

Amazon Web Services

Data Engineering Immersion Day

Exploring Data Lake with Amazon Athena and Amazon Quicksight

Nov 2019

Table of Contents

[Introduction 2](#_Toc10199277)

[Prerequisites 2](#_Toc10199278)

[Getting Started 2](#_Toc10199279)

[Query Data with Amazon Athena 3](#_Toc10199280)

[Build an Amazon QuickSight Dashboard 7](#_Toc10199281)

[Set up QuickSight 7](#_Toc10199282)

[Create QuickSight Charts 12](#_Toc10199283)

[Create QuickSight Parameters 15](#_Toc10199284)

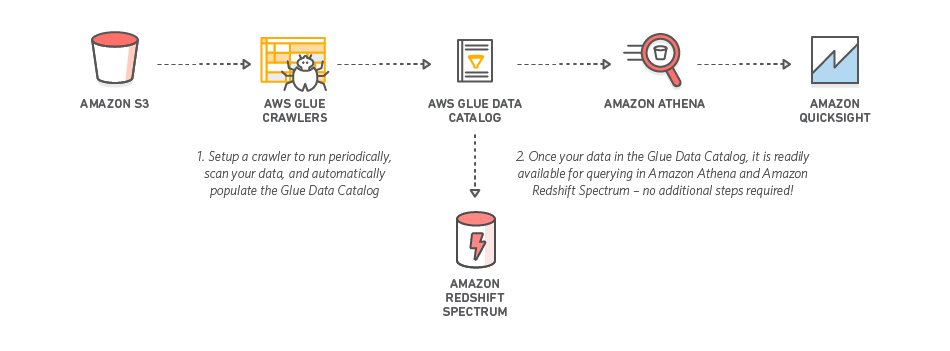
[Create a QuickSight Filter 18](#_Toc10199285)

[Add Calculated Fields 20](#_Toc10199286)

[Amazon QuickSight ML-Insights (Optional) 24](#_Toc10199287)

# 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.



## Prerequisites

The DMS Lab and Glue ETL lab is a prerequisite for this lab.

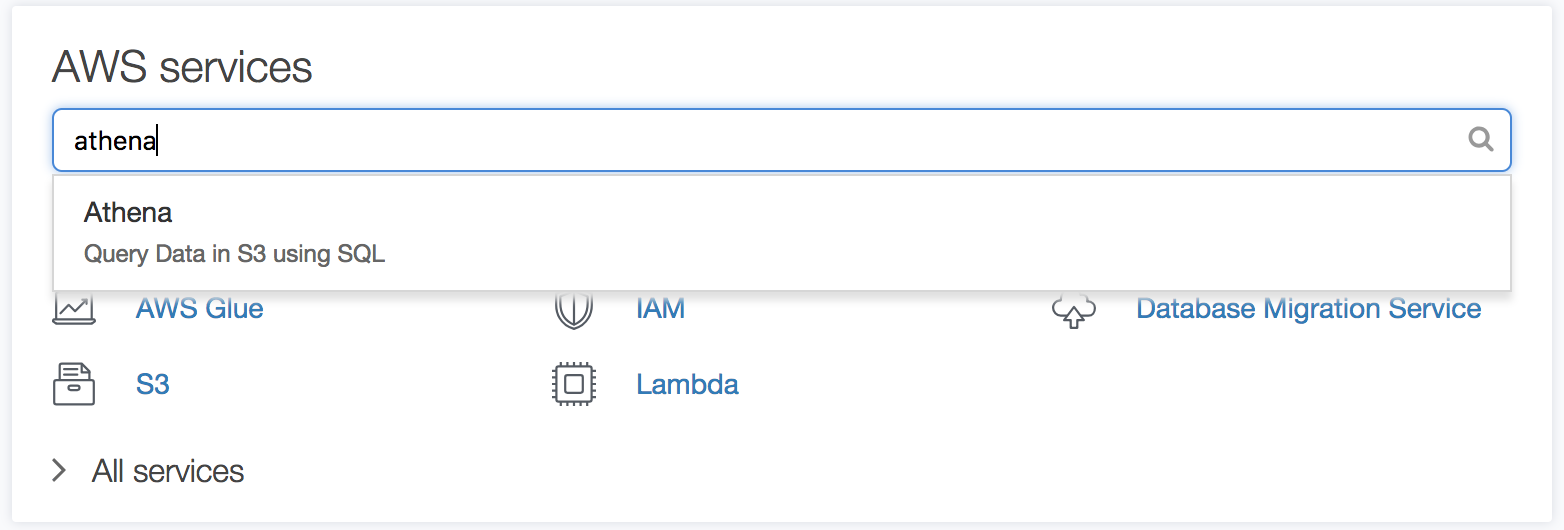
## Getting Started

In this lab, you will complete the following tasks:

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

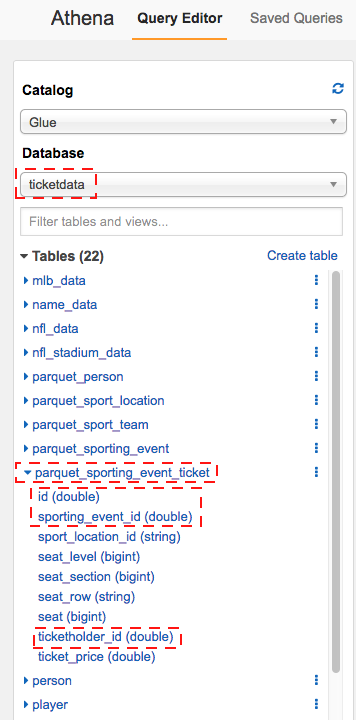
# Query Data with Amazon Athena

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



1. In the Query Editor, select your newly created database e.g., "ticketdata”.
2. 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.

1. 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

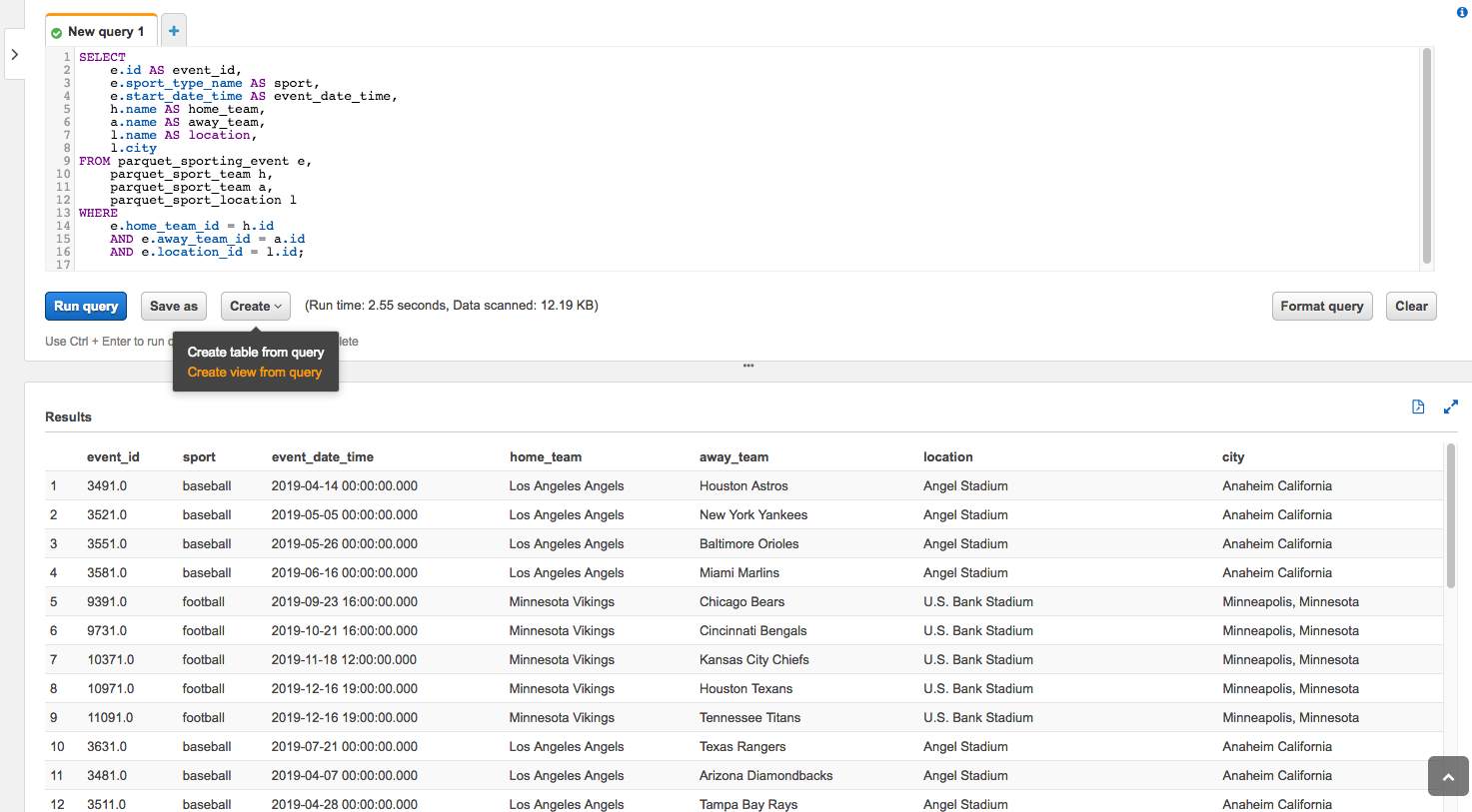
WHERE

e.home\_team\_id = h.id

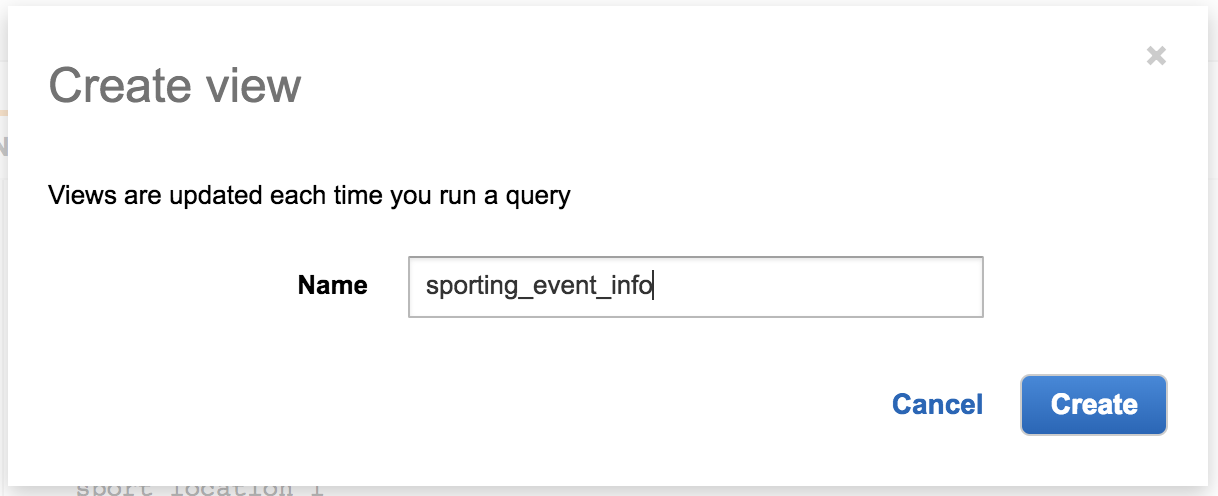
AND e.away\_team\_id = a.id

AND e.location\_id = l.id;

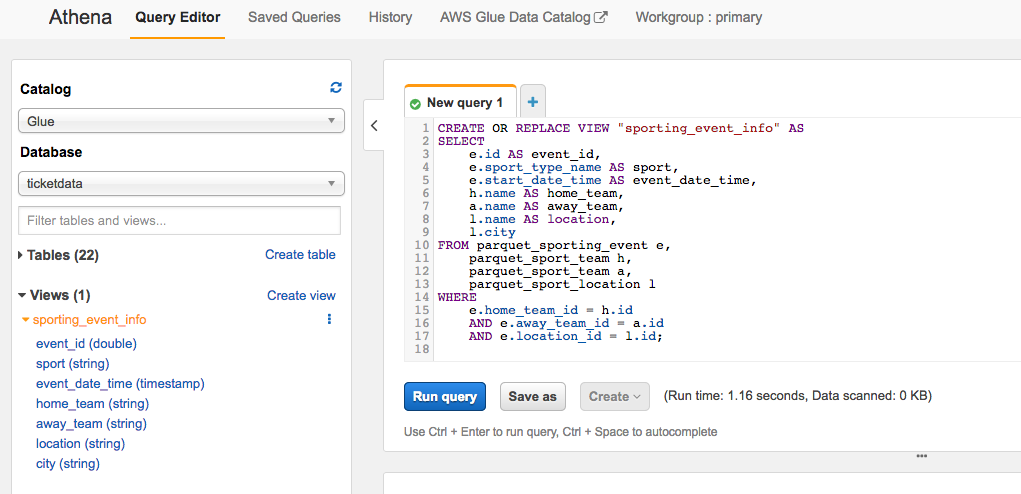
The results appear beneath the query window.



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



Your new view is created



1. Copy the following SQL syntax into the New Query 2 tab and click **Run 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,

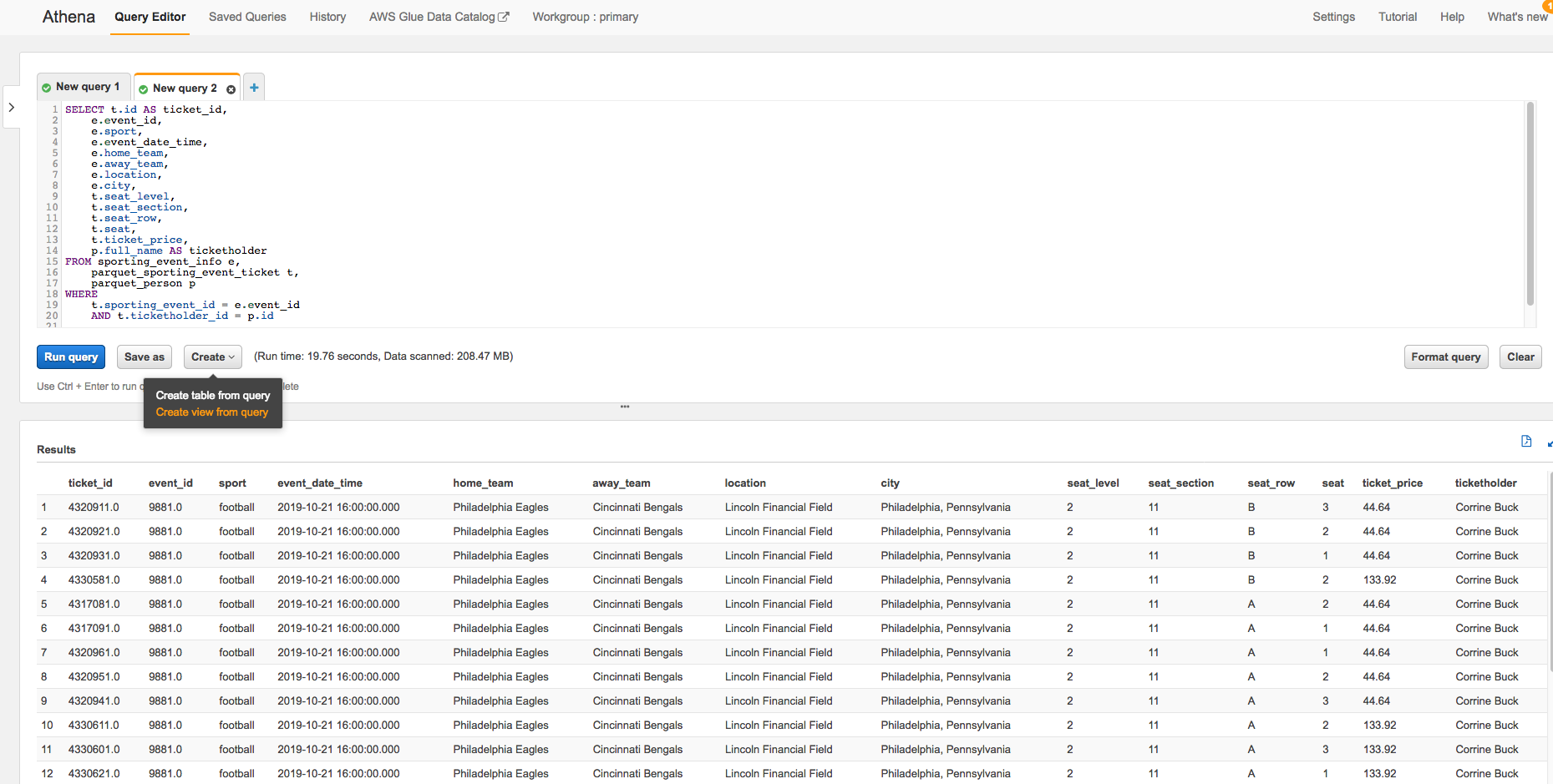
parquet\_person p

WHERE

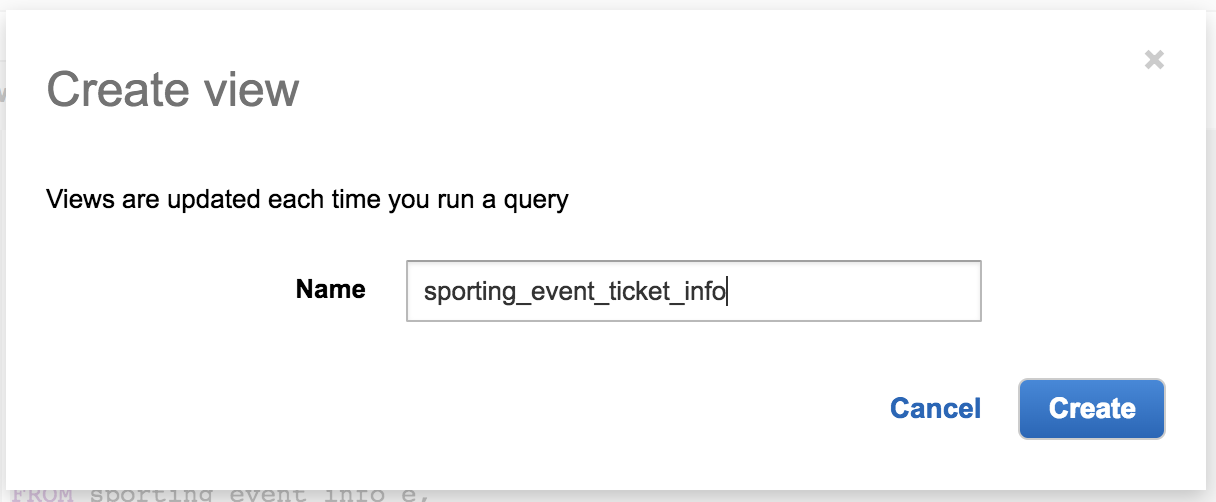
t.sporting\_event\_id = e.event\_id

AND t.ticketholder\_id = p.id

The results appear beneath the query window.



1. As shown above Click **Create view from query**.
2. Name the view "sporting\_event\_ticket\_info" and click **Create**.



1. Copy the following SQL syntax into the New Query 3 tab and click **Run Query**.

SELECT

sport,

count(distinct location) as locations,

count(distinct event\_id) as events,

count(\*) as tickets,

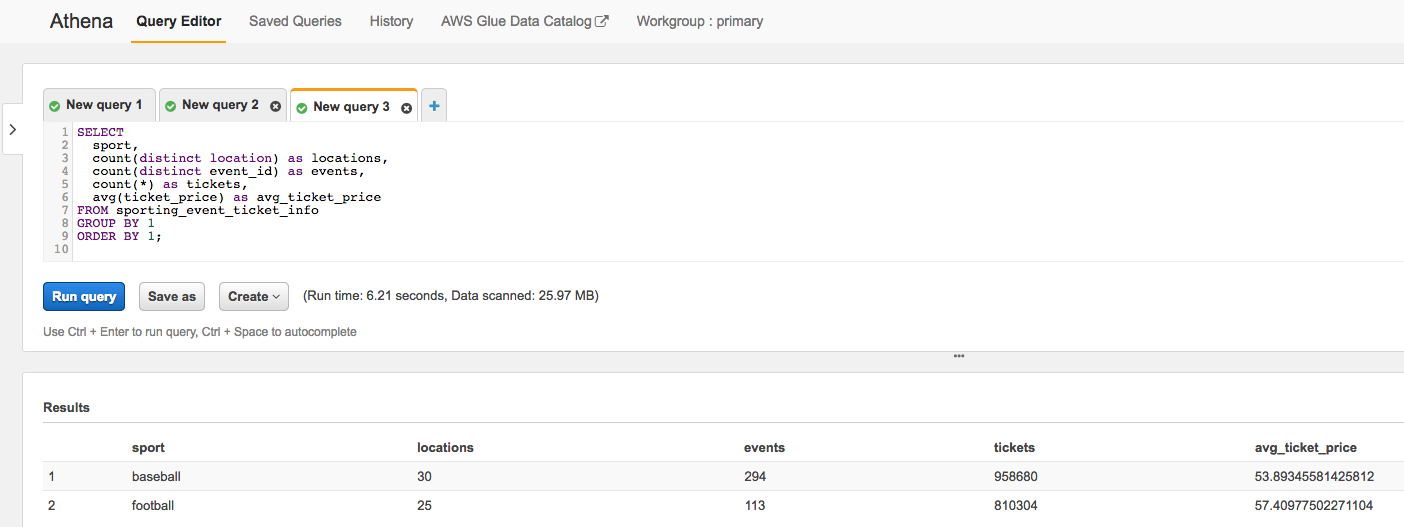
avg(ticket\_price) as avg\_ticket\_price

FROM sporting\_event\_ticket\_info

GROUP BY 1

ORDER BY 1;

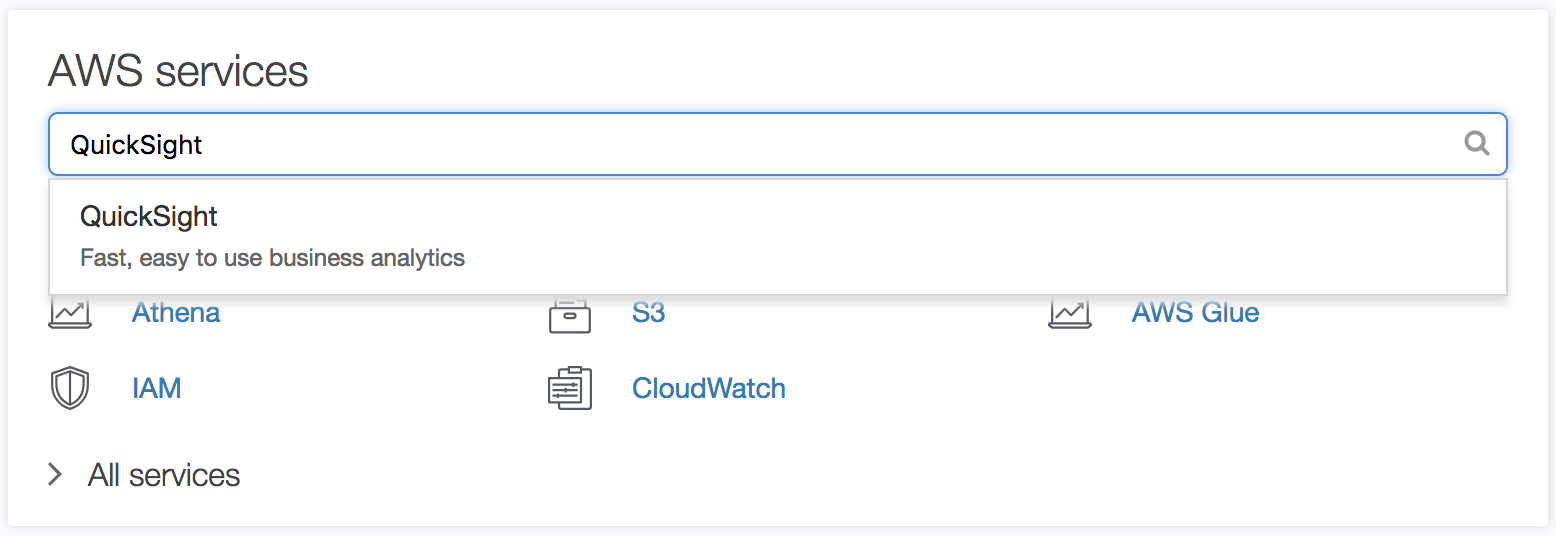
You query returns two results in approximately five seconds. The query scans 25MB of data, which prior to converting to parquet, would have been 1.59GB of CSV files.



# Build an Amazon QuickSight Dashboard

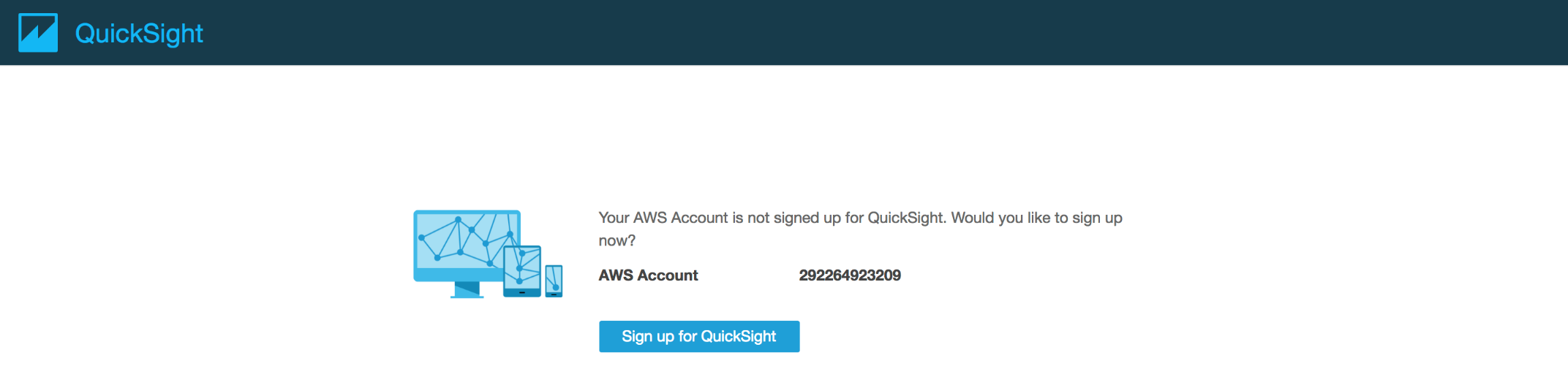
## Set up QuickSight

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

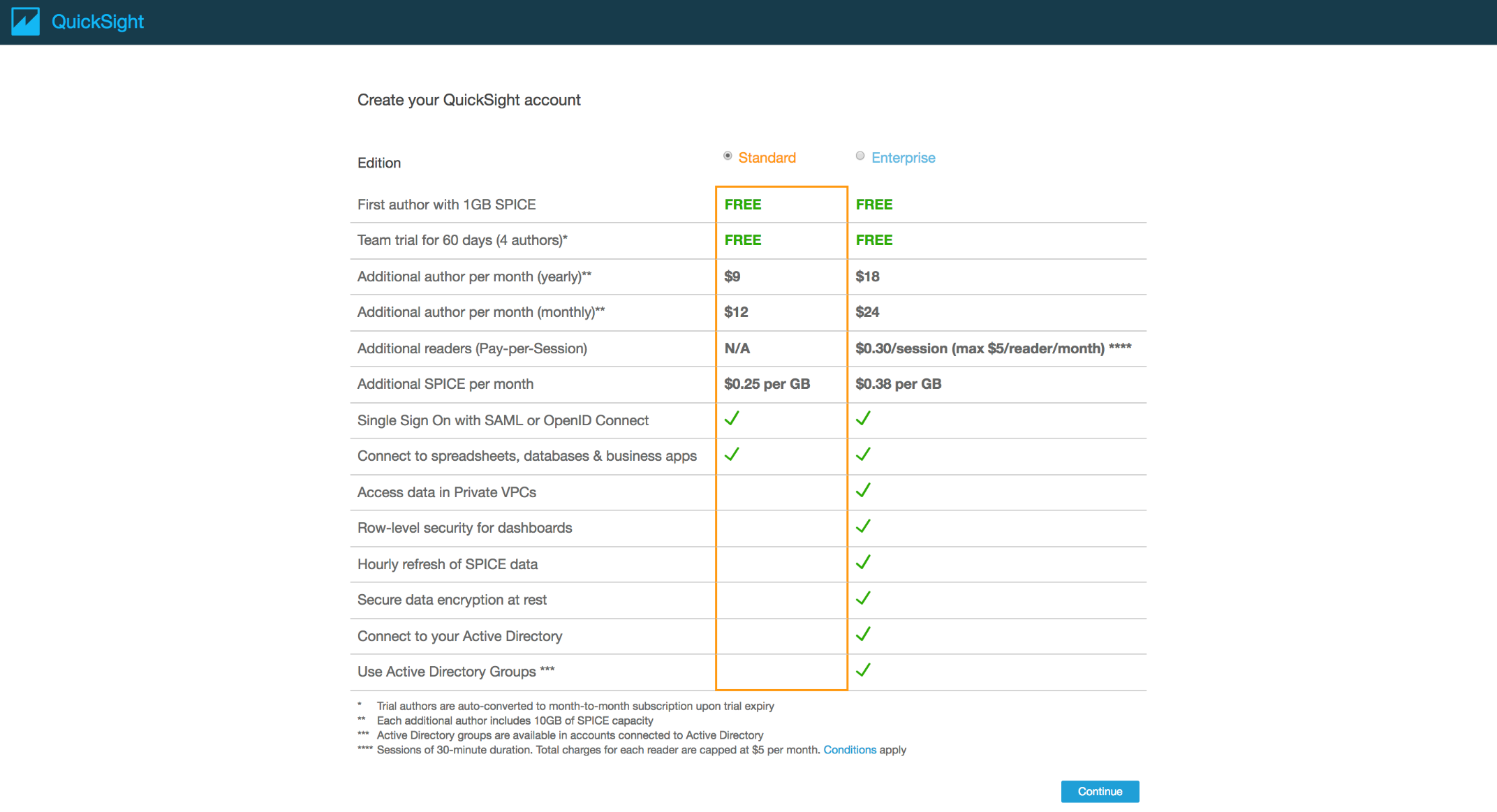


If this is the first time you have used QuickSight, you are prompted to create an account.

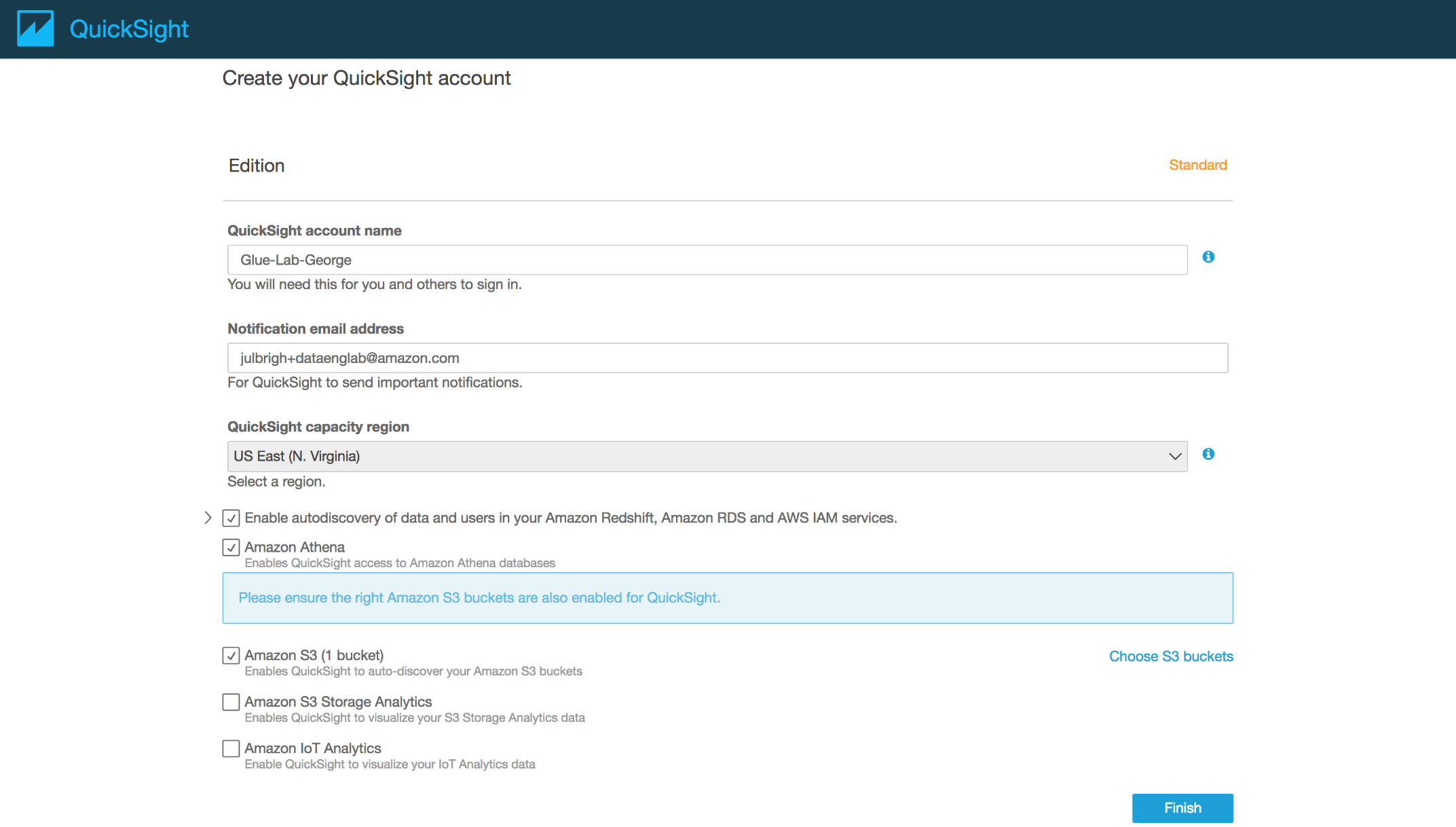
1. Click **Sign up for QuickSight**.



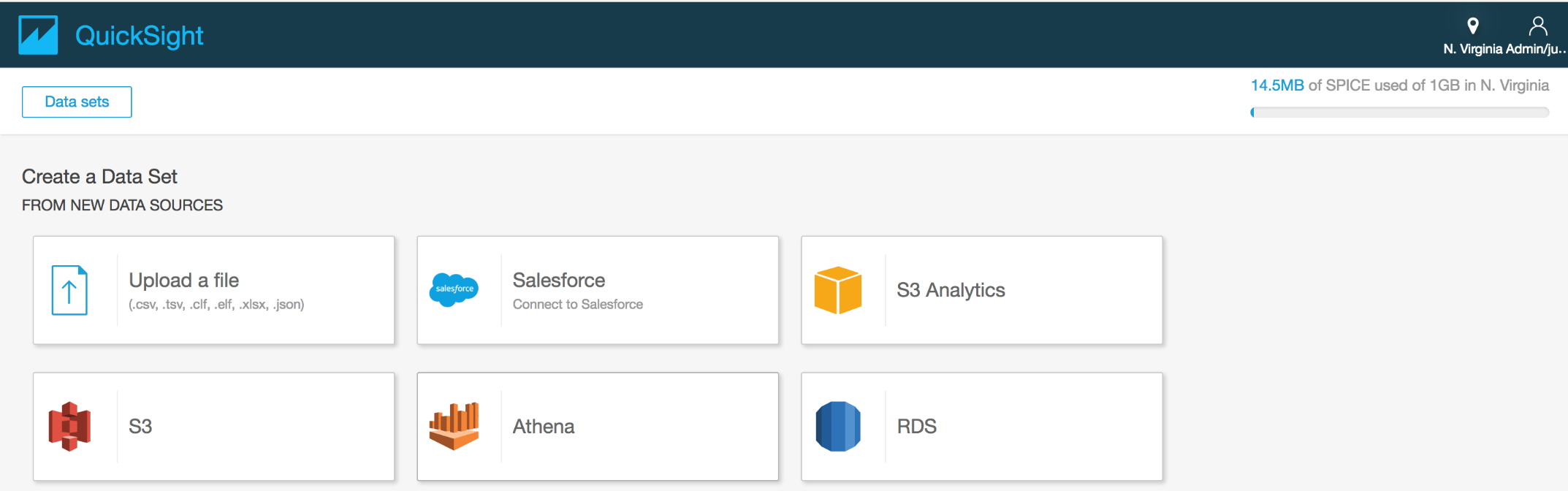
1. For account type, choose **Standard**. If you plan to complete the Bonus Exercise, please choose **Enterprise** Version
2. Click **Continue**.



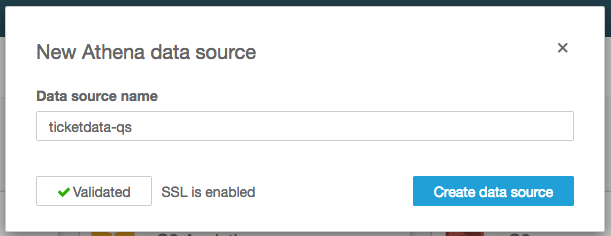
1. On the Create your QuickSight account page, fill out your name and email address.
2. Select region and the check boxes to enable autodiscovey, Amazon Athena, and Amazon S3.
3. Click **Choose S3 buckets** and select your DMS bucket (e.g., "dms-lab-george").
4. Click **Finish**.



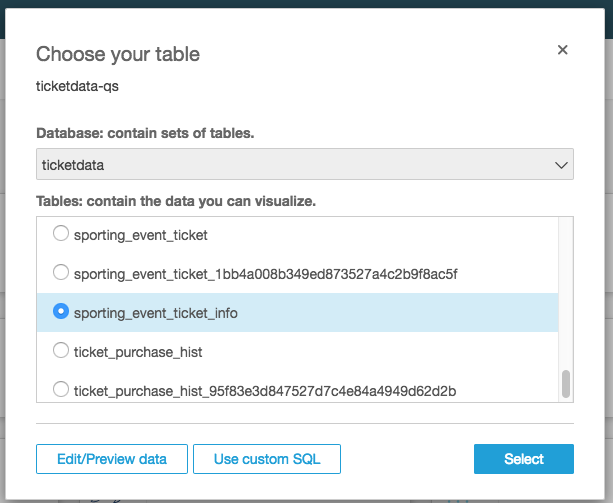
1. On the QuickSight landing page, click **Manage Data**.
2. Click **New Data Set**.
3. On the Create a Data Set page, select **Athena** as the data source.



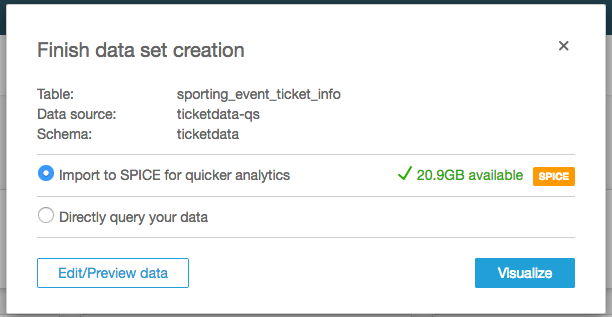
1. For Data source name, type “ticketdata-qs" and click **Validate** **connection**.
2. Click **Create data source**.



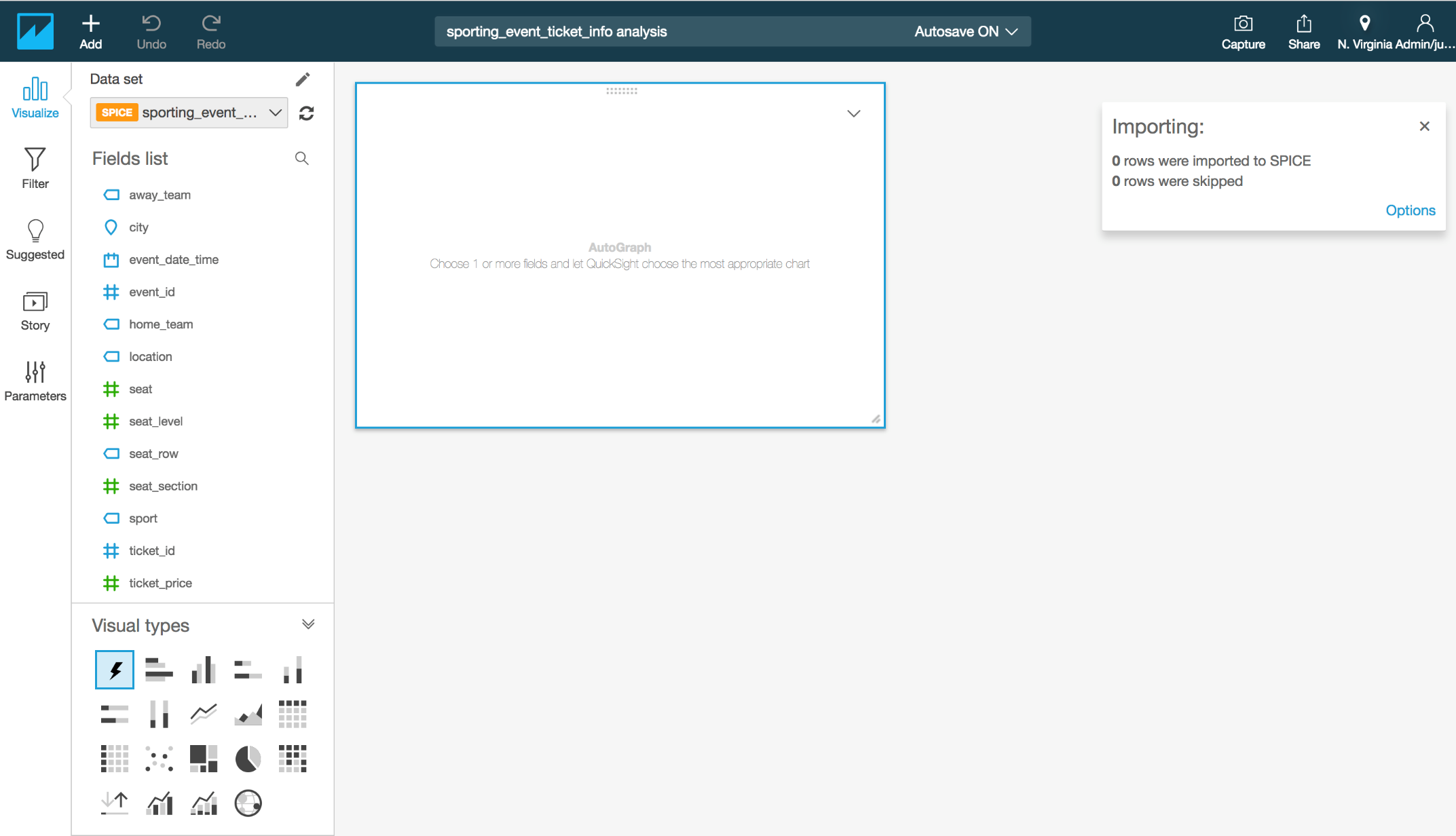
1. In the Database drop-down list, select the database name you created in the AWS Glue lab.
2. Choose the "sporting\_event\_ticket\_info" table and click **Select**.



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



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

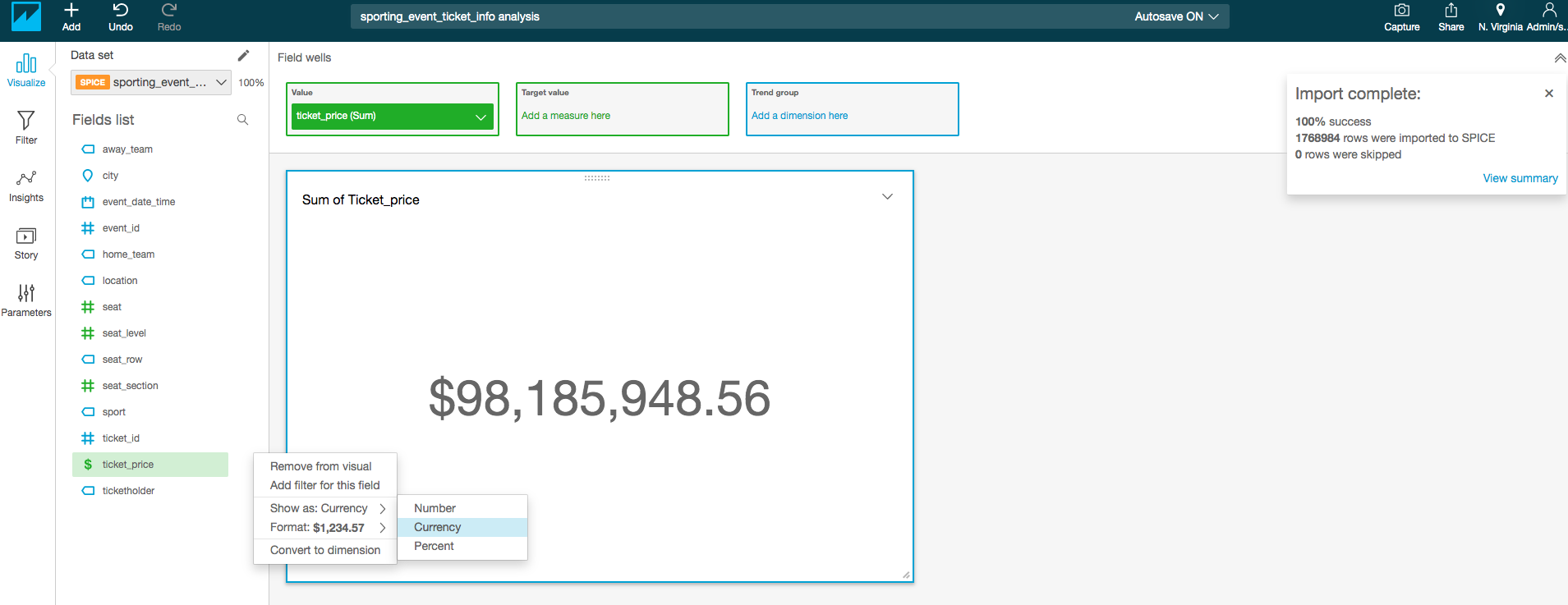


**Note**: The SPICE dataset will take a few minutes to be built, but you can continue to create some charts on the underlying data.

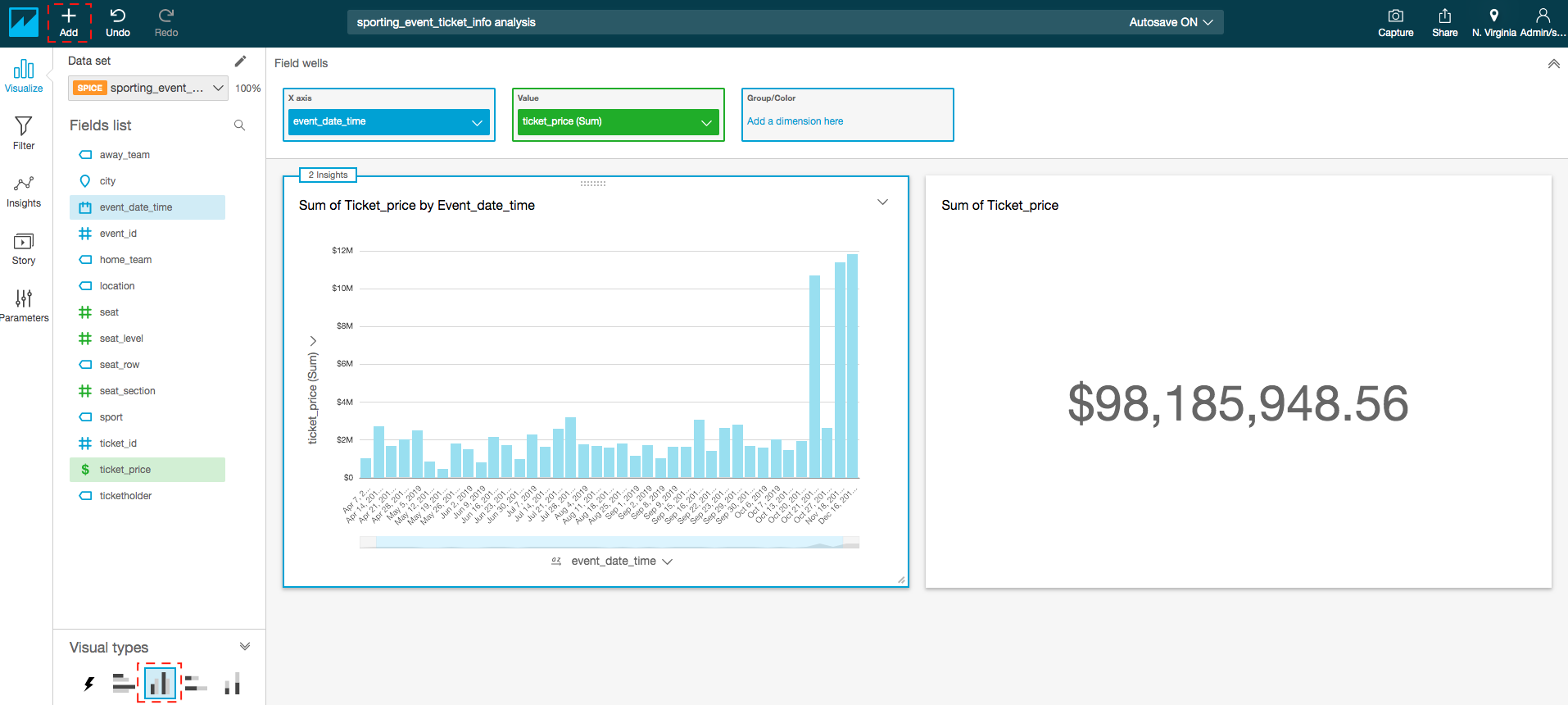
## Create QuickSight Charts

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

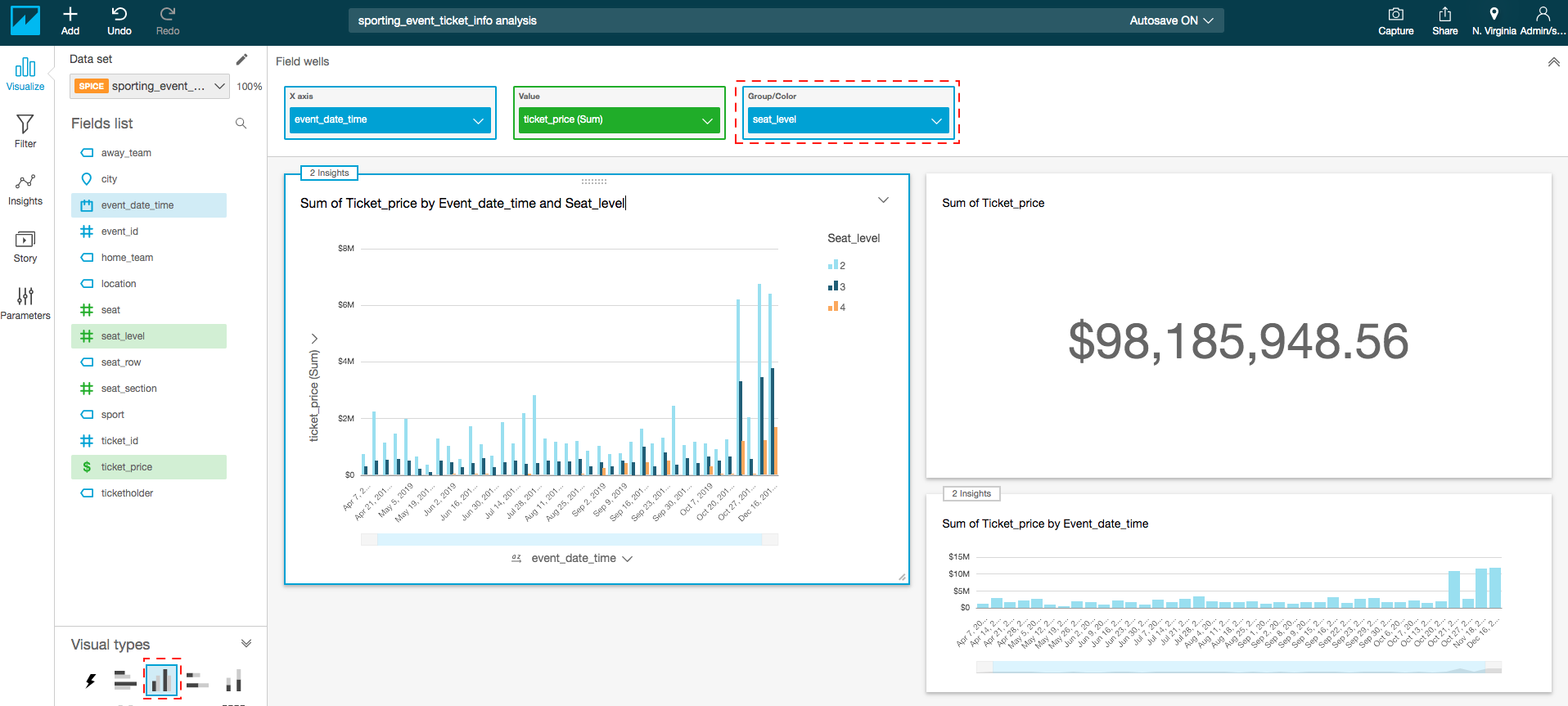
1. In the Fields list, click the "ticket\_price" column to populate the chart.
2. Click the expand icon in corner of "ticket\_price" field and select format as currency to show numbers in dollar amount.



1. 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. Click the "event\_date\_time" field and you should see the visualization update.

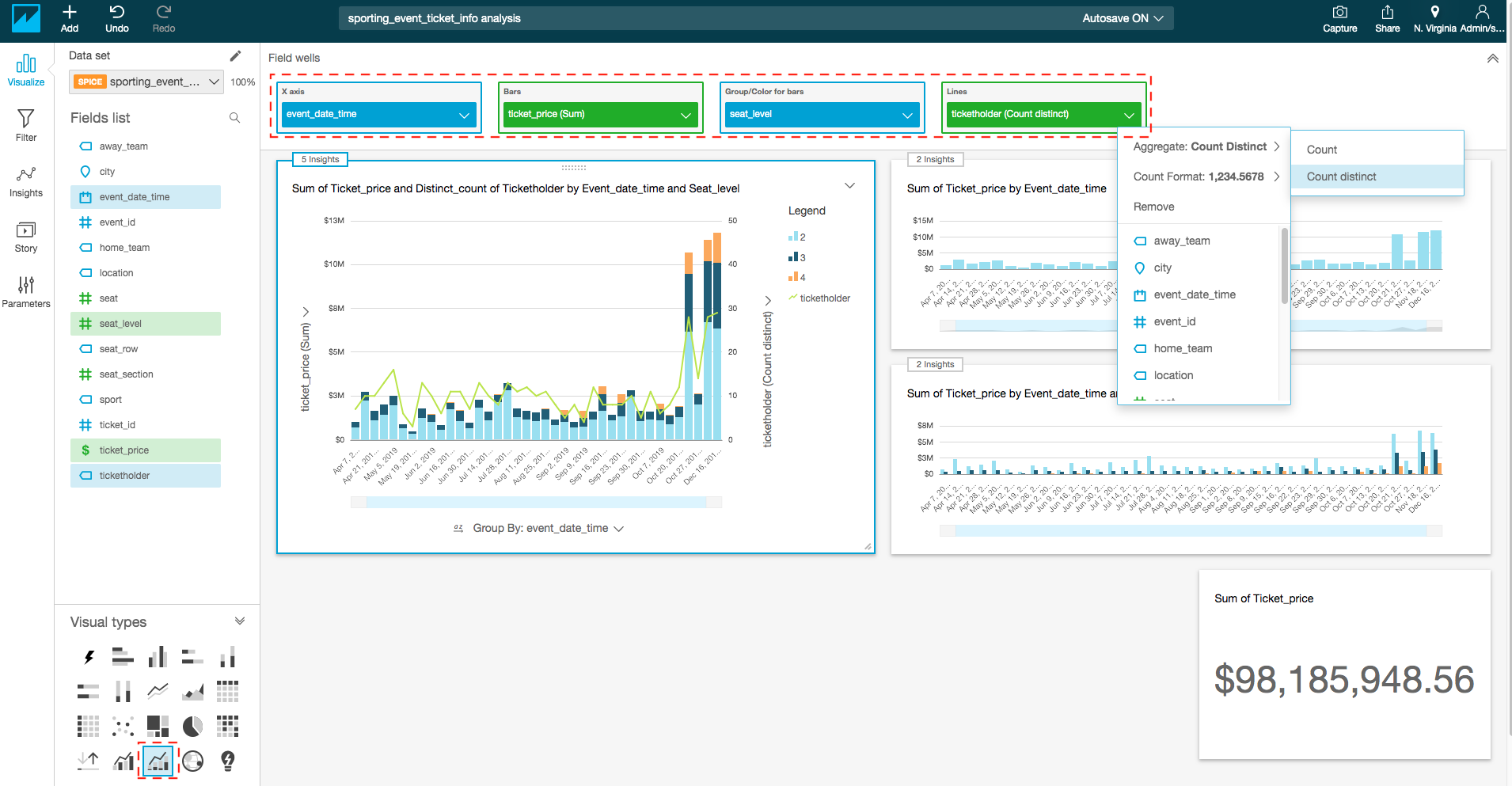


1. Add new Visual and you can drag and move other visulas to adjust space in dashboard. In the Fields list, click and drag the **seat\_level** field to the Group/Color box in the Field wells pane. You can also use the slider below the x axis to fit all of the data.

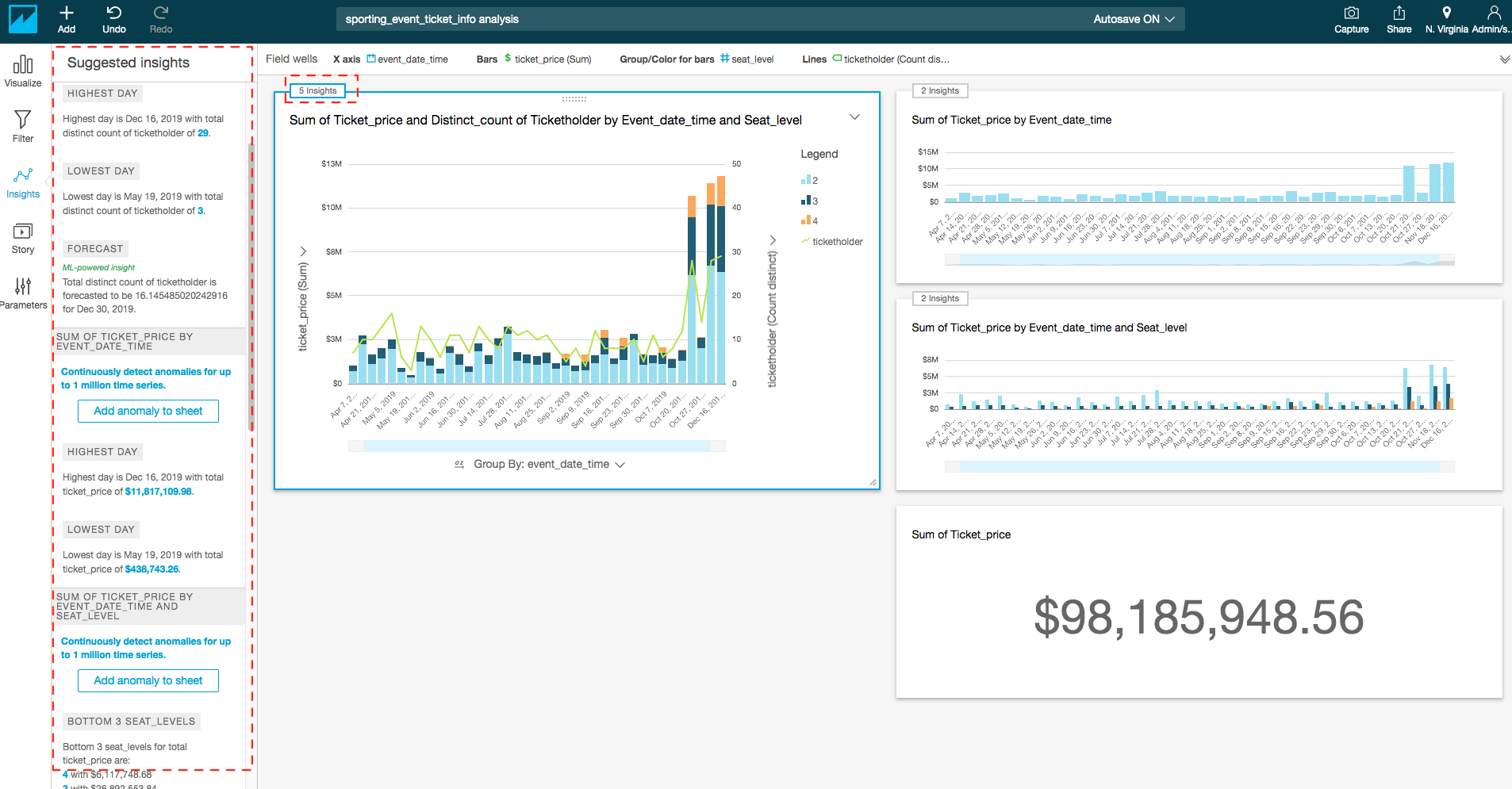


Let’s build on this one step further by changing the chart type to "Clustered bar combo chart" and adding in the ticketholder for the Lines.

1. In the Visual types area, choose the Clustered bar combo chart icon.
2. In the Fields list, click and drag the **ticketholder** field to the Lines box in the Field wells pane.
3. 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.



1. Click on insight icon on top of each chart and explore insight information in right hand pan in simple English.

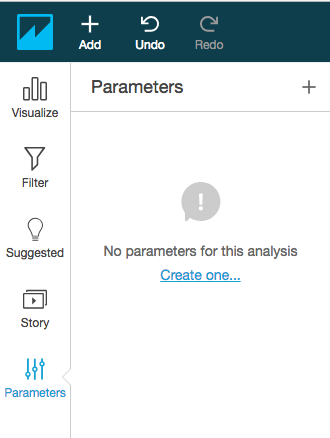


Feel free to experiment with other chart types and different fields to get a sense of the data.

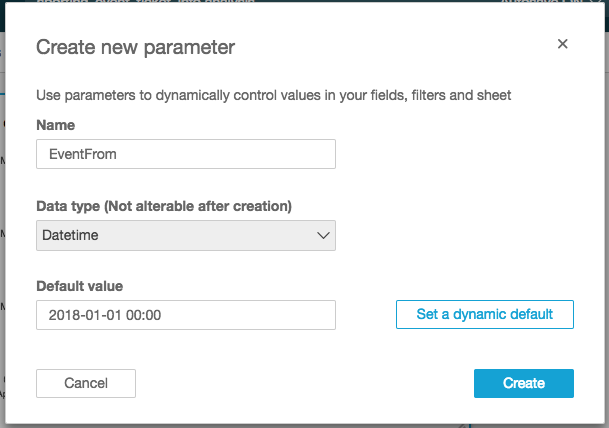
## Create QuickSight Parameters

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.

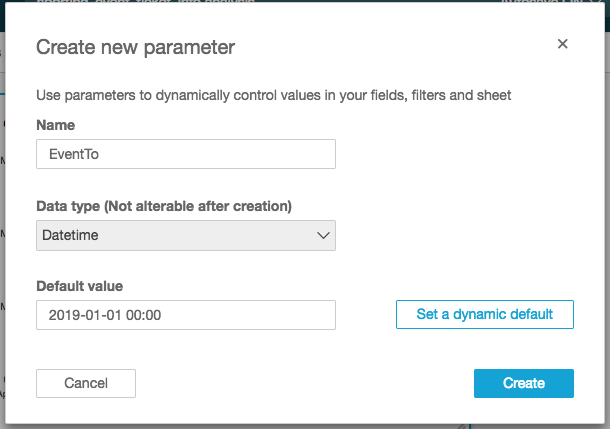
1. In the left navigation menu, select **Parameters**.



1. Click **Create one** to create a new parameter with a Name.
2. For Name, type **EventFrom**.
3. For Data type, choose **Datetime**.
4. For Default value, select the value from calendar as start date available in your graph for event\_date\_time. For example, **2018-01-01 00:00**.
5. Click **Create**, and then close the Parameter Added dialog box.

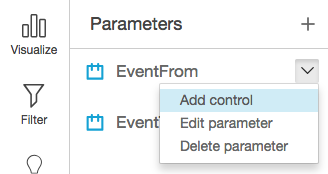


1. Create another parameter with the following attributes:
   1. Name: EventTo
   2. Data type: Datetime
   3. For Default value, select the value from calendar as end date available in your graph for event\_date\_time. For example, 2019-01-01 00:00

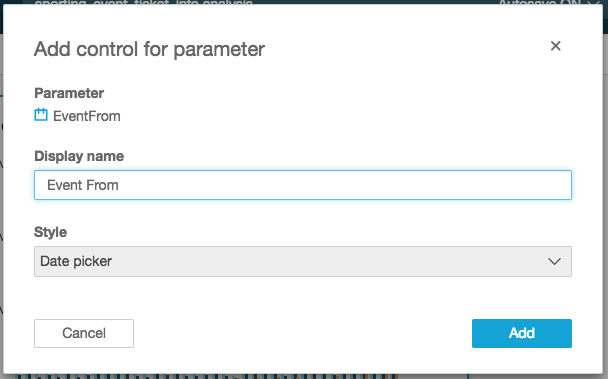


1

1. Click **Create.**
2. In theParameter Added dialog box, click **Filter** and then click **Close**.
3. Click the drop-down menu for the EventFrom parameter and choose **Add control**.

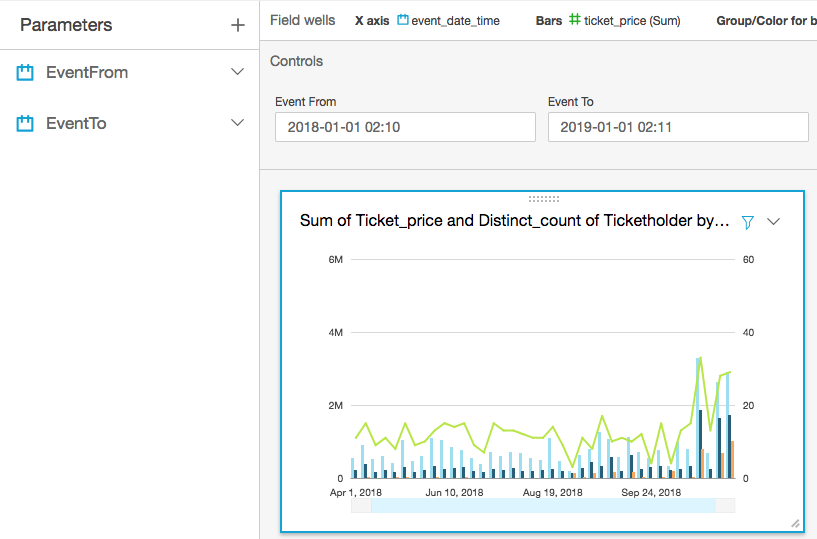


1. For Display name, specify **Event From** and click **Add**.



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

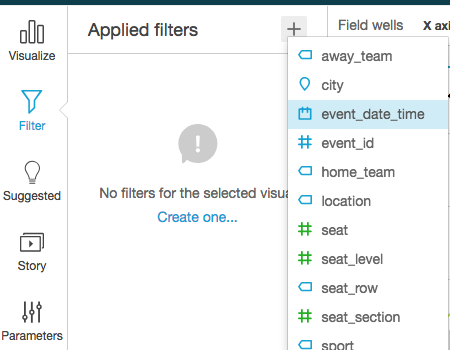
You should now be able to see and expand the Controls section above the chart.



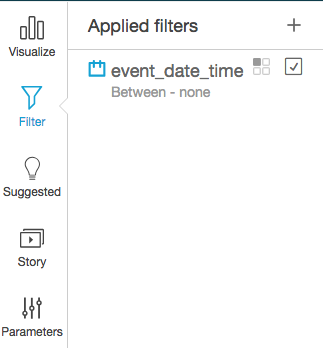
## Create a QuickSight Filter

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

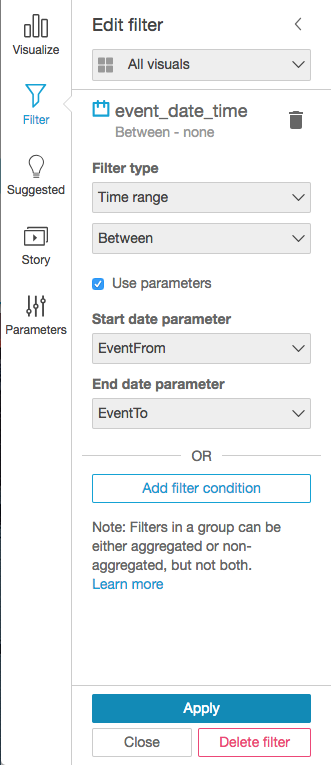
1. In the left navigation menu, choose Filter.
2. Click the plus icon (+) to add a filter for the field "event\_date\_time".



1. Click this filter to edit the properties.



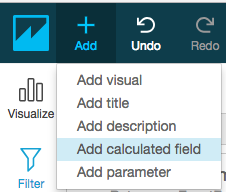
1. Choose to make this filter apply to **All visuals**.
2. For Filter type, choose **Time range** and **Between**.
3. Select option Use **Parameter.**
4. For Start date parameter, choose **EventFrom.**
5. For **End date parameter,** choose **EventTo**.
6. Click **Apply**.



## 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.

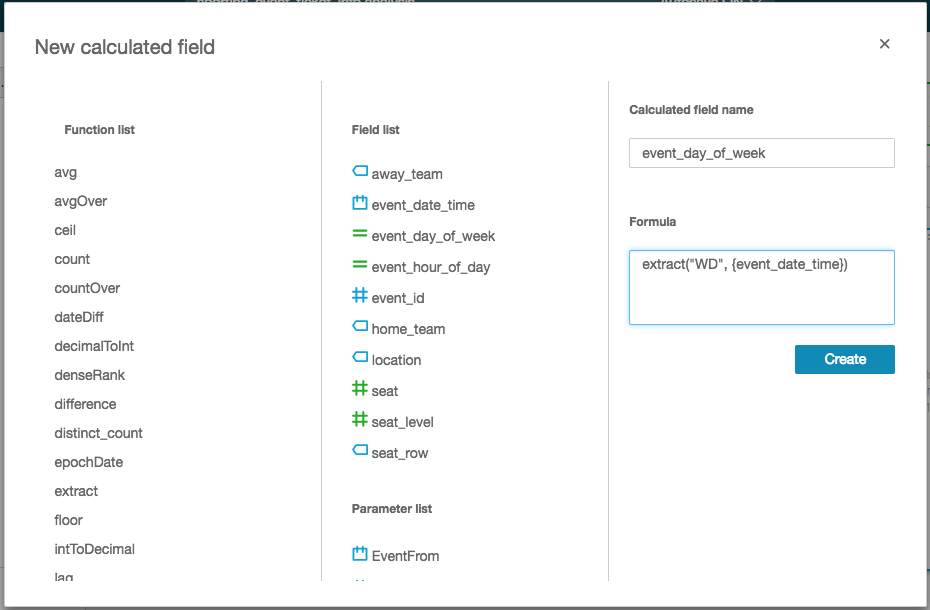
1. Click the Add button on the top left and select **Add a calculated field**.



1. For **Calculated field name** type “event\_day\_of\_week".
2. For Formula, type extract("WD", {event\_date\_time})

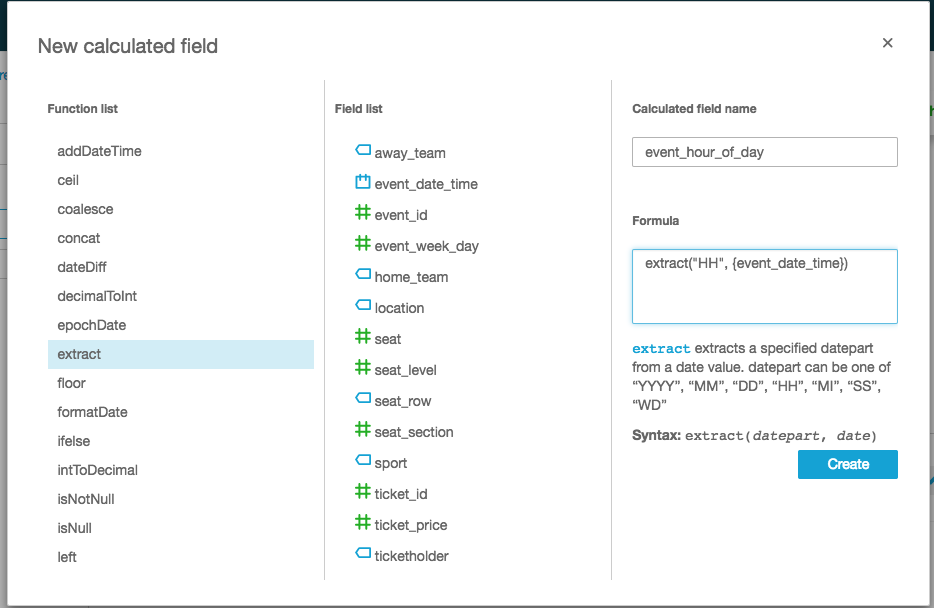
Note: extract returns a specified portion of a date value. Requesting a time-related portion of a date that doesn't contain time information returns 0. WD: This returns the day of the week as an integer, with Sunday as 1.

1. Click **Create**.

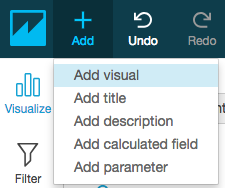


1. Add another calculated field with the following attributes:
   1. Calculated field name: "event\_hour\_of\_day"
   2. Formula: extract("HH", {event\_date\_time})

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



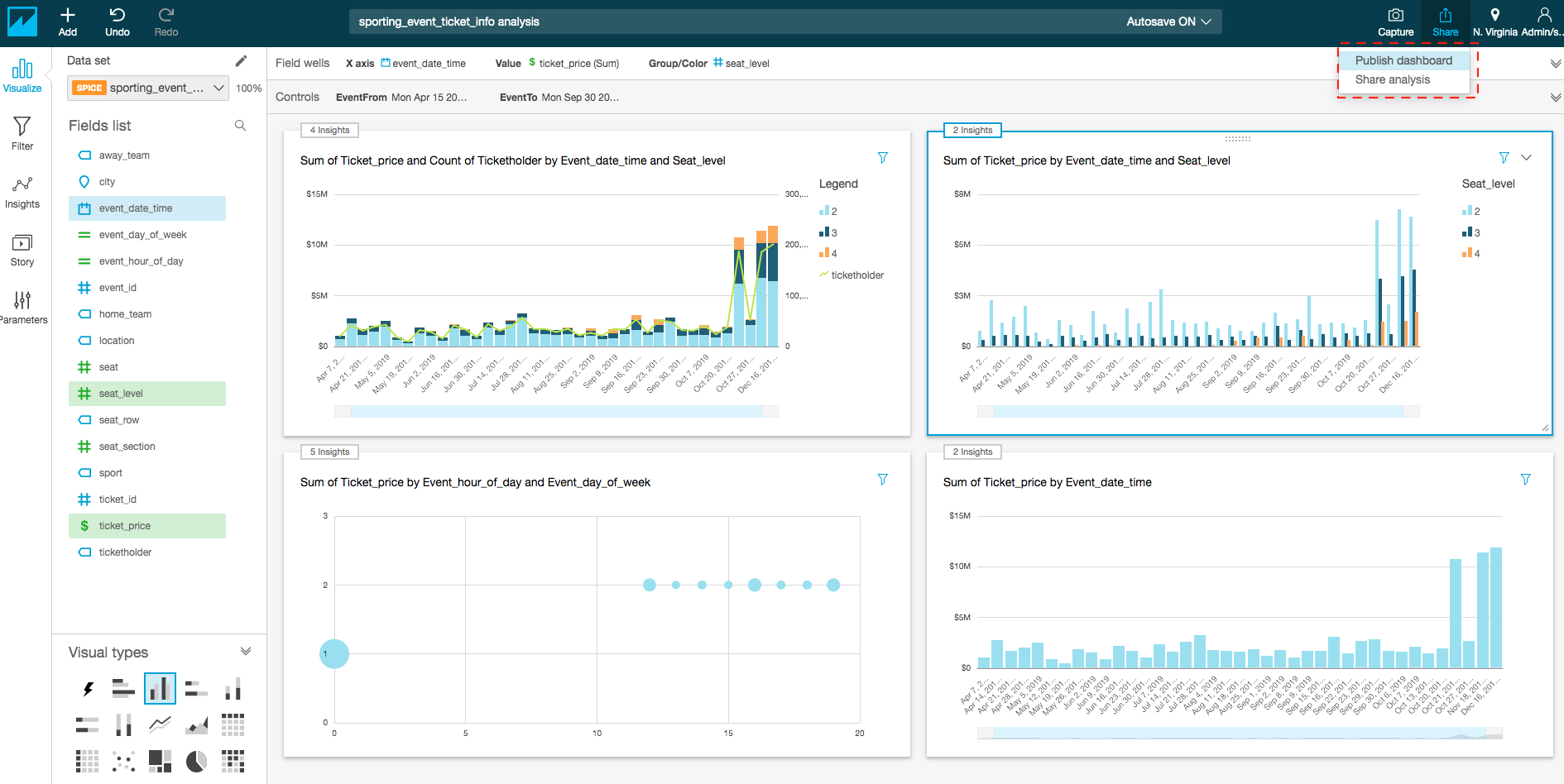
1. Click Add button in the top left and choose **Add visual**.



1. For field type, select the scatter plot.
2. In the Fields list, select and drag the following attributes to the Field wells pane to set the graph attributes:
   1. X-axis: "event\_hour\_of\_day"
   2. Y-axis: "event\_day\_of\_week"
   3. Size: "ticket\_price"



Since now you have completed your dashboard then you can publish it by clicking on top right corner of screen.



A *dashboard* is a read-only snapshot of an analysis that you can share with other Amazon QuickSight users for reporting purposes. In 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.

# Amazon QuickSight ML-Insights (Optional)

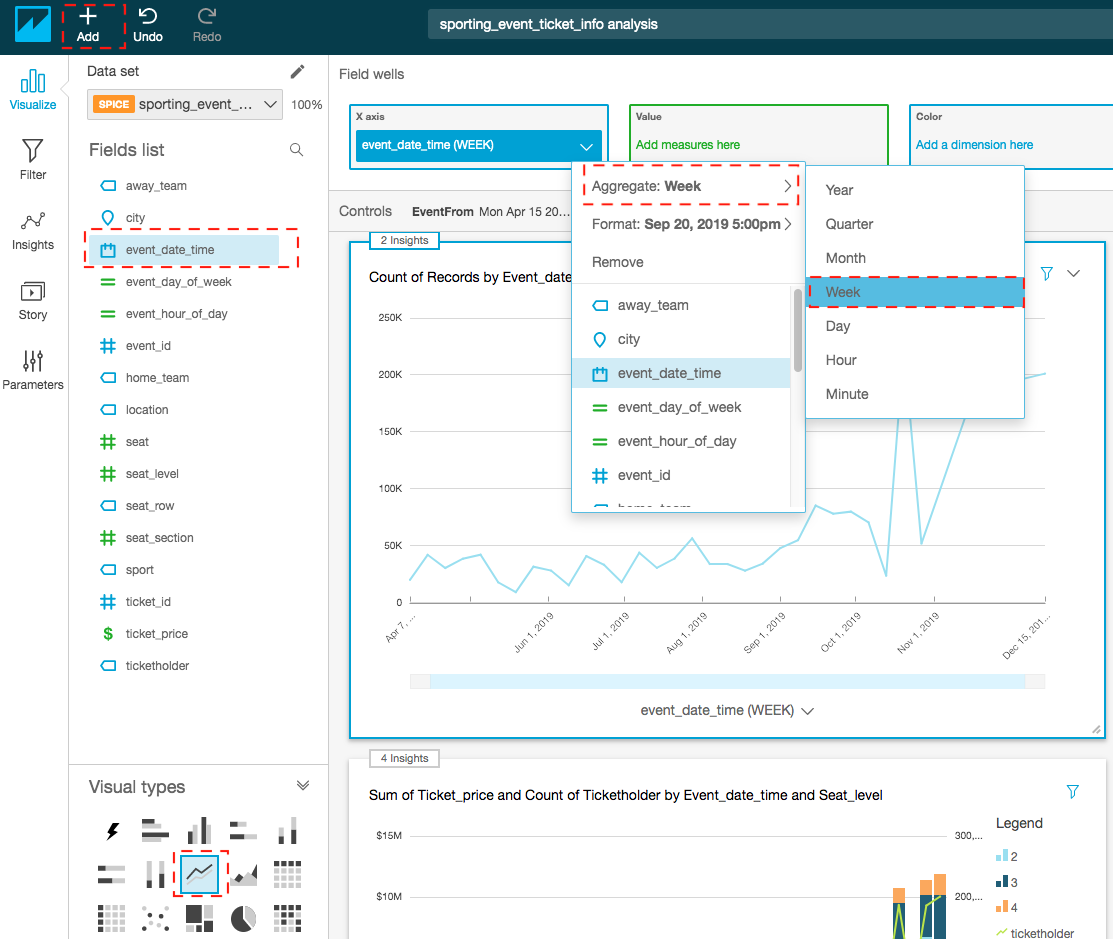
With Amazon QuickSight, you can add Machine Learning capabilities to your visuals, easily, with one click action. There are 3 types of Machine Learning Insights

* Narrative
* Anomaly Detection
* Forecasting

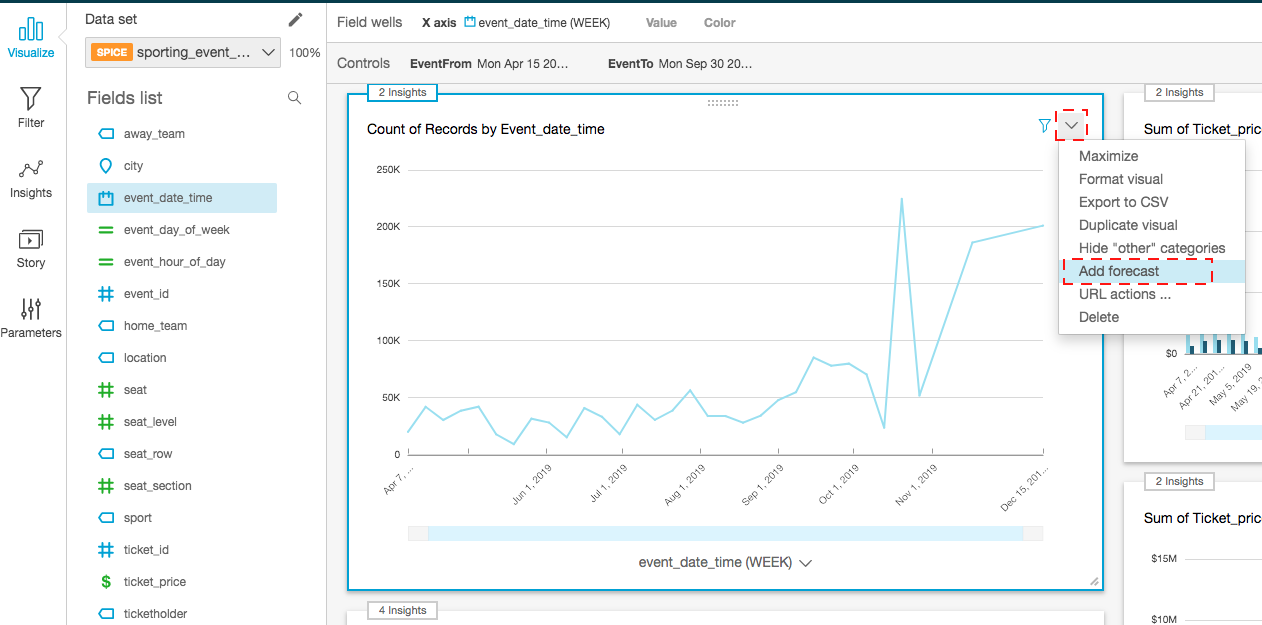
ML-Insights is only available to enterprise version of QuickSight. You will need to upgrade to Enterprise Edition before you start with the task. To upgrade your Amazon QuickSight Subscription from Standard Edition to Enterprise Edition please follow this guide <https://docs.aws.amazon.com/quicksight/latest/user/upgrading-subscription.html>

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.

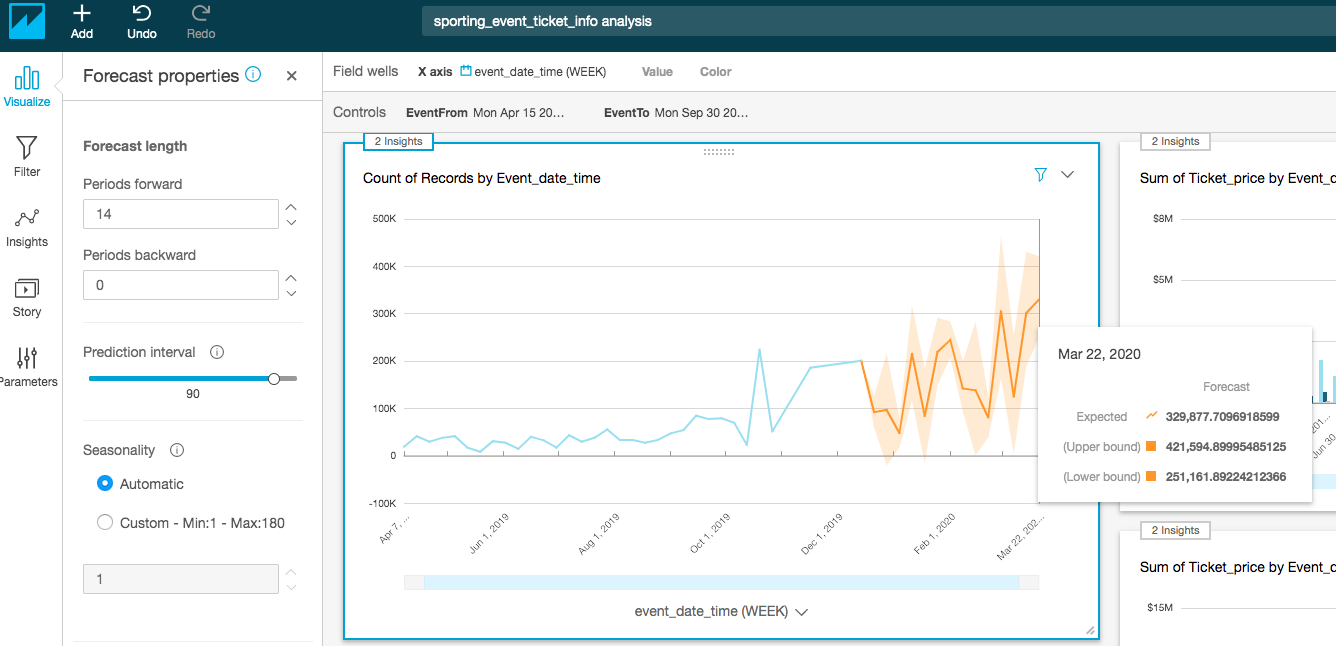
1. Click **add Visual at top left corner of screen**, and select **Line Chart** and add the **event\_date\_time** as the **x-axis** and **aggregate by week**. As shown in below screenshot



1. 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.