

-----Pick() , Ceil()-----

```
Pick(Ceil(Rand()*4), 'Received', 'Approved', 'Pending', 'Denied')
```

```
//Ceil avrundar värdet till hösta heltal
```

```
// pick väljer en av värdena 'Received', 'Approved', 'Pending', 'Denied' beroende på första indata (Ceil....)
```

-----Mapping-----

Map:

```
MAPPING LOAD * INLINE [
```

```
ID, Status
```

```
1,Received
```

```
2,Approved
```

```
3,Pending
```

```
4,Denied
```

```
];
```

Data:

```
LOAD
```

```
ApplyMap('Map',Ceil(Rand()*4)) AS Status
```

```
Autogenerate xx;
```

-----IterNo(), RecNo(), RowNo()-----

IterNo() används som räknare inom while loopar

RowNo() ger radnummer

RecNo() används som räknare för Autogenerate

```
#TempTest:
```

```
load * inline [
```

```
FIELD
```

```
one
```

```
two
```

```
three
```

```
];
```

```
FOR Each a in FieldValueList('FIELD')
```

```
Test:
```

```
LOAD
```

```
'$(a)' &'-'&RecNo() as NEWFIELD,
```

```
'$(a)' &'-'&RowNo() as NEWFIELD2 ,
```

```
'$(a)' &'-'&IterNo() as NEWFIELD3
```

```
AutoGenerate 2
```

```
while IterNo()<4;
```

```
NEXT a
```

```
Drop table #TempTest;
```

-----SUM(Total Value) Aggr(nodistinct)-----

TempTest:

load * inline [

ColA, ColB, Value

A, a, 200
A, b, 250
B, a, 300
A, b, 450
C, b, 400
C, c, 500
];

ColA	ColB	Value	=sum(Value)	=Sum(Total <ColB> Value)	=Sum(Total Value)	=Aggr(sum(Value), ColB)	=Aggr(Nodistinct sum(Value), ColB)
Totals			2100	2100	2100	-	-
A	b	450	450	1100	2100	-	1100
A	b	250	250	1100	2100	1100	1100
A	a	200	200	500	2100	500	500
B	a	300	300	500	2100	-	500
C	c	500	500	500	2100	500	500
C	b	400	400	1100	2100	-	1100

=Aggr(sum(Value), ColA)	=Aggr(Nodistinct sum(Value), ColA)	=Aggr({<ColB ={'b','a'}, ColA= {'b'} >} nodistinct sum(Value), ColA, ColB)
-	-	-
-	900	700
-	900	700
900	900	200
300	300	-
-	900	-
900	900	400

2 conditions within 1 expression

=COUNT ({< UDATE = {'>= \$(=Date(vStartDate))<= \$(=Date(vEndDate))' } , SCORECARDNUMBER = {'>= \$(=ScorecardStart)<= \$(=ScoreCardEnd)'} >} DOCUMENT_COUNT)

ColA	ColB	Value	=AVG(Total <ColA> Value)	=Stdev(Total <ColA> Value)	=Stdev({<ColB = {'b'}> Total <ColB> Value)	=Stdev(Total <ColB> Value)	
Totals			370	166,24188	198,33233	166,24188	=Sum({<ColA ={'A'}>} Value) 1,45k
A	a	200	362,5	165,2019	131,49778	131,49778	aggregated standard deviation =stdev(aggr(stdev(Value),ColA)) 84,14
A	b	250	362,5	165,2019	-	95,39392	
A	b	450	362,5	165,2019	-	95,39392	
A	c	550	362,5	165,2019	43,493295	43,493295	
B	a	100	287,5	225	131,49778	131,49778	
B	a	150	287,5	225	131,49778	131,49778	
B	b	300	287,5	225	-	95,39392	
B	c	600	287,5	225	43,493295	43,493295	
C	a	400	460	58,878406	131,49778	131,49778	
C	b	420	460	58,878406	-	95,39392	
C	c	500	460	58,878406	43,493295	43,493295	
C	c	520	460	58,878406	43,493295	43,493295	

https://help.qlik.com/en-US/sense/September2018/Subsystems/Hub/Content/Sense_Hub/ChartFunctions/ColorFunctions/color-functions-charts.htm

=Colormix1 ((Value/ MAX(Total Value)) , RGB (255, 150, 100) , RGB (100, 150, 255))

=Colormix2 ((Value/ MAX(Total Value)-0.5)*2 ,RGB (255, 100, 0) , RGB (0, 150, 100),RGB (0, 0, 0))

//=ColorMapJet (((Value-Min(Total Value)+0.01)/Max(Total Value))))

Colorized each dimension in the pivot table::::

=IF(Dimensionality()= 1

, RGB (250,250,230) //Yellow

,IF(Dimensionality()= 2

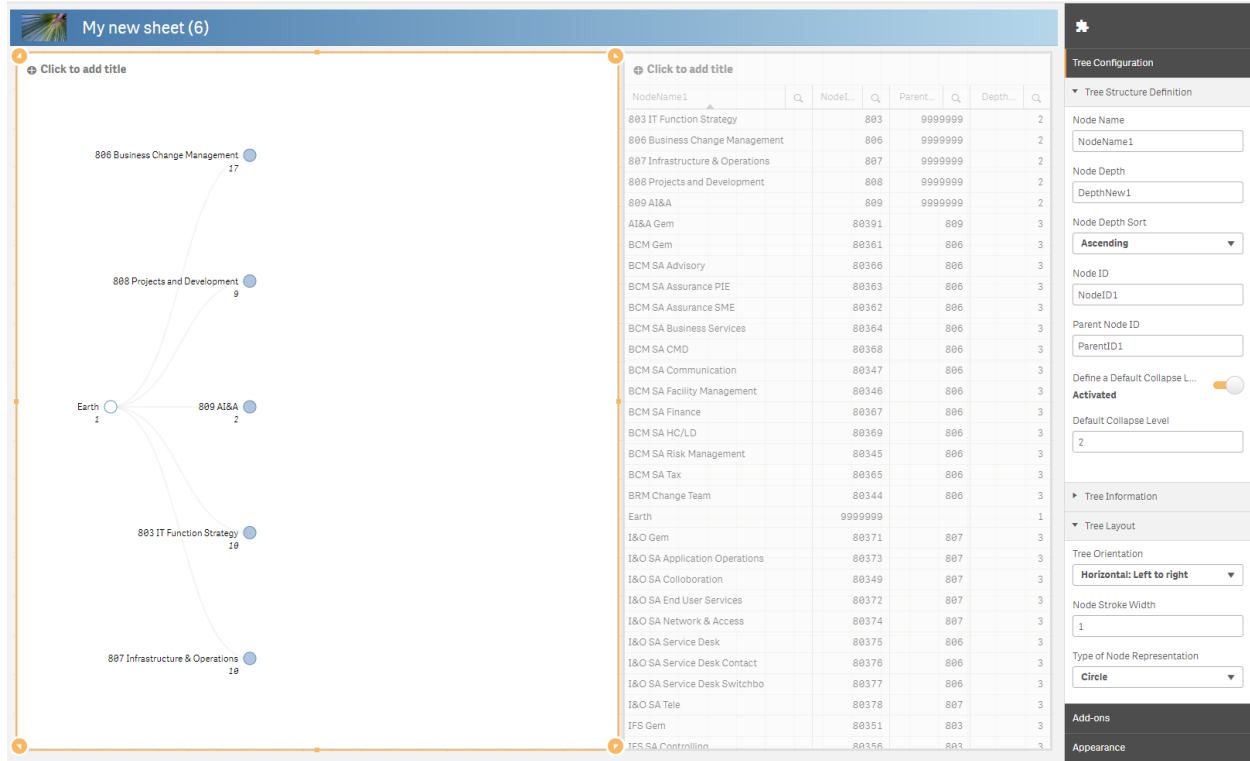
,RGB(230,250,230)// Green

,IF(Dimensionality()= 3

, RGB(230,250,250) //Blue

, RGB (250,230,230) //Red

)))



Test4:

Load distinct

Num#([Kostnadsställe]) AS NodeID1,

Num#(left("Function Area",3)) As ParentID1, //Num#(left([Kostnadsställe],3)) As ParentID1,

[KostnadsställeNamn] AS NodeName1

Resident DimOrganisation;

Concatenate(Test4)

Load

Num#(left([Function Area],3)) AS NodeID1,

9999999 As ParentID1,

[Function Area] As NodeName1

RESIDENT DimOrganisation;

Concatenate(Test4)

LOAD * inline

[
NodeID1, ParentID1, NodeName1

9999999, , Earth

];

Hierarchy (NodeID1, ParentID1, NodeName1, ParentName1, NodeName1, PathName1, '\', DepthNew1)

Load * Resident Test4;