In [1]:

```
import pandas as pd
```

In [2]:

```
data = pd.read_csv('candyhierarchy2017.csv', encoding = "ISO-8859-1" )
```

In [3]:

```
older_data = pd.read_csv('BOING-BOING-CANDY-HIERARCHY-2016-SURVEY-Responses.csv', enco
```

In [4]:

```
data.head()
```

Out[4]:

	Internal ID	Q1: GOING OUT?	Q2: GENDER	Q3: AGE	Q4: COUNTRY	Q5: STATE, PROVINCE, COUNTY, ETC	Q6 | 100 Grand Bar	Q6 | Anonymous brown globs that come in black and orange wrappers\t(a.k.a. Mary Janes)	Q6 | Any full-sized candy bar	Q6 Blac Jack
0	90258773	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaI
1	90272821	No	Male	44	USA	NM	MEH	DESPAIR	JOY	MEI
2	90272829	NaN	Male	49	USA	Virginia	NaN	NaN	NaN	NaI
3	90272840	No	Male	40	us	or	MEH	DESPAIR	JOY	MEI
4	90272841	No	Male	23	usa	exton pa	JOY	DESPAIR	JOY	DESPAI

5 rows × 120 columns

In [5]:

```
older_data.head()
```

Out[5]:

	Timestamp	Are you going actually going trick or treating yourself?	Your gender:	How old are you?	Which country do you live in?	Which state, province, county do you live in?	[100 Grand Bar]	[Anonymous brown globs that come in black and orange wrappers]	[Any full-sized candy bar]	[Black Jacks]	...
0	10/24/2016 5:09:23	No	Male	22	Canada	Ontario	JOY	DESPAIR	JOY	MEH	...
1	10/24/2016 5:09:55	No	Male	45	usa	il	MEH	MEH	JOY	JOY	...
2	10/24/2016 5:13:07	No	Female	48	US	Colorado	JOY	DESPAIR	JOY	MEH	...
3	10/24/2016 5:14:17	No	Male	57	usa	il	JOY	MEH	JOY	MEH	...
4	10/24/2016 5:14:25	Yes	Male	42	USA	South Dakota	MEH	DESPAIR	JOY	DESPAIR	...

5 rows × 123 columns

Chapter 7:

In [22]:

```
#Filling in missing data with 0's
data.fillna(0)
```

Out[22]:

	Internal ID	Q1: GOING OUT?	Q2: GENDER	Q3: AGE	Q4: COUNTRY	Q5: STATE, PROVINCE, COUNTY, ETC	Q6 | 100 Grand Bar	Q6 | Anonymous brown globs that come in black and orange wrappers\t(a.k.a. Mary Janes)	Q6 | Any full-sized candy bar	
0	90258773	0	0	0	0	0	0	0	0	
1	90272821	No	Male	44	USA	NM	MEH	DESPAIR	JOY	
2	90272829	0	Male	49	USA	Virginia	0	0	0	
3	90272840	No	Male	40	us	or	MEH	DESPAIR	JOY	
4	90272841	No	Male	23	usa	exton pa	JOY	DESPAIR	JOY	[
...	...	...	...	...	...	...	...	...	...	
2455	90314359	No	Male	24	USA	MD	JOY	DESPAIR	MEH	[
2456	90314580	No	Female	33	USA	New York	MEH	DESPAIR	JOY	
2457	90314634	No	Female	26	USA	Tennessee	MEH	DESPAIR	JOY	[
2458	90314658	No	Male	58	Usa	North Carolina	0	0	0	
2459	90314802	No	Female	66	usa	Pennsylvania	DESPAIR	DESPAIR	JOY	[

2460 rows × 120 columns

In [6]:

```
#Filling in missing data for Q3: AGE column so I can bin them
data['Q3: AGE'] = (pd.to_numeric(data['Q3: AGE'], errors='coerce').fillna(0))
```

In [7]:

```
#Binning into age groups
bins = [1, 18, 25, 30, 40, 50, 60, 70, 100]
```

In [8]:

```
age_bins = pd.cut(data['Q3: AGE'], bins)
```

In [9]:

```
age_bins
```

Out[9]:

```
0                 NaN
1        (40.0, 50.0]
2        (40.0, 50.0]
3        (30.0, 40.0]
4        (18.0, 25.0]
             ...
2455     (18.0, 25.0]
2456     (30.0, 40.0]
2457     (25.0, 30.0]
2458     (50.0, 60.0]
2459     (60.0, 70.0]
Name: Q3: AGE, Length: 2460, dtype: category
Categories (8, interval[int64, right]): [(1, 18] < (18, 25] < (25, 30] < (30, 40] <
(40, 50] < (50, 60] < (60, 70] < (70, 100]]
```

Chapter 8:

In [23]:

```
#Setting Q3: Age as index
data2 = data.set_index(['Q1: GOING OUT?'])
```

In [24]:

```
data2
```

Out[24]:

	Internal ID	Q2: GENDER	Q3: AGE	Q4: COUNTRY	Q5: STATE, PROVINCE, COUNTY, ETC	Q6 | 100 Grand Bar	Q6 | Anonymous brown globs that come in black and orange wrappers\t(a.k.a. Mary Janes)	Q6 | Any full-sized candy bar	Q6 | Black Jacks
Q1: GOING OUT?									
NaN	90258773	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
No	90272821	Male	44	USA	NM	MEH	DESPAIR	JOY	MEH
NaN	90272829	Male	49	USA	Virginia	NaN	NaN	NaN	NaN
No	90272840	Male	40	us	or	MEH	DESPAIR	JOY	MEH
No	90272841	Male	23	usa	exton pa	JOY	DESPAIR	JOY	DESPAIR
...	...	...	...	...	...	...	...	...	...
No	90314359	Male	24	USA	MD	JOY	DESPAIR	MEH	DESPAIR
No	90314580	Female	33	USA	New York	MEH	DESPAIR	JOY	NaN
No	90314634	Female	26	USA	Tennessee	MEH	DESPAIR	JOY	DESPAIR
No	90314658	Male	58	Usa	North Carolina	NaN	NaN	NaN	NaN
No	90314802	Female	66	usa	Pennsylvania	DESPAIR	DESPAIR	JOY	DESPAIR

2460 rows × 119 columns

In [18]:

```
#Creating hierarchical index
data2.index.names = ['Q1: GOING OUT?']
```

In [19]:

```
data2
```

Out[19]:

	Internal ID	Q2: GENDER	Q3: AGE	Q4: COUNTRY	Q5: STATE, PROVINCE, COUNTY, ETC	Q6 | 100 Grand Bar	Q6 | Anonymous brown globs that come in black and orange wrappers\t(a.k.a. Mary Janes)	Q6 | Any full-sized candy bar	Q6 | Black Jacks
Q1: GOING OUT?									
NaN	90258773	NaN	0.0	NaN	NaN	NaN	NaN	NaN	NaN
No	90272821	Male	44.0	USA	NM	MEH	DESPAIR	JOY	MEH
NaN	90272829	Male	49.0	USA	Virginia	NaN	NaN	NaN	NaN
No	90272840	Male	40.0	us	or	MEH	DESPAIR	JOY	MEH
No	90272841	Male	23.0	usa	exton pa	JOY	DESPAIR	JOY	DESPAIR
...	...	...	...	...	...	...	...	...	...
No	90314359	Male	24.0	USA	MD	JOY	DESPAIR	MEH	DESPAIR
No	90314580	Female	33.0	USA	New York	MEH	DESPAIR	JOY	NaN
No	90314634	Female	26.0	USA	Tennessee	MEH	DESPAIR	JOY	DESPAIR
No	90314658	Male	58.0	Usa	North Carolina	NaN	NaN	NaN	NaN
No	90314802	Female	66.0	usa	Pennsylvania	DESPAIR	DESPAIR	JOY	DESPAIR

2460 rows × 119 columns

In [6]:

```
data.fillna(0)
```

Out[6]:

	Internal ID	Q1: GOING OUT?	Q2: GENDER	Q3: AGE	Q4: COUNTRY	Q5: STATE, PROVINCE, COUNTY, ETC	Q6 | 100 Grand Bar	Q6 | Anonymous brown globs that come in black and orange wrappers\t(a.k.a. Mary Janes)	Q6 | Any full-sized candy bar
0	90258773	0	0	0	0	0	0	0	0
1	90272821	No	Male	44	USA	NM	MEH	DESPAIR	JOY
2	90272829	0	Male	49	USA	Virginia	0	0	0
3	90272840	No	Male	40	us	or	MEH	DESPAIR	JOY
4	90272841	No	Male	23	usa	exton pa	JOY	DESPAIR	JOY
...	...	...	...	...	...	...	...	...	...
2455	90314359	No	Male	24	USA	MD	JOY	DESPAIR	MEH
2456	90314580	No	Female	33	USA	New York	MEH	DESPAIR	JOY
2457	90314634	No	Female	26	USA	Tennessee	MEH	DESPAIR	JOY
2458	90314658	No	Male	58	Usa	North Carolina	0	0	0
2459	90314802	No	Female	66	usa	Pennsylvania	DESPAIR	DESPAIR	JOY

2460 rows × 120 columns

Chapter 10:

In [9]:

```
mapping = {'Q12: MEDIA [Daily Dish]': 'Regular Media', 'Q12: MEDIA [Science]': 'Regula
          'Q12: MEDIA [Yahoo]': 'Website'}
```

In [10]:

```
data.groupby(by=mapping,axis=1).sum()
```

Out[10]:

	Regular Media	Website
0	0.0	0.0
1	1.0	0.0
2	0.0	0.0
3	1.0	0.0
4	1.0	0.0
...	...	...
2455	0.0	0.0
2456	1.0	0.0
2457	1.0	0.0
2458	0.0	0.0
2459	1.0	0.0

2460 rows × 2 columns

Chapter 11:

In [45]:

```
#converting strings to dates
pd.to_datetime(older_data['Timestamp'])
```

Out[45]:

```
0      2016-10-24 05:09:23
1      2016-10-24 05:09:55
2      2016-10-24 05:13:07
3      2016-10-24 05:14:17
4      2016-10-24 05:14:25
                ...
1254   2016-10-29 16:53:53
1255   2016-10-30 06:53:55
1256   2016-10-30 11:06:11
1257   2016-10-30 16:07:27
1258   2016-10-30 17:06:46
Name: Timestamp, Length: 1259, dtype: datetime64[ns]
```

In [49]:

```
#generating time range
times = pd.date_range(start ='2016-10-24',
         end ='2016-10-30', freq ='5H')
```

In [50]:

```
times
```

```
Out[50]: DatetimeIndex(['2016-10-24 00:00:00', '2016-10-24 05:00:00',
                        '2016-10-24 10:00:00', '2016-10-24 15:00:00',
                        '2016-10-24 20:00:00', '2016-10-25 01:00:00',
                        '2016-10-25 06:00:00', '2016-10-25 11:00:00',
                        '2016-10-25 16:00:00', '2016-10-25 21:00:00',
                        '2016-10-26 02:00:00', '2016-10-26 07:00:00',
                        '2016-10-26 12:00:00', '2016-10-26 17:00:00',
                        '2016-10-26 22:00:00', '2016-10-27 03:00:00',
                        '2016-10-27 08:00:00', '2016-10-27 13:00:00',
                        '2016-10-27 18:00:00', '2016-10-27 23:00:00',
                        '2016-10-28 04:00:00', '2016-10-28 09:00:00',
                        '2016-10-28 14:00:00', '2016-10-28 19:00:00',
                        '2016-10-29 00:00:00', '2016-10-29 05:00:00',
                        '2016-10-29 10:00:00', '2016-10-29 15:00:00',
                        '2016-10-29 20:00:00'],
                       dtype='datetime64[ns]', freq='5H')
```