

At the moment it will have to sort the entire table on every execution. This will take up a lot of time the bigger the table gets. ($>O(n)$)

For speeding up this query we should create an index in ascending order on the column `last_name` of the table `customer`:

```
CREATE INDEX last_name ON customer (last_name ASC);
```

Original

```
Sort  (cost=42.62..44.12 rows=599 width=17)
  Sort Key: last_name
    -> Seq Scan on customer  (cost=0.00..14.99 rows=599 width=17)
(3 rows)

Time: 1.717 ms
```

With Index

```
Sort  (cost=42.62..44.12 rows=599 width=17)
  Sort Key: last_name
    -> Seq Scan on customer  (cost=0.00..14.99 rows=599 width=17)
(3 rows)

Time: 1.355 ms
```

Normally now we would directly get the data in order from the seq scan. That would save us the sorting.

In this case the index is not used, because the dataset is too small. If the dataset would be bigger, the index would be used.