Course 1 Task 1:

- Started by following the course material.
- In-store is nominal, converting the 0/1 values for in-store to "in-store" and "online" so it will be easier to read.
- Region is nominal, converting the 1/2/3/4 values for region to "North", "South", "East", and "West" so it will be easier to read.
- There are amounts with more than two decimal places. Since this is currency and this represents retail transactions, I am rounding the amounts to two decimal places to avoid confusion.
- There are issues with the float data type representing decimal values precisely due to the base 2 internal representation. Currency amounts should be precise, so I am converting these to decimal.
- Added histograms to visually view the values for each column. It appears that there are roughly the same amount of online and in-store transactions. The bulk of transactions appear to be from people between the ages of about 30 and 60. All transactions are 8 or less items, and the mean transaction total amount is $835.92, ranging from \$5 to $3000. The West region has the most transactions. The North region has the fewest transactions.

In [354…
```
import pandas as pd
import matplotlib
from decimal import *
```

In [355…
```
data = pd.read_csv('Demographic_Data.csv')
```

In [356…
```
data.head()
```

Out[356]:

	in-store	age	items	amount	region
0	0	37	4	281.03	2
1	0	35	2	219.51	2
2	1	45	3	1,525.70	4
3	1	46	3	715.25	3
4	1	33	4	1,937.50	1

In [357…
```
data.describe()
```

Out[357]:

	in-store	age	items	amount	region
count	80,000.00	80,000.00	80,000.00	80,000.00	80,000.00
mean	0.50	45.76	4.50	835.92	2.67
std	0.50	15.72	2.06	721.27	1.13
min	0.00	18.00	1.00	5.00	1.00
25%	0.00	33.00	3.00	285.14	2.00
50%	0.50	45.00	4.00	582.32	3.00
75%	1.00	56.00	6.00	1,233.70	4.00
max	1.00	85.00	8.00	3,000.00	4.00

In [358...]
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 80000 entries, 0 to 79999
Data columns (total 5 columns):
 #   Column    Non-Null Count  Dtype  
---  ------    --------------  -----  
 0   in-store  80000 non-null  int64  
 1   age       80000 non-null  int64  
 2   items     80000 non-null  int64  
 3   amount    80000 non-null  float64
 4   region    80000 non-null  int64  
dtypes: float64(1), int64(4)
memory usage: 3.1 MB
```

In [359...]
```
data = data.drop_duplicates()
```

In [360...]
```
data = data.dropna()
```

In [361...]
```
print(data.isnull().sum())
```

```
in-store    0
age         0
items       0
amount      0
region      0
dtype: int64
```

In [362...]
```
data.loc[data['in-store'] == 1, 'in-store-str'] = 'in-store'
data.loc[data['in-store'] == 0, 'in-store-str'] = 'online'
```

In [363...]
```
data.loc[data['region'] == 1, 'region-str'] = 'North'
data.loc[data['region'] == 2, 'region-str'] = 'South'
data.loc[data['region'] == 3, 'region-str'] = 'East'
data.loc[data['region'] == 4, 'region-str'] = 'West'
```

In [364...]
```
data.amount = data.amount.round(2)
data['amount'] = data['amount'].apply(str)
data['amount'] = data['amount'].apply(Decimal)
```

In [365...]
```
data.dtypes
```

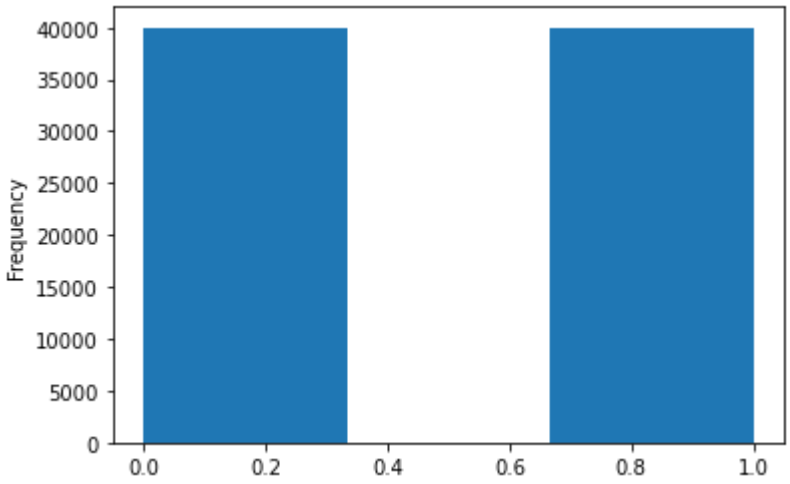
Out[365]:
```
in-store         int64
age              int64
items            int64
amount          object
region           int64
in-store-str    object
region-str      object
dtype: object
```

In [366…
```
sum(data.amount)
```

Out[366]:
```
Decimal('66848505.77')
```

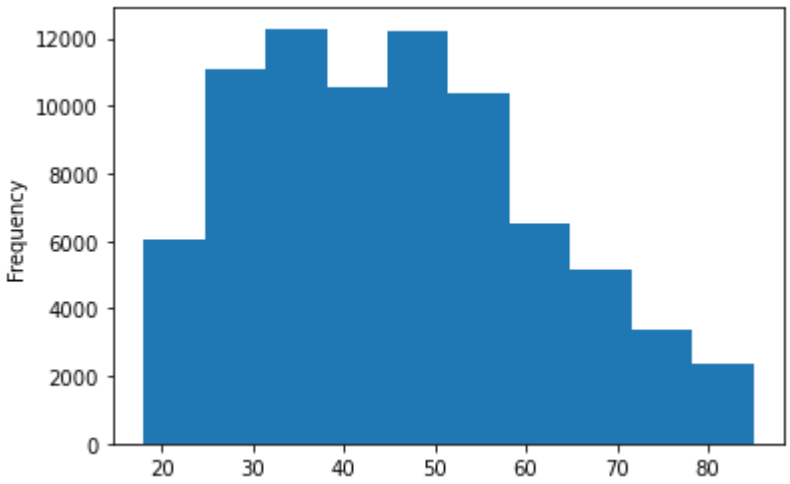
In [367…
```
data['in-store'].plot.hist(bins=3)
```

Out[367]:
```
<AxesSubplot:ylabel='Frequency'>
```



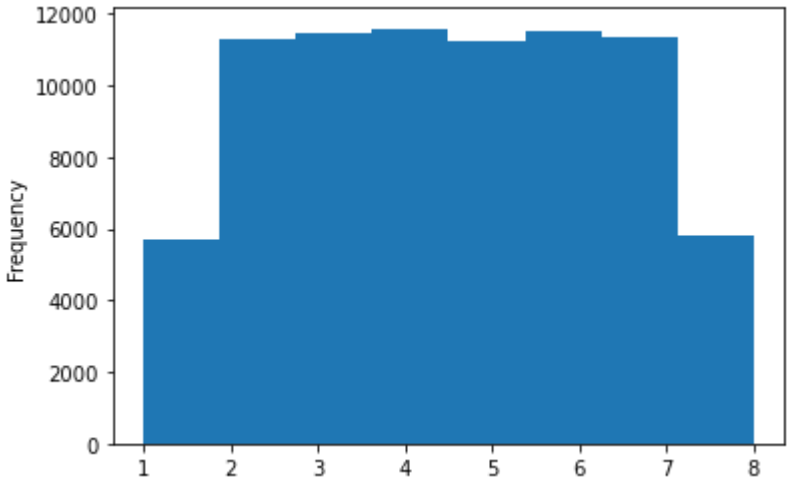
In [368…
```
data['age'].plot.hist()
```

Out[368]:
```
<AxesSubplot:ylabel='Frequency'>
```



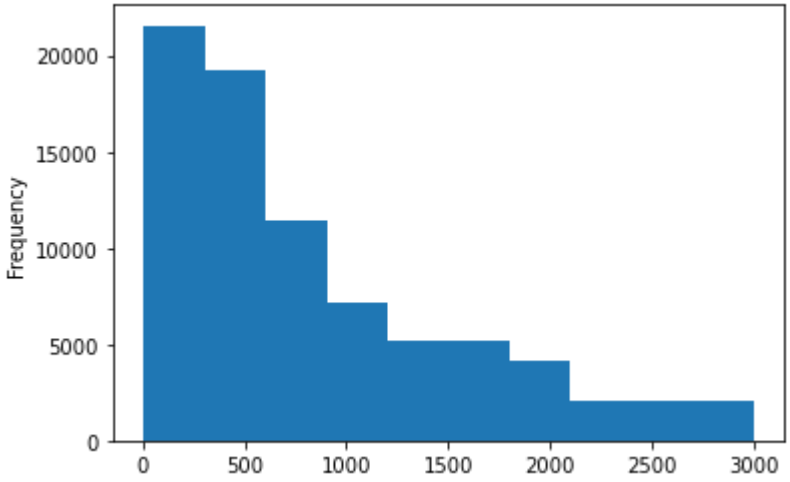
In [369…
```
data['items'].plot.hist(bins=8)
```

Out[369]:
```
<AxesSubplot:ylabel='Frequency'>
```



In [370…
```
data['amount'].apply(float).plot.hist()
```

Out[370]:
```
<AxesSubplot:ylabel='Frequency'>
```



In [371…
```
data['region'].plot.hist()
```

Out[371]:
```
<AxesSubplot:ylabel='Frequency'>
```