**windowSum:**

The windowSum function calculates the sum of the aggregated measure in a custom window that is partitioned and sorted by specified attributes. Usually, you use custom window functions on a time series, where your visual shows a metric and a date field.

Syntax:

windowSum

(

measure

, [sort\_order\_field ASC/DESC, ...]

, start\_index

, end\_index

,[ partition\_field, ... ]

)

Days wise:

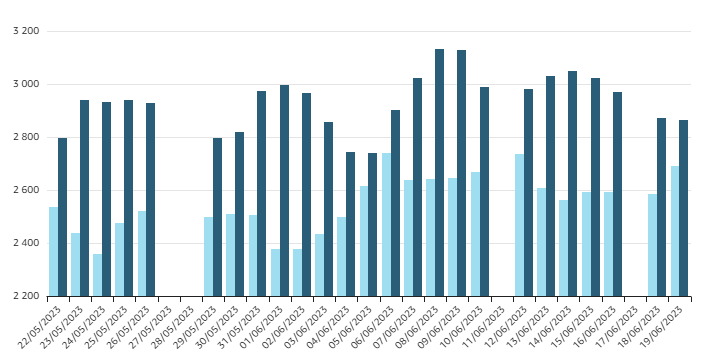
Example:

windowSum(sum(totalAmount),[orderCreated ASC],30,0)

Example:

windowSum(sum(ifelse(dateDiff(orderCreated, now(),'DD')<30, totalAmount,0),[orderCreated ASC],30,0)

output:



Example:

windowSum(sum(ifelse(dateDiff(orderCreated,AsofDate,'DD')<30, totalAmount,0)),[orderCreated ASC],30,0)

windowsum-rows wise:

The following example calculates the moving sum of sum(Revenue), sorted by SaleDate. The calculation includes two rows above and one row ahead of the current row.

Example:

windowSum

(

sum(Revenue),

[SaleDate ASC],

2,

1

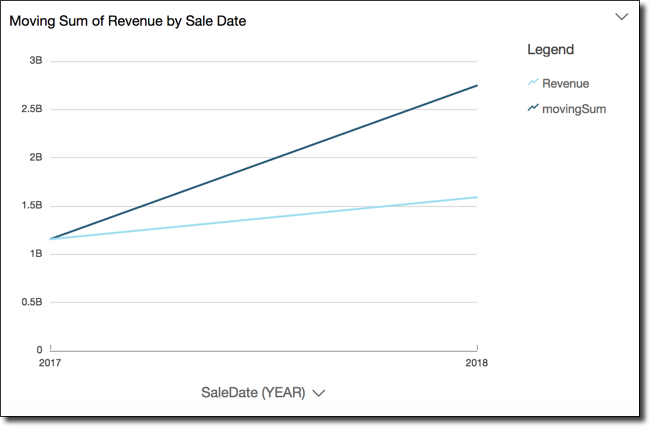
)

Windowsum\_month wise:

The following example show a trailing 12-month sum.

Example:

windowSum(sum(Revenue),[SaleDate ASC],12,0)

The following screenshot shows the results of this trailing 12-month sum example. The sum(Revenue) field is added to the chart to show the difference between the revenue and the trailing 12-month sum of revenue.

**Functionality:** The SUM() window function calculates the running or cumulative sum of values within a specified column across a defined window. In this query: The PARTITION BY clause divides the data into partitions based on user\_id . SUM() calculates the cumulative sum of spend within each user\_id partition.

**Adding drill-downs to visual data in Amazon QuickSight:**

**As you drill down each level, the data displayed is refined by the value in the field you drill down on. For example, if you drill down on the state of California, you see data on all of the cities in California.**

**Adding a drill-down**

**Use the following procedure to add drill-down levels to a visual.**

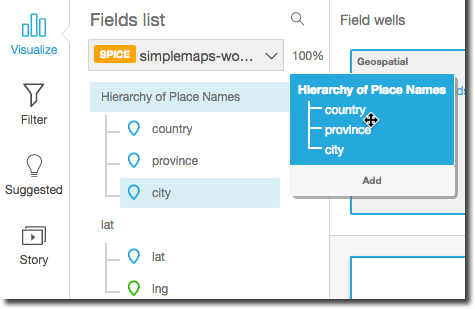
**To add drill-down levels to a visual**

1. **On the analysis page, choose the visual that you want to add drill-downs to.**

**Note**

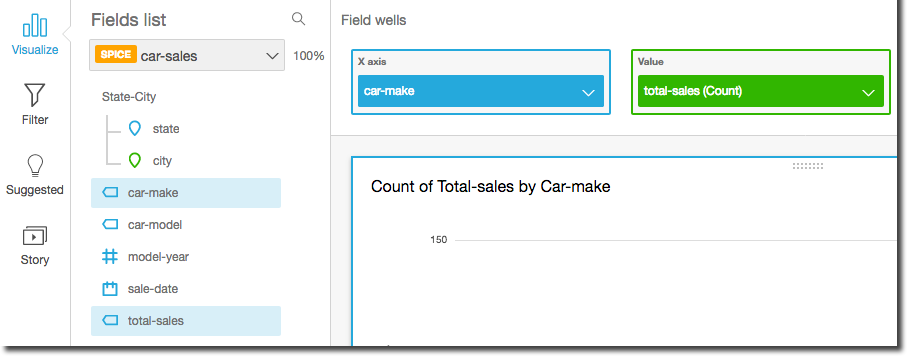
**You can't add drill-downs to pivot tables.**

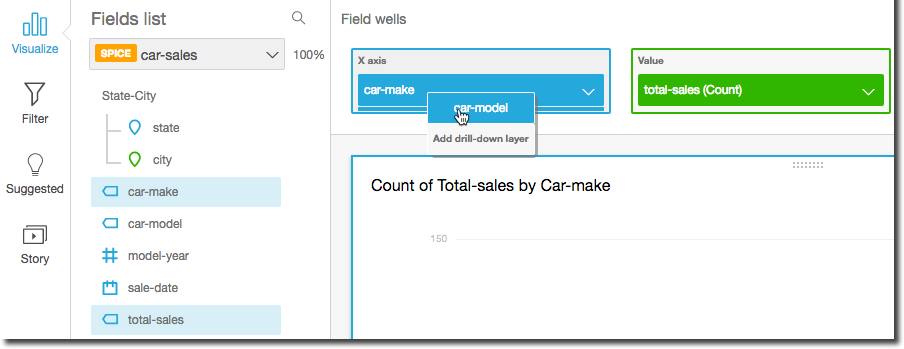
1. **Drag a field item into a Field well.**
2. **If your dataset has a defined hierarchy, you can drag the entire hierarchy into the field well as one. An example is geospatial or coordinate data. In this case, you don't need to follow the remaining steps.**

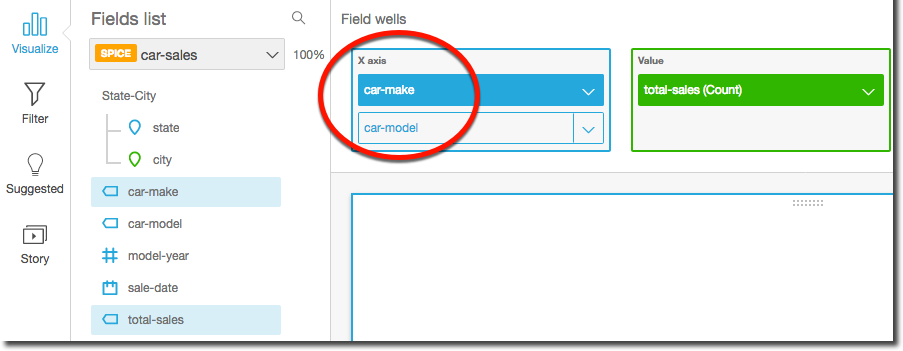
****

**If you don't have a predefined hierarchy, you can create one in your analysis, as described in the remaining steps.**

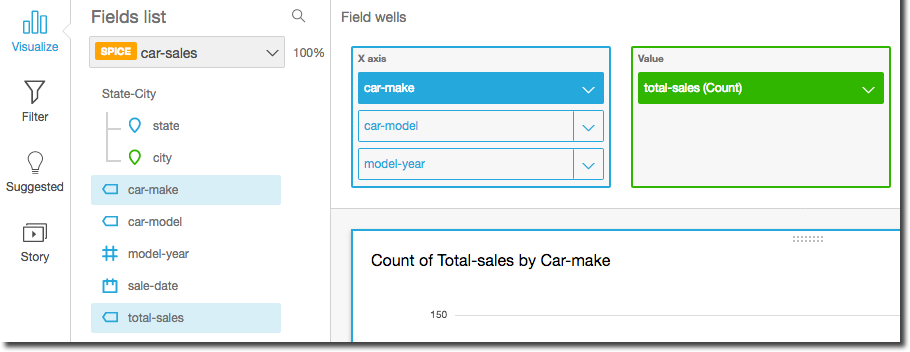
1. **Drag a field that you want to use in the drill-down hierarchy to an appropriate field well, depending on the visual type. Make sure that the label for the dragged field says Add drill-down layer. Position the dragged field above or below the existing field based on where you want it to be in the hierarchy you're creating.**

****

****

****

1. **Continue until you have added all of the levels of hierarchy that you want. To remove a field from the hierarchy, choose the field, and then choose Remove.**

****

1. **To drill down or up to see data at a different level of the hierarchy, choose an element on the visual (like a line or bar), and then choose Drill down to <lower level> or Drill up to <higher level>. In this example, from the car-make level you can drill down to car-model to see data at that level. If you drill down to car-model from the Ford car-make, you see only car-models in that car-make.**

**After you drill down to the car-model level, you can then drill down further to see make-year data, or go back up to car-make. If you drill down to make-year from the bar representing Ranger, you see only years for that model of car.**

Window sum() drilldown:

**SUM()**

**The following query uses the SUM() window function to total the amount of sales for each dealer in Q1. The word sum is a reserved keyword in Drill and must be enclosed in back ticks (``).**

**Example:**

**select dealer\_id, emp\_name, sales, sum(sales) over(partition by dealer\_id) as `sum` from q1\_sales;**

**output:**

**|------------|-----------------|--------|--------|**

**| dealer\_id | emp\_name | sales | sum |**

**|------------|-----------------|--------|--------|**

**| 1 | Ferris Brown | 19745 | 57427 |**

**| 1 | Noel Meyer | 19745 | 57427 |**

**| 1 | Raphael Hull | 8227 | 57427 |**

**| 1 | Jack Salazar | 9710 | 57427 |**

**| 2 | Beverly Lang | 16233 | 41774 |**

**| 2 | Kameko French | 16233 | 41774 |**

**| 2 | Haviva Montoya | 9308 | 41774 |**

**| 3 | Ursa George | 15427 | 37104 |**

**| 3 | Abel Kim | 12369 | 37104 |**

**| 3 | May Stout | 9308 | 37104 |**

**|------------|-----------------|--------|--------|**

**10 rows selected (0.198 seconds)**

**SUM("Qty") OVER (PARTITION BY "article", date\_trunc('month',"Date")**

**ORDER BY "Date")**

Example:

**SELECT DISTINCT date\_trunc('day', rental\_date), count(rental\_id) OVER w**

**FROM rental**

**GROUP BY rental\_date, rental\_id**

**WINDOW w AS (PARTITION BY (rental\_date BETWEEN DATE '2005-08-23' - INTERVAL '44days' AND DATE '2005-08-23')**

**ORDER BY rental\_date ROWS BETWEEN 7 PRECEDING AND CURRENT ROW)**

**ORDER BY date\_trunc('day', rental\_date) DESC;**

**Output:**

**Col1 Col2**

**date\_trunc1 count(rental\_id)**

**2006-02-21 00:00:00 182**

**2006-02-20 00:00:00 182**

**2006-02-19 00:00:00 182**

**2006-02-18 00:00:00 182**

**2006-02-17 00:00:00 182**

**2006-02-16 00:00:00 182**

**2006-02-15 00:00:00 182**

**2005-08-30 00:00:00 598**

**2005-08-29 00:00:00 1224**

**2005-08-28 00:00:00 1883**

**2005-08-27 00:00:00 2507**

**2005-08-26 00:00:00 3135**

**2005-08-25 00:00:00 3756**

**2005-08-24 00:00:00 4349**

**2005-08-23 00:00:00 3374**

**2005-08-22 00:00:00 3148**

**2005-08-21 00:00:00 2489**

**2005-08-20 00:00:00 1865**

**2005-08-19 00:00:00 1237**

**2005-08-18 00:00:00 616**

**2005-08-17 00:00:00 23**

**2005-08-16 00:00:00 0**

**2005-08-08 00:00:00 671**

**2005-08-07 00:00:00 1305**