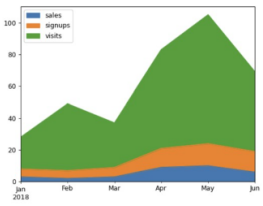
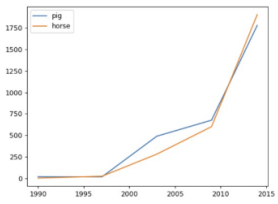
Asg 21.1

Data Visualization -- Pandas

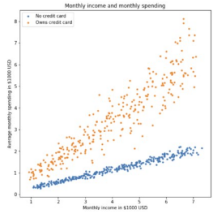
(Coding)



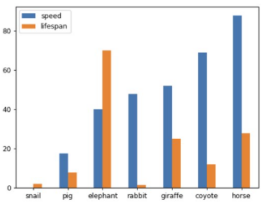
Area Plot



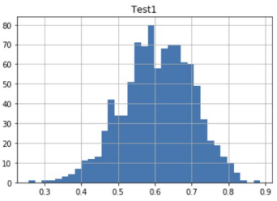
Line Plot



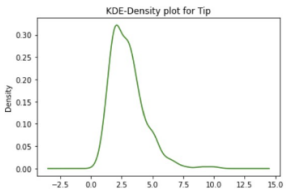
Scatter Plot



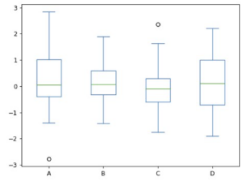
Bar Plot



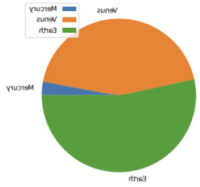
Histogram



Density Plot



Box Plot



Pie Plot

[**Link: pandas.DataFrame.plot**]

(https://pandas.pydata.org/pandas-

docs/stable/reference/api/pandas.DataFr

Files needed for this assignment:

mortality_prepped.pkl

mortality_wide.pkl

In [5]:
```
# set up notebook to display multiple output in one cell

from IPython.core.interactiveshell import InteractiveShell
InteractiveShell.ast_node_interactivity = "all"
```

```
print('The notebook is set up to display multiple output in one cell.')
```

```
The notebook is set up to display multiple output in one cell.
```

Asg 21.1: Create some plots

In [6]:
```
# conventional way to import pandas and numpy

import pandas as pd
import numpy as np
```

Get the data

In [7]:
```
mortality_data = pd.read_pickle('mortality_prepped.pkl')
mortality_data.head()
```

Out[7]:

	Year	AgeGroup	DeathRate	MeanCentered
0	1900	01-04 Years	1983.8	1790.87584
1	1901	01-04 Years	1695.0	1502.07584
2	1902	01-04 Years	1655.7	1462.77584
3	1903	01-04 Years	1542.1	1349.17584
4	1904	01-04 Years	1591.5	1398.57584

In [8]:
```
mortality_wide = pd.read_pickle('mortality_wide.pkl')
mortality_wide.head()
```

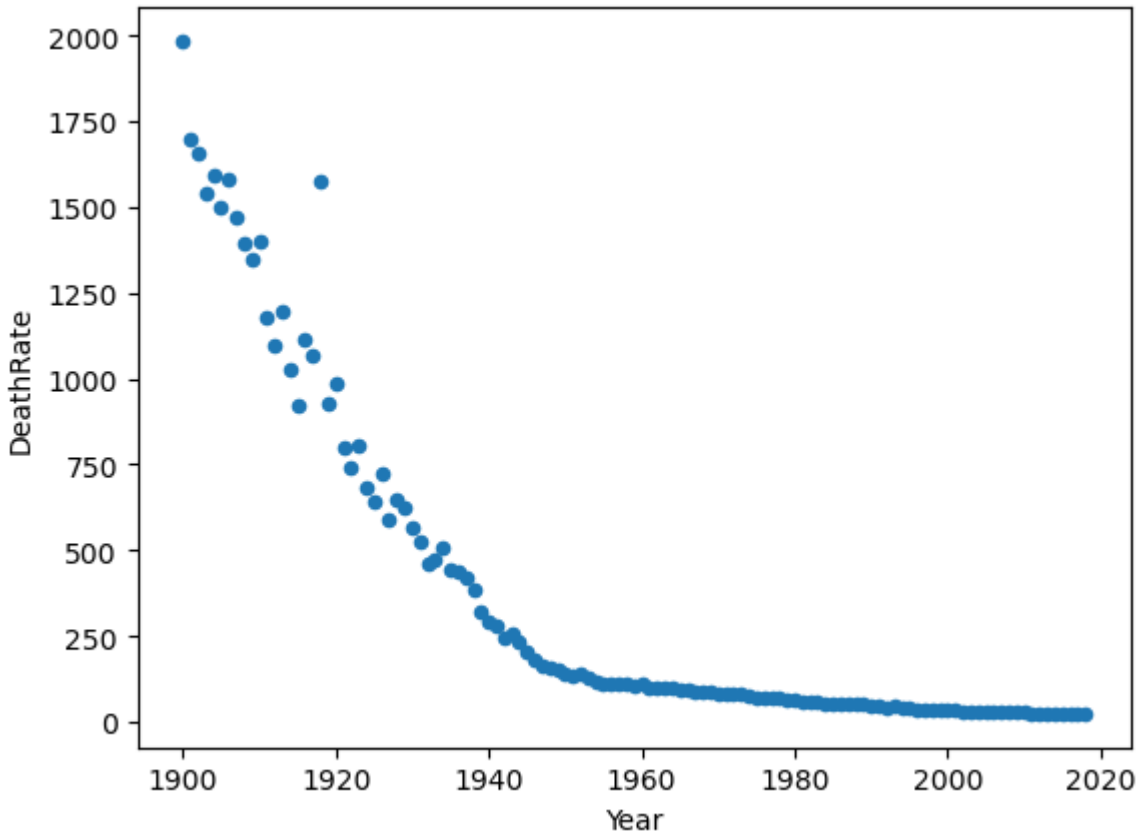
Out[8]:

AgeGroup	01-04 Years	05-09 Years	10-14 Years	15-19 Years
Year				
1900	1983.8	466.1	298.3	484.8
1901	1695.0	427.6	273.6	454.4
1902	1655.7	403.3	252.5	421.5
1903	1542.1	414.7	268.2	434.1
1904	1591.5	425.0	305.2	471.4

Visualize the data

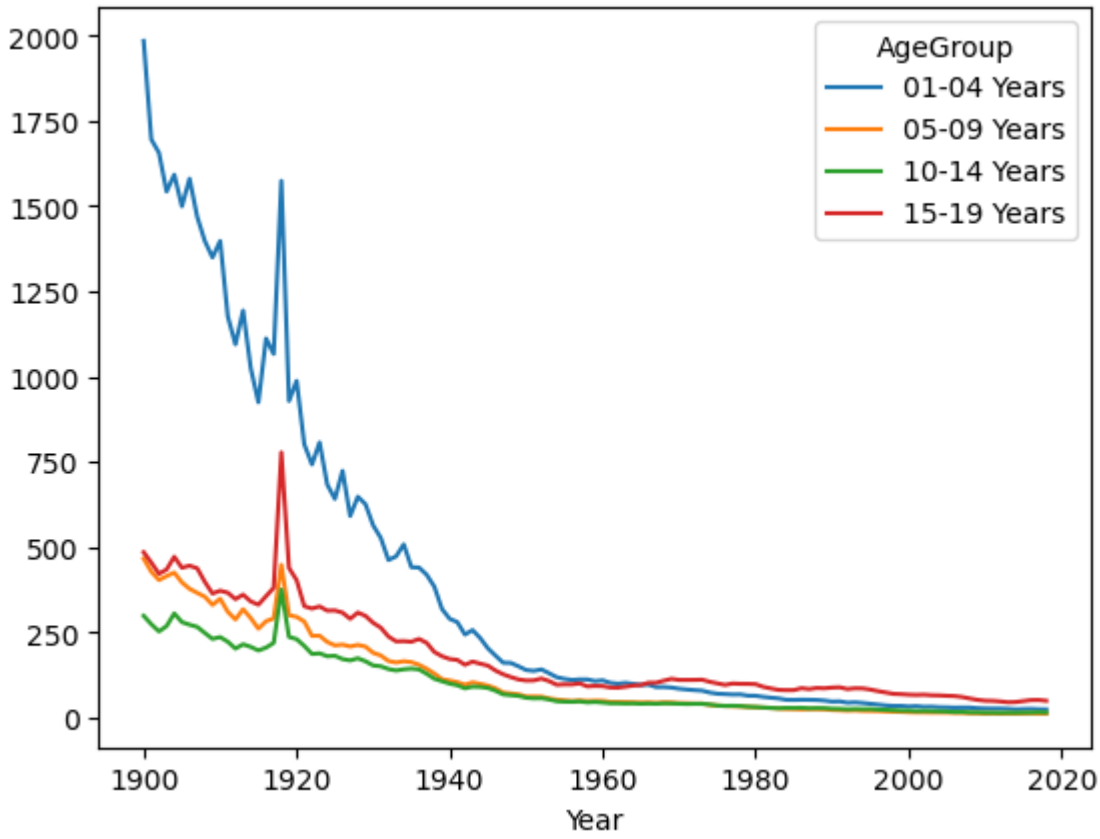
In [9]:
```
mortality_data.query('AgeGroup == "01-04 Years"').plot.scatter(x='Year', y='DeathRate'
```

Out[9]:
```
<AxesSubplot:xlabel='Year', ylabel='DeathRate'>
```



In [10]:
```
mortality_wide.plot()
```

Out[10]:
```
<AxesSubplot:xlabel='Year'>
```



NOTE:

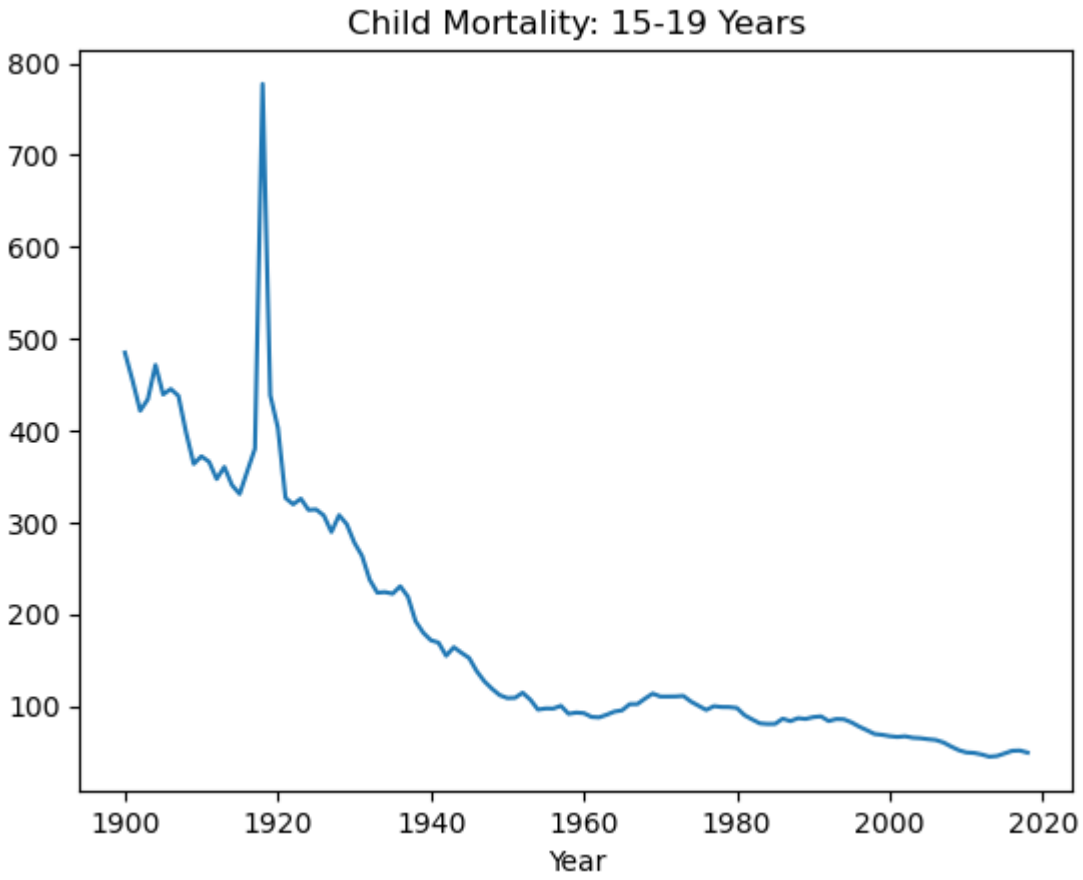
As you create the visualizations described in the steps that follow, you should not have to create any new DataFrames. Instead, you should use chaining to get the results that you want.

Question 1:

Use the wide data to create a line plot for just the data in the 15-19 age group. Include an appropriate title on the plot and remove the legend.

In [11]:
```
mortality_wide.plot.line(y='15-19 Years', title='Child Mortality: 15-19 Years', lege
```

Out[11]:
```
<AxesSubplot:title={'center':'Child Mortality: 15-19 Years'}, xlabel='Year'>
```

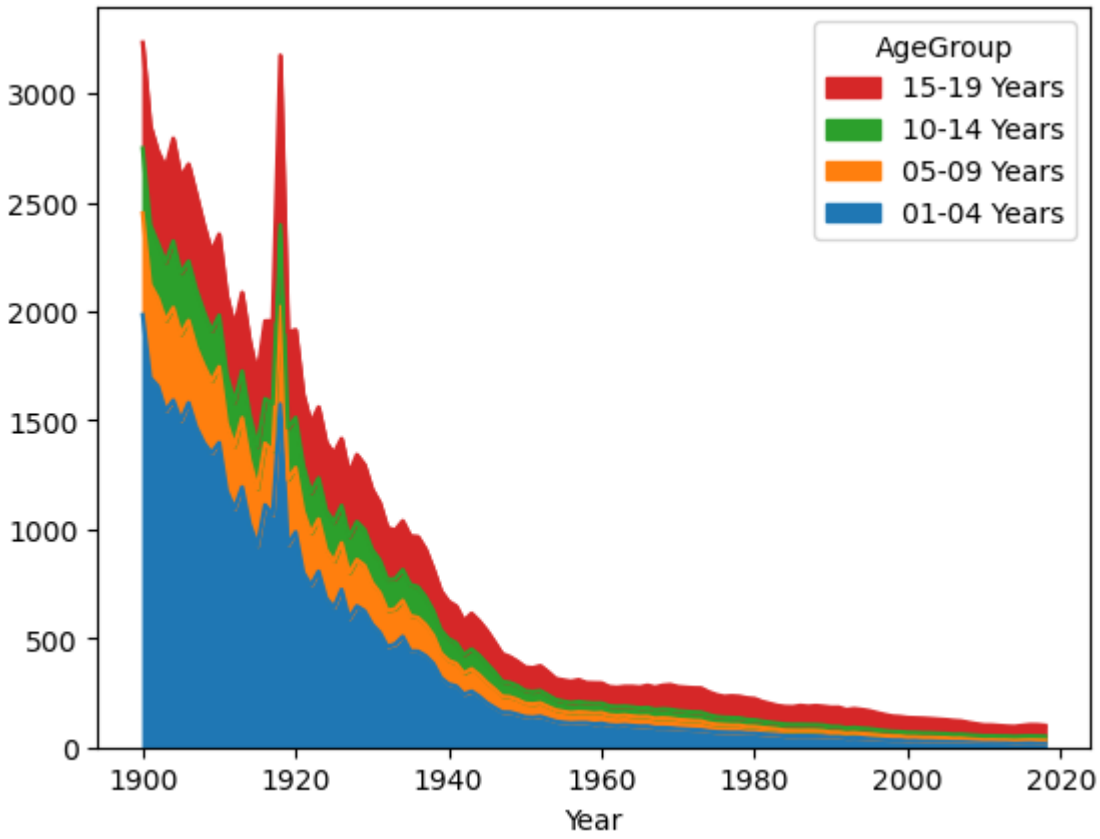


Question 2:

Use the wide data to create an area plot for all age groups, and reverse the order of the items in the legend to see how that looks.

In [12]:
```
mortality_wide.plot.area(legend='reverse')
```

Out[12]:
```
<AxesSubplot:xlabel='Year'>
```

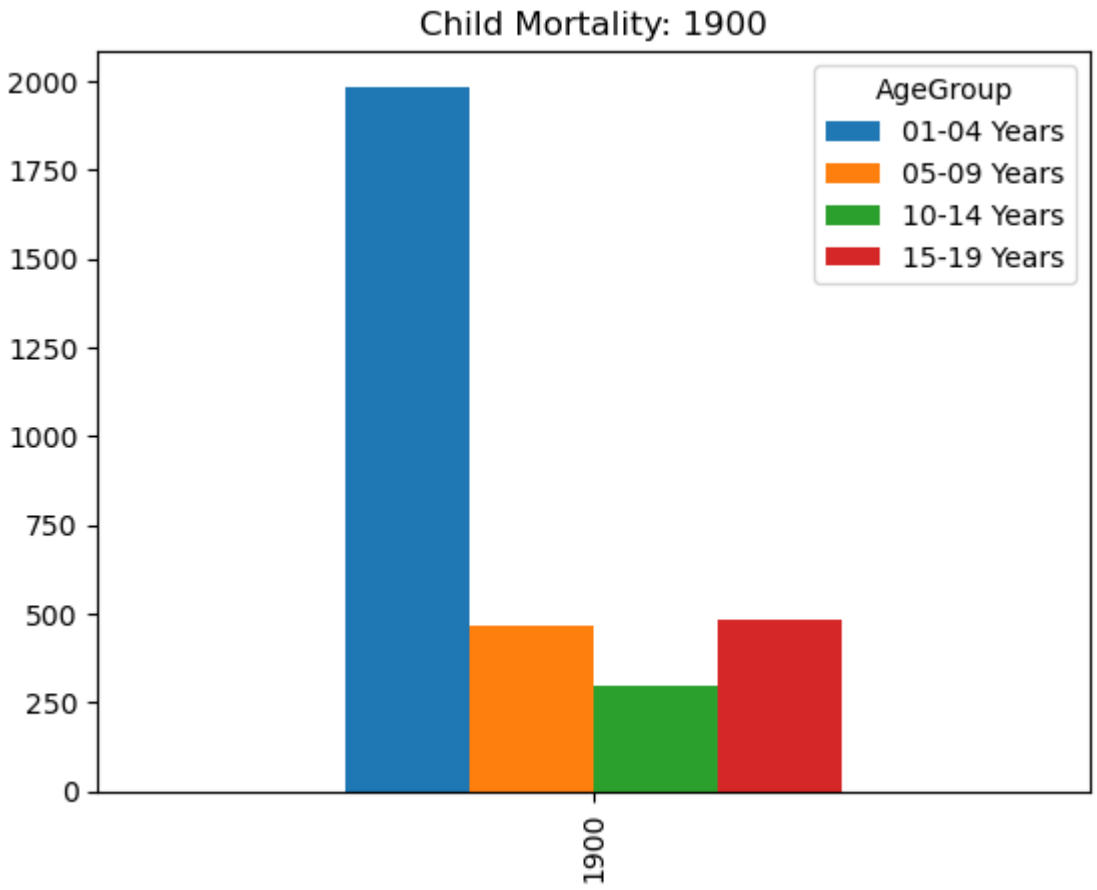


Question 3:

Use the wide data to create a bar plot for all age groups that shows the mortality rates for just the year 1900, and note the values on the y-axis.

In [14]:
```
mortality_wide.query('Year == 1900') \
    .plot.bar(title='Child Mortality: 1900', xlabel="")
```

Out[14]:
```
<AxesSubplot:title={'center':'Child Mortality: 1900'}>
```

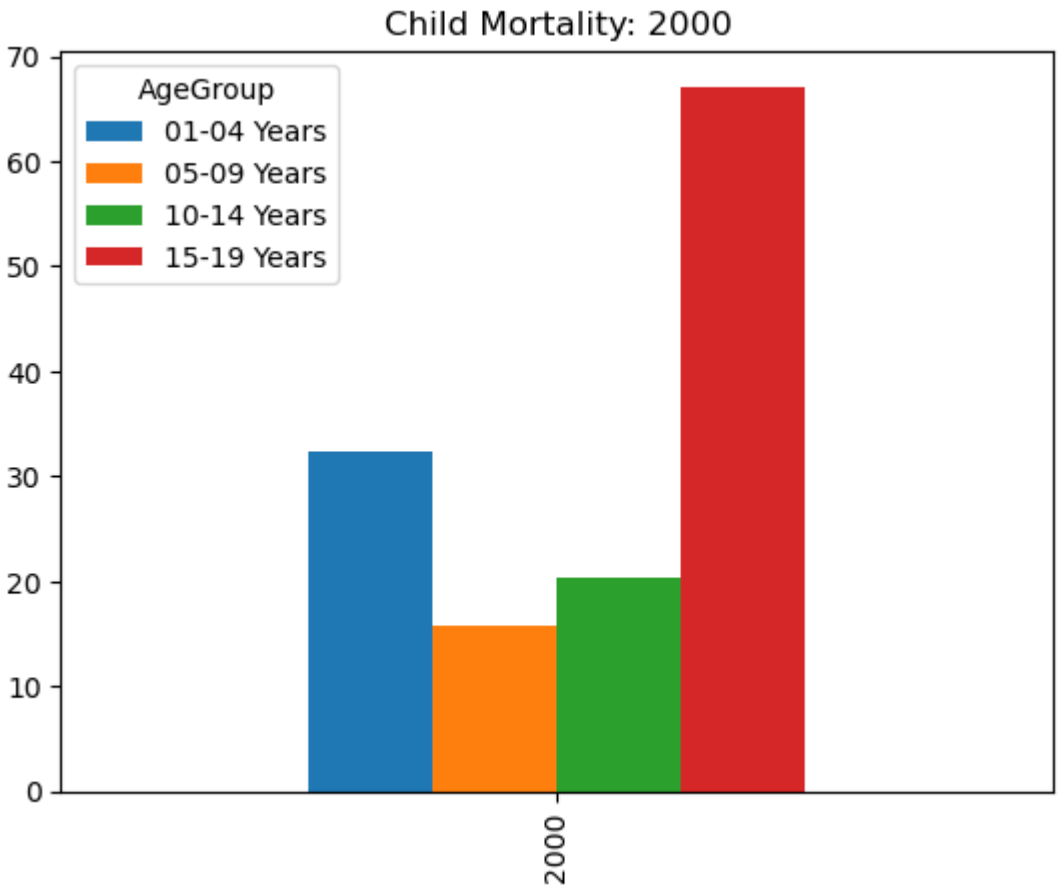


Question 4:

Change the bar plot you created in Question 3 to show the mortality rates for the year 2000, and note how the values on the y-axis change. Then, add an appropriate title to the plot and remove the label for the x-axis.

In [15]:
```
mortality_wide.query('Year == 2000') \
    .plot.bar(title='Child Mortality: 2000', xlabel="")
```

Out[15]:
```
<AxesSubplot:title={'center':'Child Mortality: 2000'}>
```

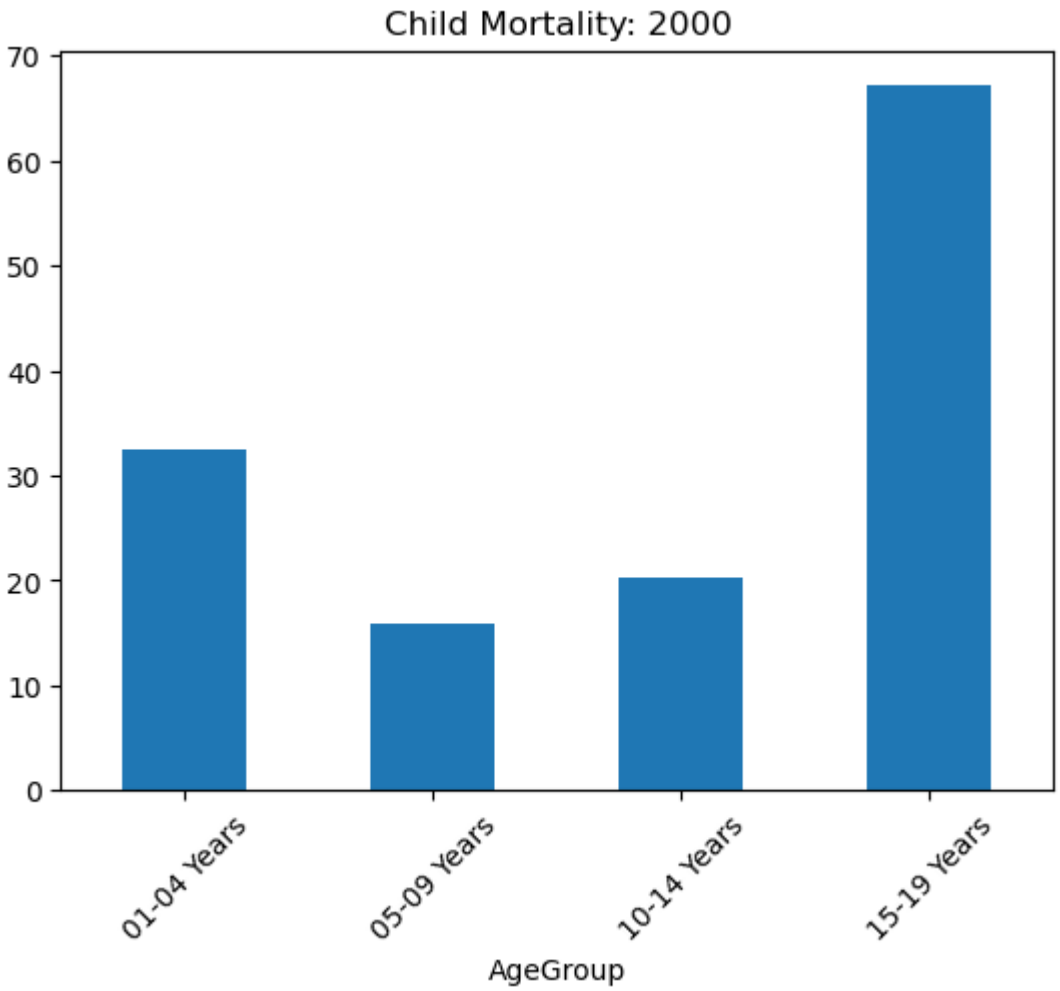


Question 5:

Use the long data to create a bar plot like the one in Question 4. To do that, you'll need to chain the pivot() method to the query) method. Compare the two bar charts. and then make improvements so the plot that uses the long data is easier to read.

In [16]:
```
mortality_data.query('Year == 2000') \
    .pivot(index='AgeGroup', columns='Year', values='DeathRate') \
    .plot.bar(title='Child Mortality: 2000', legend=False, rot=45)
```

Out[16]:
```
<AxesSubplot:title={'center':'Child Mortality: 2000'}, xlabel='AgeGroup'>
```

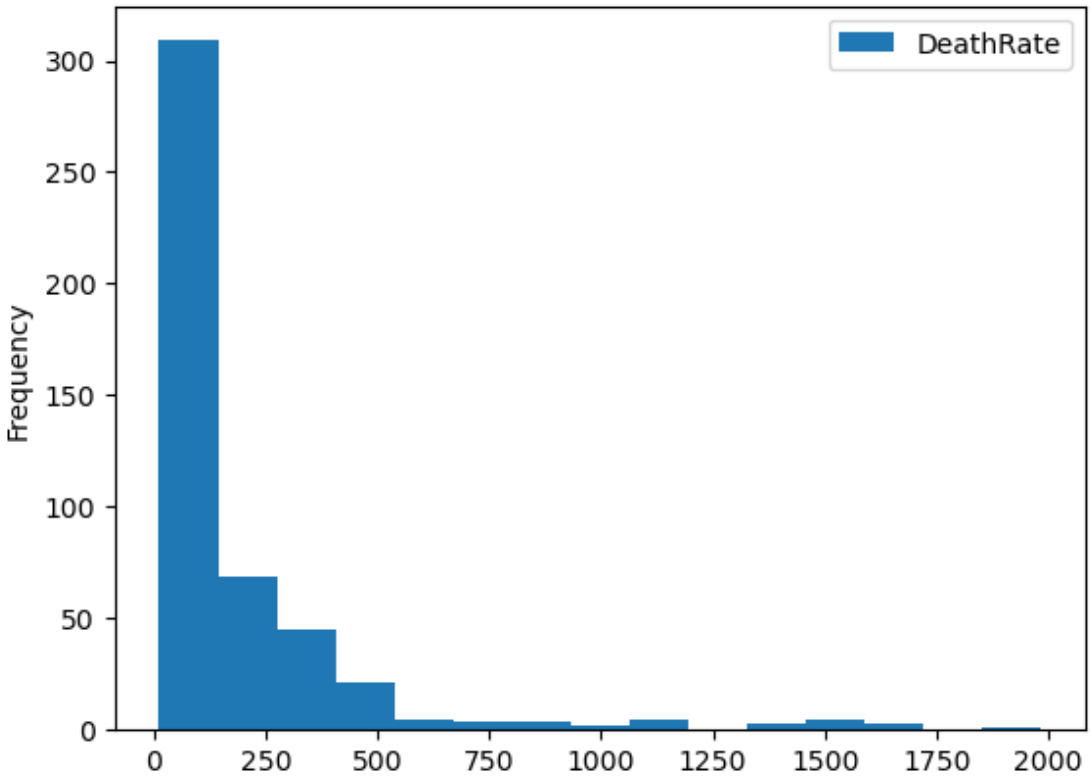


Question 6:

Use the long data to create a histogram that shows the frequency of the death rates in the default number of bins. Then, change the number of bins to 15 to see how this changes the histogram. Does this make it easier to determine the frequency at various datapoints?

In [17]:
```
mortality_data.plot.hist(y='DeathRate', bins=15)
```

Out[17]:
```
<AxesSubplot:ylabel='Frequency'>
```

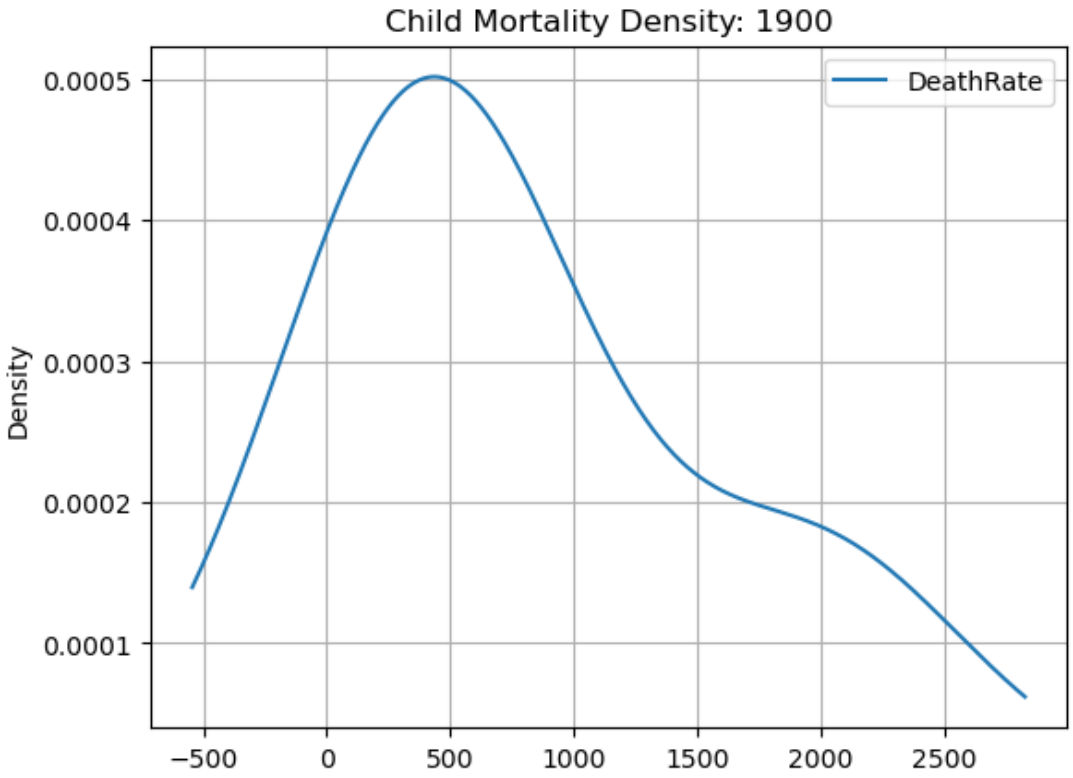


Question 7:

Use the long data to create a density plot that shows the distribution of the death rates in the year 1900. Include a title and grids in the plot to make the data easier to read.

In [18]:
```
mortality_data.query('Year==1900') \
    .plot.density(y='DeathRate', grid=True, title='Child Mortality Density:
```

Out[18]:
```
<AxesSubplot:title={'center':'Child Mortality Density: 1900'}, ylabel='Dens
ity'>
```

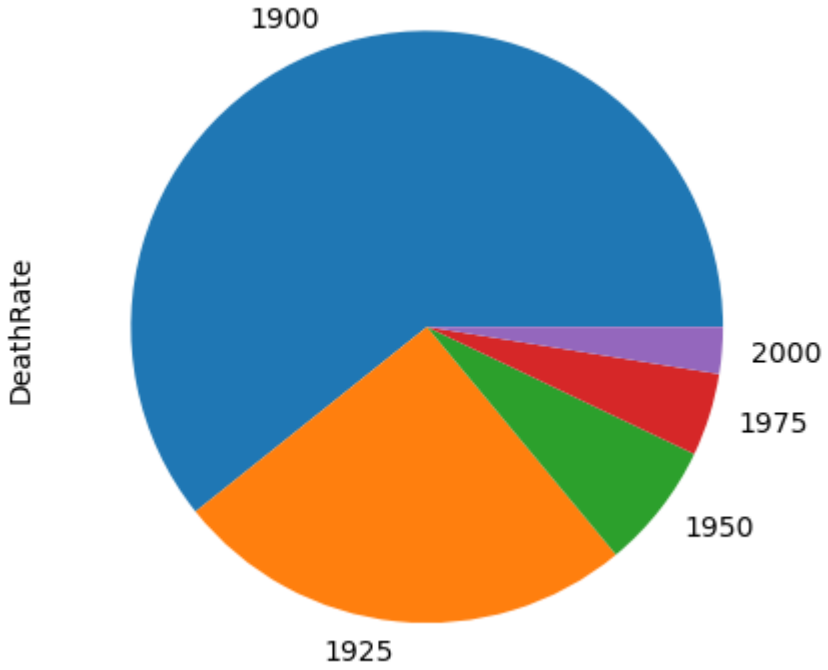


Question 8:

Use the long data to create a pie plot that shows the sum of the death rates tor the years 1900, 1925, 1950, 1975, and 2000.

In [19]:
```
mortality_data.query('Year in (1900, 1925, 1950, 1975, 2000)') \
    .groupby('Year').DeathRate.sum().plot.pie()
```

Out[19]:
```
<AxesSubplot:ylabel='DeathRate'>
```

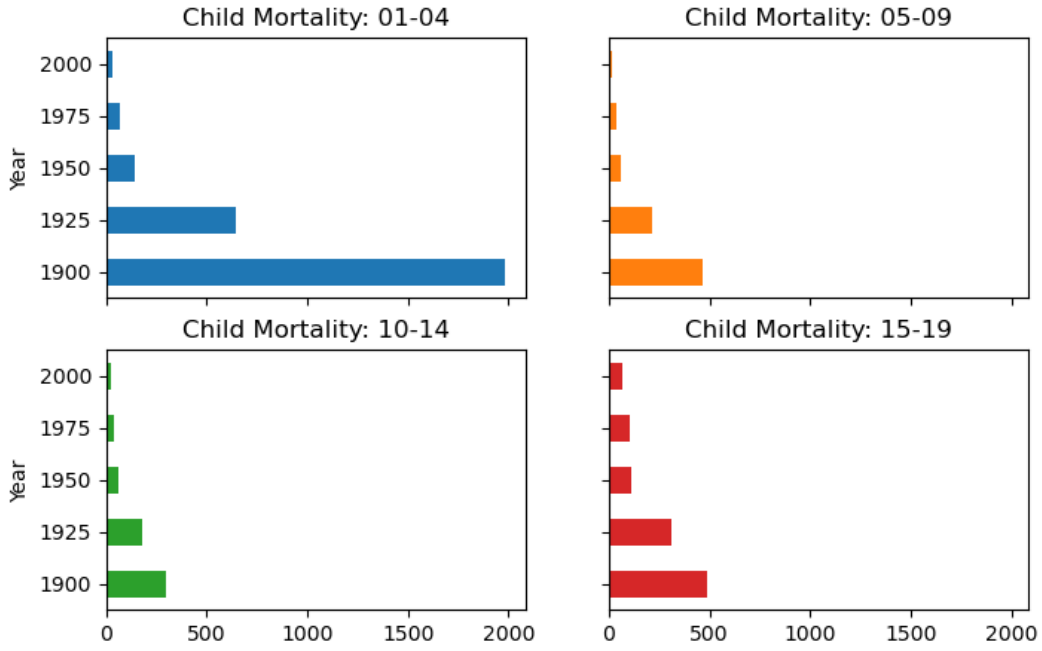


Question 9:

Create a plot with four subplots in two rows and two columns. The subplots should be horizontal bar charts that show the child mortality rates for each age group for the years 1900, 1925, 1950, 1975, and 2000. Format the subplots so they're easy to read.

In [20]:
```
mortality_wide.query('Year in (1900, 1925, 1950, 1975, 2000)').plot.barh(
    title=['Child Mortality: 01-04', 'Child Mortality: 05-09',
           'Child Mortality: 10-14', 'Child Mortality: 15-19'],
    sharey=True, legend=False, subplots=True, layout=(2, 2), figsize=(8,5
```

Out[20]:
```
array([[<AxesSubplot:title={'center':'Child Mortality: 01-04'}, ylabel='Y
ear'>,
        <AxesSubplot:title={'center':'Child Mortality: 05-09'}, ylabel='Y
ear'>],
       [<AxesSubplot:title={'center':'Child Mortality: 10-14'}, ylabel='Y
ear'>,
        <AxesSubplot:title={'center':'Child Mortality: 15-19'}, ylabel='Y
ear'>]],
      dtype=object)
```


Child Mortality: 01-04
Child Mortality: 05-09
Child Mortality: 10-14
Child Mortality: 15-19
Year
2000
1975
1950
1925
1900
0
500
1000
1500
2000