レシピ54 複数のカラムと関数のグループ分け（P149）

- ここでは以下を求める
- 曜日ごとに、全航空会社で、キャンセル便を求める
- 曜日ごとに、全航空会社で、キャンセル便と行き先変更の、数とパーセントを求める
- 出発および到着飛行場ごとに、便の総数、キャンセル便の数とパーセント、飛行時間の平均と分散を求める

In [53]:
```
flights.head(3).append(flights.tail(2))
```

Out[53]:

	MONTH	DAY	WEEKDAY	AIRLINE	ORG_AIR	DEST_AIR	SCHED_DEP	DEP_DELAY	AIR_TIME	DIST	SCHED_ARR	ARR_DELAY	DIVERTED	CANCELLED
0	1	1	4	WN	LAX	SLC	1625	58.0	94.0	590	1905	65.0	0	0
1	1	1	4	UA	DEN	IAD	823	7.0	154.0	1452	1333	-13.0	0	0
2	1	1	4	MQ	DFW	VPS	1305	36.0	85.0	641	1453	35.0	0	0
58490	12	31	4	WN	MSP	ATL	525	39.0	124.0	907	855	34.0	0	0
58491	12	31	4	OO	SFO	BOI	859	5.0	73.0	522	1146	-1.0	0	0

（1）曜日ごとに、全航空会社で、キャンセル便を求める

- 2つは同じことをやっている。上段はSeries、下段はDataFrameが戻り値になる
- `.agg({'集約カラム':'集約関数'})` だとDataFrameで戻る

In [54]:
```
flights.groupby(['AIRLINE', 'WEEKDAY'])['CANCELLED'].agg('sum').head()
```

Out[54]:
```
AIRLINE  WEEKDAY
AA       1          41
         2           9
         3          16
         4          20
         5          18
Name: CANCELLED, dtype: int64
```

In [55]:
```
flights.groupby(['AIRLINE', 'WEEKDAY']).agg({'CANCELLED':'sum'}).head()
```

Out[55]:

		CANCELLED
AIRLINE	**WEEKDAY**	
AA	**1**	41
	2	9
	3	16
	4	20
	5	18

（2）曜日ごとに、全航空会社で、キャンセル便と行き先変更の、数とパーセントを求める

- 集約カラムが複数の場合、リストの入れ子になる

In [56]:
```
flights.groupby(['AIRLINE', 'WEEKDAY'])\
    [['CANCELLED', 'DIVERTED']].agg(['sum', 'mean']).head(7)

# 複数の集約カラムの場合は、[[*, *]]となる（テキストは間違っている）
# でないとエラーになる
# FutureWarning: Indexing with multiple keys (implicitly converted to a tuple of keys) will be deprecated,
# use a list instead.
```

Out[56]:

		CANCELLED		DIVERTED	
		sum	mean	sum	mean
AIRLINE	**WEEKDAY**				
AA	**1**	41	0.032106	6	0.004699
	2	9	0.007341	2	0.001631
	3	16	0.011949	2	0.001494
	4	20	0.015004	5	0.003751
	5	18	0.014151	1	0.000786
	6	21	0.018667	9	0.008000
	7	29	0.021837	1	0.000753

（3）出発および到着飛行場ごとに、便の総数、キャンセル便の数とパーセント、飛行時間の平均と分散を求める

- 集約関数`size`：NaNを含む行の総数を返す（cf.`count`はNaNを除く）

In [57]:
```
group_cols = ['ORG_AIR', 'DEST_AIR']
agg_dict = {'CANCELLED':['sum', 'mean', 'size'],
            'AIR_TIME':['mean', 'var']}

flights.groupby(group_cols).agg(agg_dict).head()
```

Out[57]:

		CANCELLED			AIR_TIME	
		sum	mean	size	mean	var
ORG_AIR	**DEST_AIR**					
ATL	**ABE**	0	0.0	31	96.387097	45.778495
	ABQ	0	0.0	16	170.500000	87.866667
	ABY	0	0.0	19	28.578947	6.590643
	ACY	0	0.0	6	91.333333	11.466667
	AEX	0	0.0	40	78.725000	47.332692

★（補足）集約パターン

- gropubyは複数の構文があるため、わかりにくい

（ア）集約カラム`agg`でdictを使用する。これが最も柔軟性が高い！

- 集約カラム別に集約関数をセットできる
- 集約カラムが複数かつ***集約関数が別々*** の場合はこれしかない
- 集約関数が同じor1つの場合は下の記法が楽

In [58]:
```
group_cols = ['ORG_AIR', 'DEST_AIR']
agg_dict = {'CANCELLED':['sum', 'mean', 'size'],
            'AIR_TIME':['mean', 'var']}

flights.groupby(group_cols).agg(agg_dict).head()
```

Out[58]:

		CANCELLED			AIR_TIME	
		sum	mean	size	mean	var
ORG_AIR	**DEST_AIR**					
ATL	**ABE**	0	0.0	31	96.387097	45.778495
	ABQ	0	0.0	16	170.500000	87.866667
	ABY	0	0.0	19	28.578947	6.590643
	ACY	0	0.0	6	91.333333	11.466667
	AEX	0	0.0	40	78.725000	47.332692

（イ）同じ集約関数を使うのであればこれも可

- 集約カラムごとに、**別々の集約関数は適用できない**
- 複数の集約カラムの場合は`[[*, *]]`で囲む
- 集約カラムが1つの場合はSeriesで返る

In [59]:
```
flights.groupby(group_cols)[['CANCELLED','AIR_TIME']].agg(['size','count']).head()
#                            グループ分けカラム                集約カラム                （集約カラムに対し同じ）集約関数
```

Out[59]:

		CANCELLED		AIR_TIME	
		size	count	size	count
ORG_AIR	**DEST_AIR**				
ATL	**ABE**	31	31	31	31
	ABQ	16	16	16	16
	ABY	19	19	19	19
	ACY	6	6	6	6
	AEX	40	40	40	40

（ウ）集約関数を直接使いたい場合（例：.size()）

- 集約関数は**1つしか使えない**

In [60]:
```
flights.groupby(group_cols)[['CANCELLED','AIR_TIME']].size().head()
#             グループ分けカラム         集約カラム         .集約関数1つ
```

Out[60]:
```
ORG_AIR  DEST_AIR
ATL      ABE         31
         ABQ         16
         ABY         19
         ACY          6
         AEX         40
dtype: int64
```

（エ）集約カラムを指定しなければ、グループ分けカラムで集約される

In [61]:
```
flights.groupby(group_cols).size()
#                           グループ分けカラム      .集約関数
```

Out[61]:
```
ORG_AIR  DEST_AIR
ATL      ABE          31
         ABQ          16
         ABY          19
         ACY           6
         AEX          40
                    ...
SFO      SNA         122
         STL          20
         SUN          10
         TUS          20
         XNA           2
Length: 1130, dtype: int64
```