Slicers

Filters

Dy Tiles

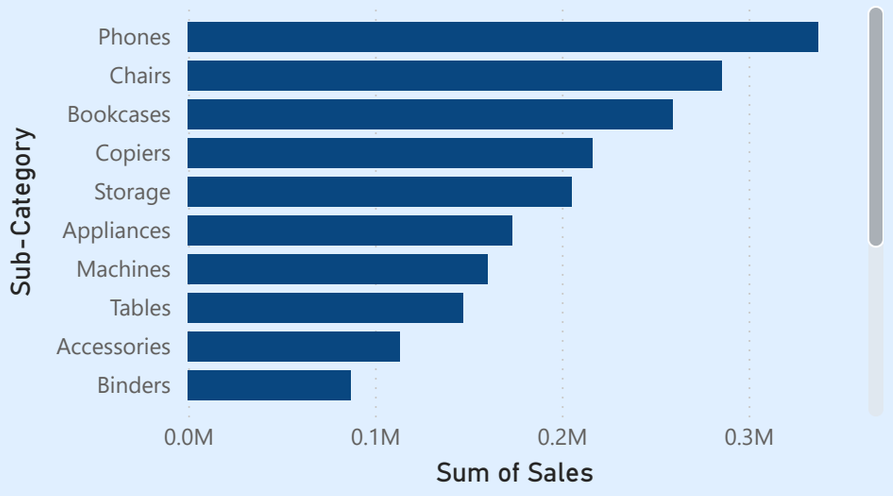
Year

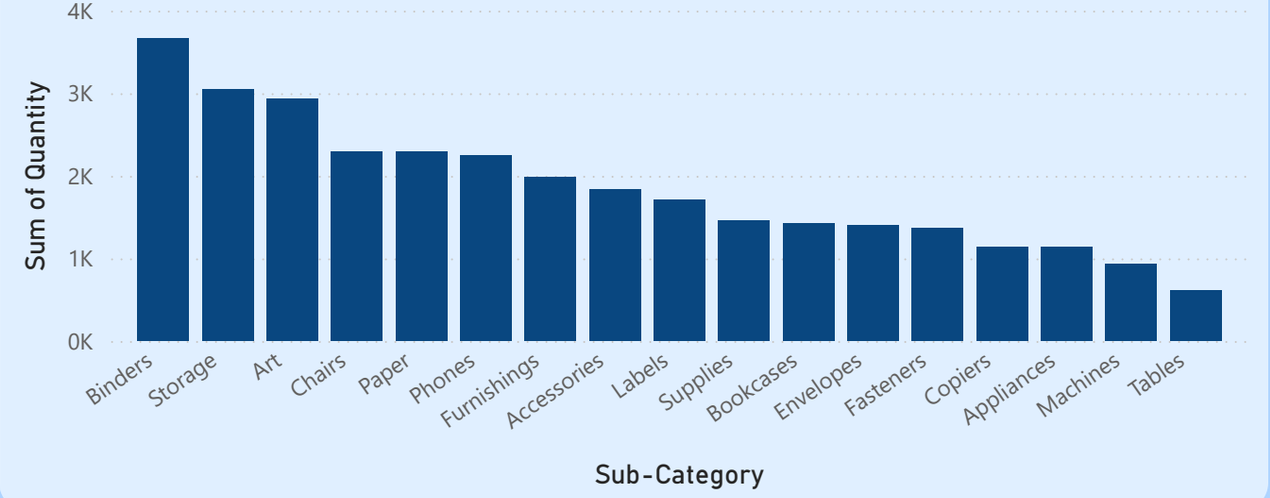
2011 | 2012 | 2013 | 2014

8998
Count of Row ID

248.94K
Sum of Profit

2.26M
Sum of Sales



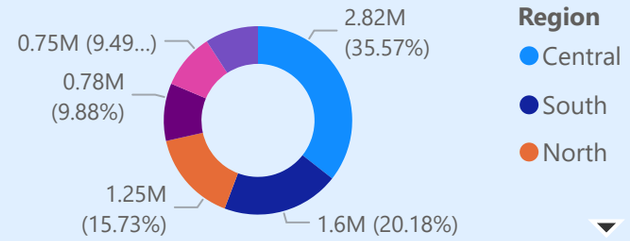


Slicers

Filters

Dy Tiles

Sum of Sales by Region

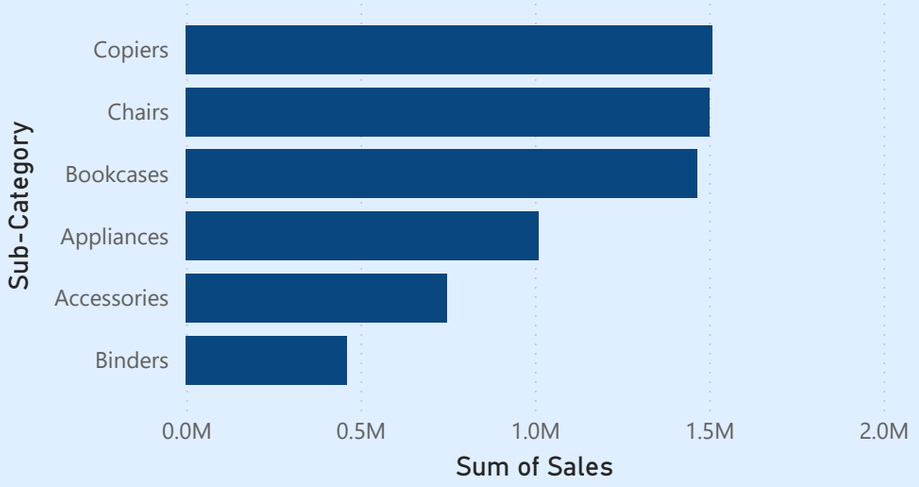


51.29K
Count of Row ID

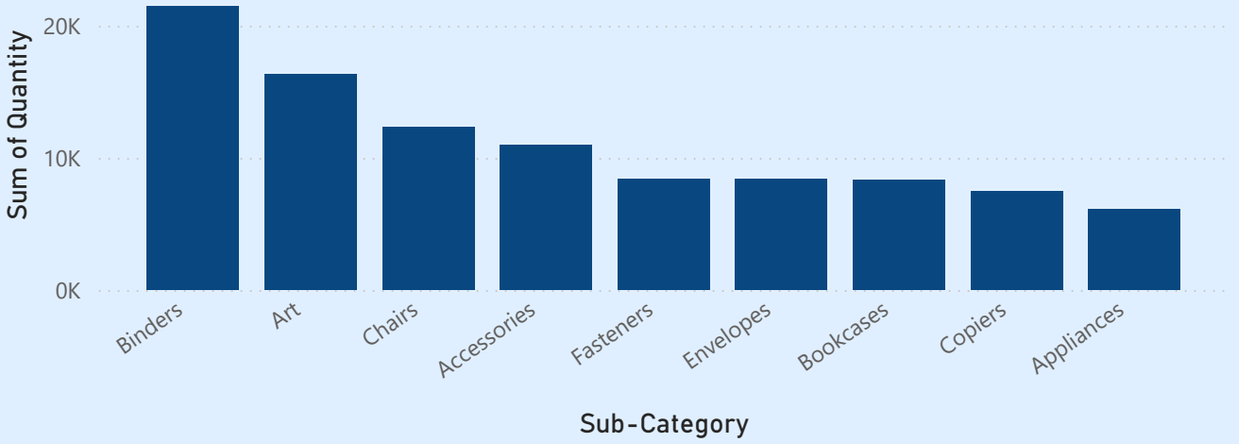
1.47M
Sum of Profit

12.64M
Sum of Sales

Sum of Sales by Sub-Category



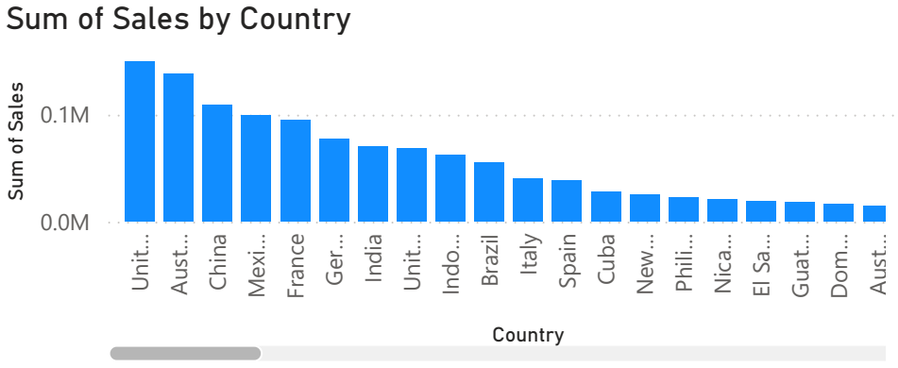
Sum of Quantity by Sub-Category



Sub-Category

- Accessories
- Appliances
- Art
- Binders
- Bookcases
- Chairs
- Copiers

Data from calculated columns

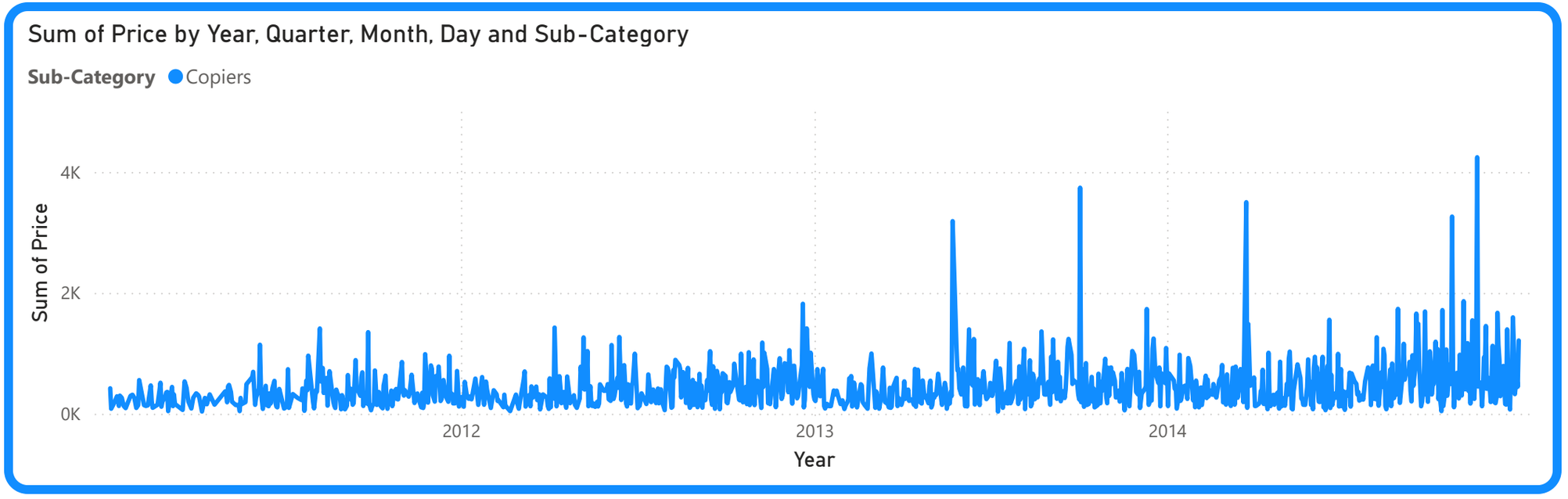


2223
Order Count

1.51M
Total sales amount

2120
distinct ordercount

679.01
avearge sales total



Slicers

Filters

Dy Tiles

x - Dynamic tiles will use new measures or calculated columns
x - Card visuals will be used
x - conditional formatting - adding appearance change on thresholds or performance metrics
x - some additional interactivity with slicers or filters

2013

Afghanistan

353.57
avearge sales total

Dynamic card - based on a calculated column can react to slicers

Country

Afghanistan	Albania	Algeria
Angola	Argentina	Armenia
Australia	Austria	Azerbaijan

Year

2011	2012	2013	2014

354
avearge sales total

Slicers

Filters

Dy Tiles

Create a measure that stores a variable you want to use for conditional formating

We will use year for that.

a measure of ProfitColor. you could use something else

```
ProfitColor =
IF (
    SUM(Store[Profit]) < 10000,
    "Red",  // Green for Profit greater than 50,000
    IF (
        SUM(Store[Profit]) >= 10000 &&
        SUM(Store[Profit]) >= 20000,
        "Blue",  // Blue for Profit between 10,000 and 50,000
        "Green"    // Red for Profit less than 10,000
    )
)
```

205.03K
Sum of Sales

48.81K
Sum of Profit

369
avearge sales total

Country

Argentina	India	New...
Nigeria	Philippines	Russia
Saudi...	Turkey	Ukraine

Year

2011	2012	2013	2014

DAX Filters

FILTER (table, expression)

filter rows in a table based on a condition, you can use FILTER. For example, if you want to sum sales values for only products that have sales greater than 30K

(Blank)

sum of Sales Over 300

```
sum of Sales Over 300 =
CALCULATE(
    SUM('Store'[Sales]),
    FILTER(
        'Store',
        'Store'[Sales] > 300
        )
)
```

```
Total Sales Over 300=
CALCULATE(
    SUM('Store'[Sales]),
    FILTER(
        'Store',
        'Store'[Sales] > 300
    )
)
```

(Blank)

Number of sales above 300

523.81

total sales sum

```
total sales sum =
CALCULATE(SUM(Store[Sales]))
```

Country

Argentina	India	New...
Nigeria	Philippines	Russia
Saudi...	Turkey	Ukraine

Year

2011	2012	2013	2014

Sub-Category

Accessories	Appliances	Art
Paper	Phones	Storage

relations

Create a new table from a previous one

then borrow columns from previous

add your own calculated columns

add your own measures

Create a new table in modeling and then:

```
Sales = SELECTCOLUMNS(
    Store,
    "Order dates", Store[Order Date],
    "Category", Store[Category],
    "sales", Store[Sales],
    "quantity", Store[Quantity]
)
```

If you then want to borrow another column from a previous one, what to do:

```
Order no. = LOOKUPVALUE(Store[Order ID], Store[Sales], Sales[sales])
```

does not work because there is no two rows uniquely

Create a new table then, with key columns:

```
Sales 2 = SELECTCOLUMNS(
    Store,
    "Order ID", Store[Order ID],
    "Order dates", Store[Order Date],
    "Category", Store[Category],
    "sales", Store[Sales],
    "quantity", Store[Quantity]
)
```

then add another column like country

use the look up command

```
Country = LOOKUPVALUE(Store[Country], Store[Row ID], Sales2[Row ID] )
```

But make sure that the references you are using are without missing values, or duplicates, otherwise it will not work

then check modelling

Create a relationship using drag and drop in modelling

then pulling data is a breeze

```
City = RELATED(Store[City])
```