https://www.ons.gov.uk/peoplepopulationandcommunity/healthandsocialcare/healthandwellbeing/articles/middle
08-13
(https://www.ons.gov.uk/peoplepopulationandcommunity/healthandsocialcare/healthandwellbeing/articles/middle
08-13)

```
In [1]: import pandas
        print('pandas',pandas.__version__)
        import numpy
        print('numpy',numpy.__version__)
```

```
pandas 0.23.4
numpy 1.13.3
```

```
In [2]: import requests

        url = "https://www.ons.gov.uk/visualisations/dvc661/drugs/datadownload.c
        sv"

        header = {
          "User-Agent": "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHT
        ML, like Gecko) Chrome/50.0.2661.75 Safari/537.36",
          "X-Requested-With": "XMLHttpRequest"
        }

        r = requests.get(url, headers=header)
```

```
In [3]: list_of_lines = [x.split(',') for x in r.text.split('\n')]
```

```
In [4]: headers = list_of_lines.pop(0)
```

```
In [5]: df = pandas.DataFrame(list_of_lines, columns=headers)
```

```
In [6]: df = df.T
```

```
In [7]: df.head()
```

Out[7]:

	0	1	2	3	4	5	6	7	8	9	...	73	74	75	76	77	78	79	80	81	82
Age	<10	10	11	12	13	14	15	16	17	18	...	82	83	84	85	86	87	88	89	90+	
1993	6	0	0	3	4	6	6	12	26	32	...	7	7	7	16	4	5	6	7	9	None
1994	4	0	1	2	5	8	7	24	22	24	...	12	10	10	13	8	9	3	8	11	None
1995	5	0	0	1	2	7	11	12	24	25	...	13	11	9	2	10	7	11	4	8	None
1996	4	0	0	2	2	9	20	18	27	40	...	12	18	9	7	7	7	7	2	16	None

5 rows × 83 columns

In [8]:
```
df.tail()
```

Out[8]:

	0	1	2	3	4	5	6	7	8	9	...	73	74	75	76	77	78	79	80	81
2013	2	0	0	0	0	2	5	9	4	12	...	6	7	10	2	4	3	2	5	19
2014	2	1	0	0	1	1	4	5	4	14	...	8	9	5	7	6	3	7	2	18
2015	1	0	0	0	1	1	4	7	8	13	...	7	6	8	3	13	6	5	5	16
2016	1	0	0	0	0	1	5	5	4	14	...	9	9	8	7	9	9	4	2	24
2017\r	1\r	0\r	0\r	0\r	0\r	2\r	2\r	9\r	8\r	14\r	...	10\r	7\r	7\r	12\r	6\r	2\r	4\r	5\r	25\r

5 rows × 83 columns

In [9]:
```
df = df.rename(index={'2017\r': '2017'})
```

In [10]:
```
# https://stackoverflow.com/questions/31328861/python-pandas-replacing-h
eader-with-top-row

new_header = df.iloc[0] #grab the first row for the header
df = df[1:] #take the data less the header row
df.columns = new_header #set the header row as the df header
```

In [11]:
```
df.head()
```

Out[11]:

Age	<10	10	11	12	13	14	15	16	17	18	...	82	83	84	85	86	87	88	89	90+	
1993	6	0	0	3	4	6	6	12	26	32	...	7	7	7	16	4	5	6	7	9	None
1994	4	0	1	2	5	8	7	24	22	24	...	12	10	10	13	8	9	3	8	11	None
1995	5	0	0	1	2	7	11	12	24	25	...	13	11	9	2	10	7	11	4	8	None
1996	4	0	0	2	2	9	20	18	27	40	...	12	18	9	7	7	7	7	2	16	None
1997	5	0	0	1	3	5	13	20	34	33	...	10	7	10	7	5	11	6	3	15	None

5 rows × 83 columns

In [12]:
```python
df.tail()
```

Out[12]:

Age	<10	10	11	12	13	14	15	16	17	18	...	82	83	84	85	86	87	88	89	90+
2013	2	0	0	0	0	2	5	9	4	12	...	6	7	10	2	4	3	2	5	19
2014	2	1	0	0	1	1	4	5	4	14	...	8	9	5	7	6	3	7	2	18
2015	1	0	0	0	1	1	4	7	8	13	...	7	6	8	3	13	6	5	5	16
2016	1	0	0	0	0	1	5	5	4	14	...	9	9	8	7	9	9	4	2	24
2017	1\r	0\r	0\r	0\r	0\r	2\r	2\r	9\r	8\r	14\r	...	10\r	7\r	7\r	12\r	6\r	2\r	4\r	5\r	25\r

5 rows × 83 columns

In [13]:
```python
df = df.apply(pandas.to_numeric)
```

In [14]:
```python
import matplotlib.pyplot as plt
```

In [15]:
```python
# https://stackoverflow.com/questions/12286607/making-heatmap-from-pandas-dataframe

plt.pcolor(df)
plt.yticks(numpy.arange(0.5, len(df.index), 1), df.index)
plt.xticks(numpy.arange(0.5, len(df.columns), 1), df.columns)
plt.xlabel('age')
plt.ylabel('year')
plt.setp(plt.gca().get_xticklabels()[1::5], visible=False)
plt.setp(plt.gca().get_xticklabels()[2::5], visible=False)
plt.setp(plt.gca().get_xticklabels()[3::5], visible=False)
plt.setp(plt.gca().get_xticklabels()[4::5], visible=False)
plt.setp(plt.gca().get_yticklabels()[1::2], visible=False)
#plt.setp(plt.gca().get_xticklabels()[::5], visible=False)
plt.show()
```