

```
import numpy as np
import pandas as pd
from datascience import *
```

```
/usr/local/lib/python3.7/dist-packages/datascience/tables.py:17: MatplotlibDeprecationWarning: The 'warn' param
matplotlib.use('agg', warn=False)
/usr/local/lib/python3.7/dist-packages/datascience/util.py:10: MatplotlibDeprecationWarning: The 'warn' paramet
matplotlib.use('agg', warn=False)
```

```
from google.colab import files
files.upload()
```

Choose Files top\_movies\_by\_title

- **top\_movies\_by\_title.csv**(text/csv) - 11026 bytes, last modified: 9/27/2021 - 100% done

Saving top\_movies\_by\_title.csv to top\_movies\_by\_title.csv

```
{'top_movies_by_title.csv': b'Title,Studio,Gross,Gross (Adjusted),Year\n101 Dalmatians,Disney,144880014,8692801
```

```
table1 = Table.read_table('top_movies_by_title.csv')
```

```
a = table1.group('Studio').sort('count', descending=True)
print(a)
col = a.where('Studio', 'Columbia')
fox = a.where('Studio', 'Fox')
print('\n{} Movies made by Columbia, {} Movies made by Fox'.format(int(col[1]), int(fox[1])))
```

```
↳ Studio | count
Warner Bros. | 29
Buena Vista (Disney) | 29
Fox | 26
Paramount | 25
Universal | 22
Disney | 11
Columbia | 10
MGM | 7
UA | 6
Sony | 6
... (14 rows omitted)
```

10 Movies made by Columbia, 26 Movies made by Fox

```
b = table1.group('Year').sort('count', descending=True)
print(b)
print('\nYears with maximum number of movies released: \n{}'.format(b.where(b['count'], np.max(b['count']))))
```

```
Year | count
2015 | 6
2009 | 6
2004 | 6
2002 | 6
2012 | 5
2007 | 5
2003 | 5
2001 | 5
1997 | 5
1982 | 5
... (60 rows omitted)
```

Years with maximum number of movies released:

```
Year | count
2015 | 6
2009 | 6
2004 | 6
2002 | 6
```

```
table2 = table1.where('Studio', are.containing('S'))
table2.pivot('Studio', 'Year')
```

```
/usr/local/lib/python3.7/dist-packages/datascience/tables.py:483: VisibleDe
values = np.array(tuple(values))
```

Year	Selz.	Sony	Sum.	TriS
1946	1	0	0	0
1985	0	0	0	1
1991	0	0	0	1
1997	0	2	0	0
2002	0	1	0	0
2004	0	1	0	0
2007	0	1	0	0
2009	0	0	1	0
2010	0	0	1	0
2012	0	1	0	0

```
table2.pivot('Studio', 'Year', 'Gross', sum)
```

```
/usr/local/lib/python3.7/dist-packages/datascience/tables.py:483: VisibleDe
values = np.array(tuple(values))
```

Year	Selz.	Sony	Sum.	TriS
1946	20408163	0	0	0
1985	0	0	0	150415432
1991	0	0	0	204843345
1997	0	423646948	0	0
2002	0	403706375	0	0
2004	0	373585825	0	0
2007	0	336530303	0	0
2009	0	0	296623634	0
2010	0	0	300531751	0
2012	0	304360277	0	0

```
table2.pivot('Studio', 'Year', 'Gross', np.average)
```

```
/usr/local/lib/python3.7/dist-packages/datascience/tables.py:483: VisibleDe
values = np.array(tuple(values))
```

Year	Selz.	Sony	Sum.	TriS
1946	2.04082e+07	0	0	0
1985	0	0	0	1.50415e+08
1991	0	0	0	2.04843e+08
1997	0	2.11823e+08	0	0
2002	0	4.03706e+08	0	0
2004	0	3.73586e+08	0	0
2007	0	3.3653e+08	0	0
2009	0	0	2.96624e+08	0
2010	0	0	3.00532e+08	0
2012	0	3.0436e+08	0	0

```
table1 = pd.DataFrame({
    'a':[9,3,3,1],
    'b':[1,2,2,10],
    'c':[3,4,5,16]
})
table1
```

	a	b	c
0	9	1	3
1	3	2	4
2	3	2	5
3	1	10	16

```
table2 = pd.DataFrame({
    'a':[9,1,1,1,4],
    'd':[1,2,2,10,3],
    'e':[3,4,5,6,9]
})
table2
```

	a	d	e
0	9	1	3
1	1	2	4
2	1	2	5
3	1	10	6
4	4	3	9

```
pd.merge(table1, table2, on='a')
```

	a	b	c	d	e
0	9	1	3	1	3
1	1	10	16	2	4
2	1	10	16	2	5
3	1	10	16	10	6

```
pd.merge(table1, table2, left_on='a', right_on='d')
```

	a_x	b	c	a_y	d	e
0	3	2	4	4	3	9
1	3	2	5	4	3	9
2	1	10	16	9	1	3

```
pd.merge(table1, table2, left_on=['a','b'], right_on=['a','d'])
```

	a	b	c	d	e
0	9	1	3	1	3
1	1	10	16	10	6

```
pd.merge(table1, table2, on='a', how='outer')
```

	a	b	c	d	e
0	9	1.0	3.0	1.0	3.0
1	3	2.0	4.0	NaN	NaN
2	3	2.0	5.0	NaN	NaN
3	1	10.0	16.0	2.0	4.0
4	1	10.0	16.0	2.0	5.0
5	1	10.0	16.0	10.0	6.0
6	4	NaN	NaN	3.0	9.0

```
pd.merge(table1, table2, on='a', how='left')
```

	a	b	c	d	e
0	9	1	3	1.0	3.0
1	3	2	4	NaN	NaN
2	3	2	5	NaN	NaN
3	1	10	16	2.0	4.0
4	1	10	16	2.0	5.0
5	1	10	16	10.0	6.0

```
pd.merge(table1, table2, on='a', how='right')
```

	a	b	c	d	e
0	9	1.0	3.0	1	3
1	1	10.0	16.0	2	4
2	1	10.0	16.0	2	5
3	1	10.0	16.0	10	6
4	4	NaN	NaN	3	9