

4. 从 Sales[月份值]中清除所有筛选器。

注意 如果你不熟悉 DAX 语言,那么深入了解这个公式的工作原理是一项很好的脑力训练,这可以帮助你更好地了解筛选上下文和变量一起工作的方式。

运行这段代码的结果如图 4-3 所示。然而,完全没必要把公式编写得这么复杂。这个公式最大的问题是无法使用为实现时间智能计算而专门内置的时间智能函数,因为时间智能函数必须依赖日期表。

行标签	SalesAmount	SalesYTD
2007	1,459,215.72	1,459,215.72
January	101,097.29	101,097.29
February	108,553.23	209,650.52
March	119,707.79	329,358.31
April	121,085.76	450,444.07
May	123,413.45	573,857.52
June	121,707.42	695,564.94
July	139,381.04	834,945.98
August	87,384.12	922,330.10
September	155,276.08	1,077,606.18
October	99,872.58	1,177,478.76
November	122,522.73	1,300,001.49
December	159,214.23	1,459,215.72
2008	1,122,534.46	1,122,534.46
2009	1,242,533.52	1,242,533.52
总计	3,824,283.70	1,242,533.52

图 4-3 [SalesYTD]计算出了正确的值,但它的公式太复杂了

如果你通过添加一个日期表(如图 4-4 所示)来更新数据模型,则公式的编写将变得更加容易。

此时,你可以使用预定义的时间智能函数来编写[SalesYTD]的公式,具体内容如下。

```
SalesYTD :=
CALCULATE (
    [SalesAmount],
    DATESYTD ( 'Date'[日期] )
)
```

注意 这不仅适用于计算 YTD。在使用日期表时,计算所有时间智能度量值都很方便。