



# **Amazon Web Services Data Engineering Immersion Day**

---

Exploring Data Lake with Amazon Athena  
and Amazon QuickSight

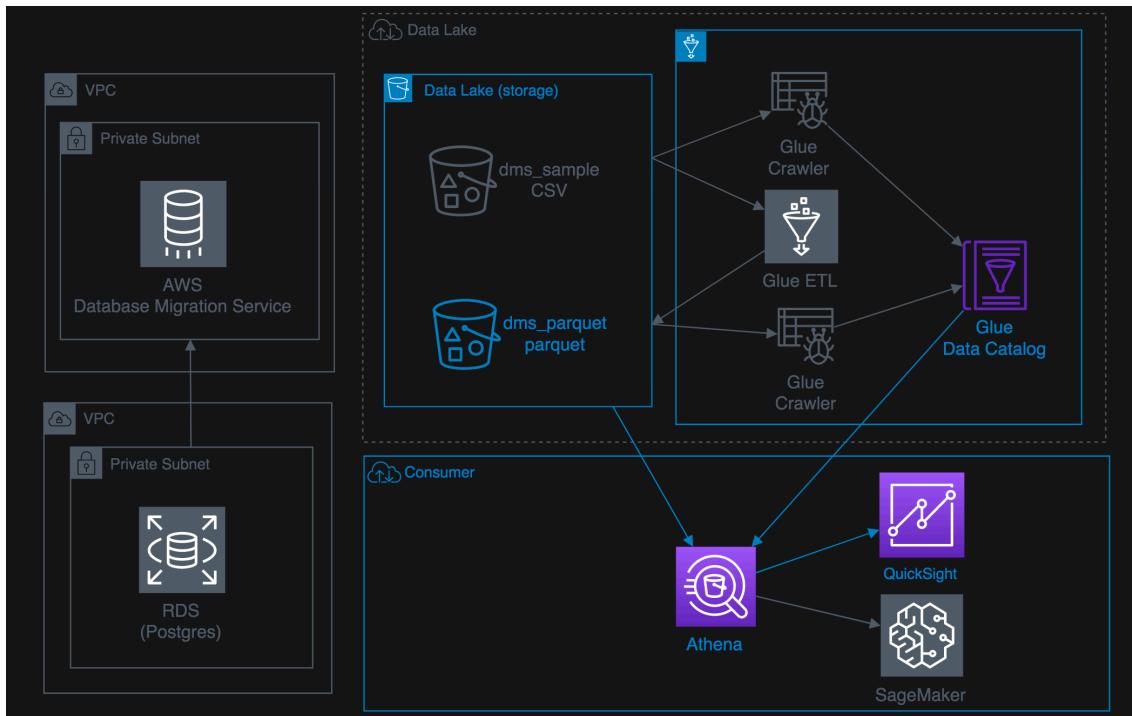
*May 2020*

## Table of Contents

<i>Introduction</i> .....	2
Getting Started.....	2
<i>Query Data with Amazon Athena</i> .....	3
<i>Build an Amazon QuickSight Dashboard</i> .....	12
Set up QuickSight.....	12
Create QuickSight Charts.....	18
Create QuickSight Parameters.....	21
Create a QuickSight Filter.....	24
Add Calculated Fields.....	26
Amazon QuickSight ML-Insights .....	29
Create a Geospatial visual.....	33
Drill down on a Pie Chart .....	33
On the fly custom metrics.....	35
Interactive filtering .....	38
Creating and sharing a dashboard.....	40

# Introduction

This lab introduces you to AWS Glue, Amazon Athena, and Amazon QuickSight. AWS Glue is a fully managed data catalog and ETL service; Amazon Athena queries data; and Amazon QuickSight provides visualization of the data you import.



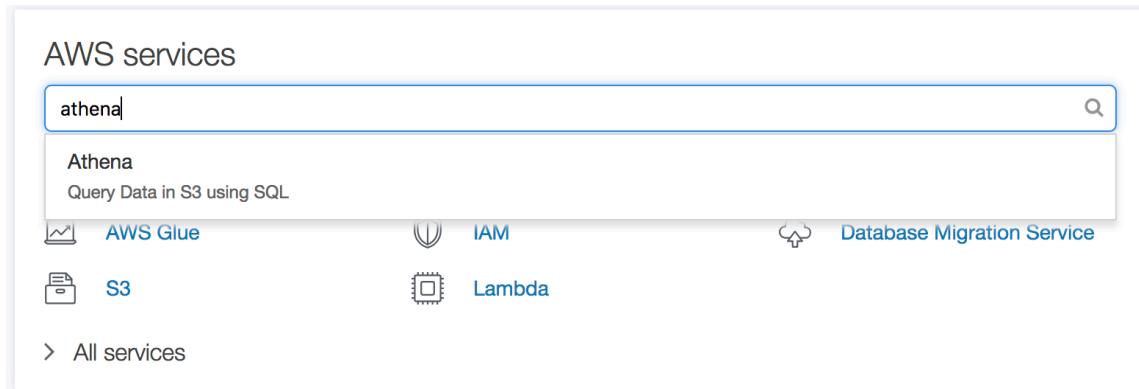
## Getting Started

In this lab, you will complete the following tasks:

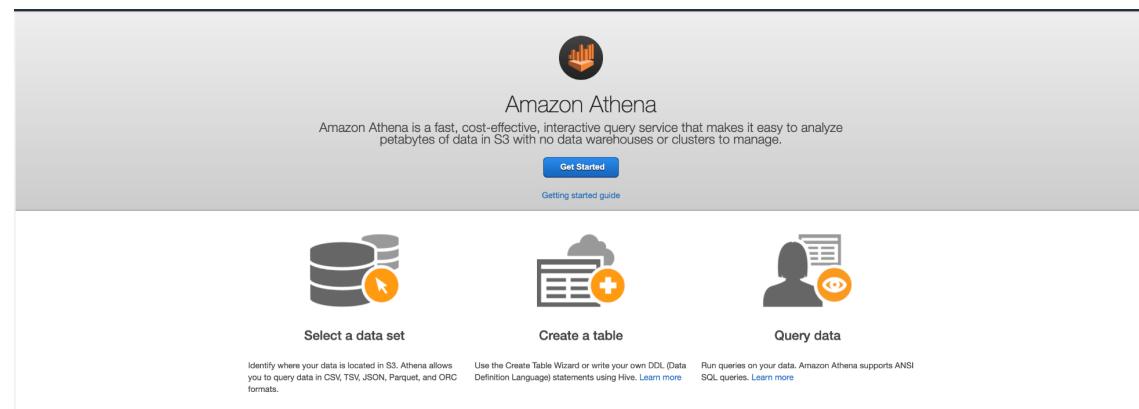
1. [Query data and create a view with Amazon Athena](#)
2. [Build a dashboard with Amazon QuickSight](#)

# Query Data with Amazon Athena

1. In the AWS services console, search for **Athena**.



2. If you are using Athena first time, click on “Get Started” button in introduction screen.



3. Athena stores query results in an Amazon S3 bucket and we should configure this before we can explore our dataset. There is already a bucket created for this purpose whose name follows the pattern: quicksight-lab-data-lake-XXX-YYYY.

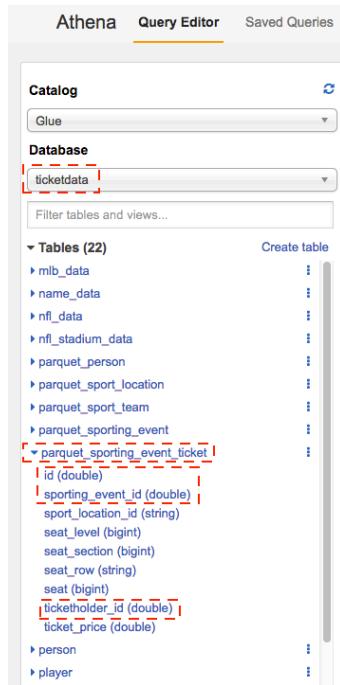
Buckets (2)		
Buckets are the fundamental container in Amazon S3 for data storage. For others to access the objects in your buckets, you'll need to explicitly grant them permissions. <a href="#">Learn more</a>		
<input type="text"/> <a href="#">Find bucket by name</a>		
Name	Region	Access
<input type="radio"/> <a href="#">quicksight-lab-athena-results-eu-west-1-141290848749</a>	EU (Ireland) eu-west-1	Objects can be public
<input type="radio"/> <a href="#">quicksight-lab-data-lake-eu-west-1-141290848749</a>	EU (Ireland) eu-west-1	Objects can be public

4. To perform such configuration, in Athena Console, click on **Settings**.

- Enter bucket name in the query result location field in the following format: s3:// <bucket-name>/ (slashes are important) and click on Save.

- From the Query Editor tab, select your "ticketdata" database if not selected already.
- Click the table named "parquet\_sporting\_event\_ticket" to inspect the fields.

**Note:** The type for fields id, sporting\_event\_id and ticketholder\_id should be (double).



Next, we will query across tables `parquet_sporting_event`, `parquet_sport_team`, and `parquet_sport_location`.

## 8. Copy the following SQL syntax into the New Query 1 tab and click **Run Query**.

```

SELECT
    e.id AS event_id,
    e.sport_type_name AS sport,
    e.start_date_time AS event_date_time,
    h.name AS home_team,
    a.name AS away_team,
    l.name AS location,
    l.city
FROM parquet_sporting_event e,
     parquet_sport_team h,
     parquet_sport_team a,
     parquet_sport_location l
WHERE
    e.home_team_id = h.id
    AND e.away_team_id = a.id
    AND e.location_id = l.id;

```

**Note:** If Run query button is not enabled, select the entire query and press Control + Enter to have it executed. This may happen in some browsers due to bug in this part of the web console. You can also re-load the page.

The results appear beneath the query window.

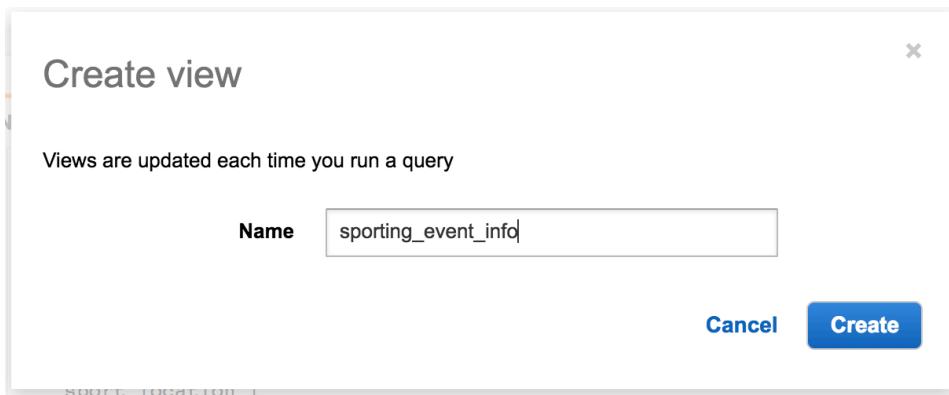
**Results**

	event_id	sport	event_date_time
1	9101	football	2020-04-14 16:00:00.000
2	9111	football	2020-02-21 16:00:00.000
3	9091	football	2020-04-17 11:00:00.000
4	9421	baseball	2020-04-17 09:00:00.000
5	9431	baseball	2020-04-18 09:00:00.000
6	9451	baseball	2020-05-10 09:00:00.000
7	6461	baseball	2020-05-17 09:00:00.000
8	6741	football	2020-09-14 19:00:00.000
9	4741	football	2020-09-14 19:00:00.000

**Details**

	Team	City	Stadium	Capacity
1	Kansas City Chiefs	Kansas City, Missouri	Arrowhead Stadium	71,200
2	Kansas City Chiefs	Kansas City, Missouri	Arrowhead Stadium	71,200
3	Kansas City Chiefs	Kansas City, Missouri	Arrowhead Stadium	71,200
4	Oakland Raiders	Oakland, California	Raiders Coliseum	72,000
5	Toronto Blue Jays	Toronto, Ontario	Hopkirk Field	25,000
6	Toronto Blue Jays	Toronto, Ontario	Maple Leaf Stadium	50,000
7	Toronto Blue Jays	Toronto, Ontario	Rogers Centre	45,000
8	Los Angeles Dodgers	Los Angeles, California	Dodger Stadium	41,000
9	Kansas City Royals	Kansas City, Missouri	Arrowhead Stadium	71,200
10	New York Jets	New York, New York	MetLife Stadium	82,500
11	Miami Dolphins	Miami Gardens, Florida	Hard Rock Stadium	65,000
12	Atlanta Falcons	Atlanta, Georgia	Merchandise Bank	71,200

9. As shown above Click **Create** and then select **Create view from query**
10. Name the view "sporting\_event\_info" and click **Create**.



Your new view is created:

```

CREATE OR REPLACE VIEW `sporting_event_info` AS
SELECT
  e.id AS event_id,
  e.sport AS sport,
  e.start_date_time AS event_date_time,
  h.name AS home_team,
  a.name AS away_team,
  l.name AS location,
  l.city
FROM ticketdata sporting_event e,
  parquet_sport_team h,
  parquet_sport_team a,
  parquet_sport_location l
WHERE
  e.home_team_id = h.id
  AND e.away_team_id = a.id
  AND e.location_id = l.id
  
```

Run query Save as Create (Run time: 0.71 seconds, Data scanned: 0 KB)

11. Copy the following SQL syntax into the New Query 2 tab and click on **Save Query**.

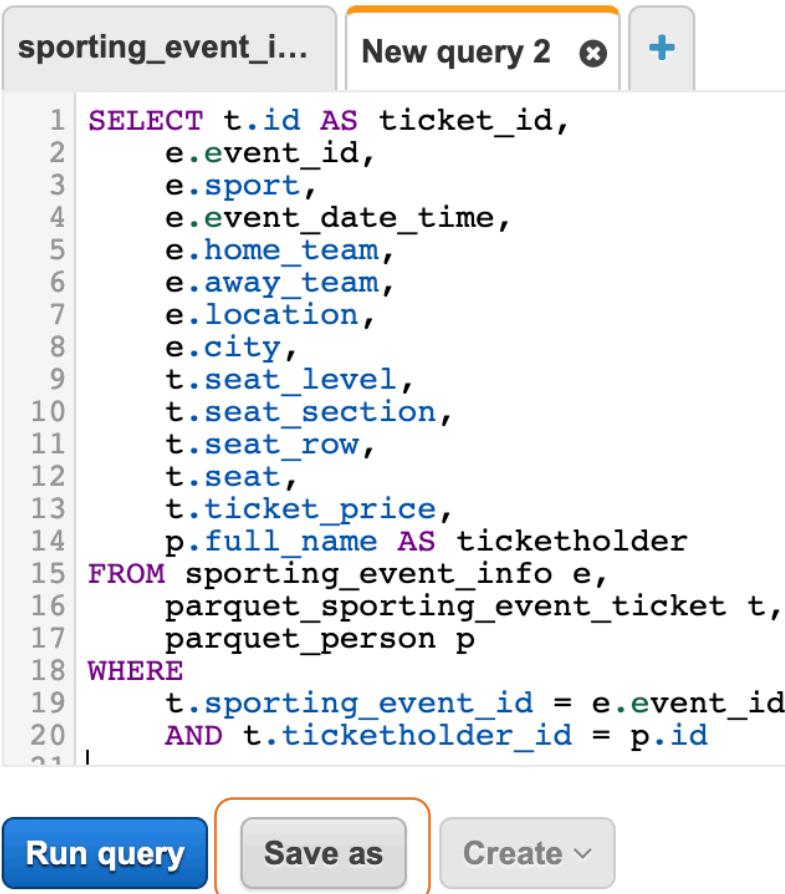
```

SELECT t.id AS ticket_id,
e.event_id,
e.sport,
e.event_date_time,
e.home_team,
e.away_team,
e.location,
  
```

```

e.city,
t.seat_level,
t.seat_section,
t.seat_row,
t.seat,
t.ticket_price,
p.full_name AS ticketholder
FROM sporting_event_info e,
parquet_sporting_event_ticket t,
parquet_person p
WHERE
t.sporting_event_id = e.event_id
AND t.ticketholder_id = p.id

```



```

1 SELECT t.id AS ticket_id,
2     e.event_id,
3     e.sport,
4     e.event_date_time,
5     e.home_team,
6     e.away_team,
7     e.location,
8     e.city,
9     t.seat_level,
10    t.seat_section,
11    t.seat_row,
12    t.seat,
13    t.ticket_price,
14    p.full_name AS ticketholder
15 FROM sporting_event_info e,
16     parquet_sporting_event_ticket t,
17     parquet_person p
18 WHERE
19     t.sporting_event_id = e.event_id
20     AND t.ticketholder_id = p.id

```

Click on **Save as** button and give this query the name **create\_view\_sporting\_event\_ticket\_info**, some description and then, click on **Save**.

## Choose a name

Name

Use 1 - 128 characters

Description

Use upto 1024 characters

Cancel

Save

Back to the query editor, you will see the query name changed. Now, click on Run Query.

sporting\_event\_i...  +

```
1 SELECT t.id AS ticket_id,
2     e.event_id,
3     e.sport,
4     e.event_date_time,
5     e.home_team,
6     e.away_team,
7     e.location,
8     e.city,
9     t.seat_level,
10    t.seat_section,
11    t.seat_row,
12    t.seat,
13    t.ticket_price,
14    p.full_name AS ticketholder
15 FROM sporting_event_info e,
16     parquet_sporting_event_ticket t,
17     parquet_person p
18 WHERE
19     t.sporting_event_id = e.event_id
20     AND t.ticketholder_id = p.id
```

**Run query** **Save as** **Create ▾**

The results appear beneath the query window.

Athena Query Editor

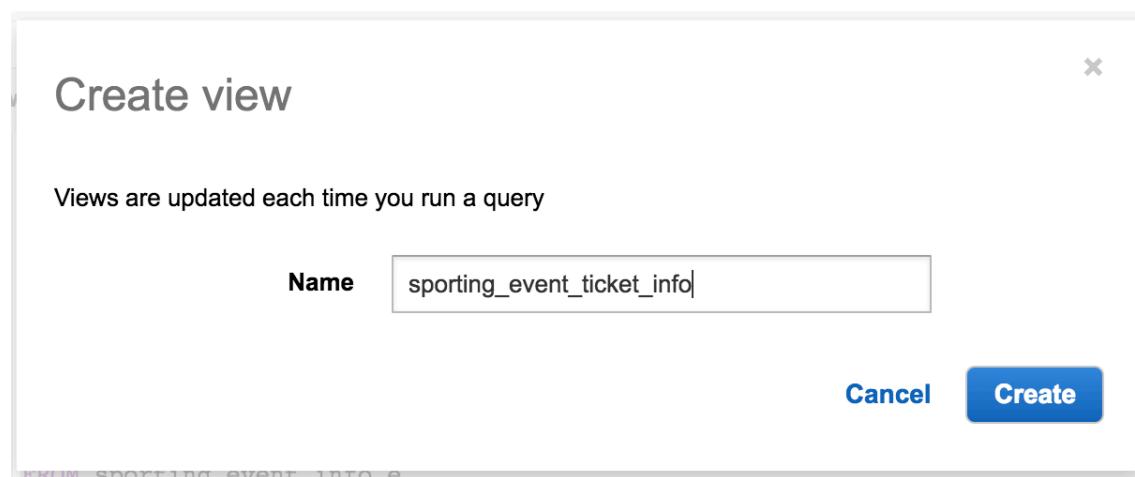
New query 1 New query 2 New query 3 New query 4

Results

#	ticket_id	event_id	sport	event_date_time	venue_name	seat_id	city	seat_level	seat_section	seat_row	ticket_price	holder	
1	760891.0	10821	football	2020-11-14 16:00:00.000	Arizona Cardinals	Kansas City Chiefs	University of Phoenix Stadium	3	17	B	2	40.00	House Guest
2	760891.0	10821	football	2020-11-14 16:00:00.000	Arizona Cardinals	Kansas City Chiefs	University of Phoenix Stadium	3	17	B	3	40.00	House Guest
3	760891.0	10821	football	2020-11-14 16:00:00.000	Arizona Cardinals	Kansas City Chiefs	University of Phoenix Stadium	3	17	B	4	40.00	House Guest
4	760891.0	10821	football	2020-11-14 16:00:00.000	Arizona Cardinals	Kansas City Chiefs	University of Phoenix Stadium	3	17	A	4	40.00	House Guest
5	760891.0	10821	football	2020-11-14 16:00:00.000	Arizona Cardinals	Kansas City Chiefs	University of Phoenix Stadium	3	17	A	1	40.00	House Guest
6	760891.0	10821	football	2020-11-14 16:00:00.000	Arizona Cardinals	Kansas City Chiefs	University of Phoenix Stadium	3	17	A	2	40.00	House Guest
7	760891.0	10821	football	2020-11-14 16:00:00.000	Arizona Cardinals	Kansas City Chiefs	University of Phoenix Stadium	3	17	A	3	40.00	House Guest
8	760891.0	10821	football	2020-11-14 16:00:00.000	Arizona Cardinals	Kansas City Chiefs	University of Phoenix Stadium	3	16	B	2	40.00	House Guest
9	760891.0	10821	football	2020-11-14 16:00:00.000	Arizona Cardinals	Kansas City Chiefs	University of Phoenix Stadium	3	16	B	3	40.00	House Guest
10	760891.0	10821	football	2020-11-14 16:00:00.000	Arizona Cardinals	Kansas City Chiefs	University of Phoenix Stadium	3	16	B	4	40.00	House Guest
11	760891.0	10821	football	2020-11-14 16:00:00.000	Arizona Cardinals	Kansas City Chiefs	University of Phoenix Stadium	3	16	B	5	40.00	House Guest
12	760891.0	10821	football	2020-11-14 16:00:00.000	Arizona Cardinals	Kansas City Chiefs	University of Phoenix Stadium	3	16	B	6	40.00	House Guest

12. As shown above Click **Create view from query**.

13. Name the view "**sporting\_event\_ticket\_info**" and click **Create**.



14. Copy the following SQL syntax into the New Query 3 tab.

```
SELECT
    sport,
    count(distinct location) as locations,
    count(distinct event_id) as events,
    count(*) as tickets,
    avg(ticket_price) as avg_ticket_price
FROM sporting_event_ticket_info
GROUP BY 1
ORDER BY 1;
```

Click on **Save as** and give the query name **analytics\_sporting\_event\_ticket\_info** and some description and then, click on **Save**.

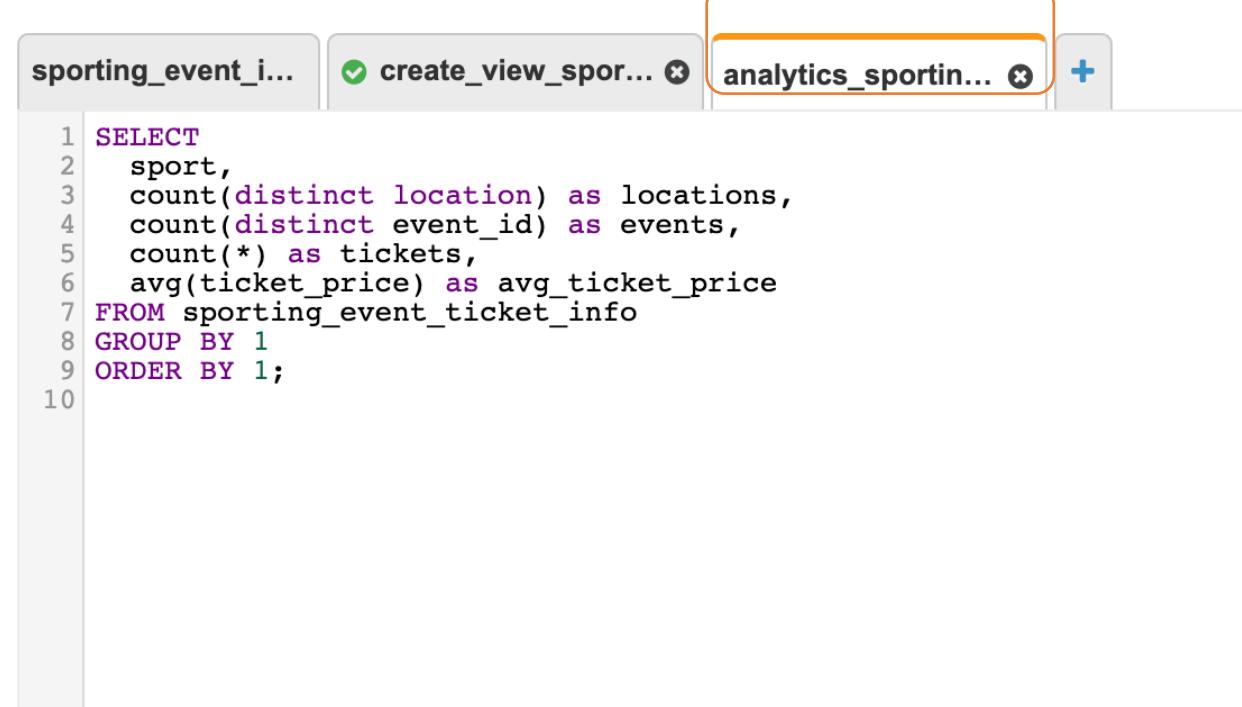
Choose a name

**Name** analytics\_sporting\_event\_ticket\_info  
Use 1 - 128 characters

**Description** to analyze the view: sporting\_event\_ticket\_info  
Use upto 1024 characters

**Cancel** **Save**

The name of the New Query 3 will be changed to one assigned in previous step. Click on **Run Query**.



```

sporting_event_i...
create_view_spor... ✎
analytics_sportin... ✎ +
```

```

1 SELECT
2   sport,
3   count(distinct location) as locations,
4   count(distinct event_id) as events,
5   count(*) as tickets,
6   avg(ticket_price) as avg_ticket_price
7   FROM sporting_event_ticket_info
8   GROUP BY 1
9   ORDER BY 1;
10

```

**Run query** **Save as** **Create**

Your query returns two results in approximately five seconds.

The screenshot shows the AWS Athena Query Editor interface. At the top, there are tabs for 'Athena', 'Query Editor' (which is selected), 'Saved Queries', 'History', and 'AWS Glue Data Catalog'. Below the tabs, there's a toolbar with buttons for 'Run query', 'Save as', and 'Create'. A dropdown menu indicates the 'Workgroup : primary'. The main area contains a code editor with the following SQL query:

```

1 SELECT
2   sport,
3   count(distinct location) as locations,
4   count(distinct event_id) as events,
5   count(*) as tickets,
6   avg(ticket_price) as avg_ticket_price
7   FROM sporting_event_ticket_info
8   GROUP BY 1
9   ORDER BY 1;
10

```

Below the code editor, status information is displayed: '(Run time: 6.21 seconds, Data scanned: 25.97 MB)'. A note at the bottom says 'Use Ctrl + Enter to run query, Ctrl + Space to autocomplete'. The results section shows a table with the following data:

	sport	locations	events	tickets	avg_ticket_price
1	baseball	30	294	958680	53.89345581425812
2	football	25	113	810304	57.40977502271104

The purpose of saving the queries is to have clear distinction between the results of the queries running on one view. Otherwise, your query results will be saved under "Unsaved" folder within the S3 bucket location provided to Athena to store query results. Please navigate to S3 bucket to observe these changes, as shown below:

The screenshot shows the Amazon S3 console. The path 'Amazon S3 > quicksight-lab-athena-results-eu-west-1-141290848749' is visible. The main page displays the following details:

- Overview**: Shows the total number of objects: 3.
- Properties**: Shows the bucket name: quicksight-lab-athena-results-eu-west-1-141290848749.
- Permissions**: Shows the bucket policy ARN: arn:aws:s3:::quicksight-lab-athena-results-eu-west-1-141290848749.
- Management**: Shows the bucket creation date: 2023-08-14T10:00:00Z.
- Access points**: Shows the endpoint URL: https://quicksight-lab-athena-results-eu-west-1-141290848749.s3.amazonaws.com/.

A search bar at the top right contains the placeholder 'Type a prefix and press Enter to search. Press ESC to clear.' Below the search bar, there are actions: 'Upload', 'Create folder', 'Download', and 'Actions' with a dropdown arrow.

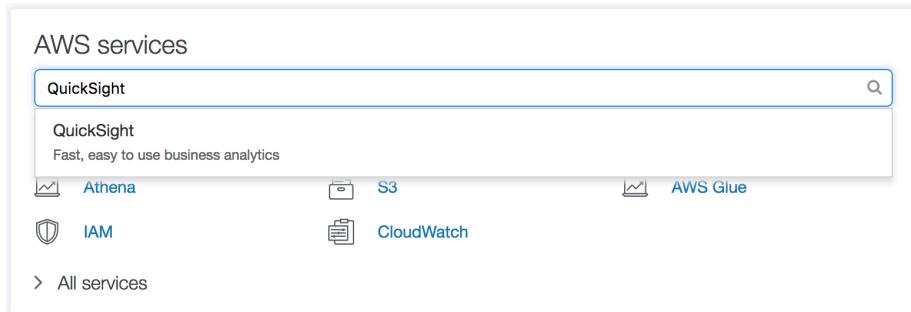
The main content area lists the objects in the bucket:

	Name	Last modified
<input type="checkbox"/>	Unnamed	--
<input type="checkbox"/>	Untitled	--
<input type="checkbox"/>	analytics_sporting_event_ticket_info	--
<input type="checkbox"/>	create_view_sporting_event_ticket_info	--

# Build an Amazon QuickSight Dashboard

## Set up QuickSight

1. In the AWS services console, search for **QuickSight**.



If this is the first time you have used QuickSight, you are prompted to create an account.

2. Click **Sign up for QuickSight**.



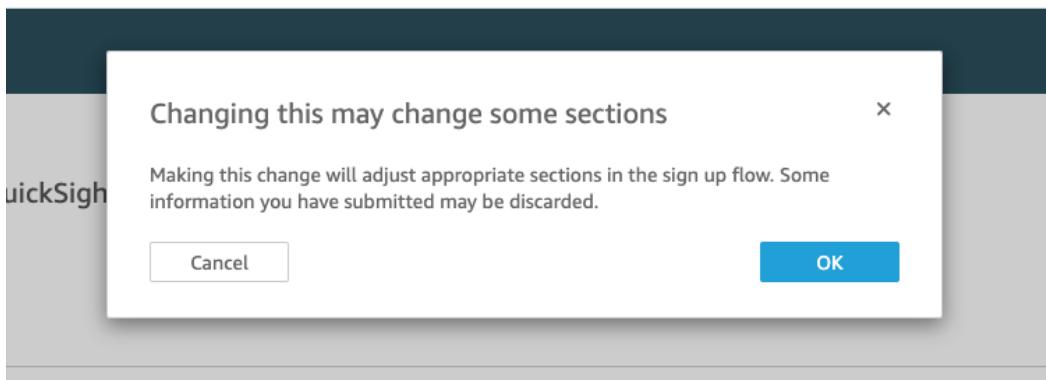
3. For account type, please choose **Enterprise Version**

Create your QuickSight account

Edition	<input type="radio"/> Standard	<input checked="" type="radio"/> Enterprise
First author with 1GB SPICE	FREE	FREE
Team trial for 60 days (4 authors)*	FREE	FREE
Additional author per month (yearly)**	\$9	\$18
Additional author per month (monthly)**	\$12	\$24
Additional readers (Pay-per-Session)	N/A	\$0.30/session (max \$5/reader/month) ****
Additional SPICE per month	\$0.25 per GB	\$0.38 per GB
Single Sign On with SAML or OpenID Connect	✓	✓
Connect to spreadsheets, databases & business apps	✓	✓
Access data in Private VPCs		✓
Row-level security for dashboards		✓
Hourly refresh of SPICE data		✓
Secure data encryption at rest		✓
Connect to your Active Directory		✓
Use Active Directory Groups ***		✓
Send email reports		✓

\* Trial authors are auto-converted to month-to-month subscription upon trial expiry

4. Click **Continue**.
5. On the *Create your QuickSight account* page, select EU (Ireland) region – Quicksight is a regional service -, accept warning if prompted and fill out your name and email address.



6. Ensure *Enable autodiscovery of data* and *Amazon Athena* are selected.
7. Click **Choose S3 buckets** and select the two available buckets.

### Create your QuickSight account

Edition Enterprise

Use Role Based Federation (SSO)  
 Use Active Directory

QuickSight region  
 Select a region. ?  
 ▼

QuickSight account name  
 ?  
 You will need this for you and others to sign in.

Notification email address  
 ?  
 For QuickSight to send important notifications.

Enable invitation by email   
 Allow inviting new users by email. This setting cannot be changed after sign-up is complete.

>  Enable autodiscovery of data and users in your Amazon Redshift, Amazon RDS, and AWS IAM services.

Amazon Athena  
 Enables QuickSight access to Amazon Athena databases

Please ensure the right Amazon S3 buckets are also enabled for QuickSight.

Amazon S3 (2 buckets selected)  
 Enables QuickSight to auto-discover your Amazon S3 buckets Choose S3 buckets

Amazon S3 Storage Analytics  
 Enables QuickSight to visualize your S3 Storage Analytics data

AWS IoT Analytics  
 Enables QuickSight to visualize your IoT Analytics data

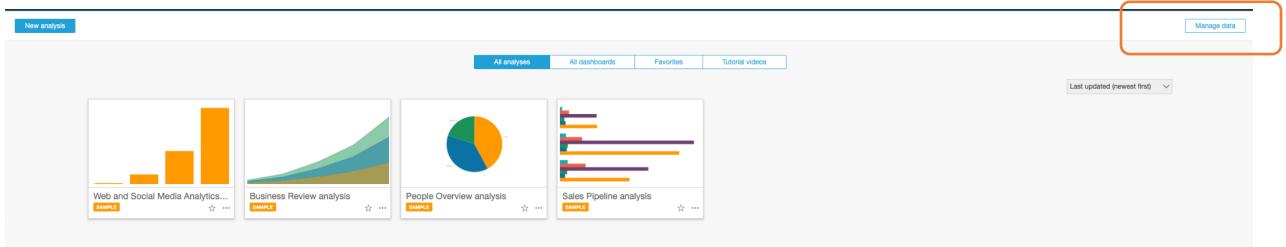
Finish

### Select Amazon S3 buckets

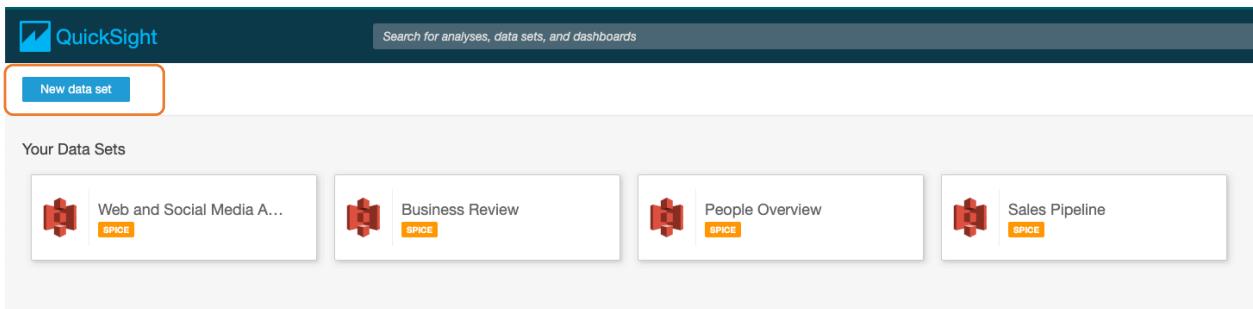
S3 Buckets Linked To QuickSight Account	S3 Buckets You Can Access Across AWS						
Select the buckets that you want QuickSight to be able to access.							
Selected buckets have read only permissions by default. However, you must give write permissions for Athena Workgroup feature.							
<input checked="" type="checkbox"/> Select all							
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="padding: 2px 10px;">S3 Bucket</th> <th style="padding: 2px 10px;">Write permission for Athena Workgroup</th> </tr> </thead> <tbody> <tr> <td style="padding: 2px 10px;"><input checked="" type="checkbox"/> quicksight-lab-athena-results-eu-west-1-131201885738</td> <td style="padding: 2px 10px;"><input type="checkbox"/></td> </tr> <tr> <td style="padding: 2px 10px;"><input checked="" type="checkbox"/> quicksight-lab-data-lake-eu-west-1-131201885738</td> <td style="padding: 2px 10px;"><input type="checkbox"/></td> </tr> </tbody> </table>		S3 Bucket	Write permission for Athena Workgroup	<input checked="" type="checkbox"/> quicksight-lab-athena-results-eu-west-1-131201885738	<input type="checkbox"/>	<input checked="" type="checkbox"/> quicksight-lab-data-lake-eu-west-1-131201885738	<input type="checkbox"/>
S3 Bucket	Write permission for Athena Workgroup						
<input checked="" type="checkbox"/> quicksight-lab-athena-results-eu-west-1-131201885738	<input type="checkbox"/>						
<input checked="" type="checkbox"/> quicksight-lab-data-lake-eu-west-1-131201885738	<input type="checkbox"/>						
<span style="border: 1px solid #0072BC; padding: 2px 10px; border-radius: 5px;">Cancel</span>	<span style="background-color: #0072BC; color: white; padding: 2px 10px; border-radius: 5px;">Finish</span>						

8. Click **Finish**. It will take some time for the new account to be created. Once it is done, go to the Amazon Quicksight dashboard.

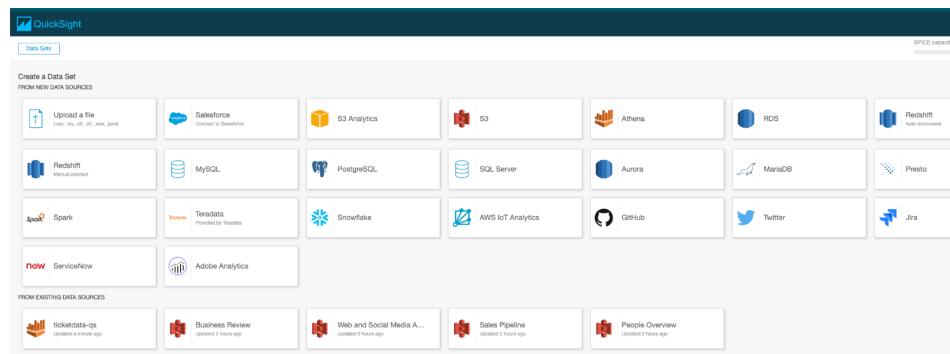
9. We are going to configure now the data set for the Quicksight analysis/dashboard that we will be building based on one of the Athena views created in the previous section. In order to do that, click **Manage Data** on the top right corner.



10. Click **New Data Set**.

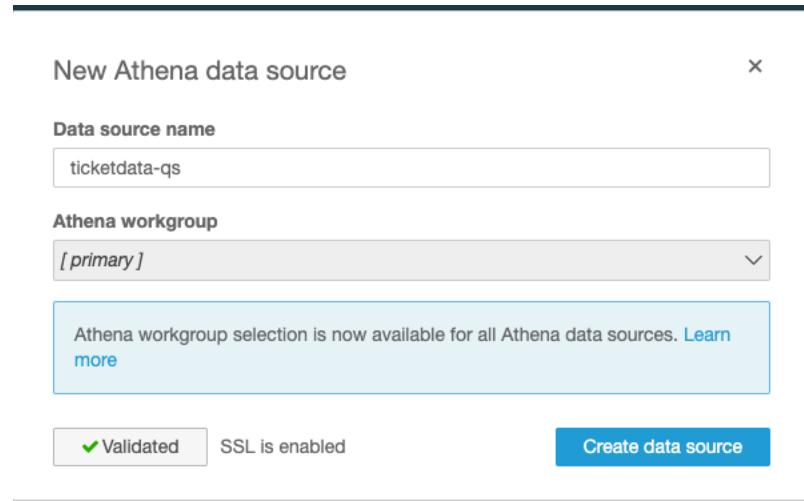


11. On the Create a Data Set page, select **Athena** as the data source.

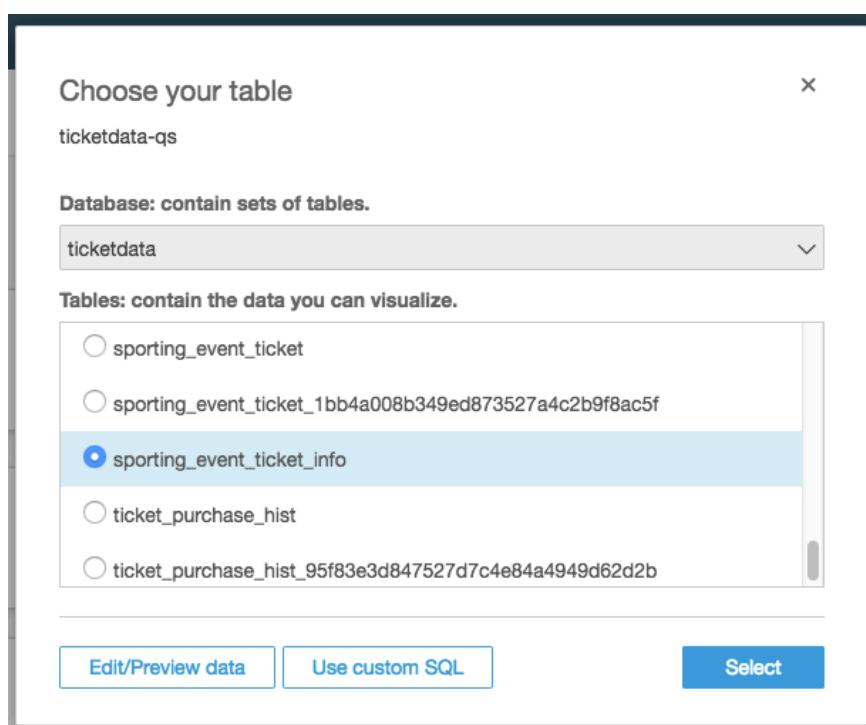


12. For Data source name, type "ticketdata-qs" and click **Validate connection**.

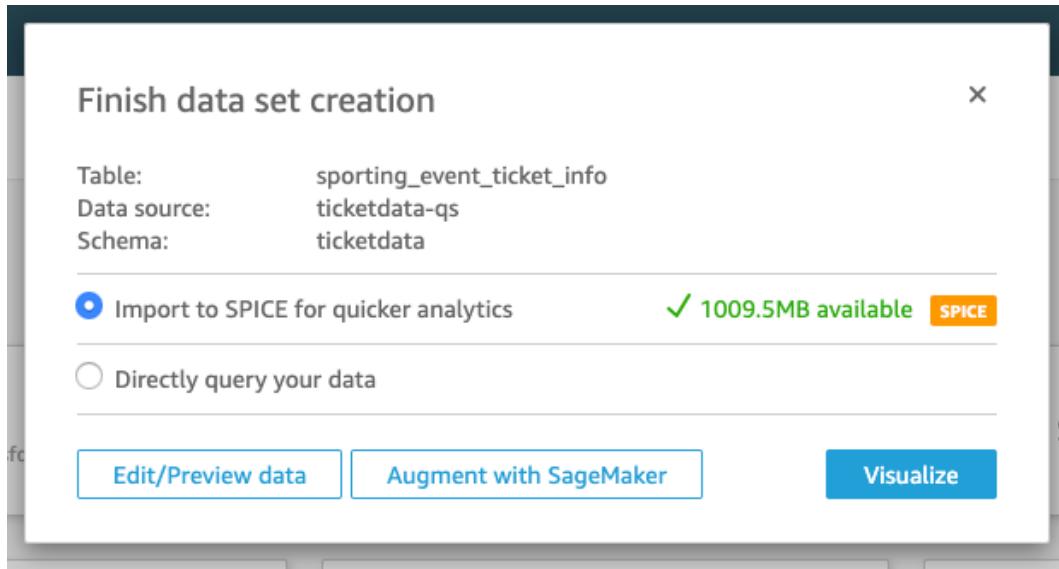
13. Once validated, click **Create data source**.



14. In the database drop-down list, select "ticketdata" database.
15. Choose the "**sporting\_event\_ticket\_info**" table and click **Select**.



16. To finish data set creation, choose the option **Import to SPICE for quicker analytics** and click **Visualize**.



*SPICE is an in-memory-calculation engine that enables achieving fast performance at scale. Data is automatically replicated for high availability allowing thousands of users to simultaneously perform fast, interactive analysis while shielding your underlying data infrastructure, saving you time and resources.*

*This lab account has been created with a default SPICE size of 1GB. If needed, it could be increased Quicksight account settings.*

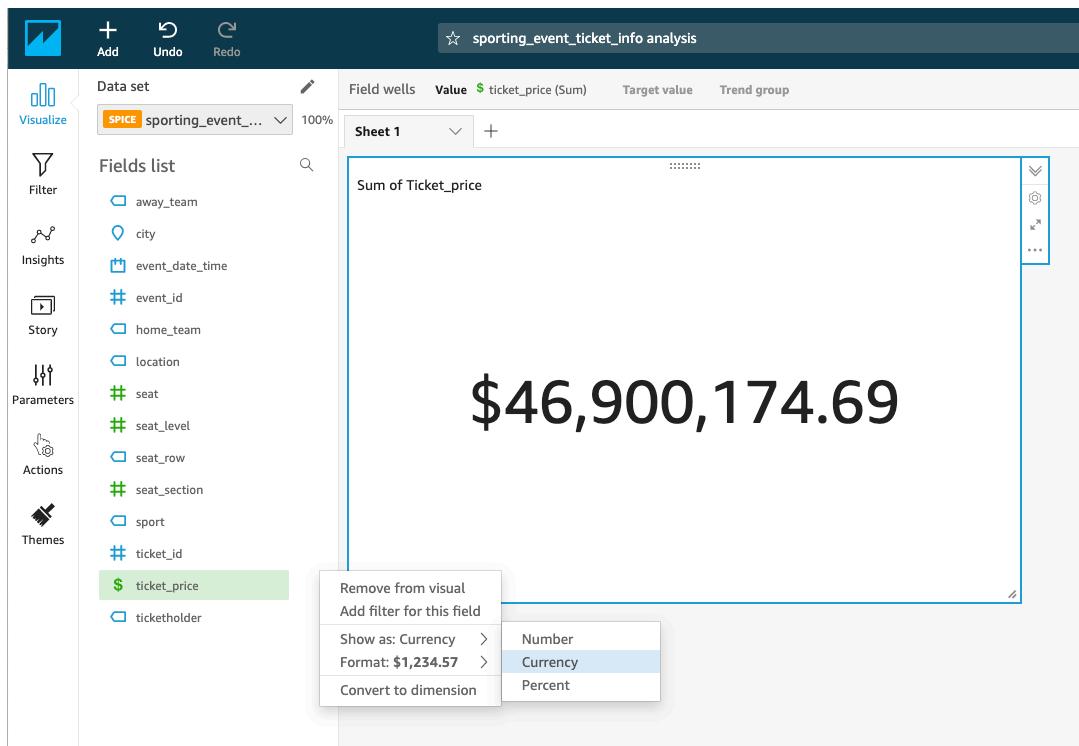
You will now be taken to the QuickSight Visualize interface where you can start building your dashboard.

**Note:** The SPICE dataset will take a few minutes to be built, but you can continue to create some charts on the underlying data.

## Create QuickSight Charts

In this section we will take you through some of the different chart types.

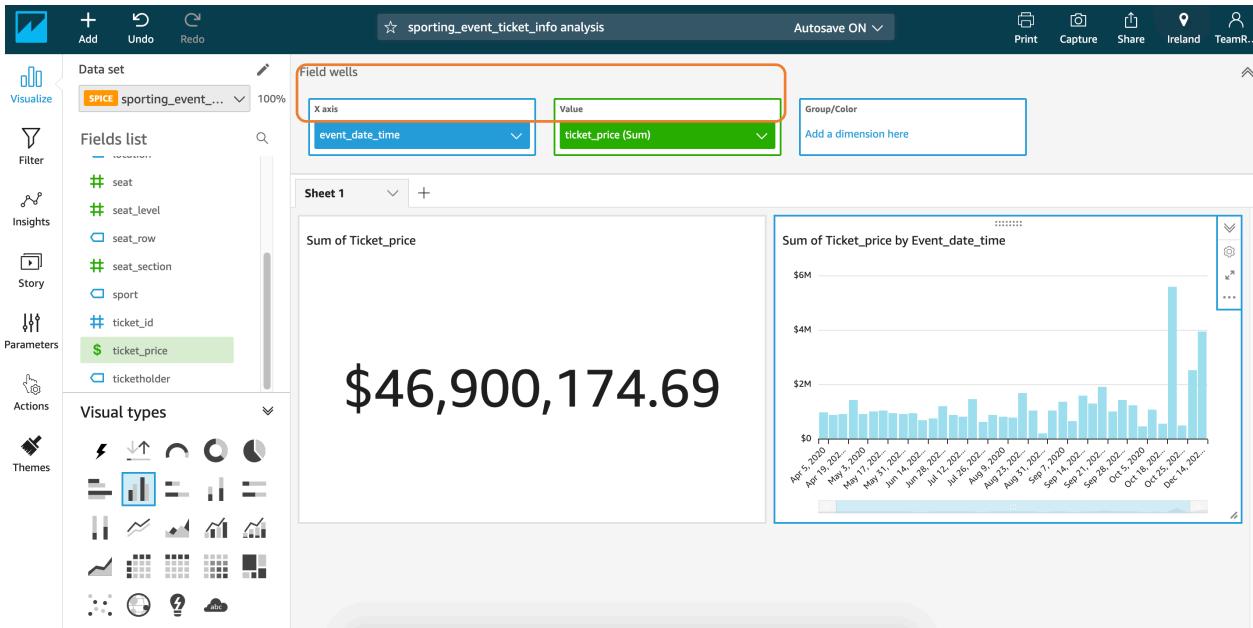
1. In the Fields list, click the "ticket\_price" column to populate the chart.
2. Click the **expand icon** in corner of "ticket\_price" field and select format as **currency** to show numbers in dollar amount.



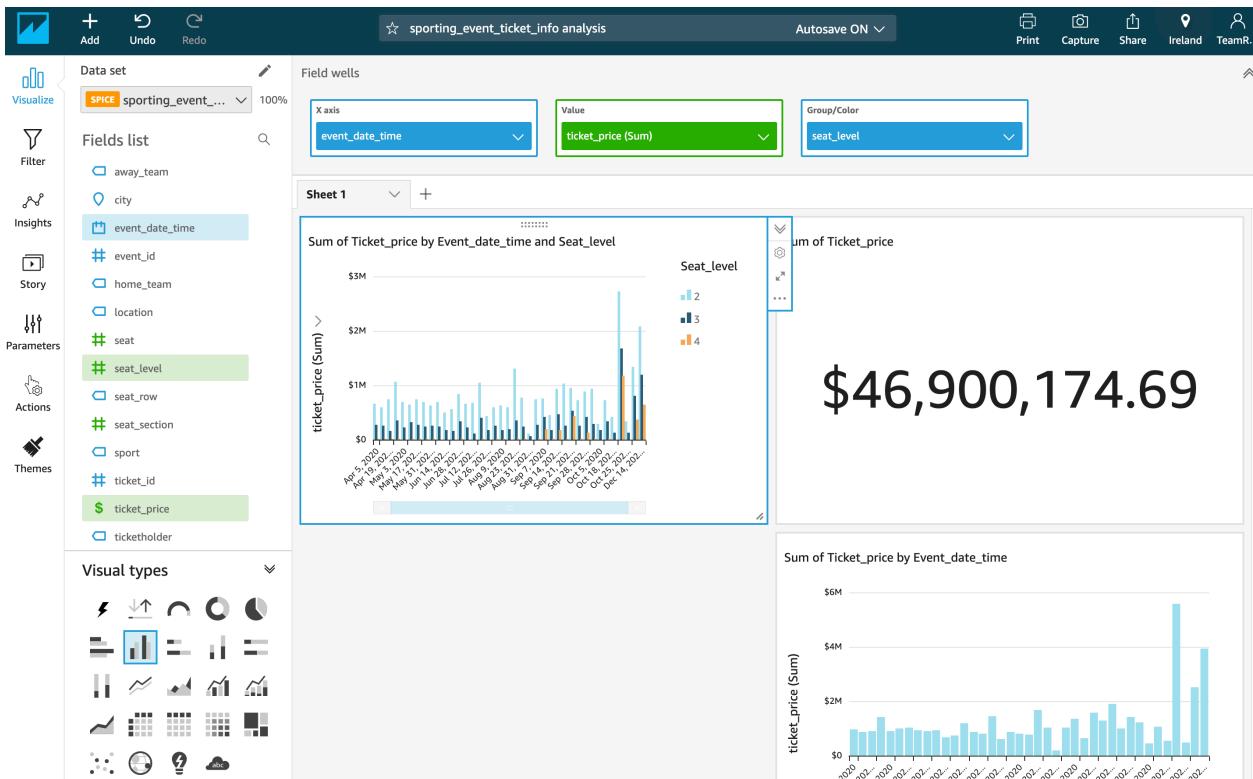
3. You can add new visual and keep building your dashboard by clicking **Add button** at top left corner of screen.

In the **Visual types** area, choose the **Vertical bar chart** icon.

This layout requires a value for the X-axis. In Fields list, select the "**event\_date\_time**" field and you should see the visualization update. For Y-axis, select "**ticket\_price**" from the Field list.

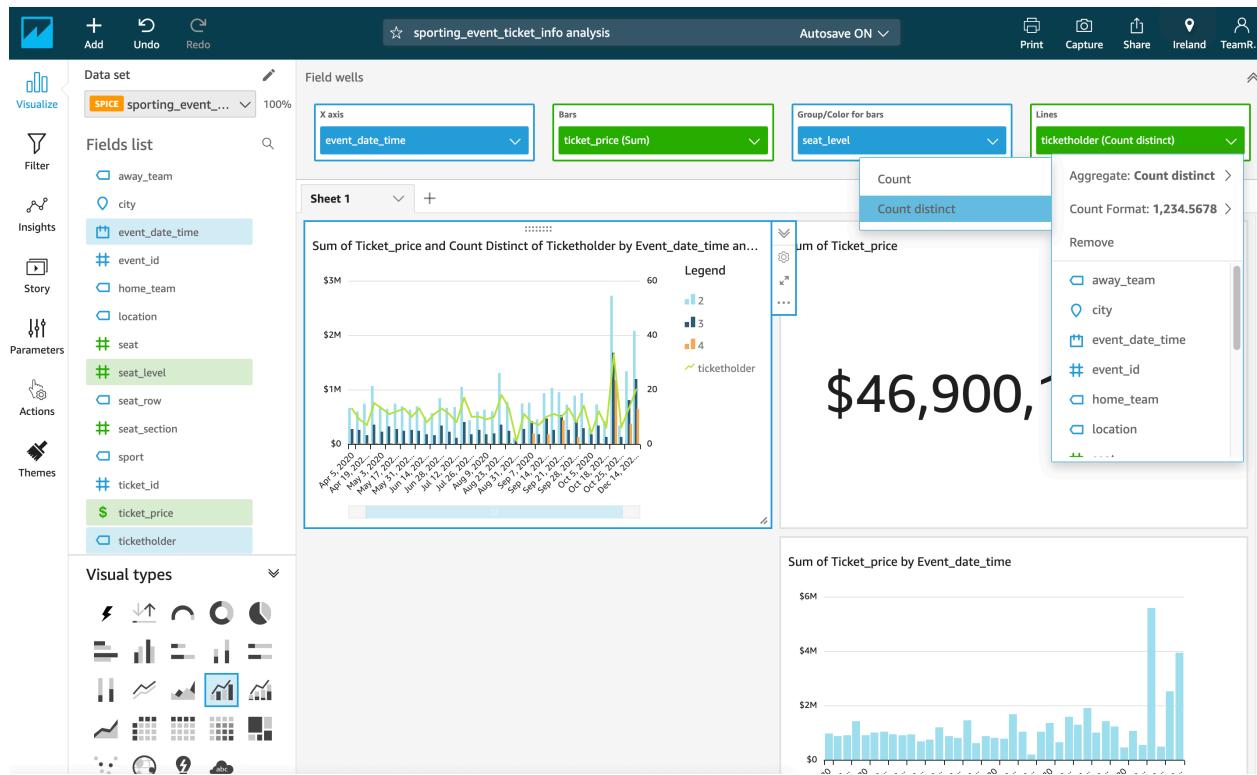


- Add another Vertical bar chart visual with same fields for both axis. You can drag and move other visuals to adjust space in dashboard. In the Fields list, click and drag the **seat\_level** field to the **Group/Color** box in the **Field wells** pane. You can also use the slider below the x axis to fit all of the data.

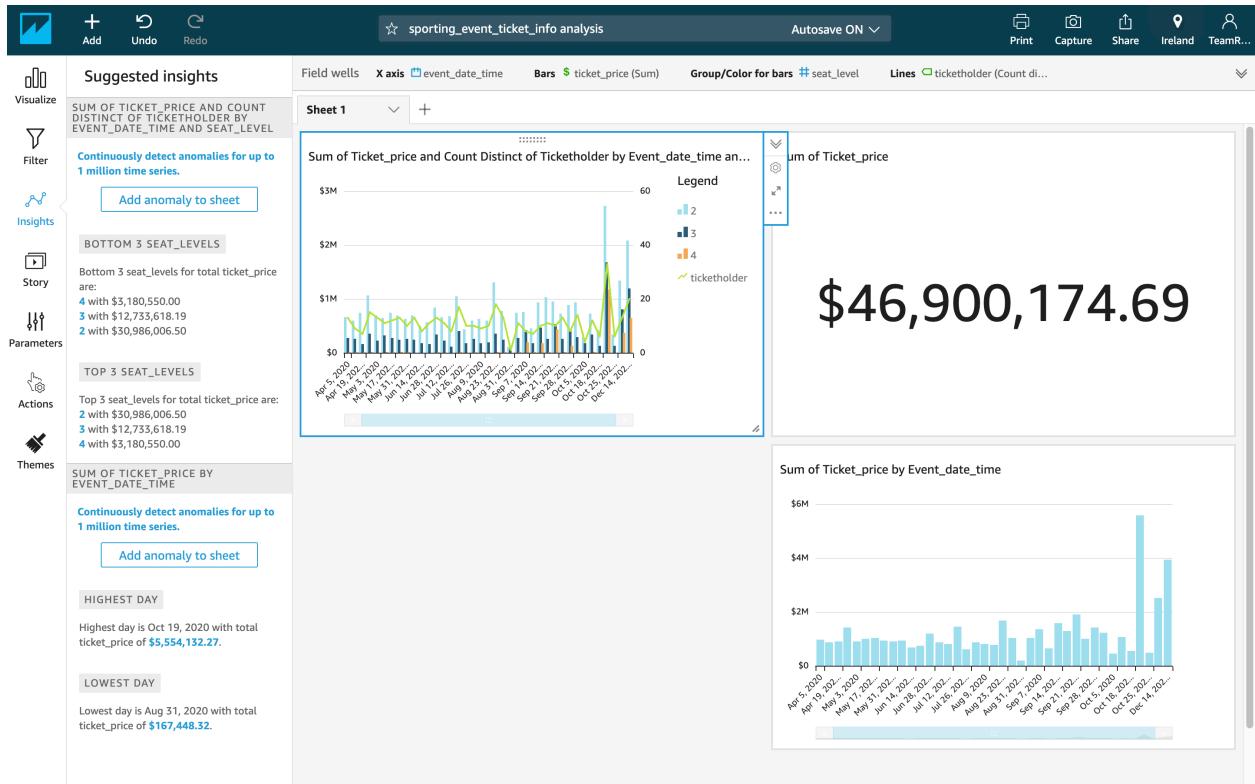


Let's build on this one step further by changing the chart type to "**Clustered bar combo chart**" and adding in the **ticketholder** field for the **Lines**.

5. In the Visual types area, choose the Clustered bar combo chart icon.
6. In the Fields list, click and drag the **ticketholder** field to the Lines box in the Field wells pane.
7. In the Field wells pane, click the Lines box and choose **Count Distinct** for Aggregate. You can then see the y-axis update on the right-hand side.



8. Click on **insight** icon on the left tabs section and explore insight information in simple English.

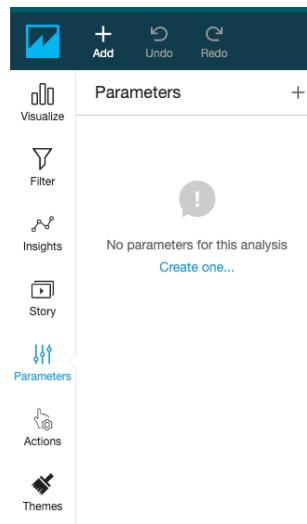


Feel free to experiment with other chart types and different fields to get a sense of the data.

## Create QuickSight Parameters

In the next section we are going to create some parameters with controls for the dashboard, then assign these to a filter for all the visuals.

1. In the left navigation menu, select **Parameters**.



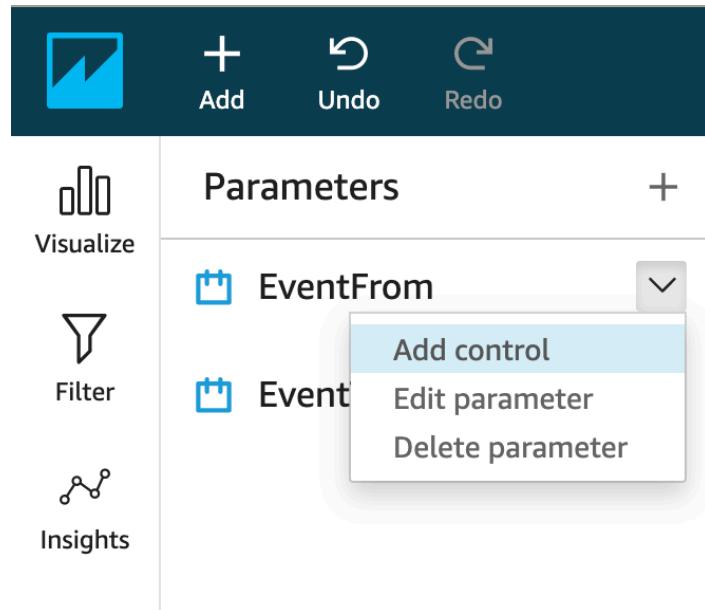
2. Click **Create one** to create a new parameter with a Name.
3. For Name, type **EventFrom**.
4. For Data type, choose **Datetime**.
5. For Default value, select the value from calendar as start date available in your graph for **event\_date\_time**. For example, **2020-01-01 00:00**.
6. Click **Create**, and then close the Parameter Added dialog box (we will be connecting it to a visual filter later).

The screenshot shows the 'Create new parameter' dialog box. At the top, it says 'Create new parameter' and has a close button 'X'. Below that is a descriptive text: 'Use parameters to dynamically control values in your fields, filters, and sheet'. The 'Name' field contains 'EventFrom'. The 'Data type (Not alterable after creation)' dropdown is set to 'Datetime'. The 'Static default value' input field contains '2020-01-01 00:00', with a link 'Set a dynamic default' next to it. At the bottom are 'Cancel' and 'Create' buttons.

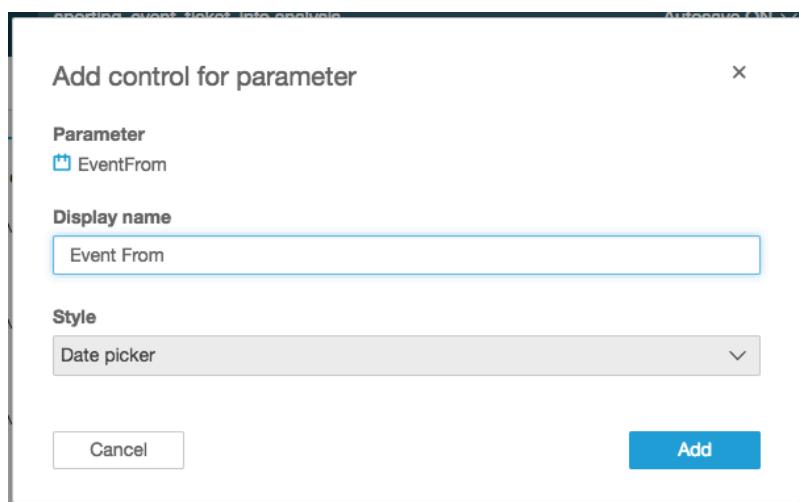
7. Create another parameter with the following attributes:
  - a. **Name:** **EventTo**
  - b. **Data type:** **Datetime**
  - c. For Default value, select the value from calendar as end date available in your graph for **event\_date\_time**. For example, **2021-01-01 00:00**.
  - d. Click **Create**

The screenshot shows the 'Create new parameter' dialog box again, this time for 'EventTo'. The interface is identical to the previous one, with 'Name' set to 'EventTo', 'Data type' set to 'Datetime', and 'Static default value' set to '2021-01-01 00:00'. The 'Cancel' and 'Create' buttons are at the bottom.

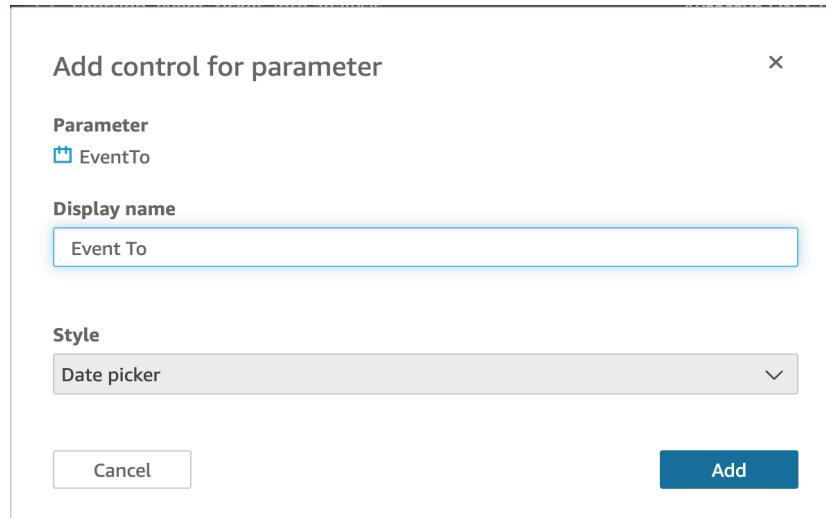
8. In next window, you can select any option to perform any operation with the parameter. Alternatively, you can click the drop-down menu for the **EventFrom** parameter and choose **Add control**.



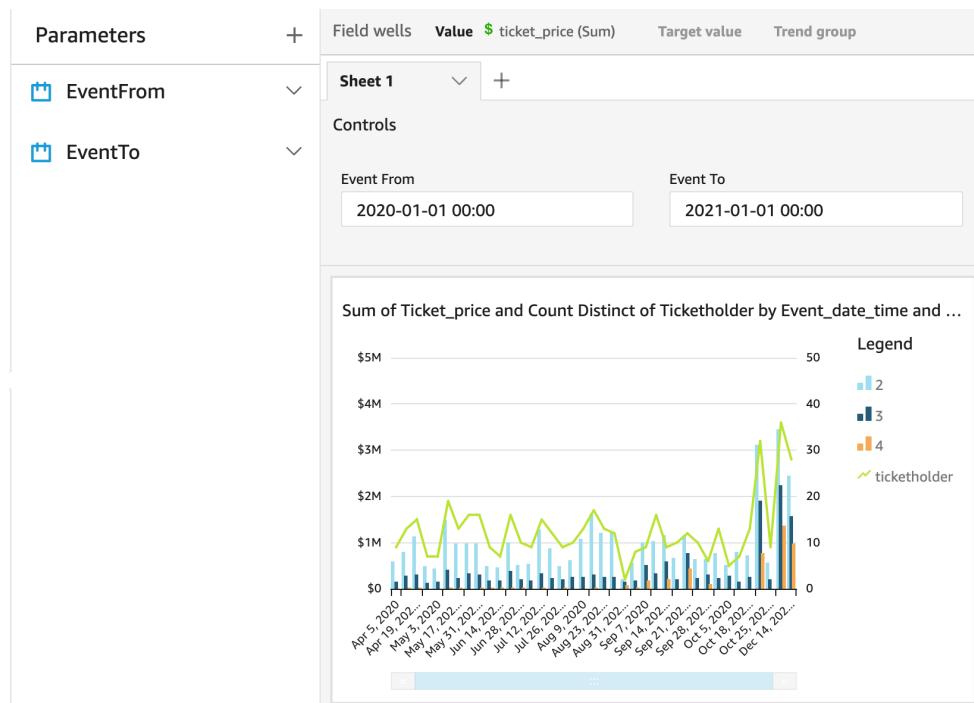
9. For Display name, specify **Event From** and click **Add**.



10. Repeat the process to add a control for **EventTo** with display name **Event To**.



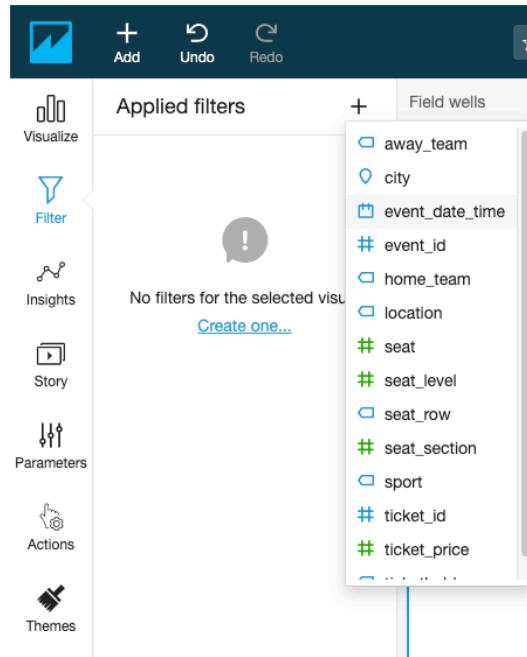
You should now be able to see and expand the Controls section above the chart.



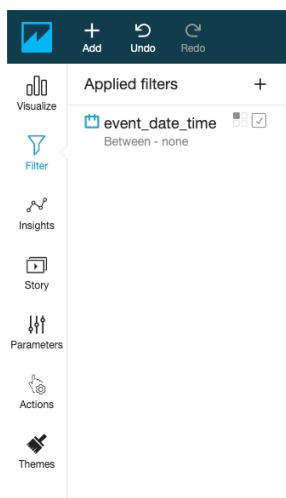
## Create a QuickSight Filter

To complete the process, we will wire up a filter to these controls for all visuals.

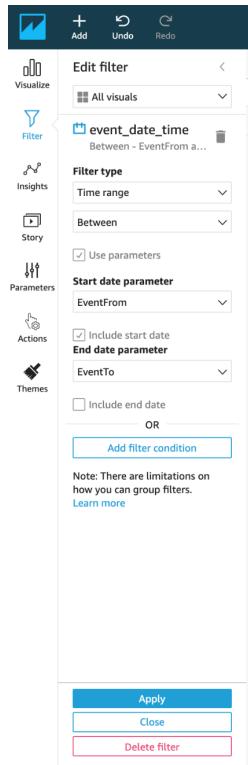
1. In the left navigation menu, choose **Filter**.
2. Click the plus icon (+) to add a filter for the field "**event\_date\_time**".



3. Click this filter to edit the properties.



4. Choose to make this filter apply to **All visuals**.
5. For Filter type, choose **Time range** and **Between**.
6. Select option **Use Parameter**.
7. For Start date parameter, choose **EventFrom**.
8. For End date parameter, choose **EventTo**.
9. Click **Apply**.

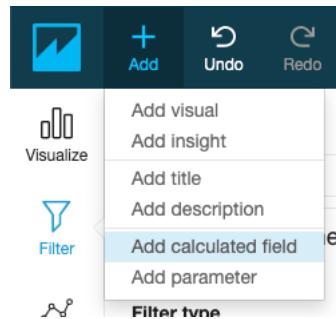


You can choose different date ranges for the Event Data control and see how all existing visuals update accordingly.

## Add Calculated Fields

In the next section, you will learn, how to add calculated fields for "day of week" and "hour of day" to your dataset and a new scatter plot for these two dependent variables.

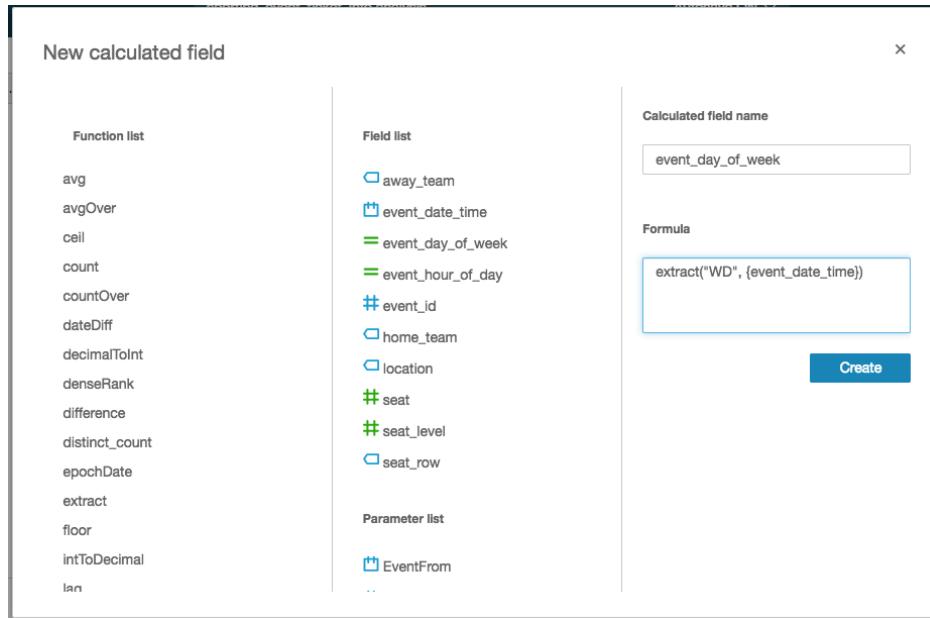
1. Click the Add button on the top left and select **Add a calculated field**.



2. For **Calculated field name** type "`event_day_of_week`".
3. For **Formula**, type `extract("WD", {event_date_time})`.

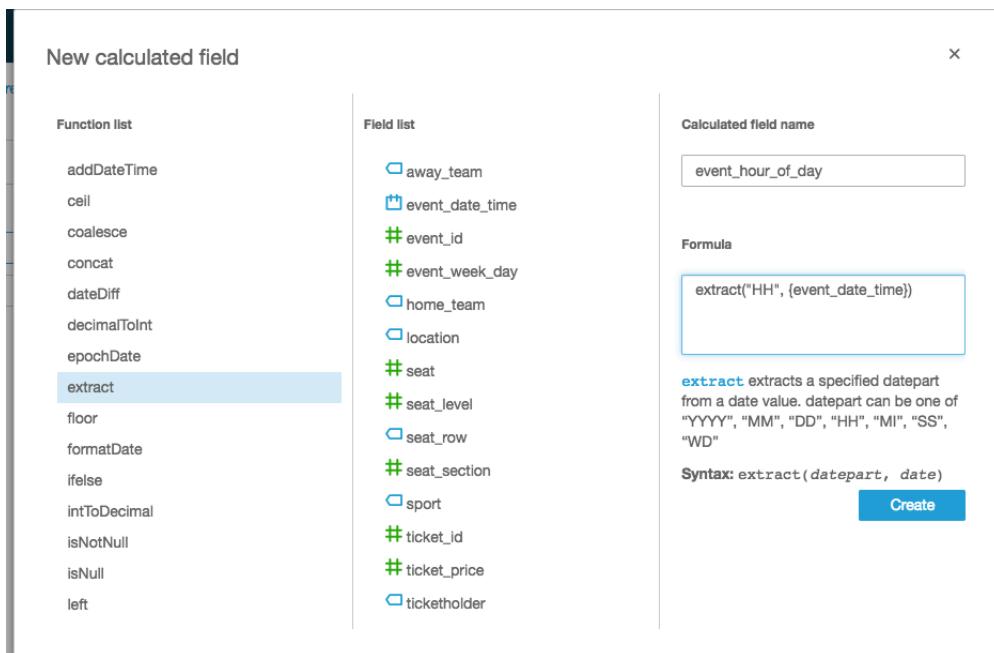
Note: `extract` returns a specified portion of a date value. Requesting a time-related portion of a date that doesn't contain time information returns 0. `WD`: This returns the day of the week as an integer, with Sunday as 1.

4. Click **Create**.

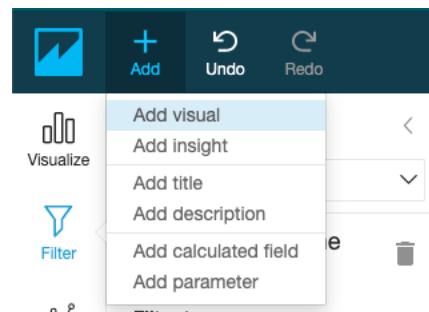


5. Add another calculated field with the following attributes:
- Calculated field name: "event\_hour\_of\_day".
  - Formula: `extract("HH", {event_date_time})`.

Note: `HH` returns the hour portion of the date.

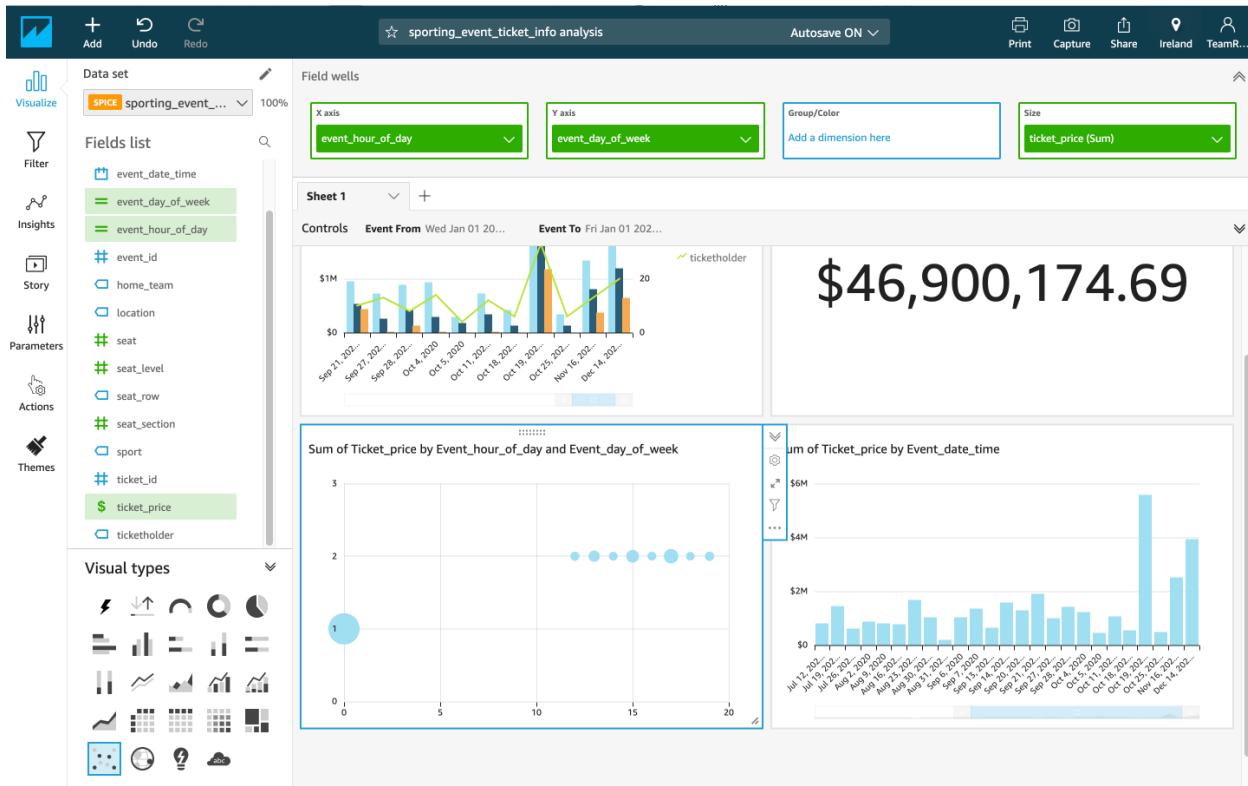


6. Click Add button in the top left and choose **Add visual**.



7. For field type, select the **scatter plot**.

8. In the Fields list, select and drag the following attributes to the Field wells pane to set the graph attributes:
  - X-axis:** "event\_hour\_of\_day"
  - Y-axis:** "event\_day\_of\_week"
  - Size:** "ticket\_price"



## Amazon QuickSight ML-Insights

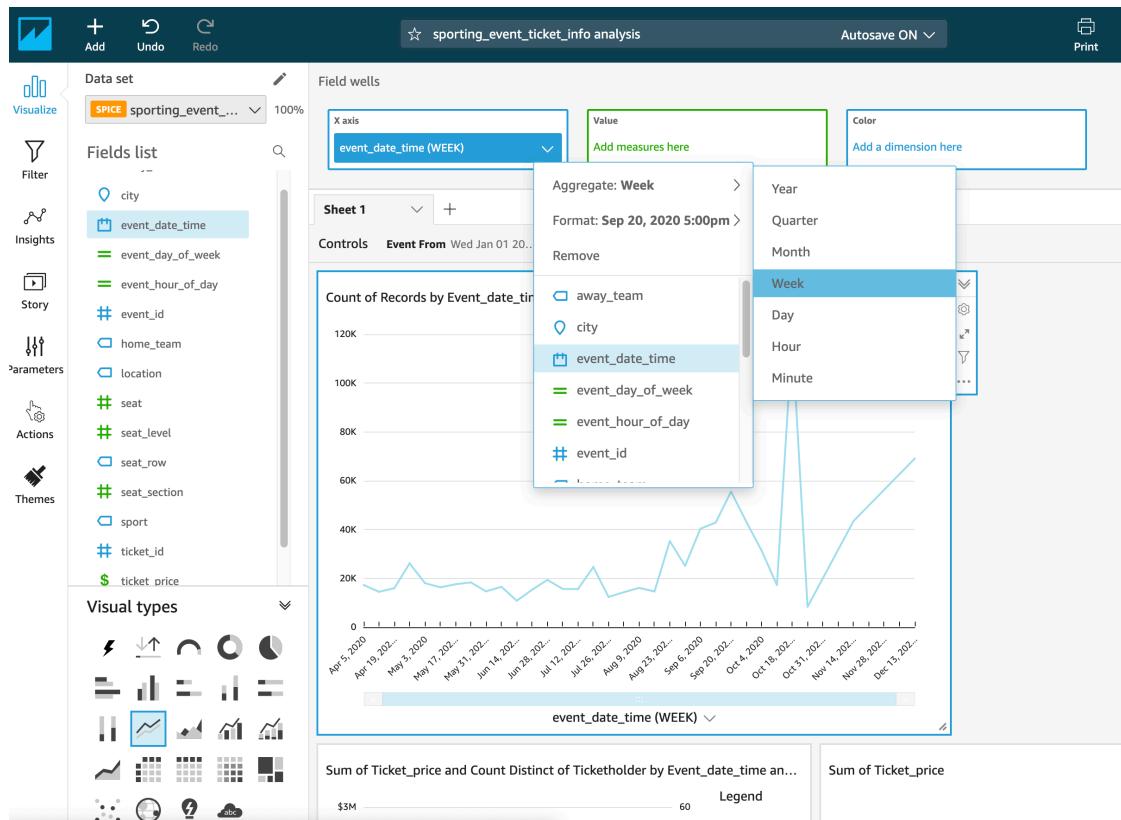
With Amazon QuickSight, you can add Machine Learning capabilities to your visuals, easily, with one click action. There are 3 types of Machine Learning Insights:

- Narrative
- Anomaly Detection
- Forecasting

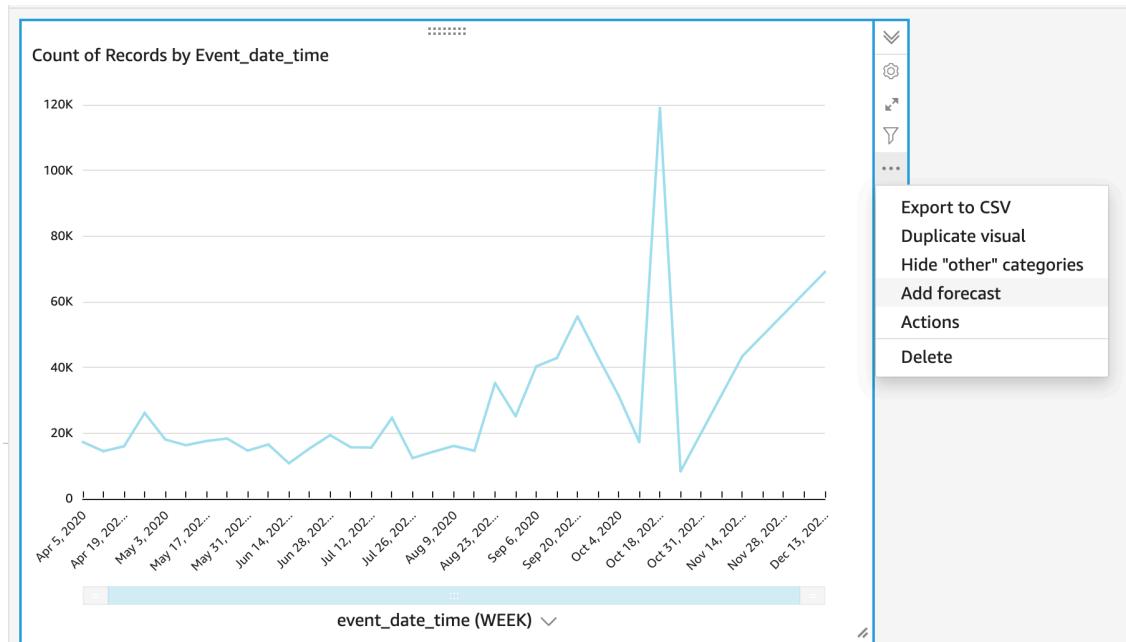
ML-Insights is only available to enterprise version of QuickSight. If you did not select enterprise at the beginning of the lab you will need to upgrade to Enterprise Edition before you start with the task. To upgrade your Amazon QuickSight Subscription from Standard Edition to Enterprise Edition please follow this guide <https://docs.aws.amazon.com/quicksight/latest/user/upgrading-subscription.html>.

Let's see how we can add a bit of forecasting in our dashboard. Forecasting works with timeseries, which is better represented with a line graph. Let's first create a line graph.

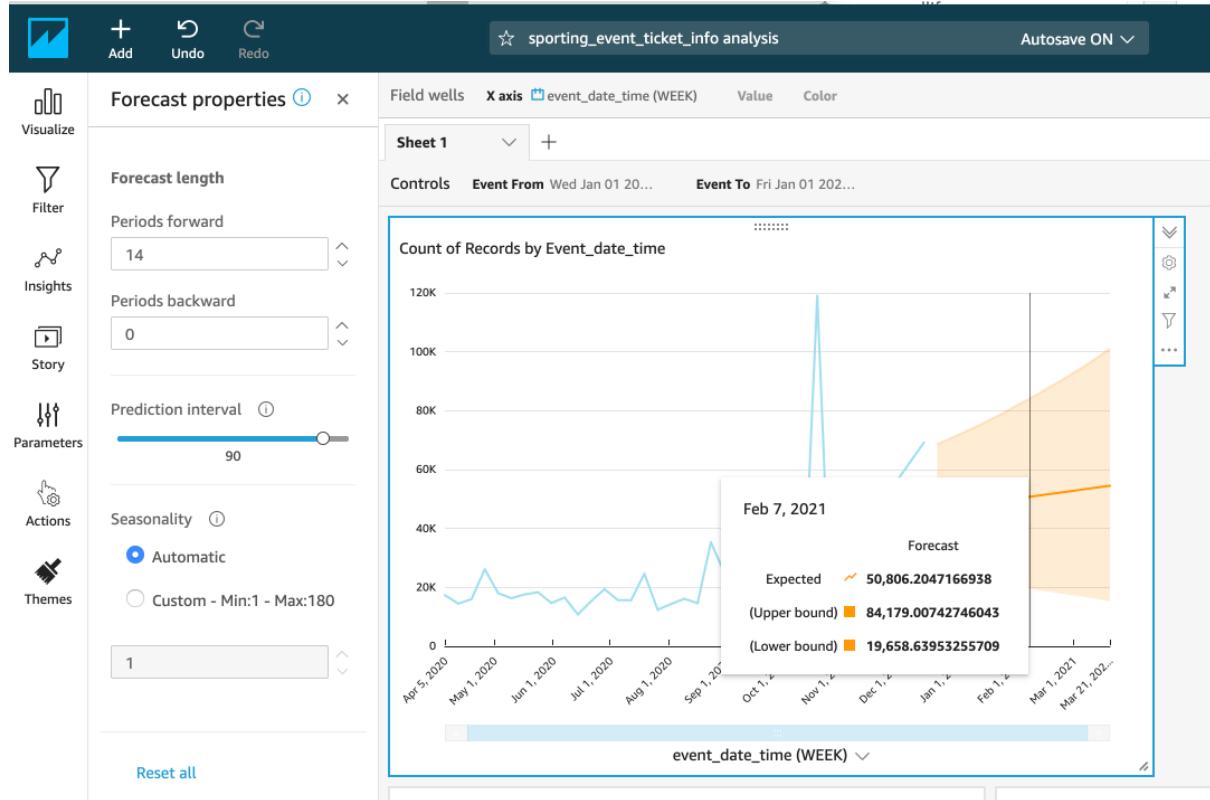
1. Click **Add Visual** at the top left corner of screen, and select **Line Chart** and add the **event\_date\_time** as the **x-axis** and **aggregate by week**. As shown in below screenshot:



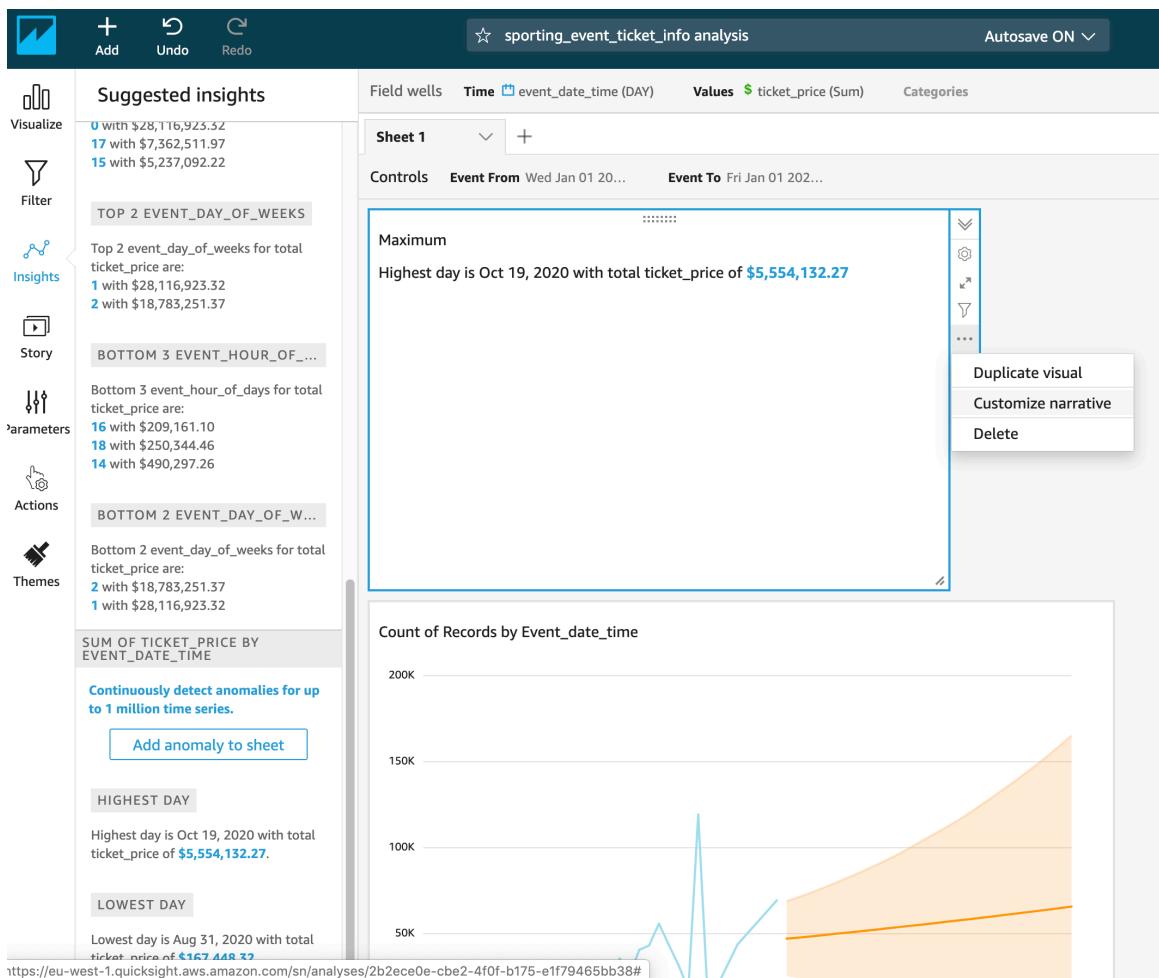
- Add forecasting to the visual. To do that, click on the drop-down arrow on the top right corner of the visual, and then click **Add forecast**.



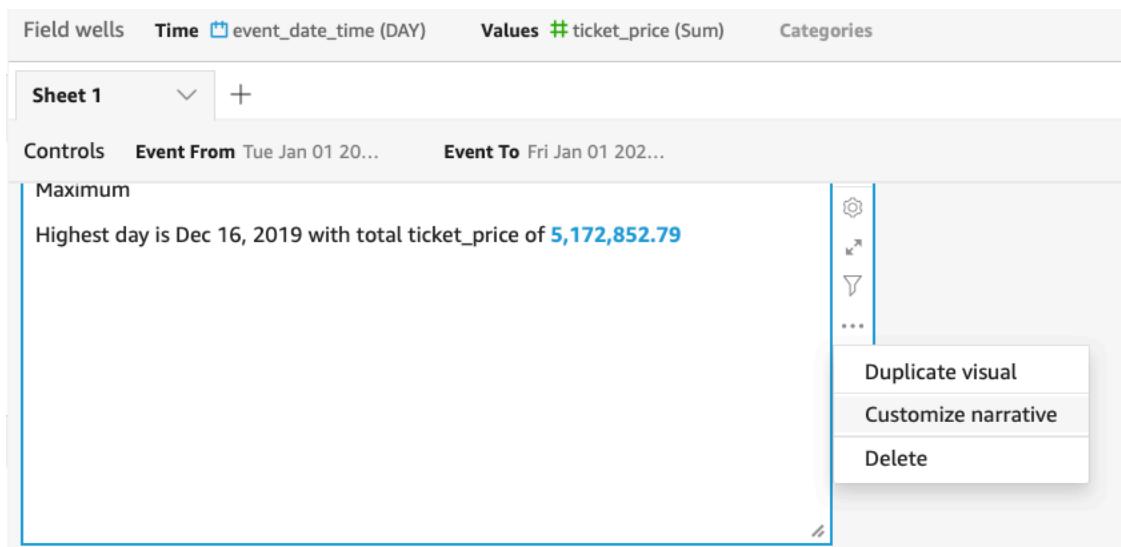
The visual will add forecast, you can hover over and explore forecasted data as shown below. Feel free to explore with the properties of the forecast algorithm.



Specific insights like max or min values from existing visuals can also be attached to our analysis. Click on insights on the left menu and then select the plus icon over the category of your preference and see how it gets added to the dashboard.



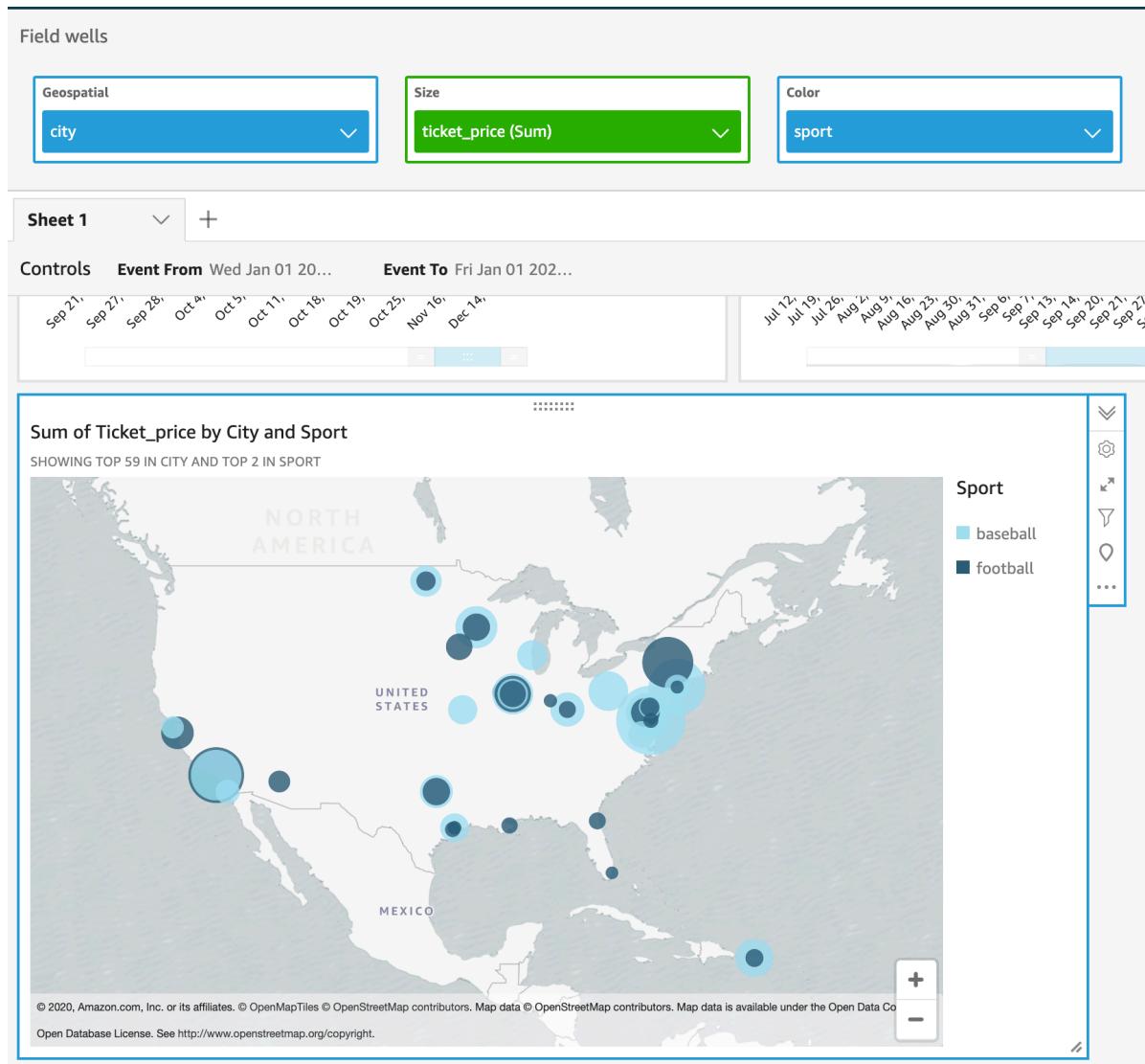
You can go ahead and change the font and text to match what you would like it to say. Go to customize Narrative from the 3 dots ... on the menu of the right of the visual and update it.



## Create a Geospatial visual

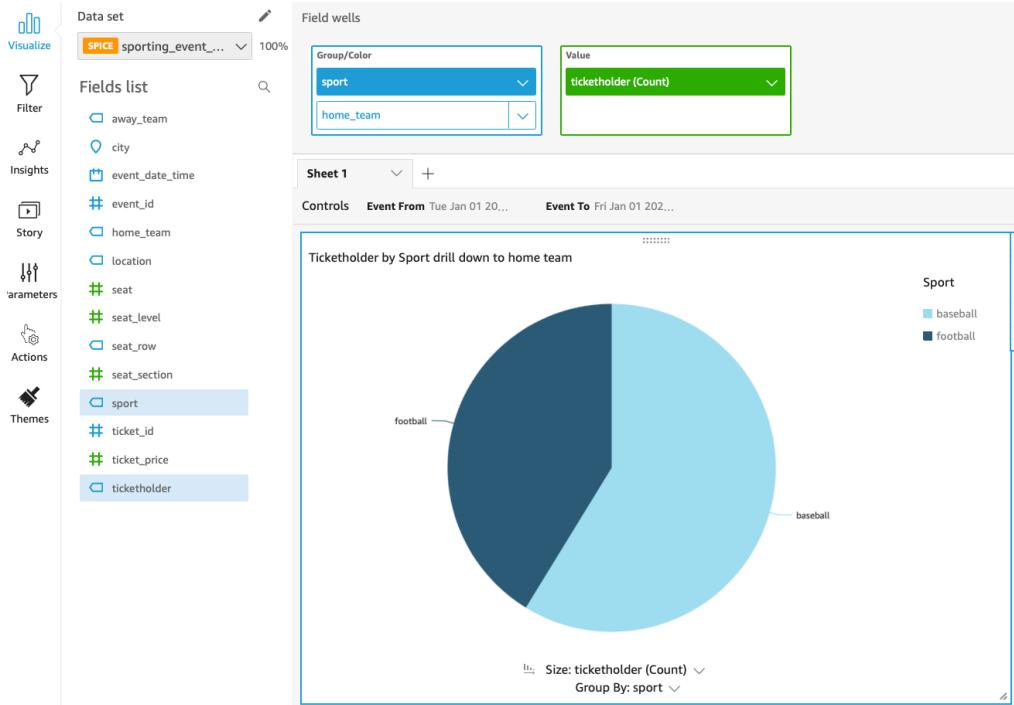
Now let's create a map with the cities where events took place and the sum of sales of tickets for each of them.

Go to Visualize and add a new visual select the Points on Map type. Then, add city for as Geospatial field, ticket price as size and sport as color as it shows below:

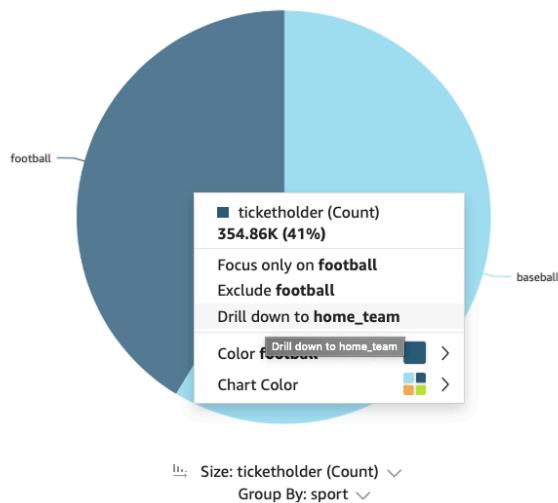


## Drill down on a Pie Chart

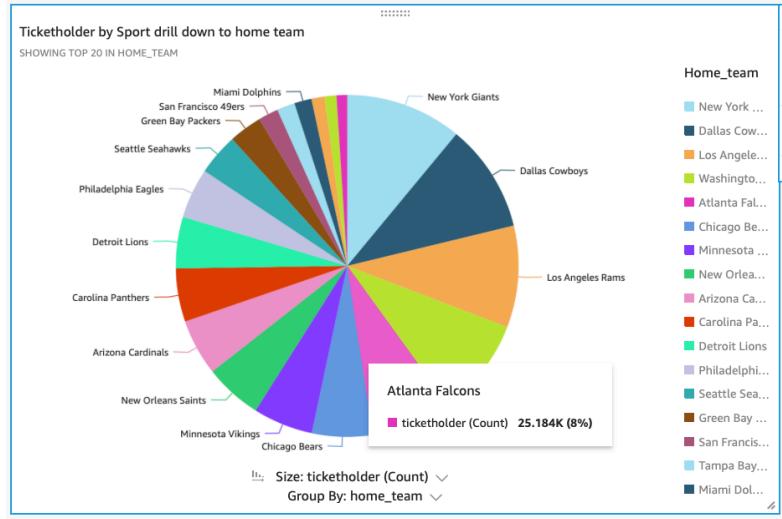
Let's add a new visual and select Pie Chart Icon. Add *sport* to group/color and also drop *home\_team* right below sport selection until it says "Add drill-down Layer". Drop it should look like the one below:



Now when you click a sport you have the option to drill down to home\_team:



And see the teams under that sport:



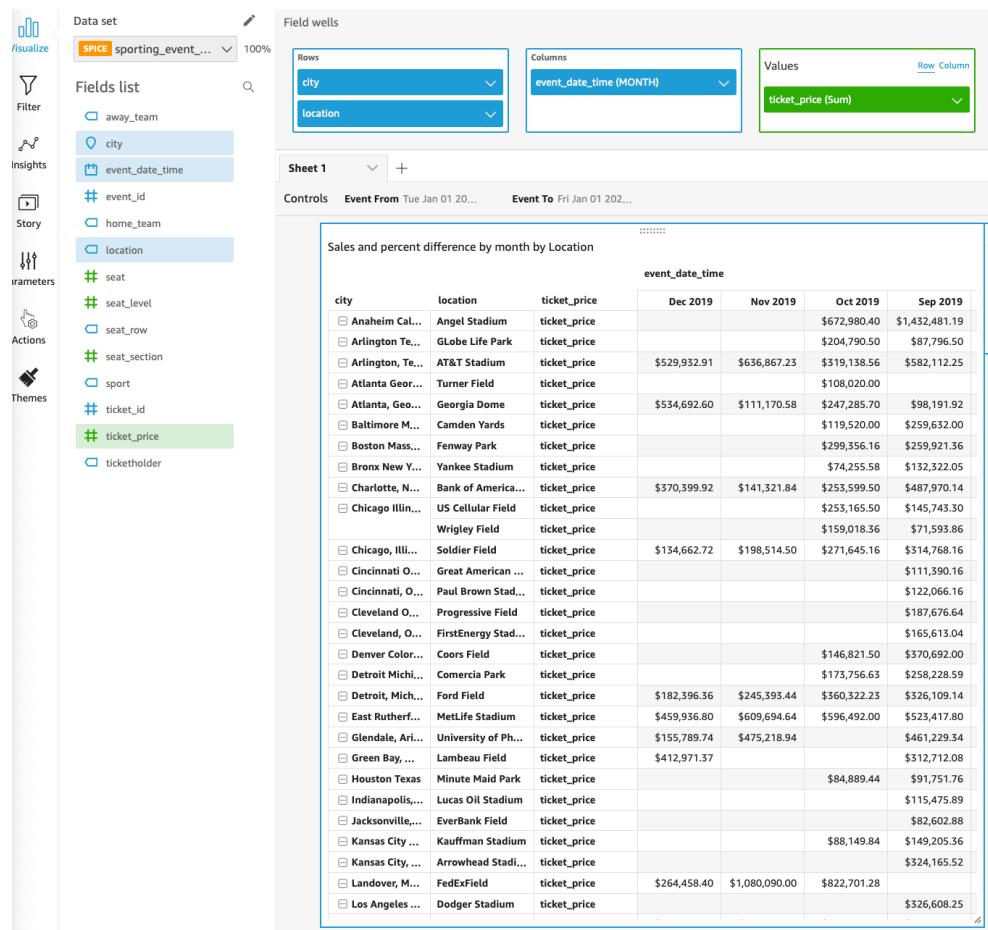
If you want to go back up you can click the arrow on the right of the visual menu.

## On the fly custom metrics

Now let's see how easy it is to make a percent difference month to month on the tickets sold by city and team.

Let's add a new visual, this time it must be of the Pivot table type.

Add the city and location on the rows, then even\_date\_time as column (aggregated by month) and ticket\_price (sum) as value (Row). It should look like this:



We can also add custom calculations to this table. In this case, we are interested in finding out how much ticket's total price varies between months for each stadium.

To do so, add again ticket\_price(sum) as a value field well. Next, add the Percent difference for this new field by clicking in the drop-down arrow and clicking on table calculation, then select Percentage difference:

The screenshot shows a Quicksight dashboard with a table titled "Sales and percent difference by month by Location". The table has columns for city, location, and event\_date\_time (Month). The values are ticket\_price (Sum). A context menu is open over a cell containing "\$672,980.40". The menu options include:

- Aggregate: Sum
- Show as: Number
- Format: 1,234,5678
- Add table calculation
- Conditional formatting
- Remove
- Running total
- Difference
- Percentage difference** (selected)
- Percent of total
- Rank
- Percentile

Although Quicksight should have changed new row format to be Percentage, confirm so through the *Show as* option:

The screenshot shows the same Quicksight dashboard with the "Show as" option set to "Percent" in the context menu. The menu now includes:

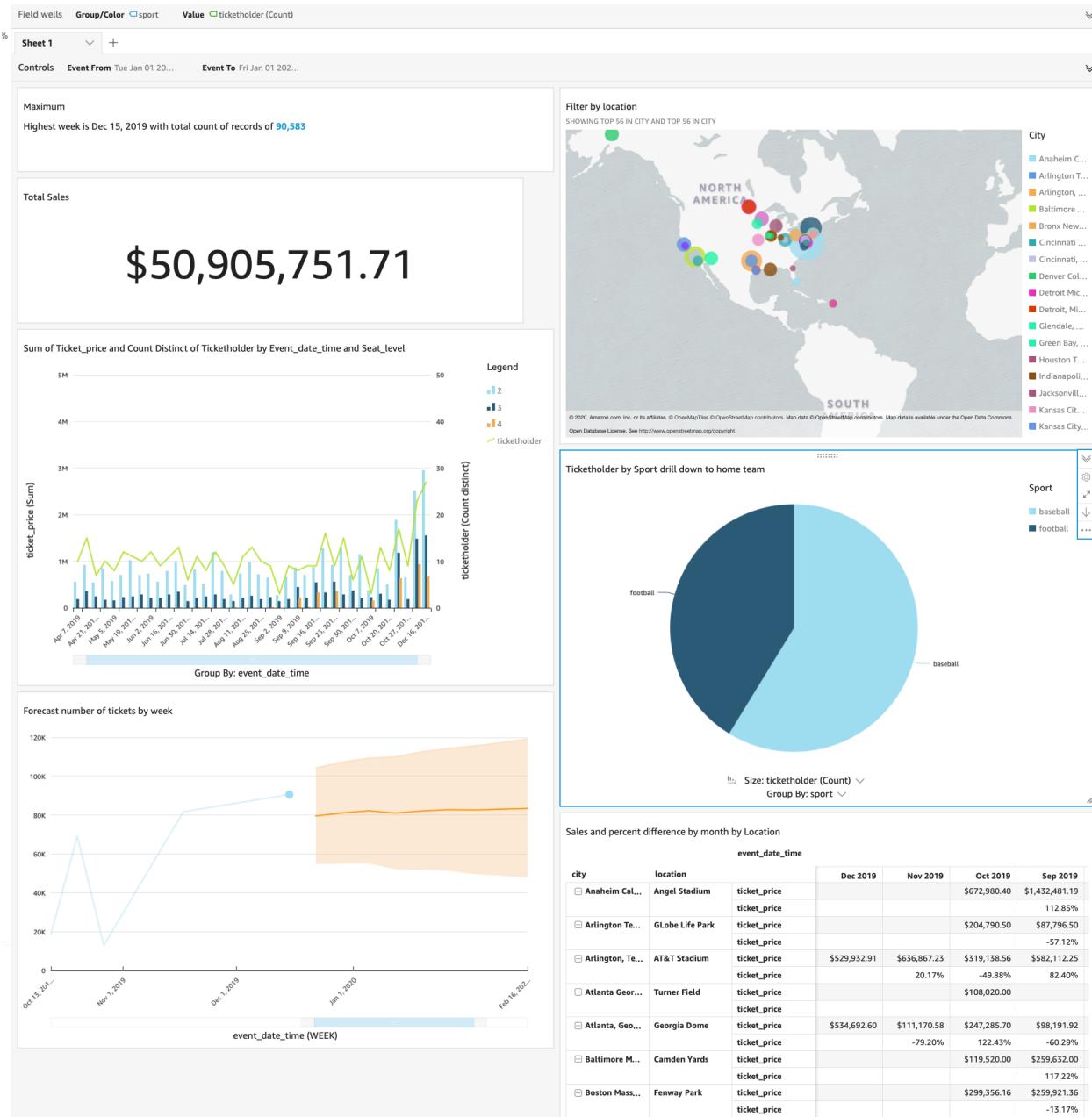
- Number
- Currency
- Percent** (selected)
- Add table calculation
- Calculate as
- Remove calculation

By now, you should be able to see the percent change compared to the previous month:

The screenshot shows the Quicksight visualizer interface. On the left, there are navigation icons for Visualize, Filter, Story, Parameters, Actions, and Themes. The main area displays the field wells for Rows (city, location) and Columns (event\_date\_time (MONTH)). Below this is a table titled "Sales and percent difference by month by Location". The table has columns for location and event\_date\_time (Month). The values are ticket\_price (Sum). The last column shows the percent difference compared to the previous month. The table data is as follows:

location	Oct 2019	Sep 2019	Aug 2019	Jul 2019	Jun 2019	May 20
Angel Stadium	\$2,980.40	\$1,432,481.19	\$260,650.68	\$838,314.10	\$84,007.32	\$321,396.
Globe Life Park		112.85%	-81.80%	221.62%	-89.97%	282.5%
AT&T Stadium	\$9,138.56	\$582,112.25	-57.12%	131.86%	-48.80%	11.05%

Reached to this point, there are a few visuals in our analysis. You can re-arrange them at any time to make it look like how you want to:



## Interactive filtering

To finish up our dashboard and make it interactive, we are going to add an action to the map we added previously so when someone selects a city that selection is taken as a filter and applied to all the other visuals.

Select the MAP visual and go to Actions on the left menu. Click on Define a custom action.

Complete the given options as follows and click on Save at the bottom of the Edit Action menu:

1. Action name: "City filter"
2. Activation: "Select"
3. Action type: "Filter Action"
4. Filter scope: "selected fields" and city
5. Target visuals: "All visuals"

The screenshot shows the 'Edit action' dialog on the left and a map visual on the right. The map displays North America with various cities highlighted by size and color according to ticket price. A legend on the right lists cities with their corresponding color-coded circles.

**Edit action**

Action name: Filter City

Activation: Select

Action type: Filter action

Filter scope: Selected fields, city

Target visuals: All visuals

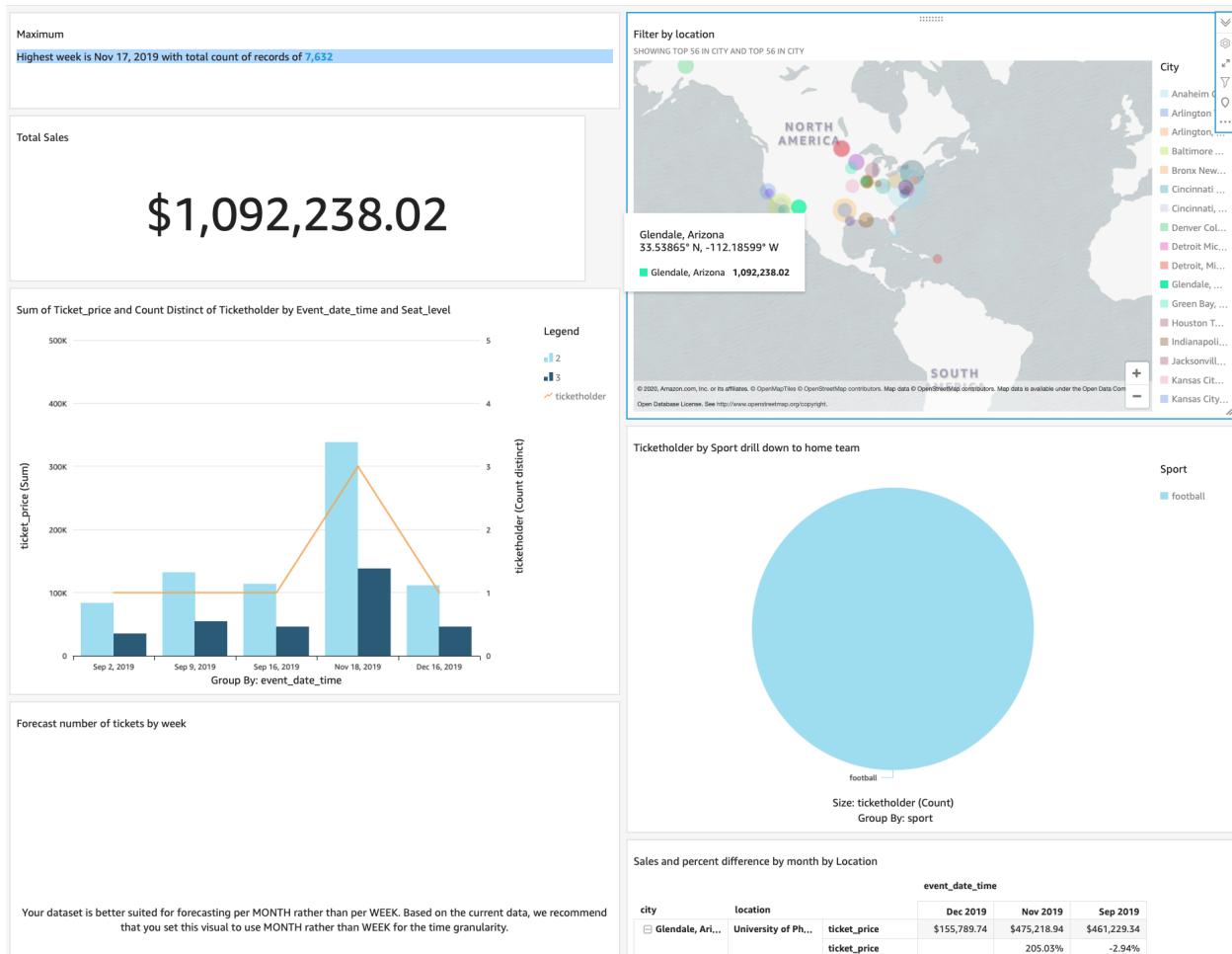
Map Visual (right):

Showing top 56 in city and top 56 in city.

Legend (City):

- Anaheim C...
- Arlington T...
- Arlington, ...
- Baltimore ...
- Bronx New...
- Cincinnati ...
- Cincinnati, ...
- Denver Col...
- Detroit Mic...
- Detroit, Mi...
- Glendale, ...
- Green Bay, ...
- Houston T...
- Indianapol...
- Jacksonvill...
- Kansas Cit...
- Kansas City...

Now when we select a city like Glendale Arizona, all the values should change to show only data for that city:



If you want to remove the filter just click anywhere on the map where there are no cities and it will display all data again.

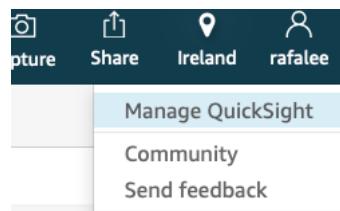
## Creating and sharing a dashboard

A *dashboard* is a read-only snapshot of an analysis that you can share with other Amazon QuickSight users for reporting purposes. In a dashboard, other users can still play with visuals and data but that will not modify dataset.

You can share an analysis with one or more other users with whom you want to collaborate on creating visuals. Analysis provide other uses to write and modify data set.

Let's create a user and share our new dashboard.

On the top right click on the user and Manage QuickSight:



Navigate now to Manage users option menu and then choose to Invite users:

Username	Email
jebeva@amazon.com	jebeva@amazon.com
rafalee	rafalee@amazon.com

Type the username of the user you would like to invite (that will be the login id) and click the plus icon on the right:

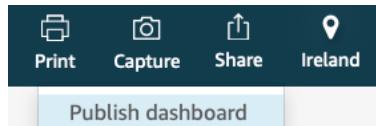
Username	Email	Role	IAM user
jane			

Once added to the table below, enter email address of the user. For testing purposes, use your own email and then use on the Invite button:

Username	Email	Role	IAM user
jane	jane@email.com	READER	No

That email address will receive an email with an invitation to join your quicksight account and you will be able to share the dashboard with them once they sign up.

Get back to our sporting\_event\_ticket\_info analysis. Using the Share option, you can publish a dashboard.



Enter its name and click on Publish dashboard.

Publish a dashboard ×

Publish new dashboard as  
My First Dashboard

Replace an existing dashboard  
QuicksightLab

[Advanced publish options ▾](#) Publish dashboard

Once clicked, you need to enter which users you would like to share it with. Type here the name of the user you created before and share the dashboard:

Share dashboard with users ×

Select users in this account.

jane

jane

Name	Email	Permission	Role

Share dashboard with users ×

Select users in this account.

Search by email address

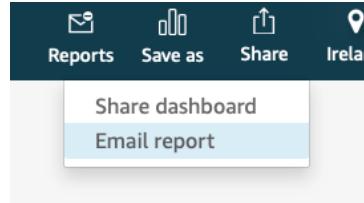
Share with all users in this account

Name	Email	Permission	Role
jebeva@amazon.com	jebeva@amazon.com	Viewer	READER

Manage dashboard access Share

If the user has activated his/her account, it should receive an email with a link to the dashboard.

You will be redirected to the dashboard you just shared. You also have the option to email it to audience users regularly according to different customizable settings:



#### Edit email report

You can send the dashboard in report form to individuals or groups, either once or on a schedule.

Users who are readers (not authors or admins) are billed one session per report. Each report includes a free dashboard session. [Learn more](#)

#### 1. Set the schedule for the report

Schedule

Send first report on

Time zone

#### 2. Customize email text and report preference

Report title

(Optional) E-mail subject line

(Optional) E-mail body text

Optimize report for

Viewing on a desktop (Preserve the dashboard layout) [\(i\)](#)

Viewing on a mobile device (Display visuals in a single column) [\(i\)](#)

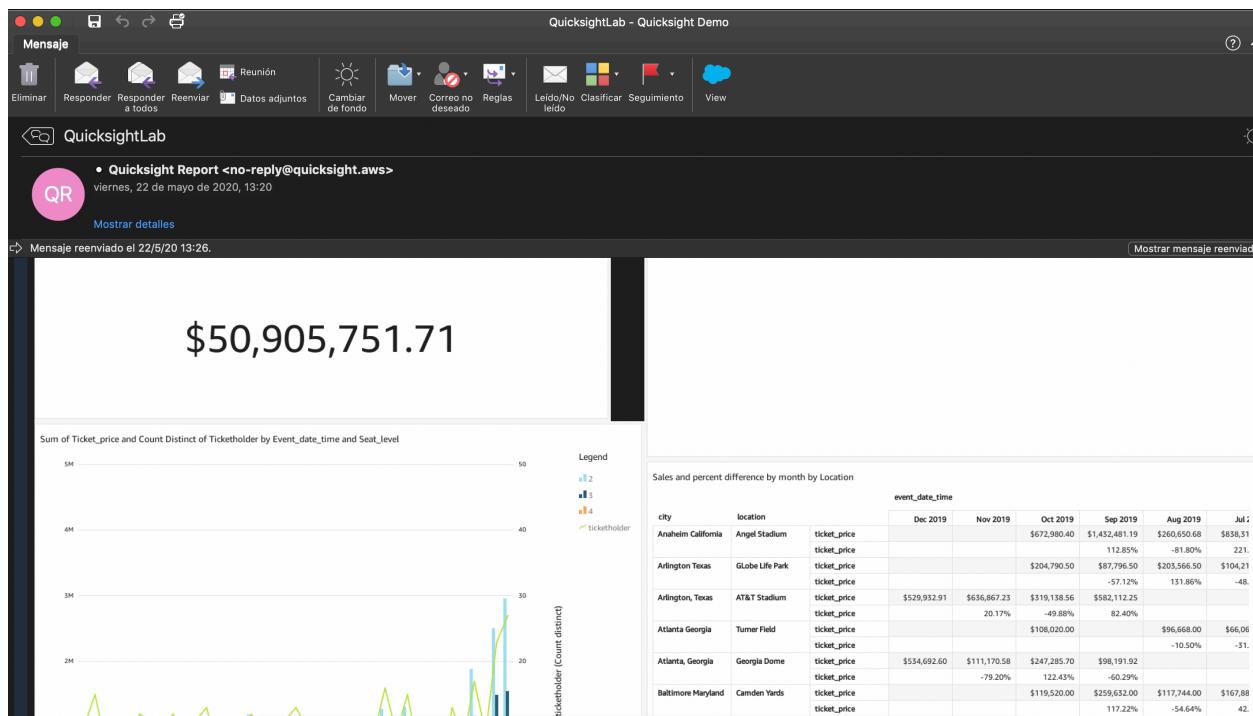
#### 3. Select recipients

##### Recipients

Reports can be sent automatically to selected recipients. Other readers can subscribe themselves via the dashboard. [Learn more](#)

Send email report to all users with access to dashboard

<input type="checkbox"/>	Username	Type	Email	Status	Action
<input checked="" type="checkbox"/>	david	Individual	rafalee@amazon.com	Subscribed	
<input checked="" type="checkbox"/>	jebeva@amazon.com	Individual	jebeva@amazon.com		
<input type="checkbox"/>	rafalee	Owner	rafalee@amazon.com	Subscribed	<input type="button" value="Send test report"/>



Congratulations!! You have successfully completed this lab.