

Using Two Maps to Generate another Map

Given:

- a map that maps a channel name to it's number of subscribers
- another map that maps a channel name to its publisher

generate a map that maps the publisher to its number of subscribers.

```
Map<String, Integer> channelToSubscribers
    = new TreeMap<>(); // channelName, numSubscribers
Map<String, String> channelToPublisher
    = new TreeMap<>(); // channelName, publisher
Map<String, Integer> publisherToSubscribers
    = new TreeMap<>(); // publisher, numSubscribers

// channel -> number of subscribers    (K, V1)
channelToSubscribers.put("JustForLaughs", 120_000);
channelToSubscribers.put("JustForGags", 10_000);
channelToSubscribers.put("ContemplationTechniques", 10_000);
channelToSubscribers.put("A New Earth", 20_000);

// channel -> publisher    (K, V2)
channelToPublisher.put("JustForLaughs", "Charlie Chaplin");
channelToPublisher.put("JustForGags", "Charlie Chaplin");
channelToPublisher.put("ContemplationTechniques", "Echhart Tolle");
channelToPublisher.put("A New Earth", "Echhart Tolle");
```

Output:

```
publisher: Charlie Chaplin; numSubscribers:130000
publisher: Echhart Tolle; numSubscribers:30000
Publisher with most subscribers: Charlie Chaplin 130000
Publisher with fewest subscribers: Echhart Tolle 30000
```

Algorithm:

1. Using a `forEach(BiConsumer)`, set up the `publisherToSubscribers` map.
2. Using a `forEach(BiConsumer)`, output `publisherToSubscribers`.
3. Calculate the publisher with the most and least subscribers. `Collections.min()` and `Collections.max()` can be useful here.