากนั้นใช้ concat

```
print(pd.concat([pokemon1, pokemon2]))
```

ได้

```
0    โป๊ปปโป
1    พีเจียน
2    พีจีอด
0    โอนิสึชิมะ
1    โอนิดริล
dtype: object
```

ซีรีส์จะถูกนำมาต่อกัน แต่ก็เห็นว่าดัชนีมีการซ้ำกันเกิดขึ้น โดยยึดตามดัชนีเดิมโดยไม่มีการเปลี่ยนให้อัตโนมัติ

~ เกียวกับเรา ~
(/naenamtua)

สารบัญ

รวมคำแปลวลีเด็ดจากญี่ปุ่น
(https://hinaboshi.com)

~ python ~ (/saraban/python)
มอดูลต่างๆ (/saraban/pythomoi)
-- numpy
-- matplotlib
(/saraban/numpy_matplotlib)
-- pandas (/saraban/pandas)
-- opencv (/saraban/opencvpytho)
-- pytorch (/saraban/pytorch)
การเรียนรู้ของเครื่อง
(/saraban/kanrianrukhangkhru)
-- โครงข่าย
ประสาทเทียม
(/saraban/khang_khai_prasat_ti)

maya (/saraban/maya)

javascript (/saraban/javascript)

ความน่าจะเป็น
(/saraban/khwamnachapen)

บันทึกในญี่ปุ่น (/saraban/nihon)

บันทึกในจีน (/saraban/zhonggu)
-- บันทึกในปักกิ่ง (/saraban/beijing)
-- บันทึกในฮ่องกง
(/saraban/hoenggong)
-- บันทึกในมาเก๊า
(/saraban/hoenggong)

บันทึกในไต้หวัน (/saraban/taiwan)

บันทึกในยุโรปเหนือ
(/saraban/nordeuropa)

หากต้องการให้เวลาที่ต่อกันแล้วสัมพันธ์เดิมทิ้งไปเลยก็ให้ใส่ ignore_index=1 เท่านั้นดัชนีก็จะจัดเรียงใหม่ไล่ตั้งแต่ 0 ไม่ว่าจะเคยเป็นอะไรมาก่อน

```
print(pd.concat([pokemon1,pokemon2],ignore_index=1))
```

```
ได้
0      โปโปโป
1      พ็เจียน
2      พ็จ๊อต
3      โอนิสซี่เมะ
4      โอนิดริล
dtype: object
```

การรวมซีรีส์เป็นเดตาเฟรม

การรวมซีรีส์นั้นนอกจากจะเอามาต่อกันให้เป็นสายยาวขึ้นแล้วยังสามารถรวมกันในลักษณะที่ใช้ซีรีส์อันหนึ่งเป็นคอลัมน์หนึ่งเพื่อสร้างเดตาเฟรมขึ้นมา

ในกรณีแบบนี้ให้ใส่ axis=1 แล้วก็ควรจะเป็นซีรีส์ที่มีการตั้งชื่อไว้ด้วย แล้วชื่อนั้นจะกลายมาเป็นชื่อคอลัมน์ ถ้าไม่ได้ตั้งก็จะถูกตั้งเองเป็นตัวเลข

ตัวอย่าง

```
saiphan = pd.Series(['อุริมุ','อิโนมุ','แมมมุ'],name='สายพันธุ์')
mailek = pd.Series([220,221,473],name='หมายเลข')
print(saiphan)
print(mailek)
```

```
ได้
0      อุริมุ
1      อิโนมุ
2      แมมมุ
Name: สายพันธุ์, dtype: object
0      220
1      221
2      473
Name: หมายเลข, dtype: int64
```



นำมา concat

```
print(pd.concat([saiphan,mailek],axis=1))
```

```
ได้


|   | สายพันธุ์ | หมายเลข |
|---|-----------|---------|
| 0 | อุริมุ    | 220     |
| 1 | อิโนมุ    | 221     |
| 2 | แมมมุ     | 473     |


```

การรวมกันในกรณี axis=1 นี้จะยึดตามเลขดัชนี คือรวมแถวที่เลขดัชนีเหมือนกันเข้าด้วยกัน ถ้าดัชนีของทั้ง ๒ ซีรีส์ต่างกันผลที่ได้ก็จะมี NaN

ตัวอย่างเช่น

```
saiphan = pd.Series(['โนคจิจิ','ฮาริเซน','ทสึโบะทสึโมะ'],index=[206,211,213],name='สายพันธุ์')
chanit = pd.Series(['ธรรมดา','น้ำพิษ','แมลง/ต่อสู'],index=[206,211,214],name='ชนิด')
print(saiphan)
print(chanit)
print(pd.concat([saiphan,chanit],axis=1))
```



บันทึกในปัยเทศอื่นๆ

日本語 (/saraban/langdaben/hi.com/nh)

中文 (https://phyblas.hinaboshi.com/zh)

เรียนภาษาจีน (/saraban/gakken)

qiita (/qiita)

บทความอื่นๆ (/saraban/uenuen)

บทความแบ่งตามหมวด (/muat)

==เลือกหมวด==

ติดตามอัปเดตของบล็อกได้ที่แฟนเพจ



ค้นหาบทความ

ค้น

บทความล่าสุด

ความน่าจะเป็นเบื้องต้นสำหรับเขียนโปรแกรม บทที่ ๒๒: การแจกแจงดิริคเล (/gasha22)

ความน่าจะเป็นเบื้องต้นสำหรับเขียนโปรแกรม บทที่ ๒๑: การแจกแจงอเนกนามและการแจกแจงแบบหมวดหมู่ (/gasha21)

ความน่าจะเป็นเบื้องต้นสำหรับเขียนโปรแกรม บทที่ ๒๐: การแจกแจงสวิตเดนดท์ (/gasha20)

ความน่าจะเป็นเบื้องต้นสำหรับเขียนโปรแกรม บทที่ ๑๙: การคาดการณ์ค่าที่พหามิเตอร์การแจกแจงมีความไม่แน่นอน (/gasha19)

ความน่าจะเป็นเบื้องต้นสำหรับเขียนโปรแกรม บทที่ ๑๘: การแจกแจงโคกำลังสอง (/gasha18)

บทความแนะนำ

ภาษาจีนแบ่งเป็นสำเนียงอะไรบ้าง มีความแตกต่างกันมากแค่ไหน (/20200520)

ทำความเข้าใจระบอบประชาธิปไตยจากประวัติศาสตร์ความเป็นมา (/prawatsat_prachathipatai)

เรียนรู้วิธีการใช้ regular expression (regex) (/20190709)

ได้

206	โนคจจิ	
211	ซาริเซน	
213	ทสึโบะทสึโมะ	
Name: สายพันธุ์, dtype: object		
206	ธรรมดา	
211	น้ำ/พืช	
214	แมลง/ต่อสู	
Name: ชนิด, dtype: object		
	สายพันธุ์	ชนิด
206	โนคจจิ	ธรรมดา
211	ซาริเซน	น้ำ/พืช
213	ทสึโบะทสึโมะ	NaN
214	NaN	แมลง/ต่อสู

การรวมเดตาเฟรมเข้าด้วยกันเป็นตารางเดียว

เช่นเดียวกันซีรีส์ เมื่อจะเอาเดตาเฟรมสองอันขึ้นไปมารวมกันก็สามารถใช้ pd.concat โดยใส่ลิสต์ของเดตาเฟรมที่ต้องการรวมลงไป

ปกติแล้วถ้าไม่ได้กำหนดคีย์เวิร์ด axis หรือกำหนด axis=0 จะเป็นการรวมในแนวนั่ง คือรวมที่มีคอลัมน์ชื่อเหมือนกันแต่ดัชนีต่างกัน ผลที่ได้จะได้เดตาเฟรมที่ต่อกันเป็นแถวยาวมีจำนวนดัชนีเท่ากับเดตาเฟรมทั้งหมดรวมกัน

ถ้ากำหนด axis=1 จะเป็นการรวมกันในแนวนอน คือรวมที่มีชื่อดัชนีเหมือนกันแต่เป็นคนละชื่อคอลัมน์ แล้วจะได้เดตาเฟรมที่มีจำนวนคอลัมน์มากขึ้น

๒ แบบนี้มีหลักการเหมือนกันเพียงแคกลับแนวตั้งแนวนอนกัน ดังนั้นในที่นี้จะยกตัวอย่างแค่กรณีแนวนั่ง

ตัวอย่าง สร้างเดตาเฟรมเก็บข้อมูลไปเกมอนขึ้นมา ๓ ตาราง

```
pokemon1 = pd.DataFrame([
    ['ธรรมดา',55,55,55],
    ['น้ำ',130,65,65],
    ['ไฟฟ้า',65,65,130],
    ['ไฟ',65,130,65]],
    index=['อีวุย','เซาเวอร์ส','ฮันเดอร์ส','นุสเดอร์'],
    columns=['ชนิด','HP','พลังโจมตี','ความไว'])
pokemon2 = pd.DataFrame([
    ['พลังจิต',65,60,110],
    ['มาร',95,110,65]],
    index=['เอย์ฟี่','แบล็กกี้'],
    columns=['ชนิด','HP','พลังป้องกัน','ความไว'])
pokemon3 = pd.DataFrame({
    'ชนิด':['พืช','น้ำแข็ง'],
    'HP':[65,65],
    'พลังโจมตี':[110,60],
    'พลังป้องกัน':[130,110]},
    index=['ลึเฟีย','เกลเซีย'])
print(pokemon1)
print(pokemon2)
print(pokemon3)
```



ได้

	ชนิด	HP	พลังโจมตี	ความไว
อีวุย	ธรรมดา	55	55	55
เซาเวอร์ส	น้ำ	130	65	65
ฮันเดอร์ส	ไฟฟ้า	65	65	130
นุสเดอร์	ไฟ	65	130	65

หลักการเขียนทับศัพท์ภาษาจีนกลาง

日本語 (<https://phyblas.hinaboshi.com/nh>)

中文 (<https://phyblas.hinaboshi.com/zh>)

หลักการเขียนทับศัพท์ภาษาจีนกลาง (/20181019)

๑ ในภาษาญี่ปุ่นออกเสียง "ก" หรือ "ง" กันแน่ (/20181017)

ทำความเข้าใจกับปัญหาประดิษฐ์และการเรียนรู้ของเครื่อง (/panyapradit)

ค้นพบระบบดาวเคราะห์ ๘ ดวง เมืองหลังความสำเร็จคือปัญญาประดิษฐ์ (AI) (/20171216)

หอดูดาวโบราณปักกิ่ง ตอนที่ ๑: แทนสังเกตการณ์และสวนดอกไม้ (/20170203)

พิพิธภัณฑ์สถาปัตยกรรมโบราณปักกิ่ง (/20151028)

เที่ยวเมืองตานตง ล่องเรือในน่านน้ำเกาหลีเหนือ (/20140927)

บันทึกการเที่ยวสวีเดน 1-12 พ.ค. 2014 (/20140514)

แนะนำองค์การวิจัยและพัฒนาการสำรวจอวกาศญี่ปุ่น (JAXA) (/20140116)

เล่าประสบการณ์ค้าขายบนเรือทางการทางดาราศาสตร์โดยโชวเคนโด 10 - 16 พ.ย. 2013 (/20131205)

ตระเวนเที่ยวตามรอยจากของอนิเมะในญี่ปุ่น (/20130820)

เที่ยวชมหอดูดาวที่ฐานสังเกตการณ์เซิงหลง (/20130529)

บันทึกการเที่ยวญี่ปุ่นครั้งแรกในชีวิต - ทุกอย่างเริ่มต้นที่สนามบินนานาชาติคันไซ (/20130118)

หลักการเขียนทับศัพท์ภาษาญี่ปุ่น (/20130109)

ทำไมจึงไม่ควรเขียนวรรณยุกต์เวลาทับศัพท์ภาษาต่างประเทศ (/20121010)

ทำไมถึงอยากมาเรียนตอนอก (/20120622)

เหตุผลอะไรที่ต้องใช้ภาษาวิบัติ? (/20101106)

บทความแต่ละเดือน

2020年	1月	2月
	(/raiduean/202001)	(/raiduean/202002)
	5月	6月
	(/raiduean/202005)	(/raiduean/202006)
	9月	10月
	(/raiduean/202009)	
2019年	1月	2月
	(/raiduean/201901)	(/raiduean/201902)
	5月	6月
	(/raiduean/201905)	(/raiduean/201906)
	9月	10月
	(/raiduean/201909)	(/raiduean/201910)
2018年	1月	2月
	(/raiduean/201801)	(/raiduean/201802)

	ชนิด	HP	พลังป้องกัน	ความไว
เอย์ฟี่	พลังจิต	65	60	110
แมลิกกี้	มาร	95	110	65

	HP	ชนิด	พลังป้องกัน	พลังโจมตี
ลีเฟีย	65	พืช	130	110
เกลเซีย	65	น้ำแข็ง	110	60

นำมารวมกัน

```
print(pd.concat([pokemon1,pokemon2,pokemon3]))
```

ได้

	HP	ความไว	ชนิด	พลังป้องกัน	พลังโจมตี
อีว	55	55.0	ธรรมชาติ	NaN	55.0
เซาเวอร์ส	130	65.0	น้ำ	NaN	65.0
ฮันเดอร์ส	65	130.0	ไฟฟ้า	NaN	65.0
บัสเตอร์	65	65.0	ไฟ	NaN	130.0
เอย์ฟี่	65	110.0	พลังจิต	60.0	NaN
แมลิกกี้	95	65.0	มาร	110.0	NaN
ลีเฟีย	65	NaN	พืช	130.0	110.0
เกลเซีย	65	NaN	น้ำแข็ง	110.0	60.0

จะเห็นว่าทั้ง ๓ ตารางก่อนรวมมีคอลัมน์ที่ทั้งเหมือนกันและต่างกัน เมื่อรวมกันแล้วคอลัมน์ที่ไม่มีจะได้ค่าเป็น NaN และลำดับของคอลัมน์ก็ถูกจัดเรียงใหม่โดยอัตโนมัติได้เป็นไปตามที่เรียงตอนแรกด้วย

บางครั้งเราอาจไม่ได้ต้องการทุกคอลัมน์แต่อาจต้องการเหลือแค่บางคอลัมน์ กรณีแบบนี้อาจใส่คีย์เวิร์ด join_axes โดยใส่ค่าเป็นซีรีส์ของชื่อคอลัมน์ที่ต้องการ เช่น

```
c = pd.Series(['พลังโจมตี', 'พลังป้องกัน', 'ความไว'])
print(pd.concat([pokemon1,pokemon2,pokemon3], join_axes=[c]))
```

ได้

	พลังโจมตี	พลังป้องกัน	ความไว
อีว	55.0	NaN	55.0
เซาเวอร์ส	65.0	NaN	65.0
ฮันเดอร์ส	65.0	NaN	130.0
บัสเตอร์	130.0	NaN	65.0
เอย์ฟี่	NaN	60.0	110.0
แมลิกกี้	NaN	110.0	65.0
ลีเฟีย	110.0	130.0	NaN
เกลเซีย	60.0	110.0	NaN

ให้หวังว่าค่าที่ต้องใส่ต้องเป็นลิสต์แม้ว่าจะมีอยู่แค่ตัวเดียวก็ตาม (ในที่นี้คือเป็น join_axes=[c] ไม่ใช่ join_axes=c) ที่จริงแล้วคีย์เวิร์ดนี้ทำให้รองรับกรณีที่เป็นแบบคอลัมน์หลายแถว ซึ่งในกรณีแบบนี้จะต้องใส่ซีรีส์ตามจำนวนเท่านั้น

ถ้าต้องการใช้คอลัมน์ของตารางแรกสักอันก็อาจใช้ได้โดยใช้ ตาราง.columns เช่น

```
print(pd.concat([pokemon1,pokemon2,pokemon3], join_axes=[pokemon3.columns]))
```

ได้

	HP	ชนิด	พลังป้องกัน	พลังโจมตี
อีว	55	ธรรมชาติ	NaN	55.0
เซาเวอร์ส	130	น้ำ	NaN	65.0
ฮันเดอร์ส	65	ไฟฟ้า	NaN	65.0
บัสเตอร์	65	ไฟ	NaN	130.0

5月 6月

ไทย

日本語 (https://phyblas.hinaboshi.com/nh)

中文 (https://phyblas.hinaboshi.com/zh)

2017年

1月 2月

(/raiduean/201701)(/raiduean/201702)(/ra

5月 6月

(/raiduean/201705)(/raiduean/201706)(/ra

9月 10月

(/raiduean/201709)(/raiduean/201710)(/ra

2016年

1月 2月

(/raiduean/201601)(/raiduean/201602)(/ra

5月 6月

(/raiduean/201605)(/raiduean/201606)(/ra

9月 10月

(/raiduean/201609)(/raiduean/201610)(/ra

ค้นบทความเก่ากว่านั้น

(/raiduean)

	HP	ชนิด	พลังป้องกัน	พลังโจมตี	ไทย
เอย์ฟี่	65	พลังจิต	60.0	NaN	日本語 (https://phyblas.hinaboshi.com/nh)
แมลิกกี้	95	มาร	110.0	NaN	中文 (https://phyblas.hinaboshi.com/zh)
ลีเฟีย	65	พืช	130.0	110.0	
เกลเซีย	65	น้ำแข็ง	110.0	60.0	

นอกจากนี้กรณีที่ต้องการเก็บเฉพาะคอลัมน์ที่ทุกเตาเฟรมต่างก็มีเหมือนกันหมดก็อาจใช้ดีวีรด์ join เพิ่มลงไป โดยใส่เป็น join='inner'

ปกติแล้วถ้าไม่ใส่ดีวีรด์นี้ลงไปจะหมายถึง join='outer' หมายถึงใช้ทุกคอลัมน์ที่มีจากทุกเตาเฟรม แต่ถ้า inner ก็จะหมายถึงใช้เฉพาะคอลัมน์ที่มีในทุกอัน ตัวอย่างเช่น

```
print(pd.concat([pokemon1,pokemon2,pokemon3],join='inner'))
```

ได้

	HP	ชนิด
อีวูบ	55	ธรรมชาติ
เซาเวอร์ส	130	น้ำ
ธันเดอร์ส	65	ไฟฟ้า
บอสเตอร์	65	ไฟ
เอย์ฟี่	65	พลังจิต
แมลิกกี้	95	มาร
ลีเฟีย	65	พืช
เกลเซีย	65	น้ำแข็ง

การรวมแล้วทำเป็นดัชนีซ้อน

ปกติเวลาที่ใช้ concat เพื่อต่อแถวเตาเฟรมหากดัชนีมีตัวที่ซ้ำกันผลที่ได้ก็จะเป็นดัชนีที่ซ้ำกัน ซึ่งอาจแก้ไขได้ด้วยการใช้ดีวีรด์ ignore_index

แต่วิธีนั้นจะทำให้สูญเสียดัชนีเดิมหมด ที่จริงแล้วมีวิธีที่ดีกว่านั้นนั่นจะทำให้ซับซ้อนขึ้นเล็กน้อย นั่นคือการเพิ่มข้อมูลที่ว่าข้อมูลแถวนั้นมาจากเตาเฟรมตัวไหนลงไปใส่เป็นดัชนีซ้อนลงไป

สามารถทำได้โดยใช้ดีวีรด์ keys แล้วตามด้วยลิสต์ของค่าดัชนีที่จะใช้แทนแต่ละตาราง

ตัวอย่าง ลองสร้างตารางข้อมูลโปเกมอนตัวหลักในแต่ละภาค

```
pokemon1 = pd.DataFrame([
    ['ฟุซิจิคาเนะ', 'พืช'], ['อีโดคาเงะ', 'ไฟ'], ['เซนิงาเมะ', 'น้ำ']],
    index=[1,4,7],columns=['สายพันธุ์', 'ชนิด'])
pokemon2 = pd.DataFrame([
    ['อีโดริดา', 'พืช'], ['อีโนอาราชิ', 'ไฟ'], ['วาณโนโกะ', 'น้ำ']],
    index=[1,4,7],columns=['สายพันธุ์', 'ชนิด'])
pokemon3 = pd.DataFrame([
    ['คิโมริ', 'พืช'], ['อาซาโม', 'ไฟ'], ['มิชิโงโร', 'น้ำ']],
    index=[1,4,7],columns=['สายพันธุ์', 'ชนิด'])
print(pokemon1)
print(pokemon2)
print(pokemon3)
```



ได้

	สายพันธุ์	ชนิด
1	ฟุซิจิคาเนะ	พืช
4	อีโดคาเงะ	ไฟ
7	เซนิงาเมะ	น้ำ

	สายพันธุ์	ชนิด
1	ชิโครีดา	พืช
4	ฮิโนอาราชิ	ไฟ
7	วานิโนโกะ	น้ำ

	สายพันธุ์	ชนิด
1	คิโมริ	พืช
4	อาซาโม	ไฟ
7	มิชิโงโรว	น้ำ

ไทย
日本語 (https://phyblas.hinaboshi.com/nh)
中文 (https://phyblas.hinaboshi.com/zh)

นำมารวมกันให้เป็นดัชนีซ้อนกัน กำหนดดัชนีเป็นตัวเลขลำดับภาค

```
print(pd.concat([pokemon1,pokemon2,pokemon3],keys=[1,2,3]))
```

ได้

		สายพันธุ์	ชนิด
1	1	ฟุจิदानะ	พืช
	4	ฮิโดคาเงะ	ไฟ
	7	เซนิงาเมะ	น้ำ
2	1	ชิโครีดา	พืช
	4	ฮิโนอาราชิ	ไฟ
	7	วานิโนโกะ	น้ำ
3	1	คิโมริ	พืช
	4	อาซาโม	ไฟ
	7	มิชิโงโรว	น้ำ

นอกจากนี้ยังมีอีกวิธีในการรวมให้ได้ดัชนีซ้อนในลักษณะแบบนี้โดยไม่ใช้คีย์เวิร์ด keys นั่นคือแทนที่จะใส่ค่าในรูปลิสต์ก็ใส่ในรูปดิกชันนารีโดยที่มีคีย์เป็นค่าดัชนีที่ต้องการ

เช่น หากต้องการผลเหมือนตัวอย่างข้างต้นก็พิมพ์

```
print(pd.concat({1:pokemon1,2:pokemon2,3:pokemon3}))
```

การต่อขยายซีรีส์

หากมีซีรีส์อยู่อันหนึ่งแล้วต้องการเอาข้อมูลจากซีรีส์อีกอันมาใส่สามารถใช้เมธอด append ได้

เพียงแต่ว่าเมธอด appand นี้ว่ากันตามจริงแล้วก็เหมือนเป็นฟังก์ชัน np.concat ที่เขียนในรูปแบบเมธอดของเดตาเฟรมอันหนึ่ง เพราะการ "เอาอันหนึ่งมาต่อกับอีกอัน" กับ "เอาสองอันมาต่อกัน" ก็คือสิ่งเดียวกัน

ต่างกันก็ตรงที่ว่า append จะใช้อันหนึ่งเป็นฐานของเมธอด และอันที่เหลือเป็นอาร์กิวเมนต์

ตัวอย่างการใช้

```
pokemon1 = pd.Series(['อุปา','บูโธ'],index=[194,195])
pokemon2 = pd.Series(['ฮัสโม','ซาสึเมโร','รุนิปโปะ'],index=[270,271,272])
pokemon3 = pd.Series(['มูยเซล','โฟลเซล'],index=[418,419])
print(pokemon1.append([pokemon2,pokemon3]))
```



ได้

194	อุปา
195	บูโธ
270	ฮัสโม
271	ซาสึเมโร

```
272   รุนบีปปะ
418   นุยเซล
419   โพลเซล
dtype: object
```

ไทย

日本語 (<https://phyblas.hinaboshi.com/nh>)中文 (<https://phyblas.hinaboshi.com/zh>)

append ไม่ได้ทำการเปลี่ยนแปลงใดๆกับเดตาเฟรมที่เป็นฐาน แต่จะทำการคืนค่าเดตาเฟรมตัวใหม่ที่อยู่ในสภาพหลังต่อขยายมาแล้วเท่านั้น

การต่อขยายเดตาเฟรม

เช่นเดียวกับซีรีส์ เดตาเฟรมก็มีเมธอด append ไว้สำหรับเอาเดตาเฟรมตัวอื่นมาต่อให้กับเดตาเฟรมตัวหนึ่ง

ตัวอย่าง

```
pokemon1 = pd.DataFrame([
    ['ดัมเบอร์', 'เหล็ก/พลังจิต', 0.6, 95.2],
    ['เมทัง', 'เหล็ก/พลังจิต', 0.6, 95.2],
    ['เมทาโกรอส', 'เหล็ก/พลังจิต', 1.6, 550.0]],
    index=[374, 375, 376],
    columns=['สายพันธุ์', 'ชนิด', 'ส่วนสูง', 'น้ำหนัก'])
pokemon2 = pd.DataFrame([
    ['ทัดลีเบย์', 'มังกร', 0.6, 42.1],
    ['โคโมรุ', 'มังกร', 1.1, 110.5],
    ['โบมันเดอร์', 'มังกร', 1.5, 102.6]],
    index=[371, 372, 373],
    columns=['สายพันธุ์', 'ชนิด', 'ส่วนสูง', 'น้ำหนัก'])
print(pokemon1)
print(pokemon2)
print(pokemon1.append(pokemon2))
```



ได้

	สายพันธุ์	ชนิด	ส่วนสูง	น้ำหนัก
374	ดัมเบอร์	เหล็ก/พลังจิต	0.6	95.2
375	เมทัง	เหล็ก/พลังจิต	0.6	95.2
376	เมทาโกรอส	เหล็ก/พลังจิต	1.6	550.0

	สายพันธุ์	ชนิด	ส่วนสูง	น้ำหนัก
371	ทัดลีเบย์	มังกร	0.6	42.1
372	โคโมรุ	มังกร	1.1	110.5
373	โบมันเดอร์	มังกร	1.5	102.6

	สายพันธุ์	ชนิด	ส่วนสูง	น้ำหนัก
374	ดัมเบอร์	เหล็ก/พลังจิต	0.6	95.2
375	เมทัง	เหล็ก/พลังจิต	0.6	95.2
376	เมทาโกรอส	เหล็ก/พลังจิต	1.6	550.0
371	ทัดลีเบย์	มังกร	0.6	42.1
372	โคโมรุ	มังกร	1.1	110.5
373	โบมันเดอร์	มังกร	1.5	102.6

append นั้นในกรณีที่ใช้กับเดตาเฟรมนอกจากใช้ตัวที่ต่อเป็นเดตาเฟรมแล้วก็ยังอาจใช้ซีรีส์มาเป็นตัวต่อได้อีกด้วย ซึ่งถ้าเป็น concat จะใช้แบบนี้ไม่ได้อีก โดยที่ซีรีส์ที่จะใช้นั้นต้องมีจำนวนแถวเท่ากับจำนวนคอลัมน์ของเดตาเฟรมพอดี

ตัวอย่าง

```
pokemon3 = pd.Series(['เลิฟคัส', 'น้ำ', 0.6, 8.7],
    index=['สายพันธุ์', 'ชนิด', 'ส่วนสูง', 'น้ำหนัก'],
    name=370)
print(pokemon1.append(pokemon3))
```



ได้

	สายพันธุ์	ชนิด	ส่วนสูง	น้ำหนัก
374	ดัมเบอร์	เหล็ก/พลังจิต	0.6	95.2
375	เมทิง	เหล็ก/พลังจิต	0.6	95.2
376	เมทาโกรส	เหล็ก/พลังจิต	1.6	550.0
370	เลฟคัส	น้ำ	0.6	8.7

ไทย

日本語 (<https://phyblas.hinaboshi.com/nh>)

中文 (<https://phyblas.hinaboshi.com/zh>)

อ้างอิง

<http://sinhrks.hatenablog.com/entry/2015/01/28/073327> (<http://sinhrks.hatenablog.com/entry/2015/01/28/073327>)
<http://d.hatena.ne.jp/Tommy1/20131211/1386719956> (<http://d.hatena.ne.jp/Tommy1/20131211/1386719956>)
<http://qiita.com/airtoxin/items/d66a22c5c7074e23be17> (<http://qiita.com/airtoxin/items/d66a22c5c7074e23be17>)
<http://pythondatascience.plavox.info/pandas/pandas> でデータフレームを結合 (<http://pythondatascience.plavox.info/pandas/pandas>)
<http://pythondatascience.plavox.info/pandas/pandas> のデータフレームに行や列を追加する (<http://pythondatascience.plavox.info/pandas/pandas>)

<< บทที่แล้ว (<https://phyblas.hinaboshi.com/yancham11>) บทถัดไป >> (<https://phyblas.hinaboshi.com/yancham13>)
หน้าสารบัญ (<https://phyblas.hinaboshi.com/saraban/pandas>)

หวัด ถูกใจ ☐ แชร์ ถูกใจ แชร์ มีหนึ่งคนถูกใจสิ่งนี้ ถูกใจเป็นคนแรกในกลุ่มเพื่อนคุณสิ



ความคิดเห็น 0 รายการ เรียงลำดับตาม เก่าสุด

เพิ่มความเห็น...

[ปลั๊กอินความคิดเห็นบน Facebook](#)
ดูสถิติของหน้านี้ (/dusathiti/yancham12)

หมวดหมู่ (/muat)

-- คอมพิวเตอร์

(/muat/%E0%B8%84%E0%B8%AD%E0%B8%A1%E0%B8%9E%E0%B8%B4%E0%B8%A7%E0%B9%80%E0%B8%95%E0%B8%AD%E0%B8%A3%E0%B9%8C)

>> เขียนโปรแกรม

(/muat/%E0%B9%80%E0%B8%82%E0%B8%B5%E0%B8%A2%E0%B8%99%E0%B9%82%E0%B8%9B%E0%B8%A3%E0%B9%81%E0%B8%81%E0%B8%A3%E0%B9%81)

>> python (/muat/python) >> pandas (/muat/pandas)

ไม่อนุญาตให้นำเนื้อหาของบทความไปลงที่อื่นโดยไม่ได้อุญาตโดยเด็ดขาด หากต้องการนำบางส่วนไปลงสามารถทำได้โดยไม่ต้องไม่ใช้การก๊อปปี้แต่ให้เปลี่ยนคำพูดเป็นของตัวเอง หรือไม่ก็เขียนในลักษณะการยกข้อความอ้างอิง และไม่ว่ากรณีไหนก็ตาม ต้องให้เครดิตพร้อมใส่ลิงก์ของบทความที่มีการใช้เนื้อหาเสมอ