Filtering Data using filter

Let us go through the details about `filter`.

- We can use `filter` on top of `iterable` to return new `iterable` with all the elements satisfying the condition.
- It takes filter logic and iterable as arguments. We can pass filter logic either as regular function or lambda function.
- `filter` returns a special iterable called as `filter`. We have to type cast to regular collection such as `list` to preview the data or we can use for loop to iterate and print the data.
- Number of elements in the `filter` object is typically less than number of elements in the original `iterable` passed to it.

```
%run 02_preparing_data_sets.ipynb
```

```
orders[:10]
```

```
['1,2013-07-25 00:00:00.0,11599,CLOSED',
 '2,2013-07-25 00:00:00.0,256,PENDING_PAYMENT',
 '3,2013-07-25 00:00:00.0,12111,COMPLETE',
 '4,2013-07-25 00:00:00.0,8827,CLOSED',
 '5,2013-07-25 00:00:00.0,11318,COMPLETE',
 '6,2013-07-25 00:00:00.0,7130,COMPLETE',
 '7,2013-07-25 00:00:00.0,4530,COMPLETE',
 '8,2013-07-25 00:00:00.0,2911,PROCESSING',
 '9,2013-07-25 00:00:00.0,5657,PENDING_PAYMENT',
 '10,2013-07-25 00:00:00.0,5648,PENDING_PAYMENT']
```

```
len(orders)
```

```
68883
```

```
order_items[:10]
```

```
['1,1,957,1,299.98,299.98',
 '2,2,1073,1,199.99,199.99',
 '3,2,502,5,250.0,50.0',
 '4,2,403,1,129.99,129.99',
 '5,4,897,2,49.98,24.99',
 '6,4,365,5,299.95,59.99',
 '7,4,502,3,150.0,50.0',
 '8,4,1014,4,199.92,49.98',
 '9,5,957,1,299.98,299.98',
 '10,5,365,5,299.95,59.99']
```

```
len(order_items)
```

```
172198
```

```
filter?
```

```
Init signature: filter(self, /, *args, **kwargs)
Docstring:
filter(function or None, iterable) --> filter object

Return an iterator yielding those items of iterable for which function(item)
is true. If function is None, return the items that are true.
Type:           type
Subclasses:
```

- We need to pass a function with filter logic and an `iterable` to filter.

```
orders[:10]
```

```
['1,2013-07-25 00:00:00.0,11599,CLOSED',
 '2,2013-07-25 00:00:00.0,256,PENDING_PAYMENT',
 '3,2013-07-25 00:00:00.0,12111,COMPLETE',
 '4,2013-07-25 00:00:00.0,8827,CLOSED',
 '5,2013-07-25 00:00:00.0,11318,COMPLETE',
 '6,2013-07-25 00:00:00.0,7130,COMPLETE',
 '7,2013-07-25 00:00:00.0,4530,COMPLETE',
 '8,2013-07-25 00:00:00.0,2911,PROCESSING',
 '9,2013-07-25 00:00:00.0,5657,PENDING_PAYMENT',
 '10,2013-07-25 00:00:00.0,5648,PENDING_PAYMENT']
```

```python
order = '1,2013-07-25 00:00:00.0,11599,CLOSED'
int(order.split(',')[2]) == 11599
```

```
True
```

- Get orders placed by customer id 12431

```python
customer_orders = filter(
    lambda order: int(order.split(',')[2]) == 12431,
    orders
)
```

```python
type(customer_orders)
```

```
filter
```

```python
customer_orders
```

```
<filter at 0x7f7986dc7898>
```

```python
list(customer_orders)
```

```
['3774,2013-08-16 00:00:00.0,12431,CANCELED',
 '3870,2013-08-17 00:00:00.0,12431,PENDING_PAYMENT',
 '4032,2013-08-17 00:00:00.0,12431,ON_HOLD',
 '22812,2013-12-12 00:00:00.0,12431,PENDING',
 '22927,2013-12-13 00:00:00.0,12431,CLOSED',
 '25614,2013-12-30 00:00:00.0,12431,CLOSED',
 '27585,2014-01-12 00:00:00.0,12431,PROCESSING',
 '28244,2014-01-15 00:00:00.0,12431,PENDING_PAYMENT',
 '29109,2014-01-21 00:00:00.0,12431,ON_HOLD',
 '29232,2014-01-21 00:00:00.0,12431,ON_HOLD',
 '45894,2014-05-06 00:00:00.0,12431,CLOSED',
 '46217,2014-05-07 00:00:00.0,12431,CLOSED',
 '49678,2014-05-31 00:00:00.0,12431,PENDING',
 '51865,2014-06-15 00:00:00.0,12431,PROCESSING',
 '63146,2014-02-13 00:00:00.0,12431,PENDING_PAYMENT',
 '67110,2014-07-14 00:00:00.0,12431,PENDING']
```

- Get orders placed by customer id 12431 in the month of 2014 January

```python
customer_orders_for_month = filter(
    lambda order: int(order.split(',')[2]) == 12431
        and order.split(',')[1].startswith('2014-01'),
    orders
)
```

```python
for rec in customer_orders_for_month:
    print(rec)
```

```
27585,2014-01-12 00:00:00.0,12431,PROCESSING
28244,2014-01-15 00:00:00.0,12431,PENDING_PAYMENT
29109,2014-01-21 00:00:00.0,12431,ON_HOLD
29232,2014-01-21 00:00:00.0,12431,ON_HOLD
```

- Get orders placed by customer id 12431 in processing or pending_payment for the month of 2014 January

```python
customer_orders_for_month = filter(
    lambda order: int(order.split(',')[2]) == 12431
        and order.split(',')[1].startswith('2014-01')
        and order.split(',')[3] in ('PENDING_PAYMENT', 'PROCESSING'),
    orders
)
```

```python
list(customer_orders_for_month)
```

```
['27585,2014-01-12 00:00:00.0,12431,PROCESSING',
 '28244,2014-01-15 00:00:00.0,12431,PENDING_PAYMENT']
```

```python
def check_order(order_details, customer_id, order_month, order_statuses):
    if int(order_details[2]) == customer_id and \
        order_details[1].startswith(order_month) and \
        order_details[3] in order_statuses:
        return True
```

```python
customer_orders_for_month = filter(
    lambda order: check_order(order.split(','), 12431, '2014-01', ('PENDING_PAYMENT',
'PROCESSING')),
    orders
)
```

```python
list(customer_orders_for_month)
```

```
['27585,2014-01-12 00:00:00.0,12431,PROCESSING',
 '28244,2014-01-15 00:00:00.0,12431,PENDING_PAYMENT']
```