Search. Observe. Protect.

Kibana Fundamentals Lab Guide

elastic.co

TABLE OF CONTENTS

KIBANA FUNDAMENTALS

Kibana is your window into the Elastic Stack. Kibana enables you to:

- **Analyze and visualize your data**. Search for hidden insights, compile a dashboard of charts, gauges, maps and other visualizations that show what you found, and share it with others.
- **Search, observe, and protect your data**. Add a search box to your app or website, analyze logs metrics, and find security vulnerabilities.
- **Manage, monitor, and secure the Elastic Stack**. Manage your indices and ingest pipelines, monitor the health of your Elastic Stack cluster, and control which users have access to which features and data.

In these labs, you will learn how to explore data in Kibana, how to create visualizations using Kibana Lens, and combine them in a dashboard.

You will use Kibana's sample datasets. One dataset describes flight information over the last 10 days. The second dataset represents orders for an e-commerce platform. You will use different Kibana visualizations to explore the data. You will gain insights to topics like typical delays by carrier and fluctuating ticket prices.

LAB 1: INTRODUCTION TO KIBANA

You will start by creating a deployment on Elastic Cloud. Next you will access Kibana, and load the sample datasets.

Creating a cloud deployment

1. Go to cloud.elastic.co
2. Click on the **Sign up** link.
3. Enter your email and choose a password.
4. Click **Start free trial**. This brings you to the Cloud console. Here, you can create a deployment to start a 14 day free trial.
5. Enter a name for the deployment, for example `kibana-workshop`.
6. Leave the other settings to their default values and click **Create deployment**. It will take a few minutes for the Elastics Stack to initialize.
7. You won't need the password that's shown to you. You will be using your Elastic Cloud account to access Kibana. Click **Continue without downloading**.
8. Wait a couple of minutes...
9. Click **Continue** when the button turns blue.

Loading sample data

1. Click **Add data**.
2. Select the **Sample data** tab.
3. Click the **Add data** buttons for all three sample data sets: **Sample eCommerce orders**, **Sample flight data**, and **Sample web logs**.
4. This action loads data in Elasticsearch indices. It also creates Kibana index patterns that describe how Kibana will use the data in Kibana. And finally, it creates the visualizations, dashboards and Canvas workpads that you will use later.

Getting to know Kibana

1. Click the menu button in the top left (it looks like three horizontal lines). This button opens Kibana's main menu. This menu has five sections:
    - **Analytics**, where you find the tools to analyze and visualize the data.
    - **Enterprise Search**, **Observability**, and **Security**, the homes for the three Elastic solutions.
    - **Management**, where you can manage your deployment.

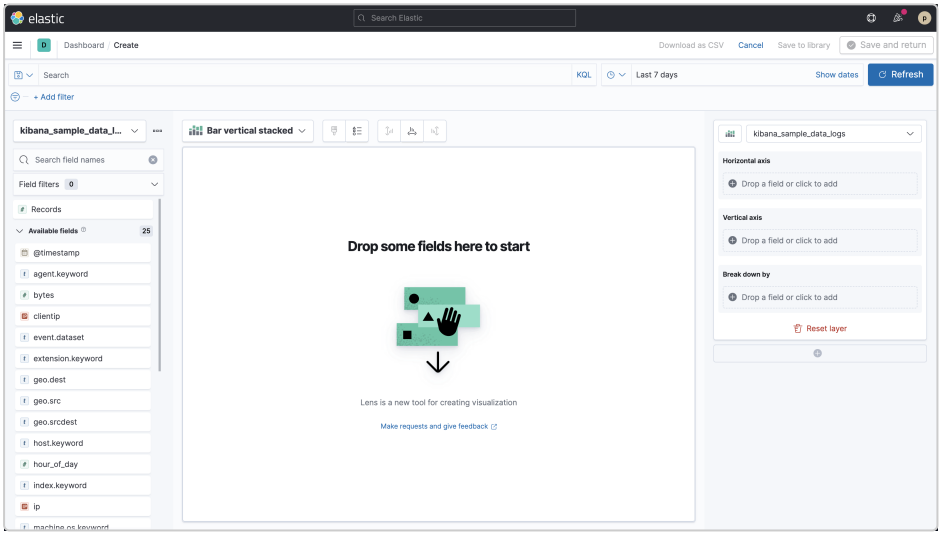
We will focus on the first section in the main menu: **Analytics**.

LAB 2: VISUALIZING DATA

Let's get started visualizing your data using Kibana Lens.

Introduction to Kibana Lens

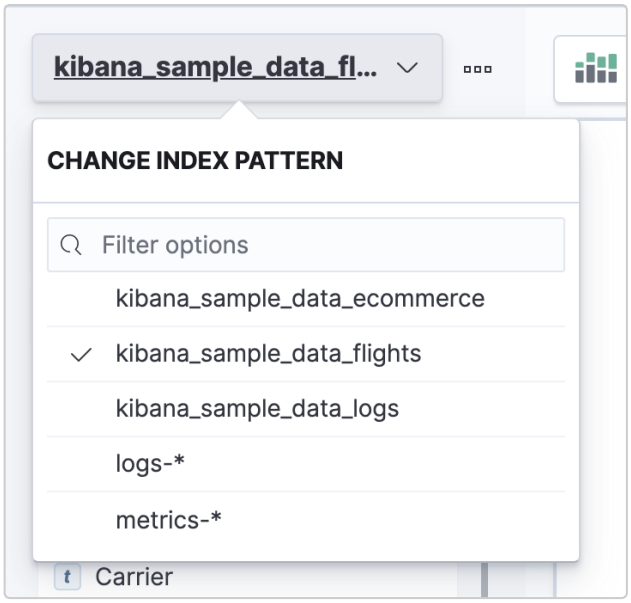
1. Open the main menu by clicking on the menu button located at the top left. Select **Dashboard** from the **Analytics** section.
2. Click on **Create dashboard** to create a new dashboard. This brings you to an empty dashboard.
3. Then click on **Create visualization** to add a new visualization to the dashboard.
4. This brings you to the Kibana Lens view:



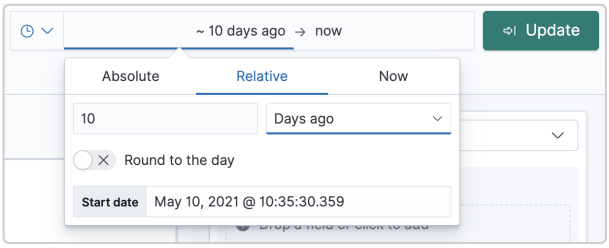
Let's break down what you see in this view:

- At the top left, you have the **query bar**, where you can write queries and create filters.
- At the top right, the **time filter** enables you to set the time period you wish to visualize.
- On the left, you see a list of the **fields** that exist in your data.
- In the middle, you have the **visualization area** where your visualization will be shown.
- And finally, on the right you find the **controls to configure** your visualization.

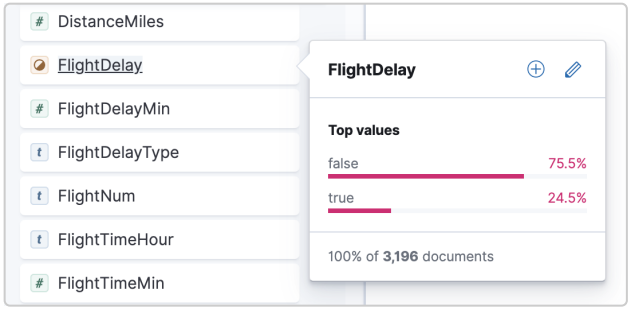
5. Select the `kibana_sample_data_flights` index pattern from the index pattern dropdown:



6. By default, Kibana visualizes the last 7 days of data. You can change that using the time filter in the top right corner:
   - Select **Relative** and set the following time period. Click **Update** for changes to take effect:

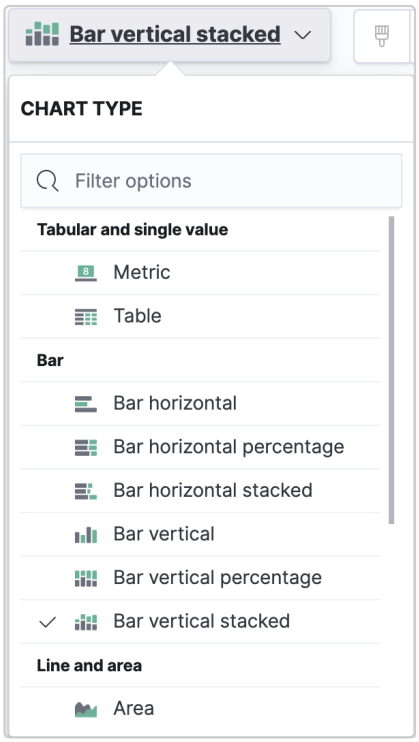


7. Note the list of fields presented on the left-hand side. You can search for a specific field or group them by type. For convenience, Lens will only show fields that contain data. Click on a field for a quick view of the top values in a field or their distribution:

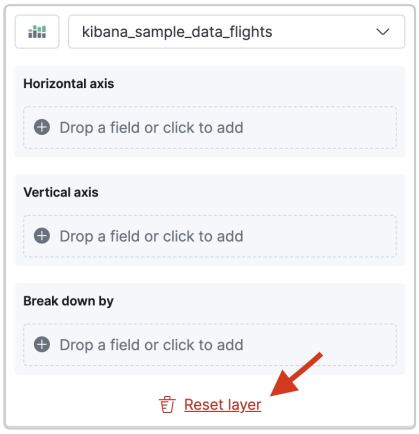


8. Click on the different fields in the index to familiarize yourself with the data.
9. To start visualizing a field, simply drag and drop it into the central visualization area of the page.
10. Below the visualization area, Lens shows some suggestions for alternate visualization types. You are not limited to those suggestions. To change the chart type, click the dropdown next to the

index name:



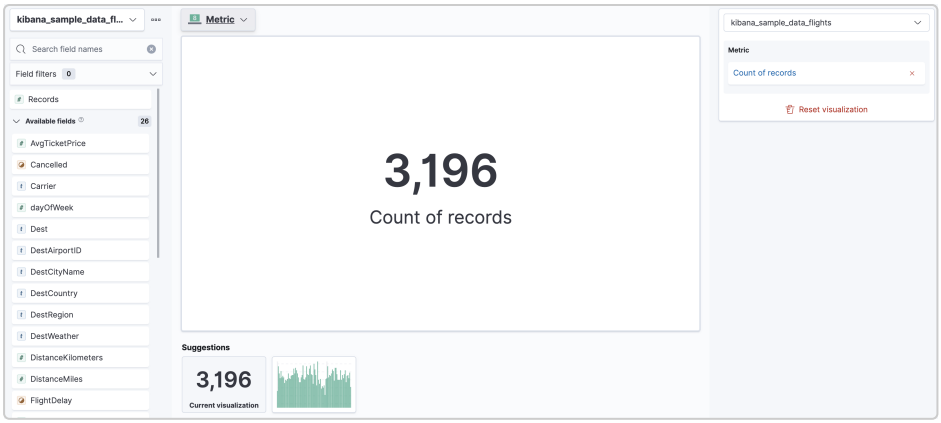
11. Reset the chart area by clicking **Reset layer**:



Metric visualization

The most basic visualization is a metric visualization. It shows a number. Let's create such a metric visualization.

1. Drag **Records** to the central visualization area.
2. Select **Metric** as the chart type.

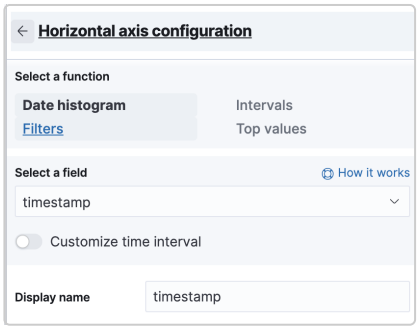


3. The count of records represents the number of flights in our dataset. Let's change the label accordingly. Click on **Count of records** on the right-hand side and enter `Flights` as the **Display name**.
4. Click **Save and return** in the top right.

Bar chart

Next, let's visualize the flight delays per carrier as a bar chart.

1. From the same dashboard used in the previous exercise, click **Create visualization** to add another visualization.
2. Drag and drop the **Records** field.
3. Click on **Add filter** to create a filter. Select **FlightDelay** is `true` and create a custom label `Delayed flights`. Click **Save** to apply the filter.
4. Next, you will change the horizontal axis to reflect carriers. Click **timestamp** on the right-hand side to open this side panel:

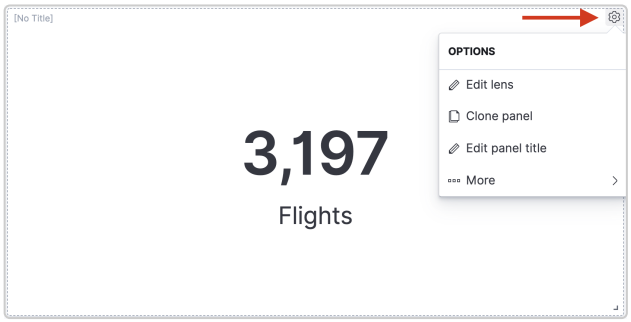


5. Select **Top values** and change the field from **timestamp** into **Carrier**. Decrease the number of values to 4.
6. Change the **Display name** into `Delays by carrier` and close the side panel.
7. Finally, click **Save and return** to return to the dashboard.

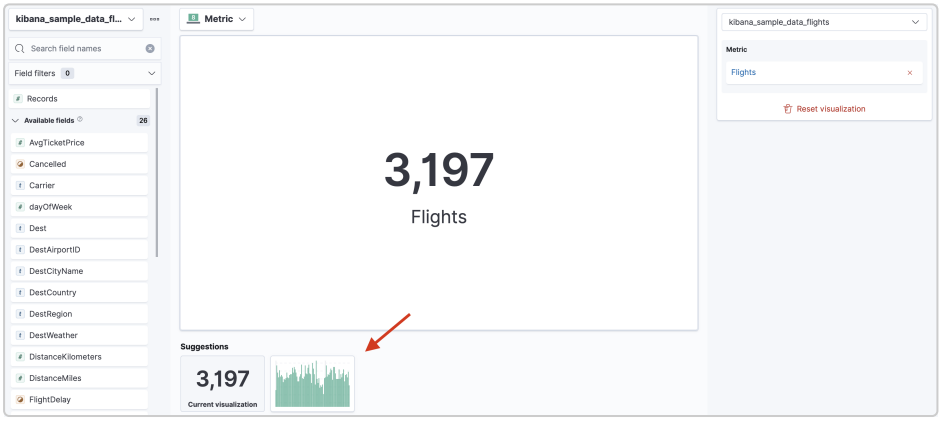
Stacked bar chart

Stacked bar charts allow you to break down the bars in a bar chart by a category. Let's build a stacked bar chart that shows the number of flights over time, broken down by delay type.

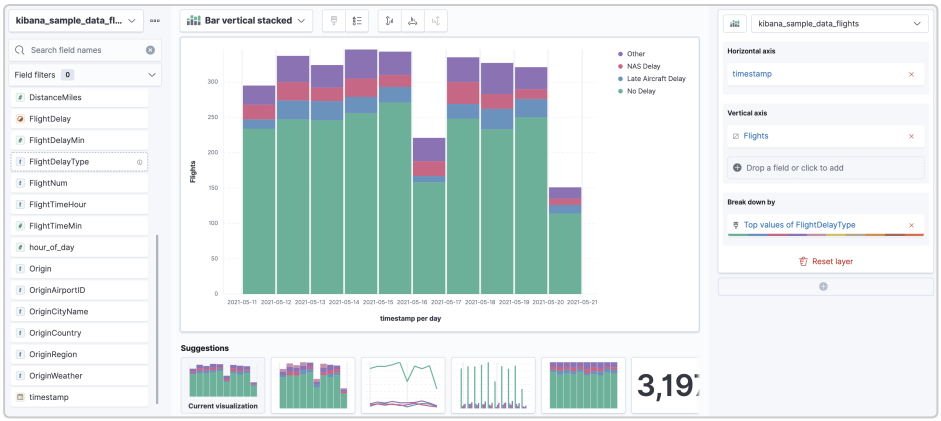
1. You don't have to start from scratch, when you want to create a new Lens visualization. It's also possible to clone an existing visualization and edit it. Click on the gear in the top right corner of the metric visualization you created earlier. This opens the panel options dialog:



2. Select **Clone panel**. This creates a copy of the metric visualization and adds it to your dashboard.
3. Click on the gear in the top right corner of the new copy, and select **Edit lens**.
4. This opens the editor for this visualization. Select the vertical bar chart under **Suggestions** below the chart:



5. The horizontal axis shows the **timestamp** field by default. Lens has selected a time interval for you. You can change the interval by clicking **timestamp** on the right. Toggle **Customize time interval** and set the **Minimum interval** to 1 day.
6. Close the side panel and drag and drop the **FlightDelayType** field into the **Break down by** area on the right.

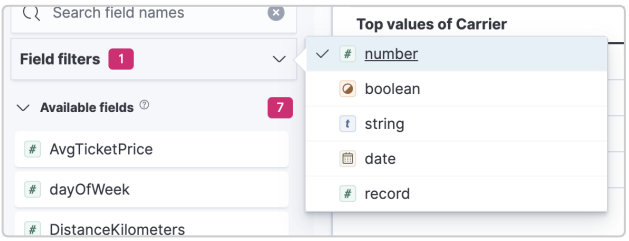


7. Click **Save and return** to return to the dashboard.

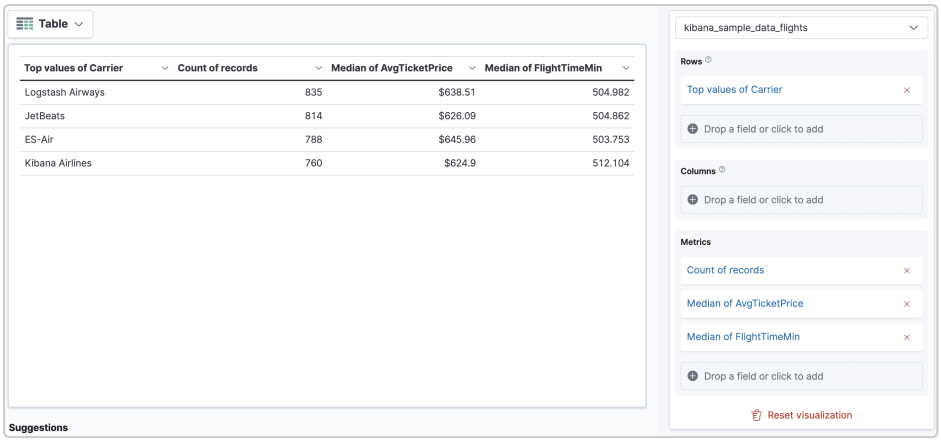
Table

Data tables display information in a grid-like format of rows and columns. Let's create one that shows various metrics for every carrier, like average ticket price.

1. From the same dashboard used in the previous exercise, click **Create visualization** to add another Lens visualization.
2. Drag and drop the **Carrier** field into the chart.
3. Select **Table** from the dropdown above the visualization area.
4. Let's add some more columns to the table. Use **Field filters** (on the left, above the list of fields) to only list numeric fields:



5. Drag the following fields to the **Metrics** area on the right-hand side: **AvgTicketPrice**, and **FlightTimeMin**. The resulting table will look like this:



6. Click **Median of AvgTicketPrice** in the header of the table, and select **Sort ascending** to sort the carriers on median ticket price.
7. Click **Save and return** to return to the dashboard.
8. Finally, save your dashboard by clicking **Save** in the top right. Give your dashboard a title, and click the **Save** button.

Downloading as CSV

Sometimes, you may want to export the data underlying a visualization. Kibana gives you the possibility to download the data for every Lens visualization as a CSV file. This enables you to open the file as a spreadsheet for further analysis for example.

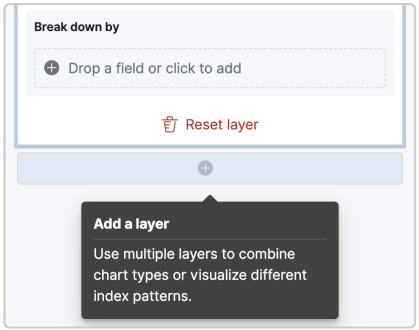
1. On the dashboard, click on the top right corner of a visualization to open that panel's options dialog (for example for the table panel that you have just created).
2. Select **More** and next **Download as CSV**.
3. Download the file and open it as a spreadsheet or a text file to view the data.

So far, you have used Kibana Lens to create basic visualizations. Next, you will build more advanced visualizations.

Multiple layers and indices

You can create multiple layers with Lens. Layers enable you to combine data from multiple sources into one visualization.

1. First, you're going to create a new dashboard. Open the main menu by clicking on the menu button located at the top left. Select **Dashboard** from the **Analytics** section.
2. Click on **Create dashboard** to create a new dashboard.
3. Click on **Create visualization** to add a new Lens visualization.
4. Select the `kibana_sample_data_flights` index, if it is not already selected. Ensure that the time filter indicates **Last 10 days** to visualize only the data from the last 10 days.
5. Change the chart type to **Area**.
6. Drag and drop **Records** into the chart area.
7. Click **timestamp** on the right-hand side. Toggle **Customize time interval** and set **Minimum interval** to 1 day. Close the side panel.
8. Click on **Count of records** on the right-hand side and change **Display name** into `Flights`. Close the side panel.
9. Click on the **+** button (below **Reset layer** on the right-hand side) to add another layer.



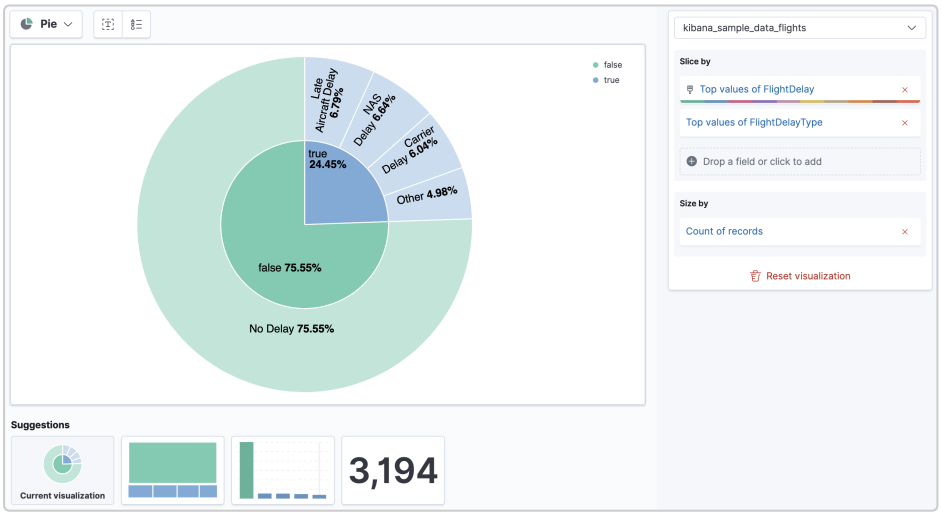
10. Select `kibana_sample_data_ecommerce`.
11. Drag and drop **order_date** on the horizontal axis.
12. Customize the time interval and set it to 1 day. Close the side panel.
13. Drag and drop **Records** on the vertical axis.
14. Change **Display name** into `Sales`.
15. Change **Axis side** into **Right**.
16. The number of flights is now shown on the left axis. The number of sales on the right axis. Click **Save and return** to return to the dashboard.

Sub-buckets

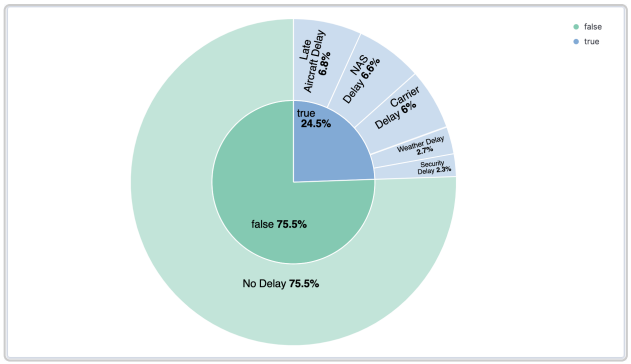
In an earlier exercise you have seen that it's possible to break down charts by the values of a field. Sometimes it's necessary to further sub-group the data. You can use visualizations that support sub-buckets for that.

Next, you will create a pie chart in which each slice is further broken down using sub-buckets.

1. From within the same dashboard used in the previous exercise, click on **Create visualization** to add another Lens visualization.
2. Select the `kibana_sample_data_flights` index.
3. Drag and drop **FlightDelay** into the chart area.
4. Change the chart type to **Pie**.
5. Drag and drop the **FlightDelayType** field into the chart area. Your chart should look like the chart below. The inner ring shows the percentage of delayed flights. The outer ring breaks down the delayed flights into the delay type.



6. You can customize the labels on this donut chart. Click on the first button to the right of the chart type dropdown (it looks like the letter "T" in a dotted square). Set the **Maximum decimal places for percent** to `1`.
7. By default, the outer ring only shows three values, and groups the other values into a slice called **Other**. To change that, click on the **Top values of FlightDelayType** (below **Slice by** on the right-hand side) and set **Number of values** to `5`.
8. The end result should look like the following chart:



9. Click **Save and return** to return to the dashboard.

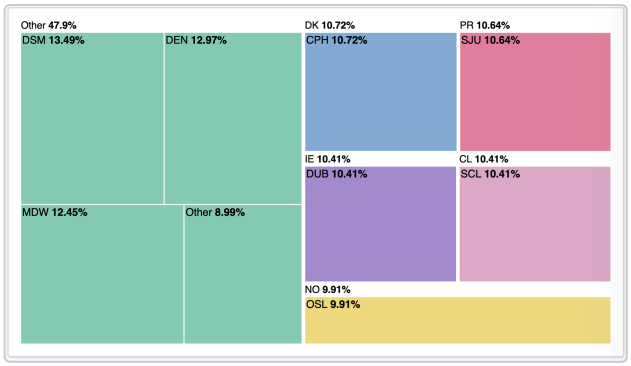
Treemap

You can also visualize hierarchical data using Lens. For instance, you can use a treemap or pie chart to display the top destination countries. Next, for each of those countries, you can further break down the top destination airports.

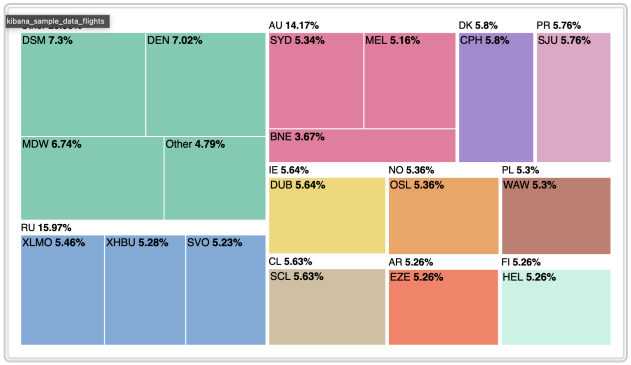
1. Click on **Create visualization** to add a new Lens visualization.
2. Select the `kibana_sample_data_flights` index, if it is not already selected.
3. Drag and drop **DestCountry** into the chart area.
4. Select the **Treemap** chart type.
5. By default, you will see a breakdown of the top countries represented by the number of records.

6. The rectangle size of the treemap represents the number of records. You can use another metric instead, like the average of the ticket price. Click on **Count of records** under **Size by** on the right. Select **Average** and **AvgTicketPrice** as the field.

7. The next step is to further break down each destination country by destination airport. Drag and drop the **DestAirportID** field into the chart area or the **Group by** section on the far right.



8. You can further customize the visualization to make it more interesting. Click on the **Top values of DestCountry** option and then set **Number of values** to `10`. You should get something like below:

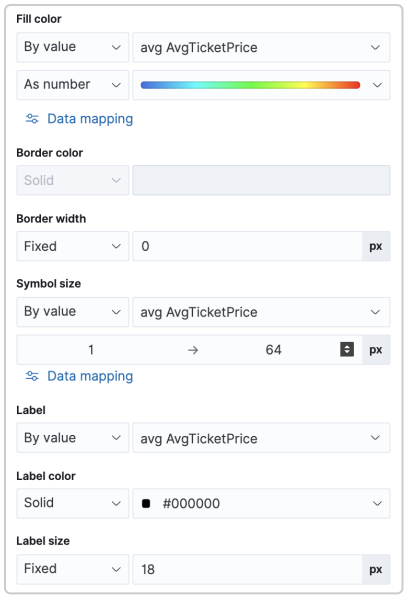


9. Click **Save and return** to return to the dashboard.
10. Finally, save your dashboard by clicking **Save** in the top right. Give your dashboard a title, and click the **Save** button.

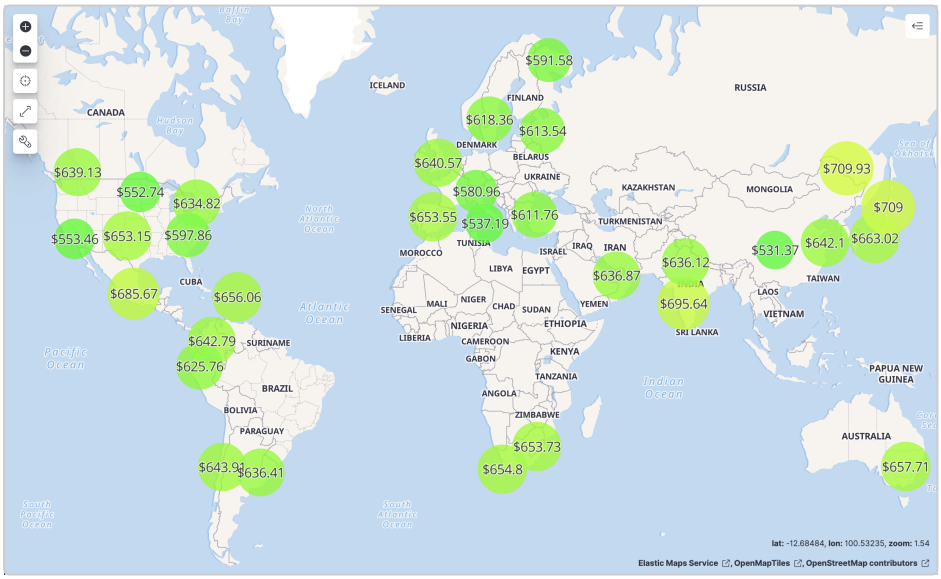
An interesting way to visualize data is using geographical maps. In this lab, you will learn how to use layers to plot your data on a map.

Getting started with maps

1. Open the main menu by clicking on the menu button located at the top left. Select **Dashboard** from the **Analytics** section.
2. Click on **Create dashboard** to create a new dashboard. This brings you to an empty dashboard.
3. Then, click on **All types** and select **Maps** to add a new maps visualization.
4. This will bring you to an empty map, to which you will add a layer.
5. If necessary, adjust the time filter in the top right so you see the last 10 days of data.
6. Click **Add Layer**. You can see the different types of layers that you can add to this map. Select **Clusters and grids**.
7. Select **kibana_sample_data_flights** from the **Index pattern** dropdown.
8. Change the **Geospatial field** value to **OriginLocation**.
9. Under **Show as**, keep **clusters** selected.
10. Click the **Add layer** button. This layer shows the flight origin locations as clusters. The size of each cluster represents the number of flight departures at each location in the data.
11. Next, you will configure the layer you have just added. Under **Metrics**, Select **Average** aggregation over the **AvgTicketPrice** field.
12. Scroll down to **Layer Style** and change the following:
    - Fill color: By Value: **avg AvgTicketPrice** and change the color to full rainbow
    - Symbol size: By Value: **avg AvgTicketPrice** and change the pixel size to **1 → 64**
    - Label: By Value: **avg AvgTicketPrice**
    - Label Size: Fixed **18**



13. Click on **Save & close**. The resulting map now shows the departure locations, with the size and color of each cluster representing the average ticket price for flights from that location:



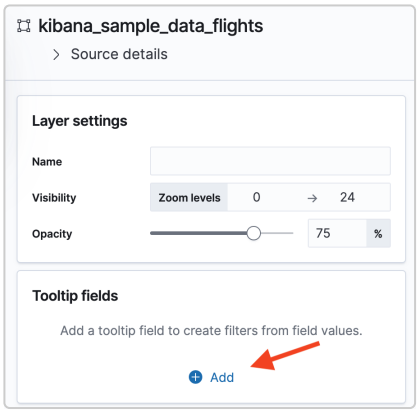
14. Click **Save and return** to return to the dashboard.

Multiple layers

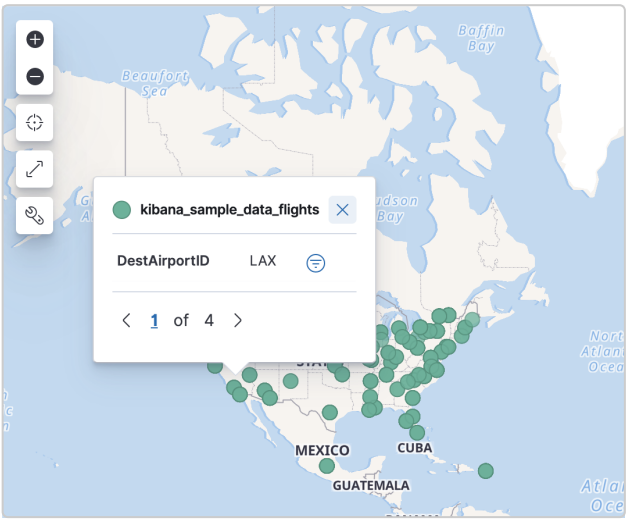
It's possible to add multiple layers to one map. That's what you will do next.

1. From your dashboard, click on **All types** and select **Maps** to add a new maps visualization.
2. This will bring you to an empty map, to which you will add two layers.
3. Click **Add layer**.

4. Select **Documents**. Individual records are called "documents" in the Elastic Stack. This layer type allows you to plot individual documents, i.e. flights.
5. Set the index pattern to **kibana_sample_data_flights**.
6. Keep **DestLocation** selected under **Geospatial field** and click **Add layer**. This layer shows individual destinations as a dot on the map.
7. Click on **Add** under **Tooltip fields** and select the **DestAirportID** field.

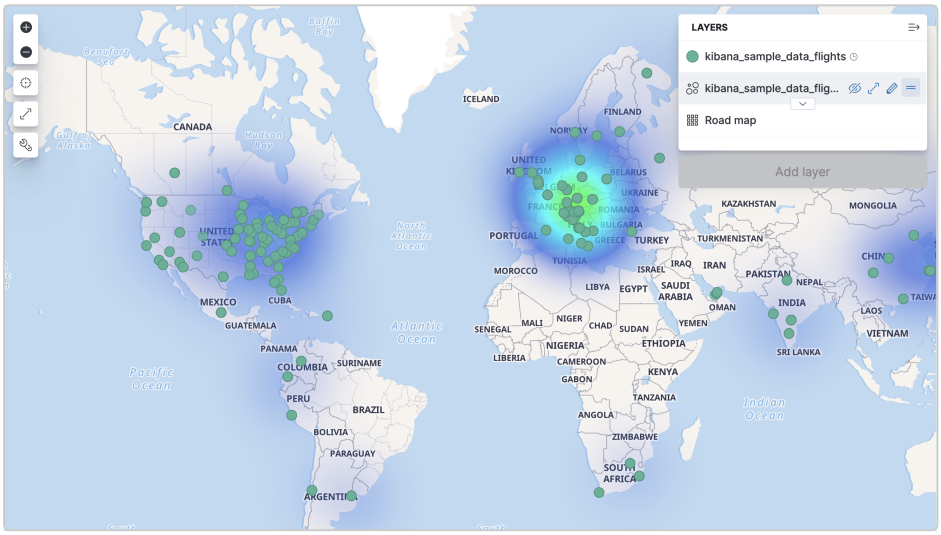


8. If you now click on one of the dots on the map, you will see the destination airport code in the tooltip:



9. Click **Save & close**.
10. Click **Add layer** to add another layer.
11. Select **Heat map**.
12. Set the index pattern to **kibana_sample_data_flights**.
13. Keep **DestLocation** selected under **Geospatial field** and click **Add layer**. You should now have two layers displayed on your map.

14. If you hover with your mouse over each of the layer names in the "LAYERS" panel, you should see a symbol appear that looks like two horizontal lines. You can use this to drag the layes, in order to change the order. Swap the order of the layers so that the documents layer is at the top and the heat map layer is at the bottom:

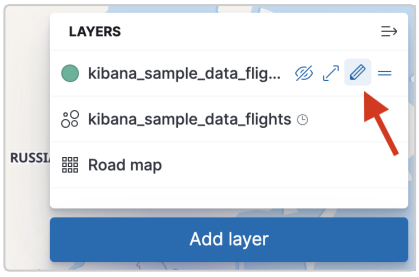


15. Click **Save and return** to return to the dashboard.

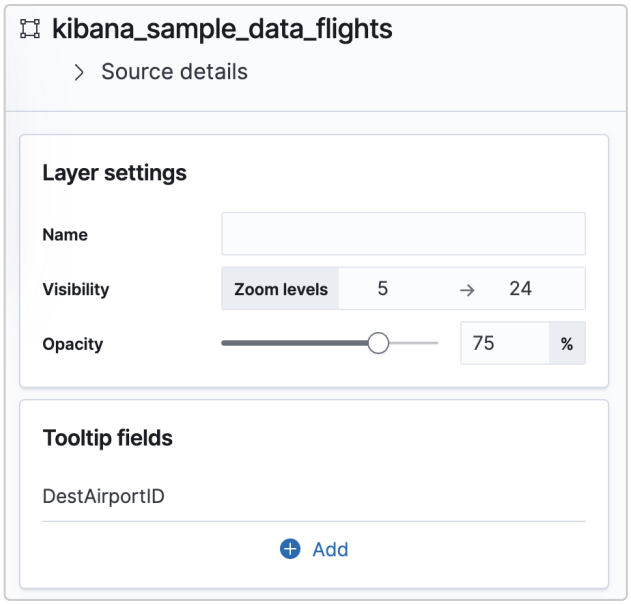
Zoom levels and layer visibility

You have just created a map with two layers. The map shows both layers by default. You can change the visibility of the layers depending on their zoom level. For instance, you may only want to make the documents visible on the map at a specific zoom level.

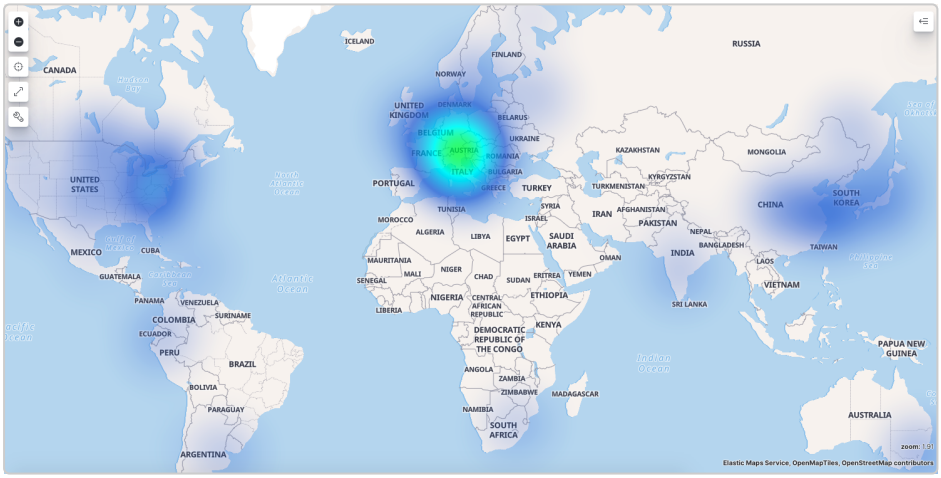
1. Open the previous map with the two layers. You can do that from the dashboard by clicking the gear icon at the top right of the panel, and selecting **Edit map**.
2. To customize the first layer, click on the edit button for that layer. It looks like a pencil, and appears when you hover over the layer name with your mouse:



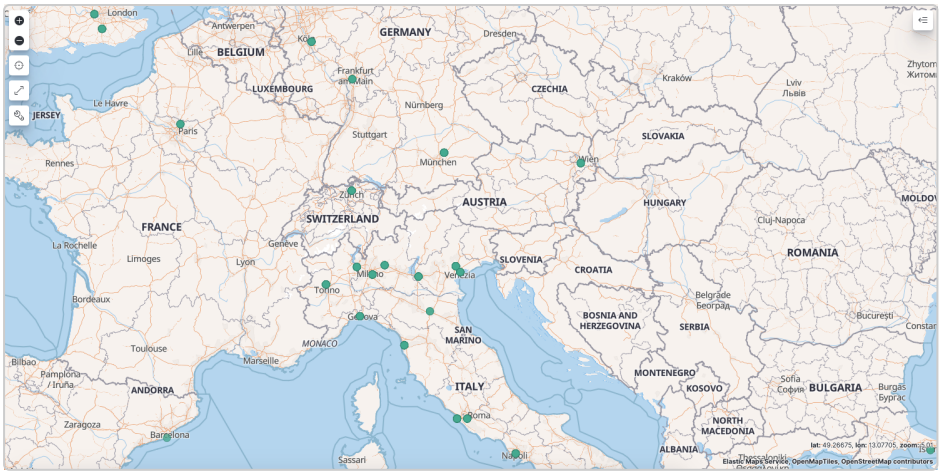
3. Set zoom levels **5** to **24**. As a result, this layer will only be visible in that range of zoom levels.



4. Click on **Save & close**.
5. Click to edit the second layer (i.e., heat map layer).
6. Set zoom levels **0** to **5**. As a result, this layer will not be visible at zoom levels higher than 5.
7. Click on **Save & close**.
8. When the map visualization is zoomed out it should look like the picture below. Notice that only the heap map layer is visible because the current zoom level is approximately 1.



9. After zooming in a few levels above 5, the heat map layer is no longer visible. Now, the documents layer become visible.

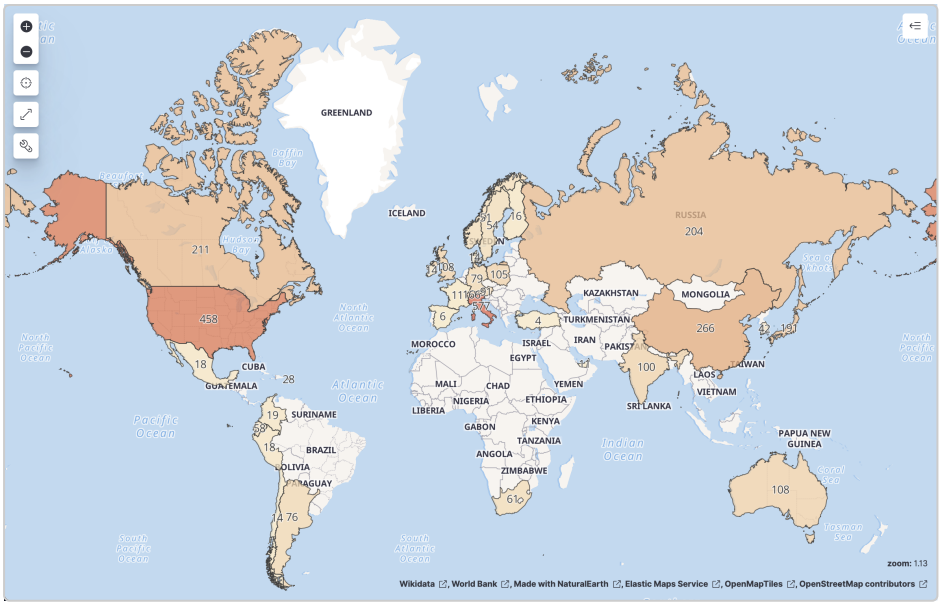


10. Click on **Save and return** to save the visualization and return to the dashboard.

Administrative boundaries

So far, you have created layers based on geo locations (longitude and latitude). You can also base a layer on administrative boundaries. An administrative boundary can be a country or region (province, county, state, ...). When you have country or region information in your data, you can use that in combination with the administrative boundaries that the Elastic Maps Service (EMS) provides.

1. From your dashboard, click on **All types** and select **Maps** to add a new maps visualization.
2. This will bring you to an empty map, to which you will add a countries layer.
3. Click **Add layer**.
4. Select **Choropleth**.
5. Select **World Countries**.
6. The flight data contains two-letter country codes. Select **ISO 3166-1 alpha-2 code** (default) to join your data with the countries in the Elastic Map Service.
7. Select **kibana_sample_data_flights** as the index pattern..
8. Select **DestCountry** as the join field and click **Add layer**.
9. The map now shows destination countries. The darker the color, the more flights go there. Click **Save & close**.



10. Save the map and return to your dashboard.
11. Finally, save your dashboard by clicking **Save** in the top right. Give your dashboard a title, and click the **Save** button.

LAB 3: DISCOVER

Next, you will learn how to use Discover to explore and understand your data.

Searching your data

"You know, for search!" Let's learn how to use the search capabilities of Elasticsearch in Kibana.

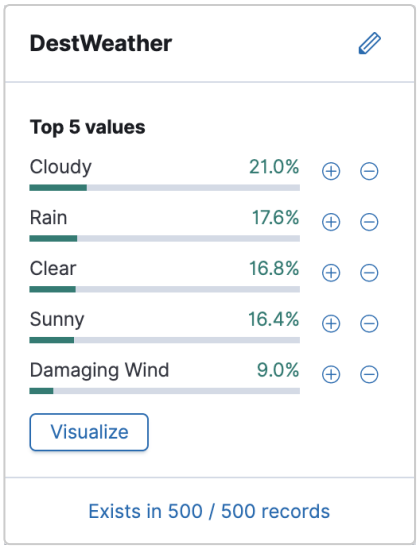
There are several ways to query your data in Kibana, including:

- Adding filters under the query bar.
- Writing a query in the query bar using the Kibana Query Language, KQL.

We will cover these here.

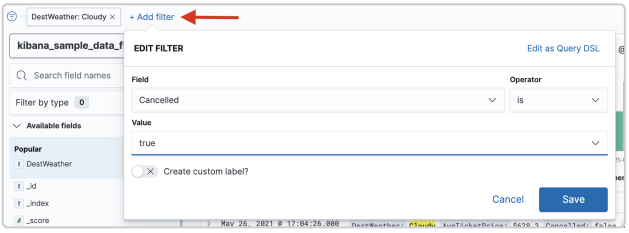
Getting to know Discover

1. Open the main menu by clicking on the menu button located at the top left. Select **Discover** from the **Analytics** section.
2. Make sure the `kibana_sample_data_flights` index is selected in the top left.
3. The Discover view consists of four sections. At the top, you can create queries and filters. You can also set the time range that you're interested in. Ensure that the time filter indicates **Last 10 days**.
4. Below the query bar, you see a date histogram that shows you how your data has been ingested over time. Click and drag in the histogram to zoom into a specific time range. Use the "back" button of your browser to return to the last 10 days.
5. Below the histogram, you see a table of the most recent records, called "documents" in the Elastic Stack. Click on the **>** symbol to expand a document and see all its fields and their values.
6. On the left, you see a list of the fields that occur in your data. Click on **DestWeather** in the list of fields on the left to view common values for that field:

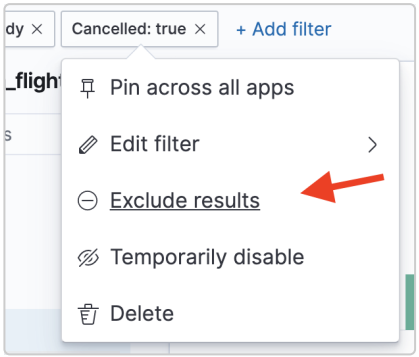


Filters

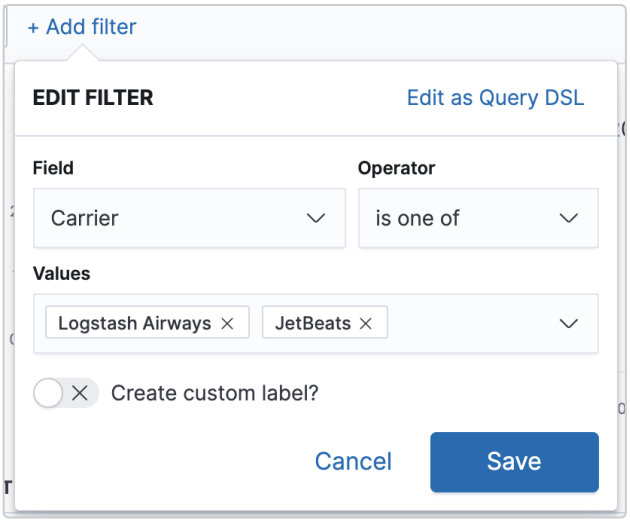
1. In the list of values for **DestWeather**, click on the **+** symbol after the **Cloudy** value. This creates a filter for this value. The results table now only shows flights where the destination weather was cloudy. Note that the filter is added at the top under the query bar.
2. Click **+ Add filter** in the top left. Select the **Cancelled** field, **is** for the operator, and next select **true** for the value. Create the filter by clicking **Save**:



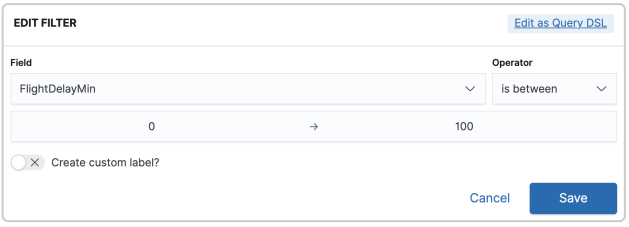
3. When you click on the filter you just created, a popup with a few actions appears. Click the **Exclude results** option to negate the filter. Now, the table only shows flights that were not cancelled.



4. Add a new filter for **Carrier** – is one of – **Logstash Airways** or **JetBeats**. Click **Save**.



5. Finally, add a filter for the **FlightDelayMin** (minutes of delay) between **0** and **100**:



Kibana Query Language (KQL)

Another way to drill down into your data is by using the query bar to write queries. These queries can be written using Kibana Query Language (KQL). You will next use the query bar to get to the same results as the filters you have just created.

1. Click **New** in the toolbar and make sure the `kibana_sample_data_flights` index is selected in the top left.
2. Enter the following query in the query bar and hit *Enter*:

```
DestWeather : Cloudy
```

   The number of hits (just above the date histogram) changes. Notice the auto complete feature for both fields and values as you type the query.

3. Add the condition on flight not being cancelled to the query:

```
and not Cancelled:true
```

   Of course you could have also set `and Cancelled:false`.

4. Add the condition for the carriers by adding the following to the query:

```
and Carrier:("Logstash Airways" or "JetBeats")
```

5. Add the following to restrict the search on flight delay:

```
and FlightDelayMin>=0 and FlightDelayMin<100
```

6. The full query should now be:

```
DestWeather:Cloudy and not Cancelled:true and
Carrier:("Logstash Airways" or "JetBeats") and
FlightDelayMin>=0 and FlightDelayMin<100
```

KQL is a powerful query language. You can also use it for full text search. For example if you look for destination city starting with B you can add `and DestCityName:B*`. This will match Bari, Bergamo, Buenos Aires, etc. You can even do a free search on any field. For example, `and Thunder*` that searches for "Thunder*" in any field. It will match both origin or destination weather.

To learn more about the KQL syntax, visit elastic.co/guide/en/kibana/current/kuery-query.html.

Saved search

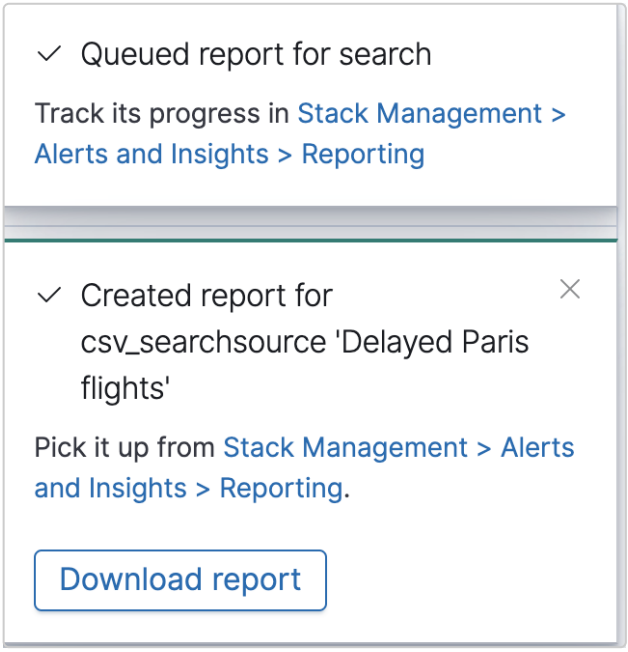
If you execute the same query often, it may be useful to save that search so you can return to it later. Next, you will customize the table view, and save it as a saved search.

1. Click **New** in the toolbar and make sure the `kibana_sample_data_flights` index is selected in the top left.
2. As you hover with your mouse over the different fields on the left hand side, a blue + appears. Click on the + for **OriginCityName**, **DestCityName**, **Carrier** and **FlightDelayMin**. This creates a custom table view that only includes these fields.
3. Use a query or filter to only show flights that have been delayed. (Hint: the field that indicates whether a flight was delayed or not is called **FlightDelay**.)
4. Click **Save** in the toolbar and enter the name **Delayed flights saved search**.
5. Click **New** to reset the table view to its original layout.
6. To return to your saved search, click **Open** and select **Delayed flights saved search**.

Exporting to CSV

Once you have used Discover to search for interesting data, you may want to export the results for further analysis.

1. Click **Open** and select **Delayed flights saved search** to return to your saved search, if it is not already open.
2. Enter **Paris** in the query bar and press Enter.
3. Save again as a new saved search called **Delayed Paris flights**.
4. Click **Share** in the toolbar. Next, select **CSV Reports** and click **Generate CSV**.
5. The CSV is being generated in the background. Once ready, a popup will appear in the bottom right corner of the screen:



6. You can retrieve generated CSVs or follow the generation progress. Select **Stack Management** from the **Management** section in the main menu. Next, select **Reporting** under **Alerts and Insights**. Wait until the report has been generated and click the download icon.
7. Wait until the CSV has been generated (if not ready) and click the download icon.

Delayed Paris flights

timestamp	OriginCityName	DestCityName	Carrier	FlightDelayMin
May 25, 2021 @ 23:11:51.000	Paris	Quito	JetBeats	300
May 25, 2021 @ 18:12:10.000	Paris	Naples	JetBeats	90
May 24, 2021 @ 22:05:32.000	Paris	Seoul	JetBeats	315
May 23, 2021 @ 20:24:36.000	Adelaide	Paris	Logstash Airways	90
May 22, 2021 @ 05:32:12.000	Jebel Ali	Paris	Logstash Airways	195
May 21, 2021 @ 17:05:51.000	Paris	Sydney	JetBeats	330
May 21, 2021 @ 05:12:24.000	Paris	Nashville	Kibana Airlines	195
May 20, 2021 @ 17:07:05.000	Brisbane	Paris	JetBeats	15
May 18, 2021 @ 22:47:56.000	Paris	Rome	JetBeats	315
May 17, 2021 @ 06:29:37.000	Abu Dhabi	Paris	ES-Air	135

Kibana Fundamentals Lab Guide

Version: 7.13