Ranking Issues in Power BI

The boss says "We are getting to many complaints from our vendors. I need the top three issues, per vendor, today." In this exercise, you will create a Power BI report that cuts through the noise and gives you the top 3 issues per vendor.

Examine the Problem

As shown below, you have an Excel file, **Issues_2023_03.xlsx**, that lists the March 2026 issues for various clients, on various dates and the related cost per instance.

	A	B	C	D
1	**Issue Date**	**Vendor Name**	**Issue Name**	**Cost**
2	3/1/2026 2:15	IronGate Maintenance	HVAC malfunction	$ 422.11
3	3/1/2026 14:55	MetroTech	Security patch required	$ 118.44
4	3/1/2026 9:22	Titan Service Group	HVAC malfunction	$ 703.88
5	3/1/2026 21:10	Northstar Repair	Network outage	$ 512.77
6	3/1/2026 11:03	Northstar Repair	User account lockout	$ 266.90
7	3/1/2026 17:44	Titan Service Group	Software bug	$ 799.22
8	3/1/2026 6:12	Titan Service Group	Network outage	$ 74.55
9	3/1/2026 19:33	Titan Service Group	Software bug	$ 381.40
10	3/1/2026 4:51	Titan Service Group	Security patch required	$ 540.66
11	3/1/2026 22:18	MetroTech	Hardware replacement	$ 777.10
12	3/2/2026 1:44	Summit Solutions	Hardware replacement	$ 311.55
13	3/2/2026 16:22	PrimePath Support	Network outage	$ 509.33
14	3/2/2026 8:10	SwiftPoint Solutions	Hardware replacement	$ 129.77
15	3/2/2026 23:41	Velocity Support	User account lockout	$ 244.19

Your goal is to create a report with two tables. The first table should show each vendor's name, a count of their issues and their total cost, plus the top three issues that vendor is experiencing.

Stats per Vendor

Vendor Name	Issue Count	Total Cost	Issue1	Issue2	Issue3
Velocity Support	188	82,747.17	Network outage	Security patch required	System performance degradation
Titan Service Group	126	51,392.79	Network outage	Security patch required	System performance degradation
SwiftPoint Solutions	92	35,799.02	Network outage	Security patch required	User account lockout
Summit Solutions	82	30,911.42	Security patch required	Network outage	User account lockout
RedRock Maintenance	77	33,811.12	Network outage	Security patch required	System performance degradation
Pioneer TechWorks	75	33,760.33	Network outage	Security patch required	System performance degradation
Granite Peak Services	59	26,962.42	Network outage	System performance degradation	Security patch required
PrimePath Support	59	24,434.23	Network outage	Hardware replacement	Security patch required
MetroTech	52	21,890.02	Network outage	System performance degradation	Hardware replacement
Northstar Repair	48	20,926.20	Network outage	Security patch required	System performance degradation
Total	**1050**	**447,314.62**	**Network outage**	**Security patch required**	**System performance degradation**

The second table should rank all the issues for the selected vendor, and include the count and total per issue:

Stats per Issue - SwiftPoint Solutions

Issue Name	Issue Count	Total Cost	Issue Rank
Network outage	21	8,216.10	1
Security patch required	12	4,540.50	2
User account lockout	12	4,155.81	2
Hardware replacement	10	3,031.16	3
System performance degradation	9	4,867.06	4
HVAC malfunction	7	3,264.07	5
Software bug	7	2,495.07	5
Email delivery issue	6	3,345.74	6
Database error	4	561.20	7
Printer failure	4	1,322.31	7
Total	**92**	**35,799.02**	**1**

Step 1: Load the data into Power BI

1. Start a new report in Power BI.
2. Load the Excel file's March sheet into Power BI.
3. Rename the **March** table to **Issue Log**.

Step 2: Update Data Types

Select each field in the Data pane and if needed change its data type and format as shown below:

Field Name	Data Type	Format	Decimal Places
Cost	Decimal Number	Currency	2
Issue Date	Date	mm/dd/yyyy	
Issue Name	Text	Text	
Vendor Name	Text	Text	

Step 3: Create Cost and Count Measures

1. Create the following measure to count the number of records.

```
Issue Count = COUNTROWS('Issue Log')
```

2. To understand the Issue Count measure, add a table visual with the Issue Name and the Issue Count. Sort the table by the Issue Count in descending order. Then add "Count per Issues" as the Title of the table. Notice the number of rows counted depends on the issue name.

3. Create a second table visual that counts the issues per vendor name.

Count per Issue

Issue Name	Issue Count
Network outage	264
Security patch required	183
System performance degradation	130
User account lockout	99
Hardware replacement	88
HVAC malfunction	70
Software bug	69
Email delivery issue	68
Printer failure	61
Database error	18
Total	**1050**

Count per Vendor

Vendor Name	Issue Count
Velocity Support	188
Titan Service Group	126
SwiftPoint Solutions	92
Summit Solutions	82
RedRock Maintenance	77
Pioneer TechWorks	75
Granite Peak Services	59
PrimePath Support	59
MetroTech	52
Northstar Repair	48
Total	**1050**

4. Create the following measure to sum the total cost.

```
Total Cost = SUM('Issue Log'[Cost])
```

5. Add the **Total Cost** to the two table visuals. Then rename the tables as "Stats per Issue" and "Stats per Vendor".

Stats per Issue

Issue Name	Issue Count	Total Cost
Network outage	264	109,935.97
Security patch required	183	77,017.86
System performance degradation	130	59,152.34
User account lockout	99	41,195.50
Hardware replacement	88	39,176.01
HVAC malfunction	70	27,188.50
Software bug	69	30,266.43
Email delivery issue	68	27,419.50
Printer failure	61	29,299.74
Database error	18	6,662.77
Total	**1050**	**447,314.62**

Stats per Vendor

Vendor Name	Issue Count	Total Cost
Velocity Support	188	82,747.17
Titan Service Group	126	51,392.79
SwiftPoint Solutions	92	35,799.02
Summit Solutions	82	30,911.42
RedRock Maintenance	77	33,811.12
Pioneer TechWorks	75	33,760.33
Granite Peak Services	59	26,962.42
PrimePath Support	59	24,434.23
MetroTech	52	21,890.02
Northstar Repair	48	20,926.20
Total	**1050**	**447,314.62**

Step 4: Create a Ranking Measure

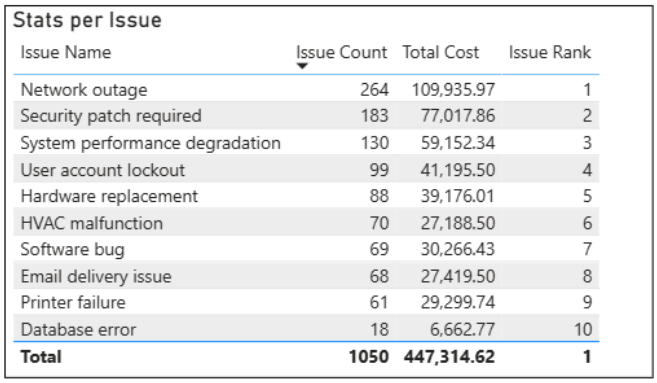
1. Add a new measure to rank each issue.

```
Issue Rank =
    RANKX(
        ALL('Issue Log'[Issue Name]),
        [Issue Count],
        ,
        DESC,
        DENSE
    )
```

   This ranks issues based on Issue Count in descending order. The ranking is dense meaning no gaps allowed (1,2,3,4 not 1,2,4).

   The purpose of the `ALL('IssueLog'[Issue Name])` expression is to remove any existing filter on the Issue Name field, resulting in a list of all possible issue names.

2. Add the Issue Rank to the Stats per Issue table visualization. Notice the issue with the most records is ranked number.



Stats per Issue

Issue Name	Issue Count	Total Cost	Issue Rank
Network outage	264	109,935.97	1
Security patch required	183	77,017.86	2
System performance degradation	130	59,152.34	3
User account lockout	99	41,195.50	4
Hardware replacement	88	39,176.01	5
HVAC malfunction	70	27,188.50	6
Software bug	69	30,266.43	7
Email delivery issue	68	27,419.50	8
Printer failure	61	29,299.74	9
Database error	18	6,662.77	10
Total	**1050**	**447,314.62**	**1**

Step 5. Create an Issue1 Measure

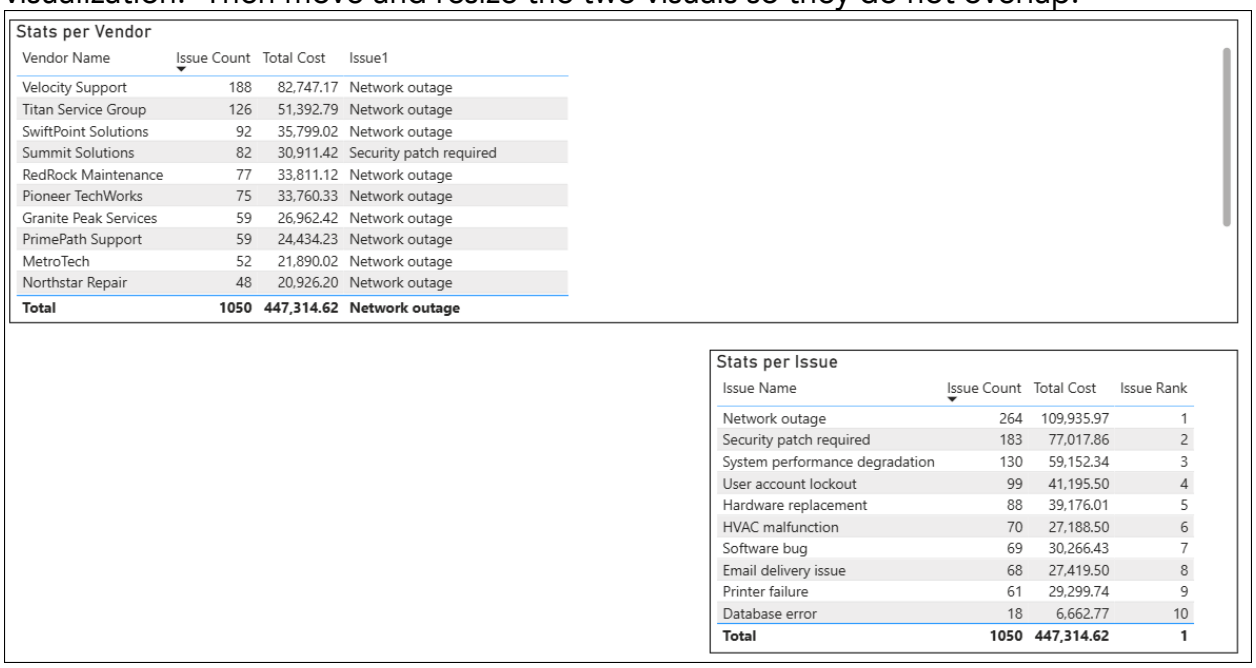
1. Create a new measure for the first ranking issue.

```
Issue1 =
VAR IssuesWithCounts =
    ADDCOLUMNS(
        VALUES('Issue Log'[Issue Name]),
        "Cnt", [Issue Count]
    )
VAR TopRow =
    TOPN(1, IssuesWithCounts, [Cnt], DESC, 'Issue Log'[Issue Name], ASC)
RETURN
    MAXX(TopRow, 'Issue Log'[Issue Name])
```

This measure first creates two variables.

- The `IssuesWithCounts` variable creates a virtual table.
  - The `VALUES('Issue Log'[Issue Name])` returns a one-column table of unique issues names.
  - The `ADDCOLUMNS` function is used to add columns to the table of Issue Names.
  - `Cnt` is the name of the second column.
  - `[Issue Count]` defines the value of the second column.
  - Therefore the `IssuesWithCounts` variable represents a table of Issue Names and their counts.
- The `TopRow` variable returns another virtual table with the first row of the `IssuesWithCounts` table based on the data sort by `[Cnt]` in `DESC`ending order and subsorted by `'Issue Log'[Issue Name]` in `ASC`ending order.
- The measure finally uses the `MAXX` function to return the `'Issue Log'[Issue Name]` field from the `TopRow` variable.

2. As shown below, add the Issue1 measure to the Stats per Vendor table visualization. Then move and resize the two visuals so they do not overlap.



Stats per Vendor

Vendor Name	Issue Count	Total Cost	Issue1
Velocity Support	188	82,747.17	Network outage
Titan Service Group	126	51,392.79	Network outage
SwiftPoint Solutions	92	35,799.02	Network outage
Summit Solutions	82	30,911.42	Security patch required
RedRock Maintenance	77	33,811.12	Network outage
Pioneer TechWorks	75	33,760.33	Network outage
Granite Peak Services	59	26,962.42	Network outage
PrimePath Support	59	24,434.23	Network outage
MetroTech	52	21,890.02	Network outage
Northstar Repair	48	20,926.20	Network outage
Total	**1050**	**447,314.62**	**Network outage**

Stats per Issue

Issue Name	Issue Count	Total Cost	Issue Rank
Network outage	264	109,935.97	1
Security patch required	183	77,017.86	2
System performance degradation	130	59,152.34	3
User account lockout	99	41,195.50	4
Hardware replacement	88	39,176.01	5
HVAC malfunction	70	27,188.50	6
Software bug	69	30,266.43	7
Email delivery issue	68	27,419.50	8
Printer failure	61	29,299.74	9
Database error	18	6,662.77	10
Total	**1050**	**447,314.62**	**1**

3. Experiment with selecting vendors in the Stats per Vendor visual, and notice the rankings in Stats per Issue visual.

Step 6. Create an Issue2 Measure

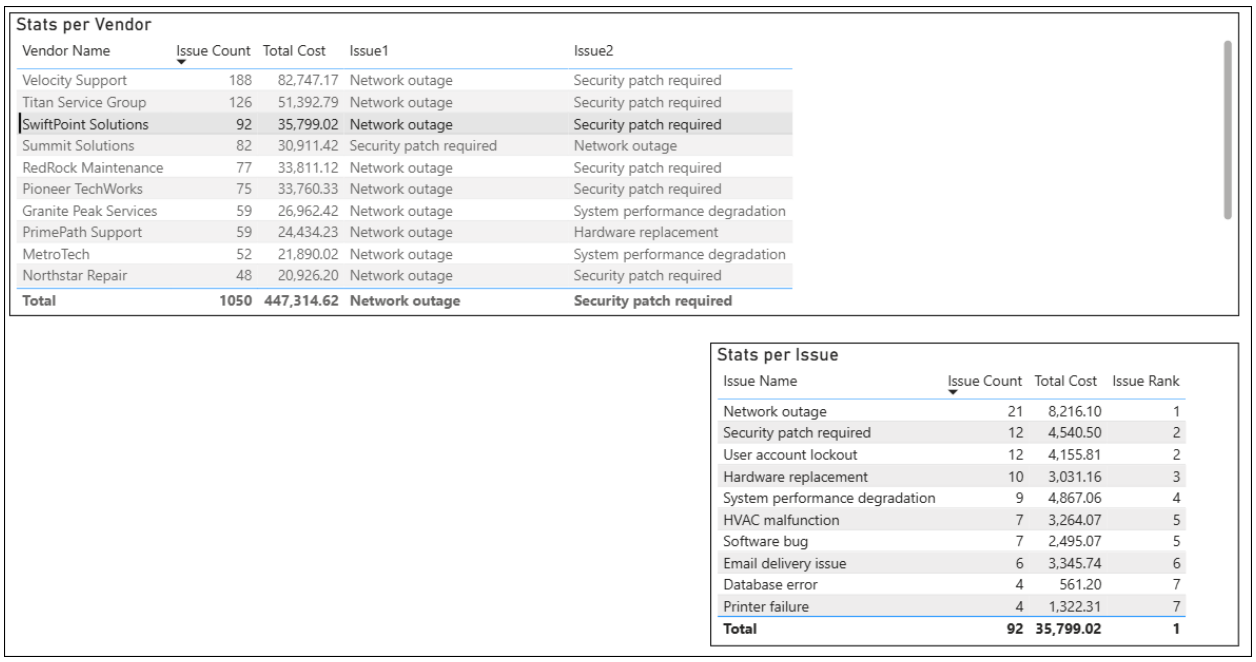
1. Create a new measure for the second ranking issue.

```
Issue2 =
VAR IssuesWithCounts =
    ADDCOLUMNS(
        VALUES('Issue Log'[Issue Name]),
        "Cnt", [Issue Count]
    )
VAR Top2 =
    TOPN(2, IssuesWithCounts, [Cnt], DESC, 'Issue Log'[Issue Name], ASC)
VAR Top1 =
    TOPN(1, IssuesWithCounts, [Cnt], DESC, 'Issue Log'[Issue Name], ASC)
Var Row2 =
    EXCEPT(Top2, Top1)
RETURN
    MAXX(Row2, 'IssueLog'[Issue Name])
```

This measure first creates four variables.

- The `IssuesWithCounts` variable is the same as the one created previously.
- The `Top2` variable returns a virtual table with the top **two** rows of the `IssuesWithCounts` table. Its sorting is also based on `[Cnt]` in `DESC`ending order and by `'Issue Log'[Issue Name]` in `ASC`ending order.
- The `Top1` variable returns a virtual one row table based on the `IssuesWithCounts` table using the same sorting criteria.
- The `Row2` variable returns the records that are in the `Top2 EXCEPT` it does not include those in the `Top1`.
- The measure finally uses the `MAXX` function to return the `'Issue Log'[Issue Name]` field from the `Row2` variable.

2. As shown below, add the Issue2 measure to the Stats per Vendor table visualization.



Stats per Vendor

Vendor Name	Issue Count	Total Cost	Issue1	Issue2
Velocity Support	188	82,747.17	Network outage	Security patch required
Titan Service Group	126	51,392.79	Network outage	Security patch required
SwiftPoint Solutions	92	35,799.02	Network outage	Security patch required
Summit Solutions	82	30,911.42	Security patch required	Network outage
RedRock Maintenance	77	33,811.12	Network outage	Security patch required
Pioneer TechWorks	75	33,760.33	Network outage	Security patch required
Granite Peak Services	59	26,962.42	Network outage	System performance degradation
PrimePath Support	59	24,434.23	Network outage	Hardware replacement
MetroTech	52	21,890.02	Network outage	System performance degradation
Northstar Repair	48	20,926.20	Network outage	Security patch required
Total	**1050**	**447,314.62**	**Network outage**	**Security patch required**

Stats per Issue

Issue Name	Issue Count	Total Cost	Issue Rank
Network outage	21	8,216.10	1
Security patch required	12	4,540.50	2
User account lockout	12	4,155.81	2
Hardware replacement	10	3,031.16	3
System performance degradation	9	4,867.06	4
HVAC malfunction	7	3,264.07	5
Software bug	7	2,495.07	5
Email delivery issue	6	3,345.74	6
Database error	4	561.20	7
Printer failure	4	1,322.31	7
Total	**92**	**35,799.02**	**1**

3. Notice when you select the SwiftPoint Solutions vendor that the **Stats per Issue** visual shows there are two issues that tie for second place.

   Notice the **Issue2** value for SwiftPoint Solutions is the first of the two second place issues.

   Remember, to break ties, both the Top2 and Top1 variables sort by `[Cnt]` and then by `[Issue Name]`.

   Please note that the `Top2` variable does not return one record. It returns both the first and second ranking items. Thus, the `Row2` variable is created with the `EXCEPT` function to return only the second ranking item.

Step 7: Create an Issue3 Measure

1. Create a new measure for the second ranking issue.

```
Issue3 =
VAR IssuesWithCounts =
    ADDCOLUMNS(
        VALUES('IssueLog'[Issue Name]),
        "Cnt", [Issue Count]
    )
VAR Top3 =
    TOPN(3, IssuesWithCounts, [Cnt], DESC, 'IssueLog'[Issue Name], ASC)
VAR Top2 =
    TOPN(2, IssuesWithCounts, [Cnt], DESC, 'IssueLog'[Issue Name], ASC)
VAR Row3 =
    EXCEPT(Top3, Top2)
RETURN
    MAXX(Row3, 'IssueLog'[Issue Name])
```

   This measure uses the same logic as the Issue2 measure.

2. As shown below, add the Issue3 measure to the Stats per Vendor table visualization.

Stats per Vendor

Vendor Name	Issue Count	Total Cost	Issue1	Issue2	Issue3
Velocity Support	188	82,747.17	Network outage	Security patch required	System performance degradation
Titan Service Group	126	51,392.79	Network outage	Security patch required	System performance degradation
SwiftPoint Solutions	92	35,799.02	Network outage	Security patch required	User account lockout
Summit Solutions	82	30,911.42	Security patch required	Network outage	User account lockout
RedRock Maintenance	77	33,811.12	Network outage	Security patch required	System performance degradation
Pioneer TechWorks	75	33,760.33	Network outage	Security patch required	System performance degradation
Granite Peak Services	59	26,962.42	Network outage	System performance degradation	Security patch required
PrimePath Support	59	24,434.23	Network outage	Hardware replacement	Security patch required
MetroTech	52	21,890.02	Network outage	System performance degradation	Hardware replacement
Northstar Repair	48	20,926.20	Network outage	Security patch required	System performance degradation
Total	**1050**	**447,314.62**	**Network outage**	**Security patch required**	**System performance degradation**

Step 8: Enhance the Stats per Issue table's title

You will now enhance the title for the Stats per Issue table, so it includes the name of the selected vendor from the Stats per Vendor table. But if no vendor is selected, then the title should just be "Stats per Issue".

1. Create the following new measure:

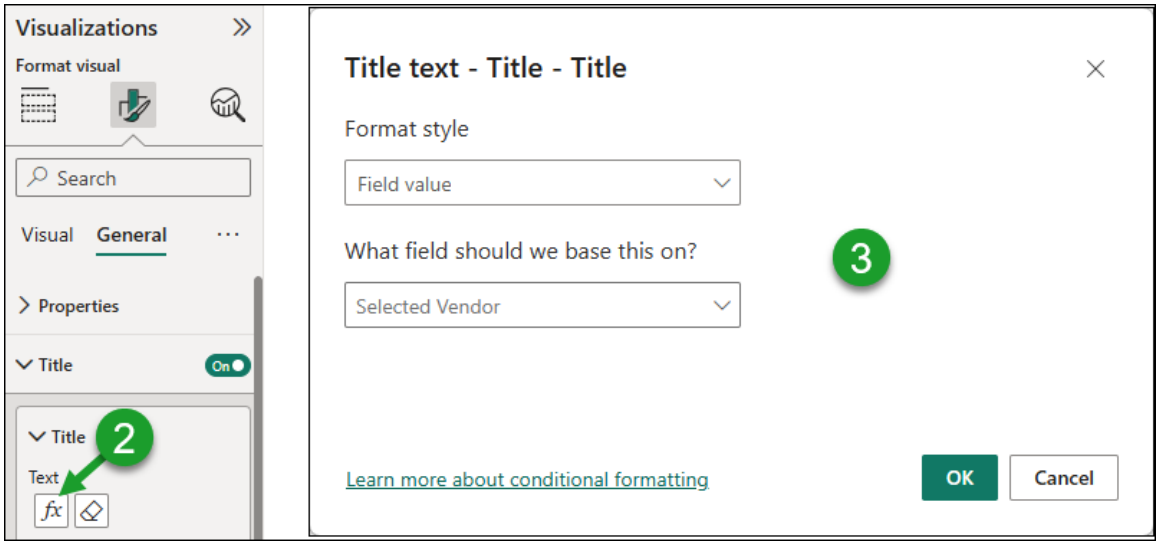
```
Selected Vendor =
    VAR SelectedVendor = SELECTEDVALUE('Issue Log'[Vendor Name])
    Return
    IF(
        NOT ISBLANK(SelectedVendor),
        "Stats per Issue - " & SelectedVendor,
        "Stats per Issue"
    )
```

   This measures creates a `SelectedVendor` variable that uses the `SELECTEDVALUE` function to find out vendor name was selected.

   The measure then uses the `IF` function to `Return` either "Stats per Issue –" with the name from the `SelectedVendor` variable or simply "Stats per Issue" when the variable is blank.

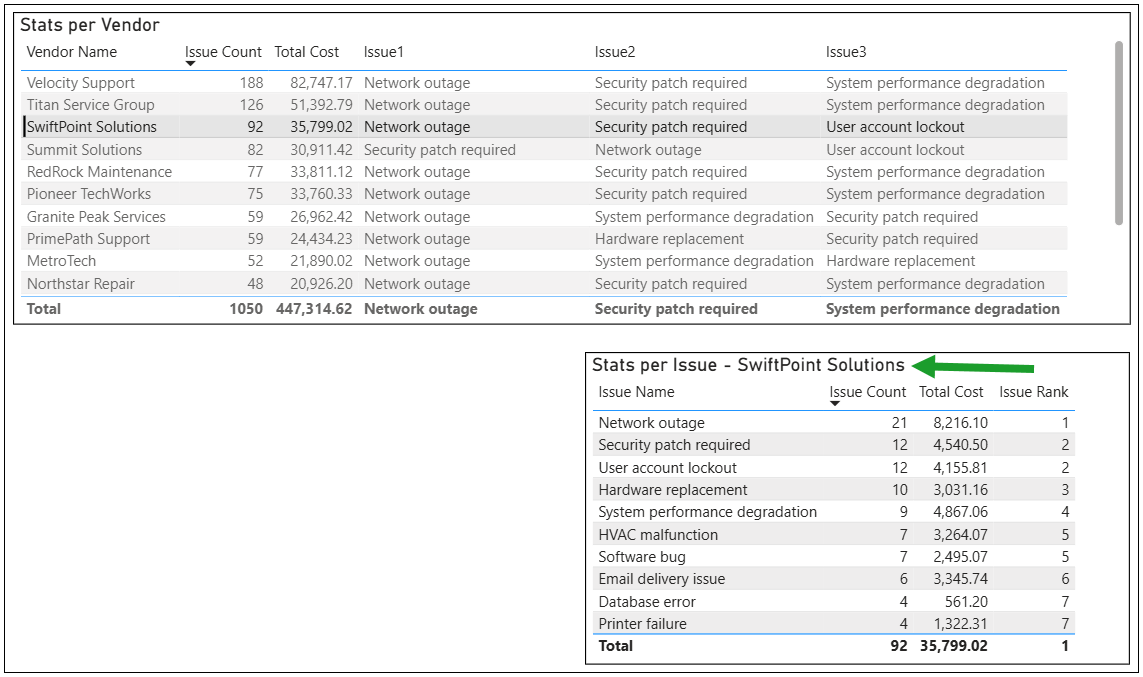
2. Select the **Stats per Issue** table. Then from the table's **Format** tab, expand the **Title** area, and for the Title's **Text** click the fx button.

3. Set the **Format style** to **Field value** and set the **What field should we base this on?** to the **Selected Vendor** measure. Finally click **OK**.



4. In the **Stats per Vendor** table, choose a vendor.

   The **Stats per Issue** table should now display that vendor's name in its title.



Stats per Vendor

Vendor Name	Issue Count	Total Cost	Issue1	Issue2	Issue3
Velocity Support	188	82,747.17	Network outage	Security patch required	System performance degradation
Titan Service Group	126	51,392.79	Network outage	Security patch required	System performance degradation
SwiftPoint Solutions	92	35,799.02	Network outage	Security patch required	User account lockout
Summit Solutions	82	30,911.42	Security patch required	Network outage	User account lockout
RedRock Maintenance	77	33,811.12	Network outage	Security patch required	System performance degradation
Pioneer TechWorks	75	33,760.33	Network outage	Security patch required	System performance degradation
Granite Peak Services	59	26,962.42	Network outage	System performance degradation	Security patch required
PrimePath Support	59	24,434.23	Network outage	Hardware replacement	Security patch required
MetroTech	52	21,890.02	Network outage	System performance degradation	Hardware replacement
Northstar Repair	48	20,926.20	Network outage	Security patch required	System performance degradation
Total	**1050**	**447,314.62**	**Network outage**	**Security patch required**	**System performance degradation**

Stats per Issue - SwiftPoint Solutions

Issue Name	Issue Count	Total Cost	Issue Rank
Network outage	21	8,216.10	1
Security patch required	12	4,540.50	2
User account lockout	12	4,155.81	2
Hardware replacement	10	3,031.16	3
System performance degradation	9	4,867.06	4
HVAC malfunction	7	3,264.07	5
Software bug	7	2,495.07	5
Email delivery issue	6	3,345.74	6
Database error	4	561.20	7
Printer failure	4	1,322.31	7
Total	**92**	**35,799.02**	**1**