Introduction to Dashboards - Lab Exercises

Overview

Welcome to the Splunk Education lab environment. These lab exercises will guide you through the process of creating basic dashboards in Splunk's Dashboard Studio.

Scenario

You will use data from a chain of video gaming cafes named Buttercup Games Cafe. Customers can purchase video games to play, coffee drinks and pastries using a mobile device, standard computer or the cafe's internal point-of-sale system.

> **IMPORTANT:** Save dashboards you create to the Introduction to Dashboards app with permissions set to Private. If you copy text from this document, please note that character formatting and artifacts created by the PDF generation process can cause errors in the XML. Consider using a text editor as an interim step.

Typographical Conventions

- **Blue** text indicates text to **add**
- **Red** text indicates text to **remove**
- **Grey** text provides context for edits

Lab Connection Info

Access labs using the server URL, user name and password shown in your lab environment.

SERVERS	LAB DOCUMENT	CHECK MY WORK	HELP

Lab Server Info:

SERVER URL	PUBLIC IP	SPLUNK USER NAME	PASSWORD	DOWNLOAD	STATUS
https://11-195-15-aio.class.splunk.com	3.23.114.109	powerUser		link	DEPLOYED

Source Types

The source types used in these exercises are referred to by the type of data they represent.

Type	Index	Source type	Interesting Fields
Cafe Food	cafefood	access_combined_cf	action, bytes, categoryId, clientip, itemId, JSESSIONID, price_large, price_med, product_name, productId, referer, referer_domain, roast, status, user, useragent
Cafe Games	cafegames	access_combined_cg	action, bytes, clientip, JSESSIONID, player1name, player1score, player2name, player2score, price, product_name, productId, referer, referer_domain, sale_price, status, user, useragent

splunk>

Lab Exercise 1 – Convert a Dashboard

Description

When converting a classic dashboard to the Splunk Dashboard Studio, it is important to remember what will convert and what will not. For example, the dashboard studio does not support radio inputs. These are converted to drop-down inputs. In the following tasks, you clone a dashboard to the Dashboard Studio, then name the search queries used on the dashboard and adjust visualization placement.

Scenario: **The Buttercup Cafe sales team wants a classic simple xml dashboard updated to the dashboard studio.**

Steps

Task 1: Change the account name and time zone.

Set up your lab environment to fit your time zone and the app you will be working in. This also allows the instructor to track your progress and assist you if necessary.

1. Navigate to **User Menu** > **Account Settings**.
2. In the Full name box, enter your name: `<Firstname Lastname>`
   For example: Mitch Fleischman
3. Click **Save** and reload your browser.
4. Navigate to **User Menu > Preferences.**
5. Enter the following settings:
   - Time zone: `<your local time zone>`
   - Default application: Introduction to Dashboards
6. Click **Apply**.
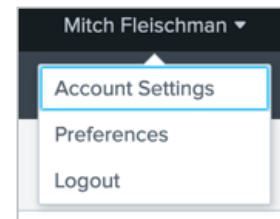
Task 2: Clone a dashboard into Dashboard Studio.

7. Click the **Splunk logo** in the upper left to go to the Introduction to Dashboards app.

NOTE: Since your default application is now Introduction to Dashboards, clicking the Splunk logo is the same as navigating to Apps > Introduction to Dashboards.

8. Click **Dashboards**.
9. Click **Lab 1: Game Sales – Classic**.
10. Test the radio button input by selecting a game.
11. Click **Hide Filters** to display only the visualizations
12. Click the **More…** button on the upper right.
13. Select **Clone in Dashboard Studio**.
14. In the Clone in Dashboard Studio modal window, delete the words "- Classic Copy" from the title.
    For example: Lab 1: Game Sales
15. Set permissions to: Private
16. Click **Convert & Save**.

17. Examine the Configuration panel. Notice the Display Mode is set to Fit to Width and the zoom level is not 100%.
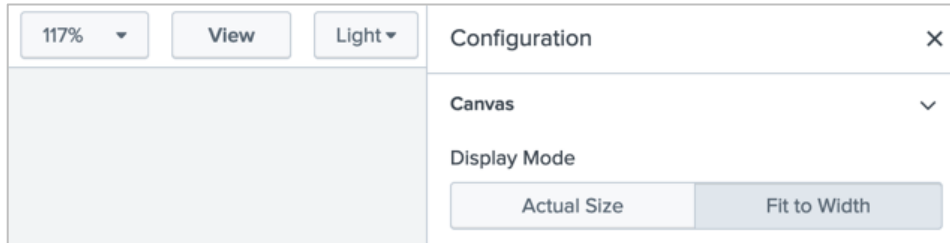
18. Set the Display Mode to Actual Size and notice the zoom level is now 100%.

19. Examine the dashboard. Notice the radio button input has been converted to a drop-down input. Test the input. Adjust visualization sizes and positions.

Task 3: Name search queries.

20. Click the **Games Purchased – Last 7 days** chart.

21. On the Configuration side panel, locate the Data Configurations section.

22. Click the **pencil** icon beside Unnamed.
    The Edit Data Source side panel appears.
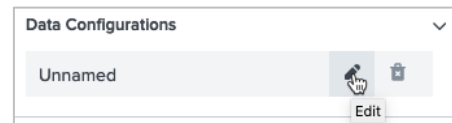
23. Change the Data Source Name from Unnamed to: `Games Purchased`

24. Click **Run & Save**.

25. Repeat steps above for **Total Sales – Last 7 days** chart and name its Data Source: `Total Sales`

26. Click the **All Game Sales – Last 7 days** visualization.

27. On the Configuration side panel, locate the Data Configurations section.

28. Click the **pencil** icon beside BCG_CafeGames.

29. Notice you cannot edit the Data Source Name. This is because the data source is a report (saved search) that your account does not have permission to edit.

    If you needed to edit the query or the report name, you could clone it to gain ownership and then make changes. Or you could contact the owner and have them make the changes.

30. Click **Save** to save the dashboard.

31. Click **View**.

32. Test the drop-down menu. Changes should update the two column chart visualizations.

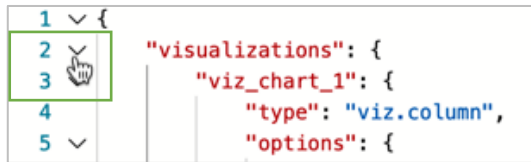33. Adjust the magnification level.
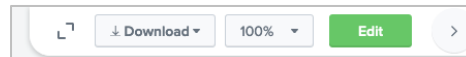
34. Click **Edit**.

35. Click the **Source icon**.

36. Examine the source code.

37. Move your mouse over the number on the left and notice the down arrows appear.

38. Click the down arrow beside the `visualizations` section.
    Notice the source code in that section is hidden, or *folded*, under the section title.

39. Fold the source code under the remaining dashboard definition sections: `dataSources`, `defaults`, `inputs`, and `layout`.

```
1 ⌄ {
2 ⌄        "visualizations": {
3            "viz_chart_1": {
4                "type": "viz.column",
5 ⌄            "options": {
```

Lab 1: Game Sales

```
1    {
2 >      "visualizations": {⋯
76       },
77 >     "dataSources": {⋯
108      },
109 >    "defaults": {⋯
117      },
118 >    "inputs": {⋯
173      },
174 >    "layout": {⋯
215      },
216      "description": "",
217      "title": "Lab 1: Game Sales"
218  }
```

40. Click **Back** to return to the visual editor.

41. Click **View**.

Example:

Lab Exercise 2 – Create a Prototype

Description

When you create a dashboard, it is best to first create a prototype, have that reviewed by stakeholders; then, build something more elaborate. In this lab exercise you will create a prototype for the Buttercup Cafe sales team.

Scenario: **The sales team wants a dashboard that shows cafe sales. They want to see when sales are at their highest and how customers are placing their orders, in person or using the cafe's app.**


Steps

Task 1: Create a basic dashboard.

1. Navigate to the dashboards page and click **Create New Dashboard**.
2. In the Dashboard Title box enter: `Lab 2: Cafe Sales`
3. Click **Dashboard Studio**.
4. Set permissions to **Private**.
5. Select **Grid Layout**.
6. Click **Create**.
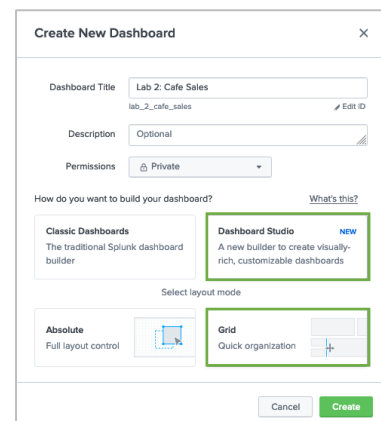
Task 2: Add a single value visualization.

In this prototype you'll use the `makeresults` command to generate sample search results in temporary memory. In later lab exercises you'll use live data.

7. On the new dashboard, click the **Add Chart** icon and select **Single Value** near the bottom.
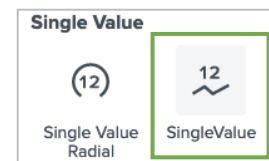
8. On the New Data Source side panel, locate the Data Source Name box and enter: `Customers`
9. In the Search with SPL box enter:

    `| makeresults count=12 | streamstats count`

10. Click **Run & Save**.
11. On the Configuration side panel, locate the Visualizations Options section.
12. In the Title box, enter: `Customers`
13. Scroll down to the Major Value & Trend section and in the Trend Display menu select **Percent**.
14. Save the dashboard.

Introduction to Dashboards
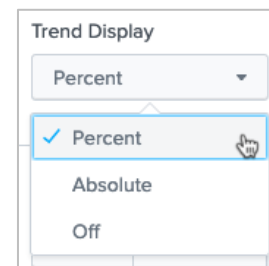
**Task 3: Add a column chart.**

15. Click the **Add Chart** icon and select **Column**.

16. In the Data Source Name box enter: `Category Sales`

17. In the Search with SPL box enter:

```
| makeresults count=12 | streamstats count
| eval _time=_time-(count*3600)
| eval drip =(random () % 3) + 1
| eval espresso =(random () % 3) + 1
| eval baked_good =(random() % 4) + 1
| fields - count
```

18. Click **Run & Save**.

19. On the Configuration side panel, in the Title box, enter: `Category Sales`

20. Click and drag the column chart to position it on the right of the single value visualization.

21. With the column chart still selected, adjust its width to be twice as wide as the single value visualization by clicking and dragging its left edge.

22. Make the row height slightly taller by clicking and dragging the dotted line beneath the visualizations.
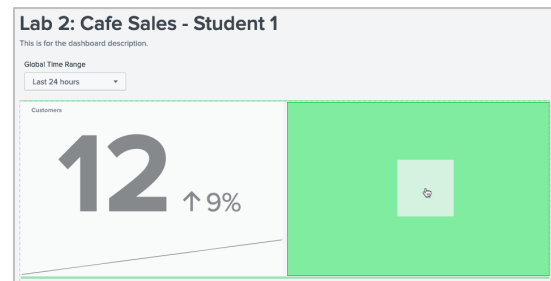
23. On the Configuration side panel, locate the Legend section.

24. Select **Bottom** from the drop-down menu.

25. Save the dashboard.

26. Click **View**.

Example:

Task 4: Clone a visualization.

27. Click **Edit**.
28. Click on the **Category Sales** visualization.
29. Click the **Clone** button on its Action panel.
30. On the Configuration side panel, in the Title box, enter: `Device Used for Purchase`
31. On the Configuration side panel, in the Data Configurations section, click the **pencil** icon beside Copy of Category Sales.
32. Rename the data source as: `Device Used for Purchase`
33. Replace the search query with:

```
| makeresults count=12
| streamstats count
| eval _time=_time-(count*3600)
| eval android =(random() % 3) + 1
| eval iOS =(random() % 3) + 1
| eval inCafe =(random() % 4) + 1
| fields - count
```

34. Click **Run & Save**.
35. Save the dashboard.
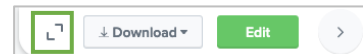
Example:
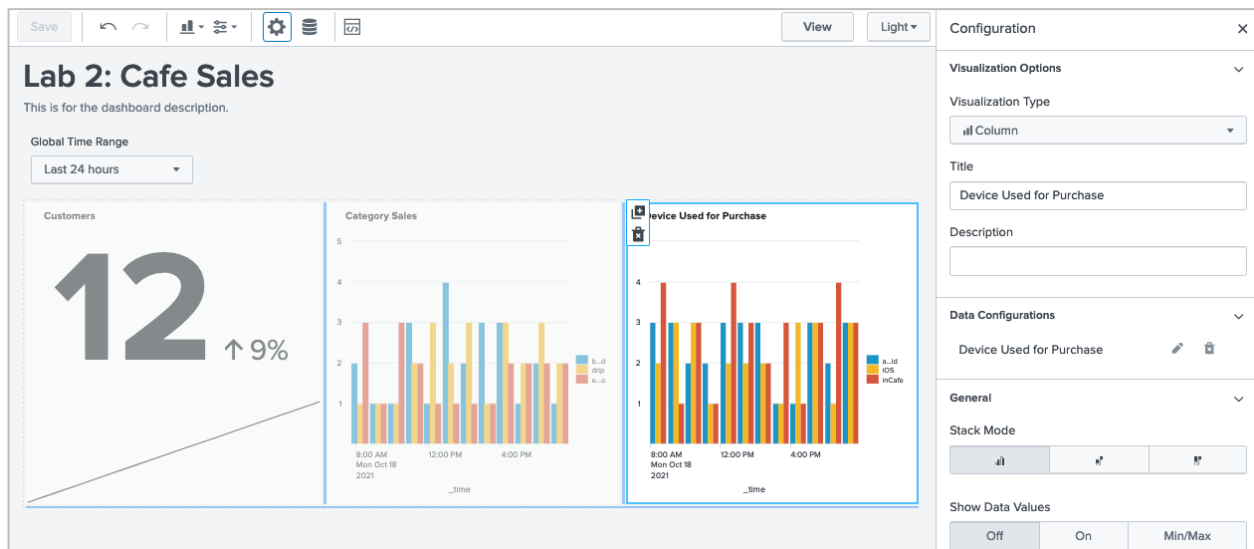
Task 5: Set the default time range.

36. Select the Global Time Range input.

37. On the Configuration side panel, in the Title box, enter: `Select a time range:`

38. On the Configuration side panel, in the Default Value dropdown menu, select **Last 7 days**.

Task 6: Add a table.

39. Click the **Add Chart** icon and select **Table** at the bottom of the list.
    Notice the table is added to a new row.

40. On the New Data Source side panel, in the Data Source Name box, enter: `Web Server Errors`

41. In the Search with SPL box enter:
```
| makeresults count=3 | streamstats count
| eval host = case(count=1, "www1", count=2, "www2", count=3, "www3", count=4, null())
| eval 406 = case(count=1, 25, count=2, 39, count=3, 31, count=4, null())
| eval 500 = case(count=1, 25, count=2, 39, count=3, 31, count=4, null())
| eval 503 = case(count=1, 25, count=2, 39, count=3, 31, count=4, null())
| table host, 406, 500, 503
```

42. Click **Run & Save**.

43. On the Configuration side panel, in the Title box, enter: `Web Server Errors`

44. Select the Device Used for Purchase column chart and drag it to the lower row, positioning it to **the left** of the Web Server Errors table.

45. Adjust the width of the column chart to be twice as wide as the table visualization by clicking and dragging its right edge.

46. Make both rows a similar height by clicking and dragging the dotted line beneath the visualizations line up or down.

47. Save the dashboard and refresh your browser.

48. Click **View**.

49. Expand the dashboard to full screen by clicking the **scale arrows**.

Example:

Introduction to Dashboards   October 18, 2021   8
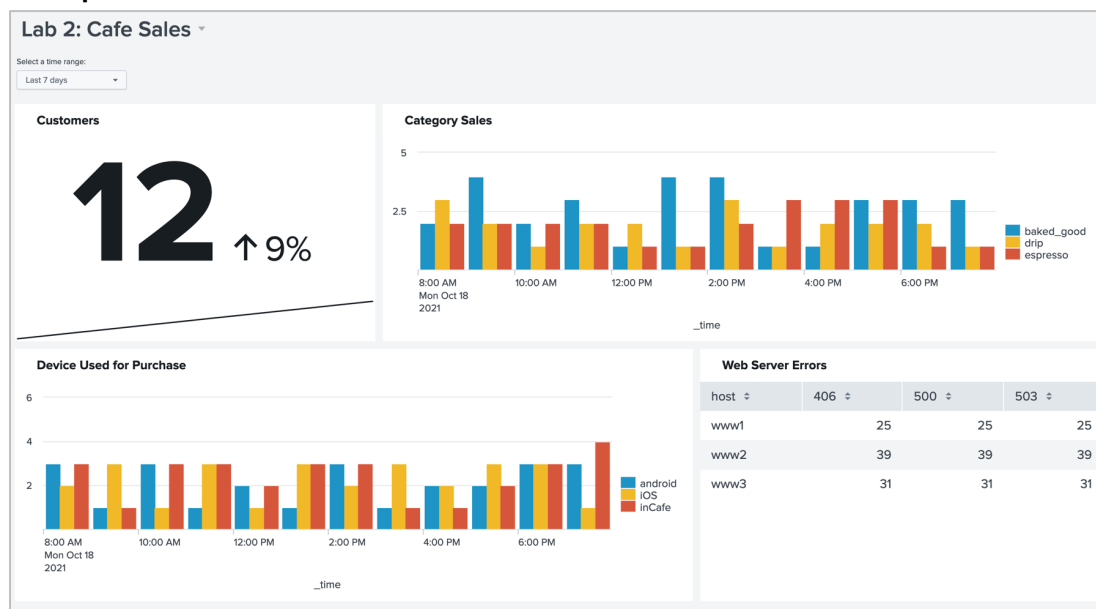
Lab Exercise 3 – Add Dynamic Coloring

Description

After stakeholders approve the prototype, begin building your dashboard with live data. Refinements can be made later. In this lab exercise you will create a dashboard based on the prototype with one refinement, you will add dynamic coloring to the table.
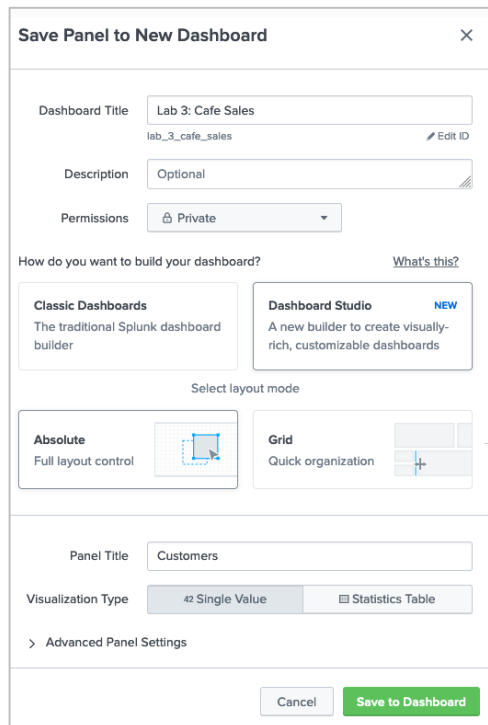

Scenario: **The sales team has given approval to build a dashboard based on the prototype. One change they want is to have the web server errors table highlighted by color for any unusually high counts.**

Steps

Task 1: Create a dashboard.

1. Navigate to the Introduction to Dashboards app.

2. Search over the **last 7 days** for:

   ```
   index=cafefood sourcetype=access_combined_cf
   action=purchase status=200
   | timechart count as sales | accum sales
   ```

3. Select the **Visualization** tab.

4. From the Visualization Type menu, select **Single Value**

5. Click the **Format** menu.

6. Select Show Trend in **Percent**.

7. Select Compared to **7 days before** and close the format window.

8. Select **Save As** > **New Dashboard**.

9. On the Save Panel to New Dashboard window, in the Dashboard Title box enter:

   ```
   Lab 3: Cafe Sales
   ```

10. Set permissions to **Private**.

11. Click **Dashboard Studio**.

12. Select **Absolute Layout**.

13. In the Panel Title box enter: `Customers`

14. Click **Save to Dashboard**.

15. Click **View Dashboard**.

Task 2: Name the data source.

16. Click **Edit**.

17. On the Configuration side panel, set the Display Mode to Actual Size.

18. Click on the **single value visualization**.

19. On the Configuration side panel, under Data Configurations, click the **pencil** icon beside Unnamed.

20. Change the Data Source Name from Unnamed to: `Customers`

21. Click **Run & Save**.

22. Save the dashboard and reload your browser.

    Reloading your browser refreshes the data source reference and removes the Selected Data Field warning.
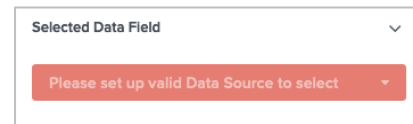
Task 3: Add Dynamic Coloring.

23. Click the **single value visualization**.

24. On the Configuration side panel, locate the Coloring section.

25. In the Dynamic Elements menu, select **Major Value & Trend**.

26. Save the dashboard.

Task 4: Set the default time range.

When adding a visualization to a dashboard, the query associated with it retains the original time range. To have the visualization use the dashboard's time range picker, you need to delete the visualization's queryParameters for time in the source code.

27. Select the Global Time Range input.

28. On the Configuration side panel, in the Title box, enter: `Select a time range:`

29. Locate the Default Value dropdown menu and select **Last 7 days**.

30. In the Visual Editor, click the **Source** icon.

31. Locate the Customers search and delete the **comma before and the `queryParameters`**.

```
1  { "dataSources": {
2      "ds_search_1": {
3        "type": "ds.search",
4        "options": {
5        "query": " index=cafefood ... | timechart count ... | accum sales",
6        "queryParameters": {
7          "earliest": " 7d@h",
8          "latest": "now"
9        }
10     },
11     "name": "Customers"
12    }
13   },
14  ...
```

32. Click **Back**.

33. Save the dashboard.

Task 5: Add a column chart.

34. Click the **Add Chart** icon.

35. Select **Column**.

36. On the New Data Source side panel, in the Data Source Name box, enter: `Category Sales`

37. In the Search with SPL box enter:

```
index=cafefood sourcetype=access_combined_cf action=purchase status=200
| timechart count by categoryId
```

38. Click **Run & Save**.

39. On the Configuration side panel, locate the Title box and enter: `Category Sales`

40. Locate the Position & Size section, set the column chart width to `800` and height to `300`.

41. Locate the Legend section and in the Position drop-down menu, select **Bottom.**

42. Select the **single value visualization**.

43. Set its width to `400` and its height to `300`.

44. Click and drag the column chart to the upper row, positioning it to **the right** of the single value.

45. Save the dashboard.

46. Click **View**.

Example:

Task 6: Clone a visualization.

47. Click **Edit**.

48. Select the Category Sales visualization and click the **Clone** button on its Action panel.

49. Revise the cloned visualization's title to: `Device Used for Purchase`

50. Revise the cloned data source name to: `Device Used for Purchase`

51. Replace the Device Used for Purchase query with:

```
index=cafefood sourcetype=access_combined_cf action=purchase status=200
useragent="*"
| stats count(eval(match(useragent, "POS"))) as "POS",
count(eval(match(useragent, "Android"))) as "Android",
count(eval(match(useragent, "iPhone"))) as "iPhone"
| transpose | sort by useragent
```

52. Click **Run & Save**.

53. On the Configuration side panel, locate the Position & Size section and set the X Position to `0` and Y Position to `300`.

54. Locate the Legend section and in the Position dropdown menu, select **None**.
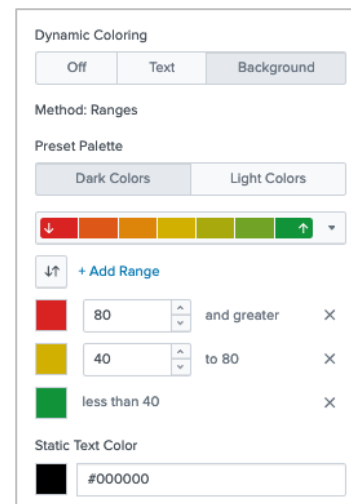
55. Save the dashboard.

Task 7: Add a table.

56. Click the **Add Chart** icon and select **Table** at the bottom of the list.

57. On the New Data Source side panel, in the Data Source Name box, enter: `Web Server Errors`

58. In the Search with SPL box enter:

```
index=cafefood sourcetype=access_combined_cf status>399
| chart count by host, status limit=3 useother=f
```

59. Click **Run & Save**.

60. On the Configuration side panel, locate the Title box and enter: `Web Server Errors`

61. Locate the Position & Size section and set the table X Position to `800` and Y Position to `300`.

Task 8: Add Dynamic Coloring.

62. Locate Column Formatting section and in the *+Add column to format* menu, select **400 - number**.

63. Click the **pencil icon** beside 400 – *number.*

64. In the Dynamic Coloring section of the window, select **Background**. The window expands to show Method: Ranges and Preset Palette.

65. Click the **range switch** button ⬍ to show values progressing from lowest being green to the highest being red.

66. Delete any two ranges.

67. Set the top range to red and `80`.

68. Set the middle range to yellow and `40`.

69. Repeat the above steps for the two remaining values: 404 and 504

70. Save the dashboard.

71. Click **View**.

72. Expand the dashboard to full screen by clicking the **scale arrows**.

Example: