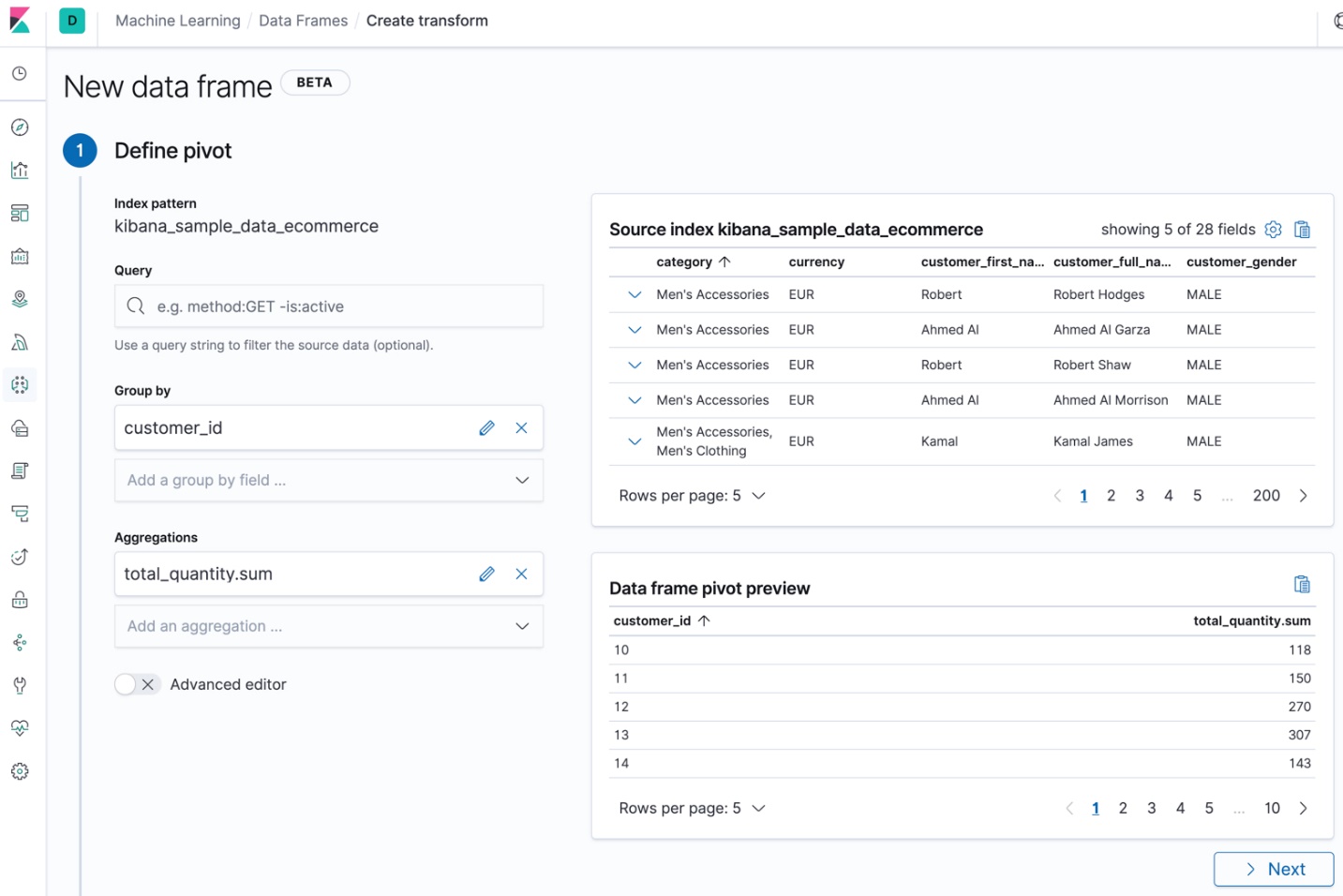
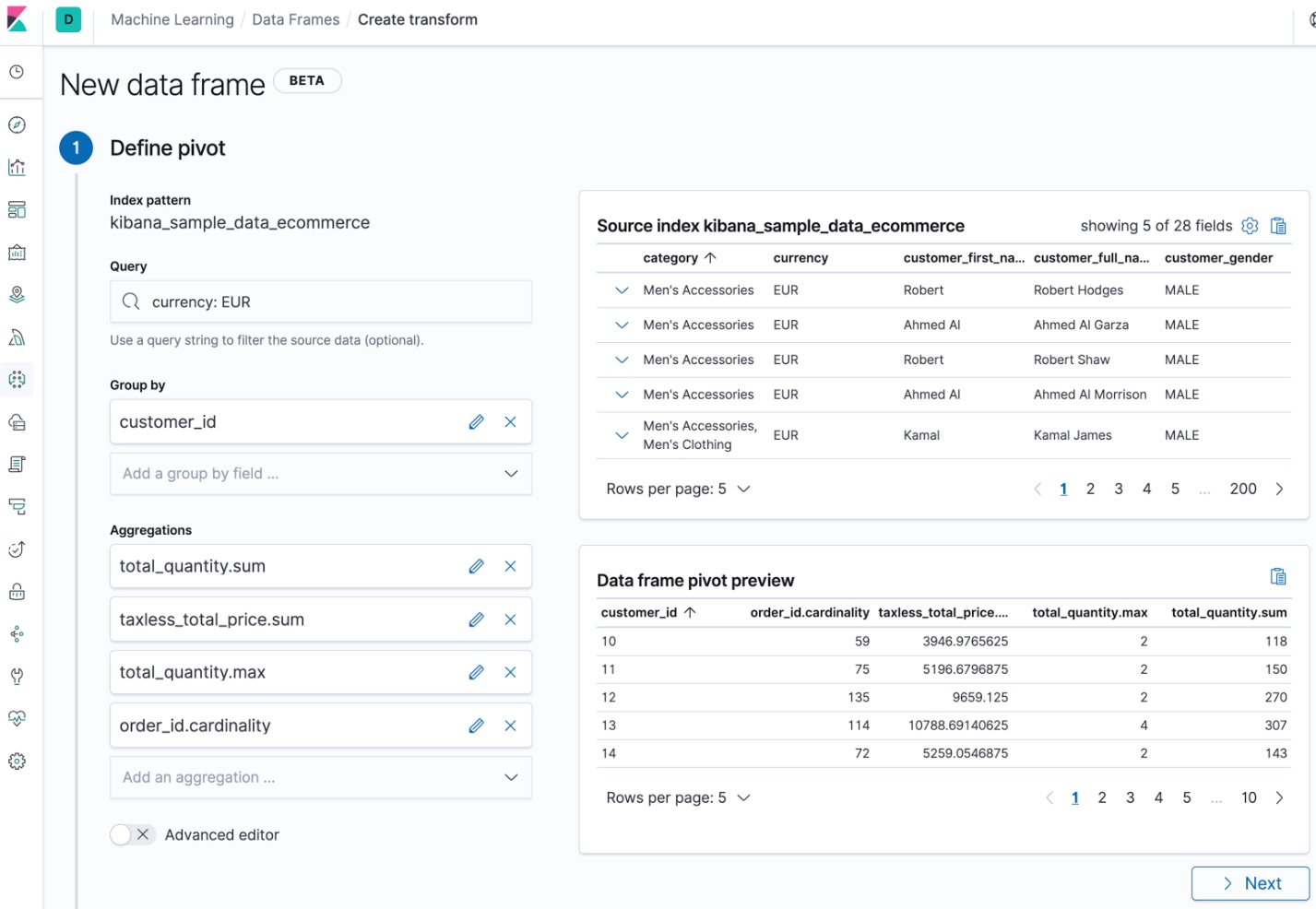
# Lab 5

1. Go to **Machine Learning** > **Data Frames** in Kibana and use the wizard to create a transform:



In this case, we grouped the data by customer ID and calculated the sum of products each customer purchased.

Let’s add some more aggregations to learn more about our customers' orders. For example, let’s calculate the total sum of their purchases, the maximum number of products that they purchased in a single order, and their total number of orders. We’ll accomplish this by using the [sum aggregation](https://www.elastic.co/guide/en/elasticsearch/reference/7.4/search-aggregations-metrics-sum-aggregation.html) on the taxless\_total\_price field, the [max aggregation](https://www.elastic.co/guide/en/elasticsearch/reference/7.4/search-aggregations-metrics-max-aggregation.html) on the total\_quantity field, and the [cardinality aggregation](https://www.elastic.co/guide/en/elasticsearch/reference/7.4/search-aggregations-metrics-cardinality-aggregation.html) on the order\_id field:



If you prefer, you can use the [preview transforms API](https://www.elastic.co/guide/en/elasticsearch/reference/7.4/preview-transform.html):

POST \_data\_frame/transforms/\_preview

{

"source": {

"index": "kibana\_sample\_data\_ecommerce",

"query": {

"bool": {

"filter": {

"term": {"currency": "EUR"}

}

}

}

},

"pivot": {

"group\_by": {

"customer\_id": {

"terms": {

"field": "customer\_id"

}

}

},

"aggregations": {

"total\_quantity.sum": {

"sum": {

"field": "total\_quantity"

}

},

"taxless\_total\_price.sum": {

"sum": {

"field": "taxless\_total\_price"

}

},

"total\_quantity.max": {

"max": {

"field": "total\_quantity"

}

},

"order\_id.cardinality": {

"cardinality": {

"field": "order\_id"

}

}

}

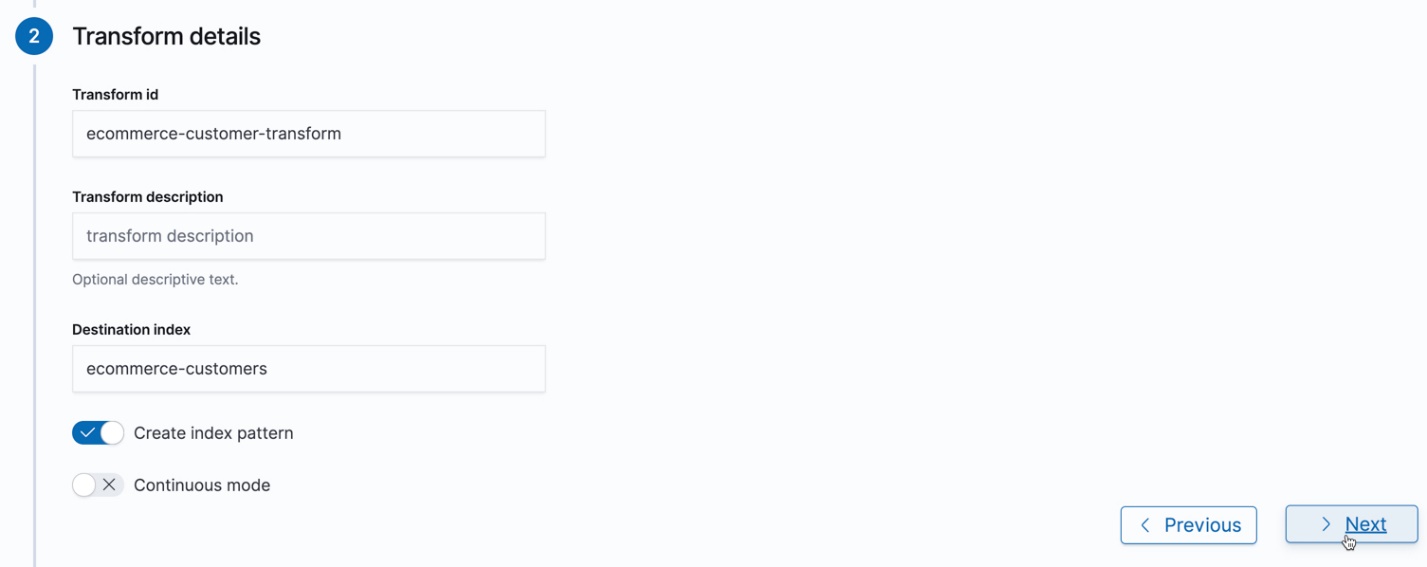
}

}

Copy as cURL[View in Console](http://localhost:5601/app/kibana#/dev_tools/console?load_from=https://www.elastic.co/guide/en/elasticsearch/reference/7.4/snippets/1040.console)

1. When you are satisfied with what you see in the preview, create the transform.
   1. Supply a job ID and the name of the target (or *destination*) index. If the target index does not exist, it will be created automatically.
   2. Decide whether you want the transform to run once or continuously.

Since this sample data index is unchanging, let’s use the default behavior and just run the transform once.



If you prefer, you can use the [create transforms API](https://www.elastic.co/guide/en/elasticsearch/reference/7.4/put-transform.html). For example:

PUT \_data\_frame/transforms/ecommerce-customer-transform

{

"source": {

"index": [

"kibana\_sample\_data\_ecommerce"

],

"query": {

"bool": {

"filter": {

"term": {

"currency": "EUR"

}

}

}

}

},

"pivot": {

"group\_by": {

"customer\_id": {

"terms": {

"field": "customer\_id"

}

}

},

"aggregations": {

"total\_quantity.sum": {

"sum": {

"field": "total\_quantity"

}

},

"taxless\_total\_price.sum": {

"sum": {

"field": "taxless\_total\_price"

}

},

"total\_quantity.max": {

"max": {

"field": "total\_quantity"

}

},

"order\_id.cardinality": {

"cardinality": {

"field": "order\_id"

}

}

}

},

"dest": {

"index": "ecommerce-customers"

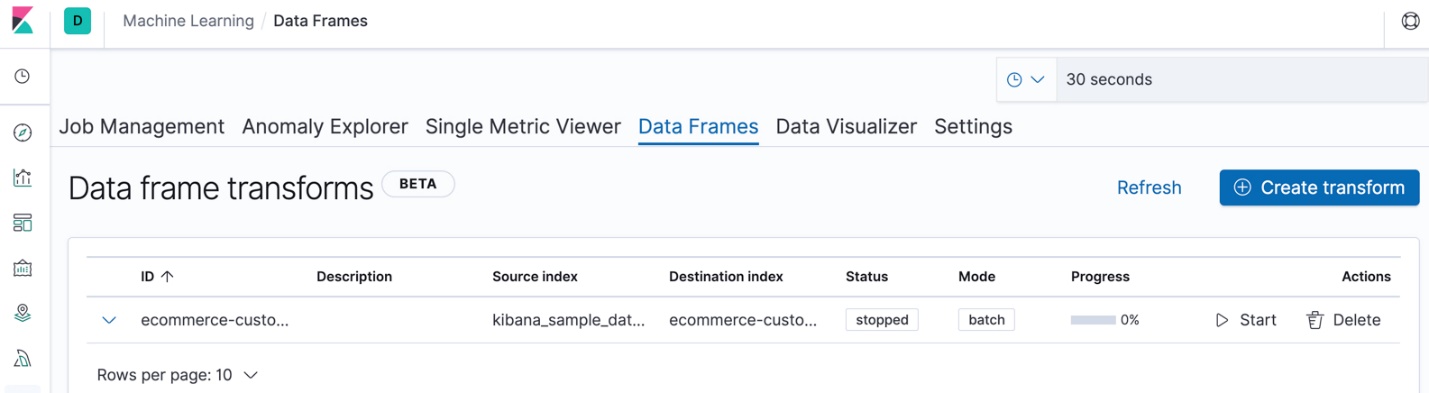
}

}

Copy as cURL[View in Console](http://localhost:5601/app/kibana#/dev_tools/console?load_from=https://www.elastic.co/guide/en/elasticsearch/reference/7.4/snippets/1041.console)

1. Start the transform.

You can start, stop, and manage transforms in Kibana:



Alternatively, you can use the [start transforms](https://www.elastic.co/guide/en/elasticsearch/reference/7.4/start-transform.html) and [stop transforms](https://www.elastic.co/guide/en/elasticsearch/reference/7.4/stop-transform.html) APIs. For example:

POST \_data\_frame/transforms/ecommerce-customer-transform/\_start

Copy as

1. Explore the data in your new index.

For example, use the **Discover** application in Kibana:

