#### Data Warehouse Design & Develop



Masoud Mirzakhani Senior DW/ ETL/ BI Architect

#### Microsoft SQL Server 2019 Design & Develop



## Masoud Mirzakhani Senior DW/ ETL/ BI Architect

- Master of Science in Information Technology
- Bachelor of Science in Information Technology
- md.mirzakhani@gmail.com
- @MasoudMirzakhani
- linkedin.com/in/masoudmirzakhani



## اجزای محیط توسعه



#### Development Workstation \*

- ❖ماشینی که بر روی آن ابزار های توسعه را نصب می کنیم.
- ❖ بعد از اتمام توسعه مدل، بایستی آن را بر روی یک سرور بارگزاری نماییم.
- 💠 این بارگزاری با استفاده از یک زبان DDL ای به نام Tabular Model Scripting Language (TMSL) صورت می گیرد.

#### Development Server \*

- لا سروری که SSAS بر روی آن نصب شده است.
- ❖داده پس از Process به این سرور انتقال داده می شود.

#### Workspace Database ❖

- 💠 یک دیتابیس موقت است که هنگام توسعه مدل ایجاد می شود.
  - این دیتابیس می تواند در دو حالت زیر ایجاد گردد:
- 💠 Integrated workspace: از سرویس SSASداخلی Visual Studio استفاده می کند.
- ❖ Workspace server؛ از یک سرویس مستقل SSAS که بر روی ماشین نصب شده است بهره می برد.

# فرایند نصب SSAS



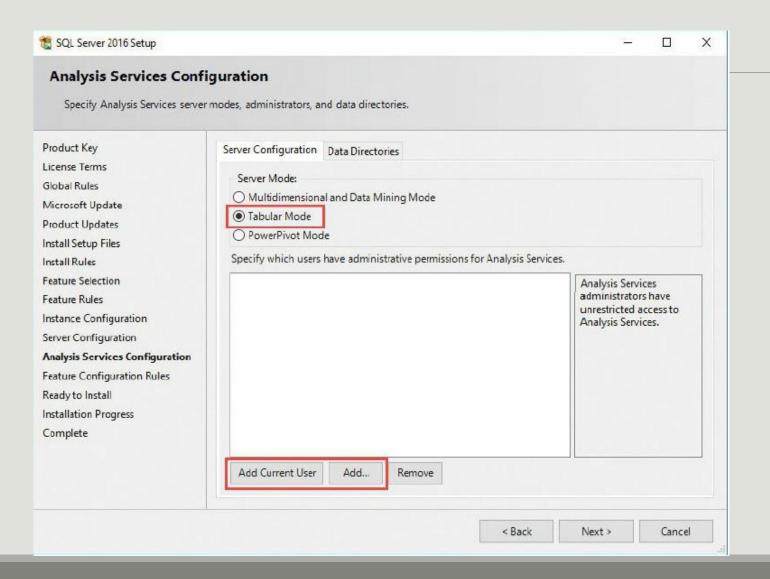
Feature Selection  Select the Evaluation features	to install.					
Product Key License Terms Global Rules Microsoft Update Product Updates Install Setup Files Install Rules Feature Selection Feature Rules Instance Configuration Server Configuration Analysis Services Configuration Feature Configuration Rules Readyto Install	Features:    Database Engine Services   SQL Server Replicatio   R Services (In-Databa   Full-Text and Semant   Data Quality Services   PolyBase Query Services   Reporting Services - Nati   Shared Features   R Server (Standalone)   Reporting Services - Shar   Reporting Services Add-i	n se) ic Extractions for Sea ce for External Data ve	instance feature isolated from ot Server instances Prerequisites for Already installed i Microsoft .N To be installed f	on and operation of of a SQL Server inst her SQL Server inst can operate side-b selected features: d: lET Framework 4.0 from media:	tance is ances. SQL py-side on	^ >
Installation Progress Complete	Select All Unselect All Instance root directory: Shared feature directory (x86):	C:\Program Files\Mic C:\Program Files\Mic C:\Program Files (x86)	rosoft SQL Server\		···	

نصب بر روی Development server

https://go.microsoft.com/fwlink/?linkid=866662

# فرایند نصب SSAS





ادامه...

## فرایند نصب SSAS



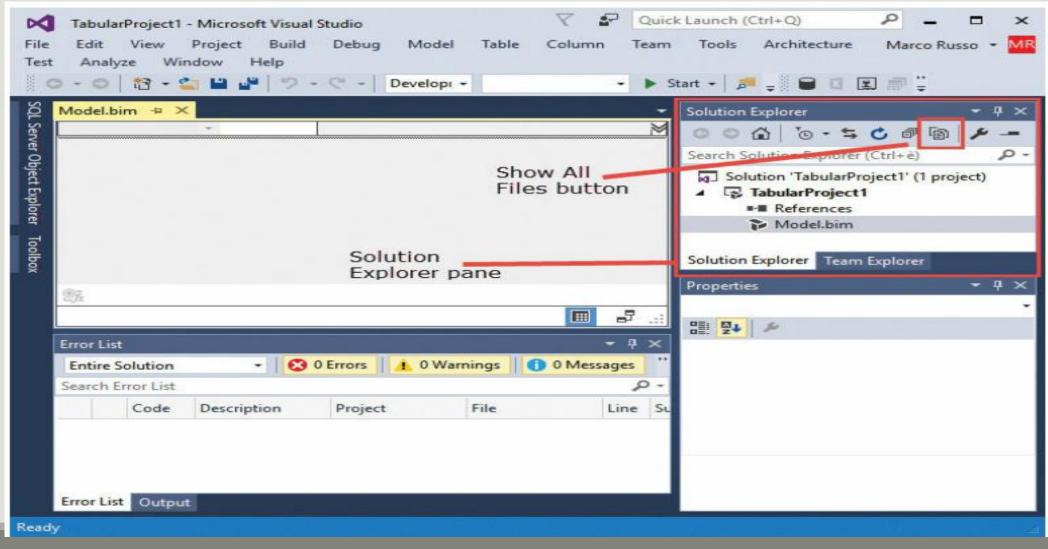
نصب بر روی Development workstation

- ❖ Visual Studio Community Version
- https://visualstudio.microsoft.com/thank-you-downloading-visual-studio/?sku=Community&rel=16
- ❖ Visual Studio "Microsoft Analysis Services Projects" Extension
- https://marketplace.visualstudio.com/items?itemName=ProBITools.MicrosoftAnalysisServicesModeling
   Projects

### Tabular Project

#### **Visual Studio**





### اضافه کردن داده به مدل



- ∜منابع داده قابل استفاده:
- Relational databases \*
- Multidimensional sources \*
  - Data feeds \*
- Open Data Protocol (OData) 🧇
  - Reporting Services \*
    - Text files ❖

## اضافه کردن داده به مدل

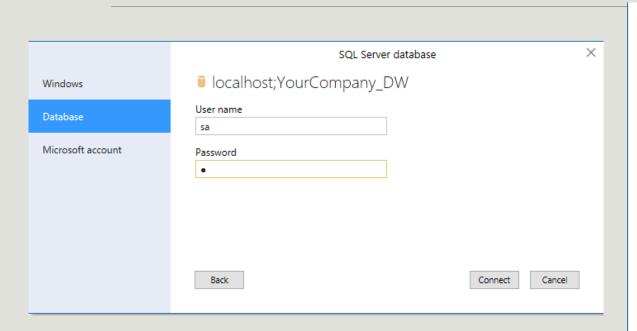


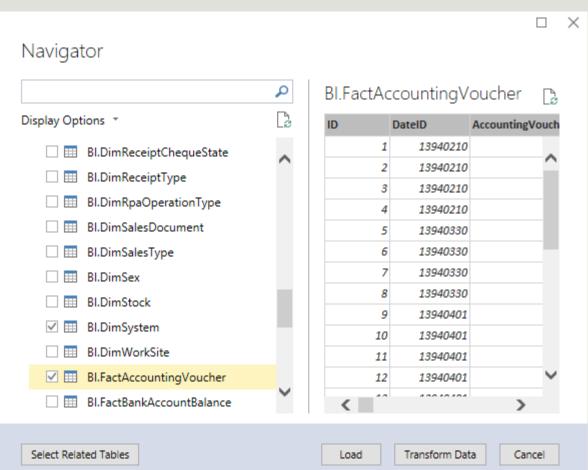
Search	Database
All	SQL Server database
File	Access database
Database	
Azure	Oracle database
Online Services	MySQL database
Other	PostgreSQL database
	Sybase database
	Teradata database
	SAP HANA database
	■ IBM Informix database
	SAP Business Warehouse Application Server

SQL Server database			
Server (i)			
Database (optional)			
YourCompany_DW			
Advanced options			
Advanced options			

## اضافه کردن داده به مدل

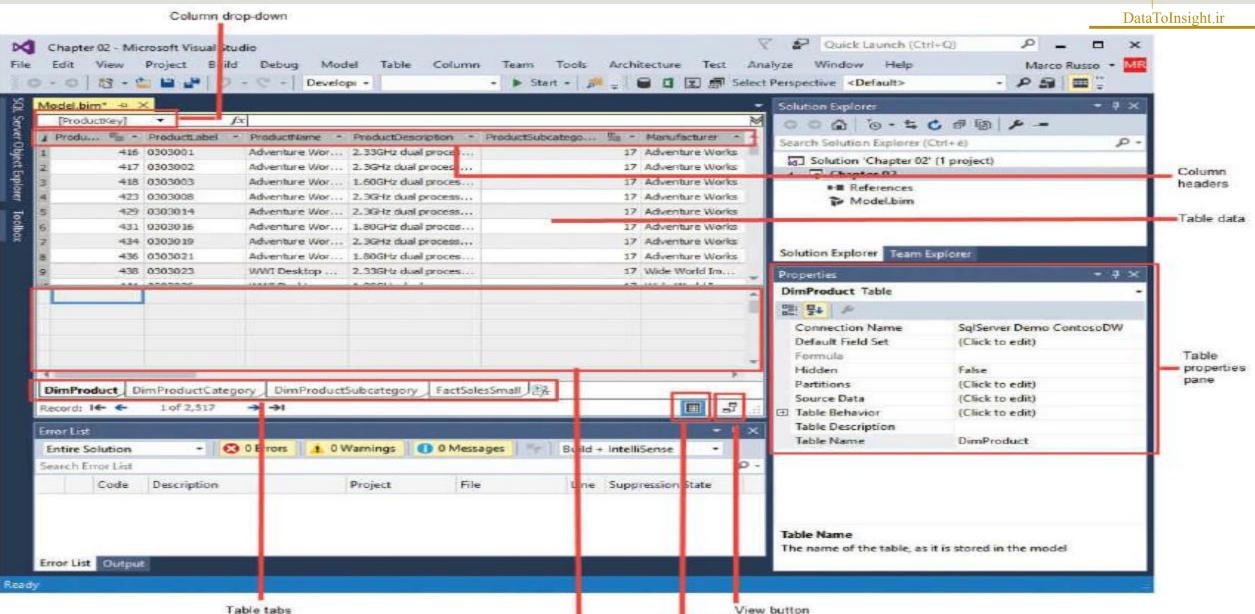






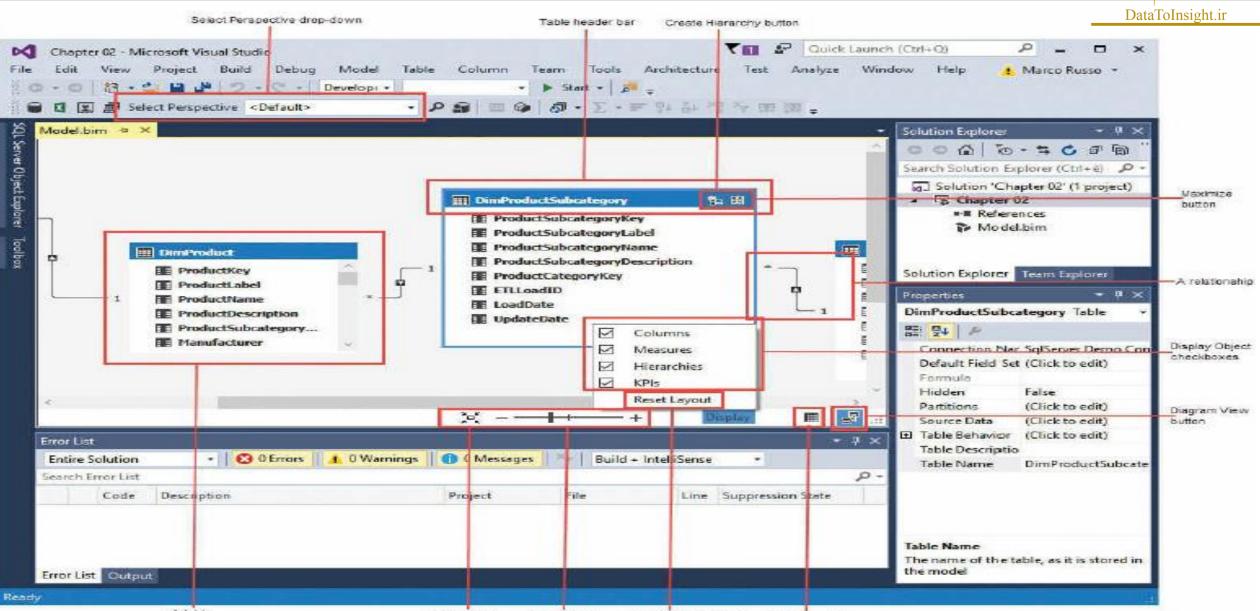
#### **Grid View**





#### Model View





### DimDate



Mark as Dat	e Table		?	×
	ted column must be o	unique identifier for the of the date data type a		
Date	Date			~

### Calculated Column



oim* ⊅ X											→ 🌣	Tab	ular Model Explorer
tBalance]	-	$f_X$ = BI FactAcco	untingVo	ucher'[Credit] - 'BI Fac	tAccountingVou	cher'[Debit]					₩	+	□ Search
VoucherSt	B ▼ Sys	圆▼ Acco	<u>F</u>	DetailedAcco 児	· Curre 圆	ı ▼ Debit	~ Credit ~	DebitBalance *	BalanceBasedOnNature 1	CreditBalance	Add Column -		■ KPIs
	2	2	229			-1 500	0 0	5000	5000	-5000		4	Measures
	2	2	229			-1 500	0 0	5000	5000	-5000			Sum of DebitBalance
	2	2	229			-1 500	0 0	5000	5000	-5000			Perspectives
	2	2	229			-1 500	0 0	5000	5000	-5000		▷	Relationships
	2	2	229			-1 500	0 0	5000	5000	-5000			Roles
	2	2	229			-1 500	0 0	5000	5000	-5000		1	Tables
	2	2	229			-1 500	0 0	5000	5000	-5000			
	2	2	229			-1 500	0 0	5000	5000	-5000			_
	2	2	229			-1 500	0 0	5000	5000	-5000			
	2	2	229			-1 500	0 0	5000	5000	-5000			D III BI DimDate
	2	2	229			-1 500	0 0	5000	5000	-5000			
	2	2	229			-1 500	0 0	5000	5000	-5000			
	2	2	229			-1 500	0 0	5000	5000	-5000			■ BI FactAccountingvoucher
	2	2	229			-1 500	0 0	5000	5000	-5000		н	✓ Golu Add Colum
	2	2	229			-1 500	0 0	5000	5000	-5000		l	<b>■</b> 4
	2	2	229			-1 500	0 0	5000	5000	-5000			
	2	2	229			-1 500	0 0	5000	5000	-5000			
	2	2	229			-1 500	0 0	5000	5000	-5000			E C L'
	2	2	229			-1 500	0 0	5000	5000	-5000	Ų		TreditBalance

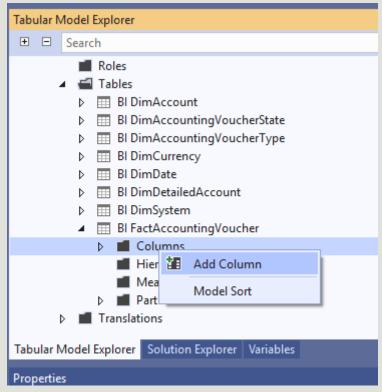
#### Calculated Column



\* = 'BI FactAccountingVoucher'[Credit] - 'BI FactAccountingVoucher'[Debit]

روش دوم

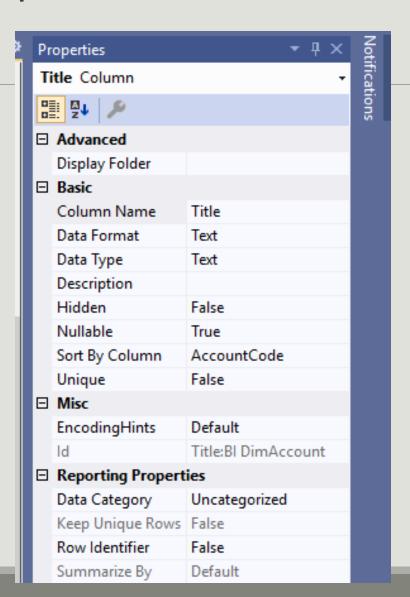
- = YEAR('BI DimDate'[GregorianDate])
- = "Y " & YEAR('BI DimDate'[GregorianDate])



### Column Properties

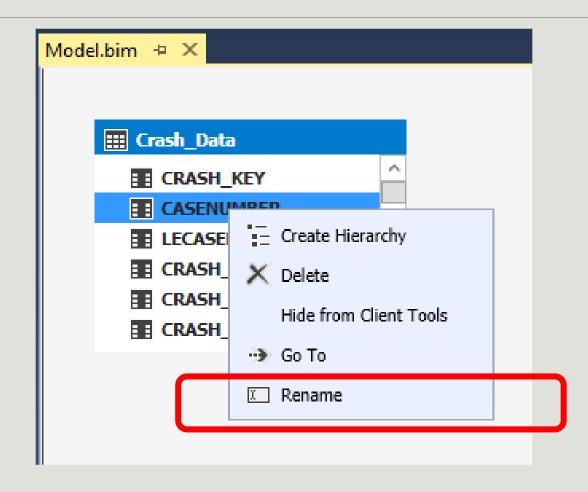
DataToInsight.ir

- Hidden
- Sort By Column
- Summarized By
- Display Folder



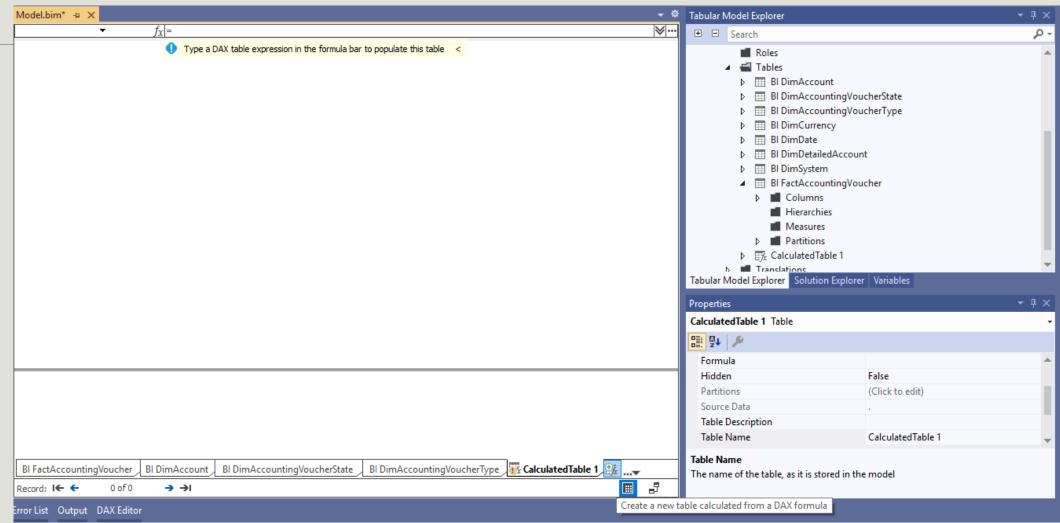
## Renaming Column





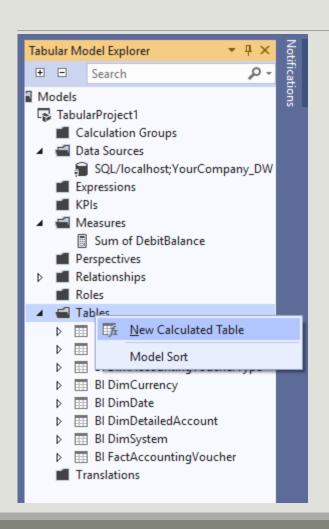
### Calculated Table





#### Calculated Table

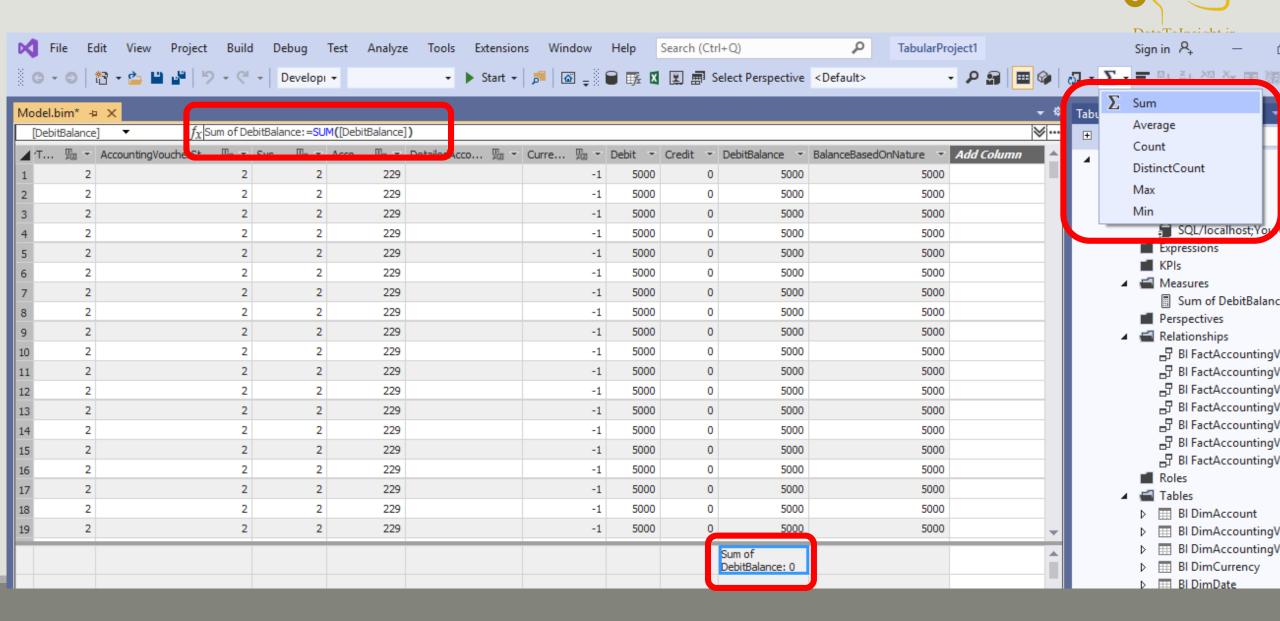




- **❖** =CALENDAR("1/1/2020","1/1/2021")
- =YEAR(DateTable[Date])
- =MONTH(DateTable[Date])
- =FORMAT(DateTable[Date],"MMM")
  - ❖"M": returns month number
  - ❖"MM": returns the 2-digit month number
  - ❖"MMM": returns the first three characters of the month name
  - ❖"MMMM": returns the full length of the month name
- = WEEKDAY(DateTable[Date],1)
- =WEEKDAY(DATEADD(DateTable[Date],1,DAY))
- =FORMAT(WEEKDAY(DateTable[Date]), "DDD")

روش دوم

#### Measure



#### Measure



- Sum\_debit\_balance:= SUM('BI FactAccountingVoucher'[DebitBalance])
- count\_records:= COUNT('BI FactAccountingVoucher'[ID])
- SSAS column summarized by options:
  - **.** Sum
  - Average
  - Count
  - DistinctCount
  - Max
  - **♦** Min

## **Key Performance Indicator**



	Ke	y Performance In	dicator (KPI)		?   ;
KPI base measure (valu	ue): Sum_of_Fatalities				V
KPI Status					
Target					
Measure:	Last_Year_Fatalities				
O Absolute value:					
		<b>②</b>	9(		
<b>←</b>					<b>→</b>
				Target	
Select icon style:					
	<b>(3)</b>		•	0 •	
	(1)		<u> </u>	0	
				Ŏ	
		_		•	
<b>♠</b> Descriptions					
				OK	Cancel

KPIها، مقادیری برای سنجیدن اهداف کسب و کار هستند

از آن ها می توانیم برای مانیتور کردن عملکرد واقعی سازمان در مقابل اهداف برنامه ریزی شده یا بوجه های پیش بینی شده استفاده کرد.

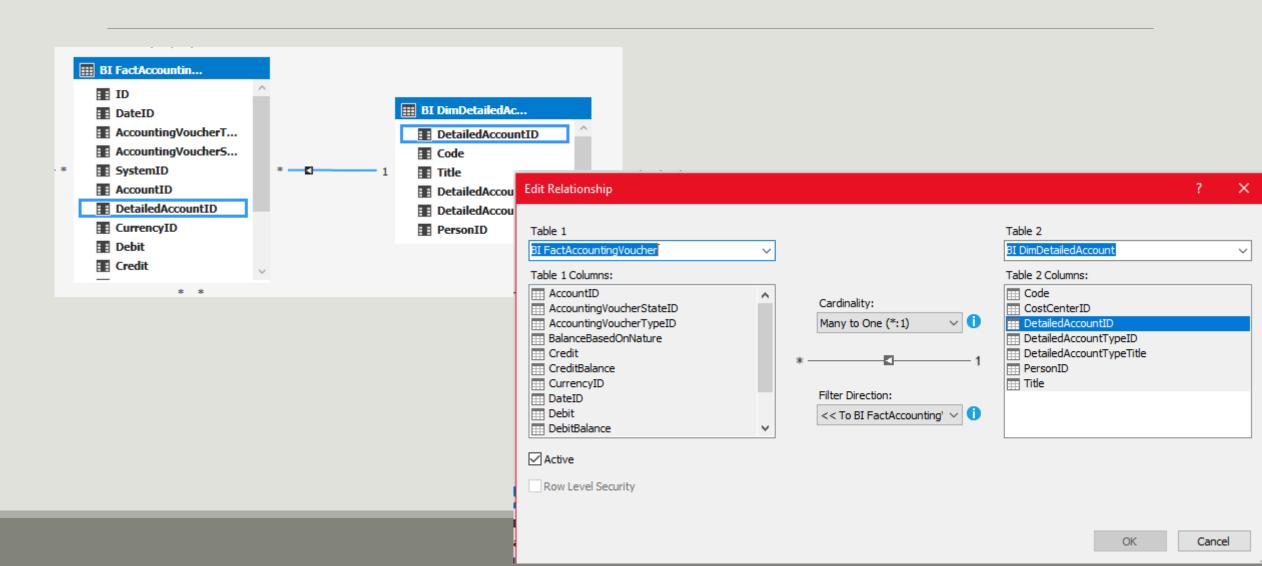
ویژگی های KPI:

**Value** 

Goal **∜** 

**Status** \*







#### **Cardinality \***

Many to One (\*:1)❖

One to Many (1:\*)

One to One (1:1) \*

#### **Filter Direction \***

single direction .

Bidirectional \*

**Active \*** 



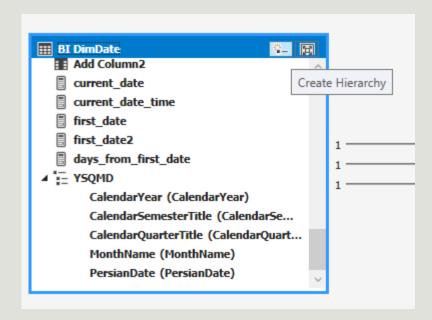
ایجاد روابط به صورت دستی به صورت اتوماتیک



Role Playing Dimension پیاده سازی

### Hierarchy





You can think of them as predefined pathways through your data that help your users explore down from one level of granularity to another in a meaningful way.

You should build hierarchies when one-to-many relationships exist between the columns in a single table because this usually indicates the existence of an underlying pattern in the data itself.

when certain columns are frequently grouped together in reports. For example, a retailer might want to drill down from Product Category to Brand to Style to Color to Product, even if there is a many-to-many relationship among Brand, Style, and Color.

### **Translations**



https://www.sqlbi.com/tools/ssas-tabular-translator/

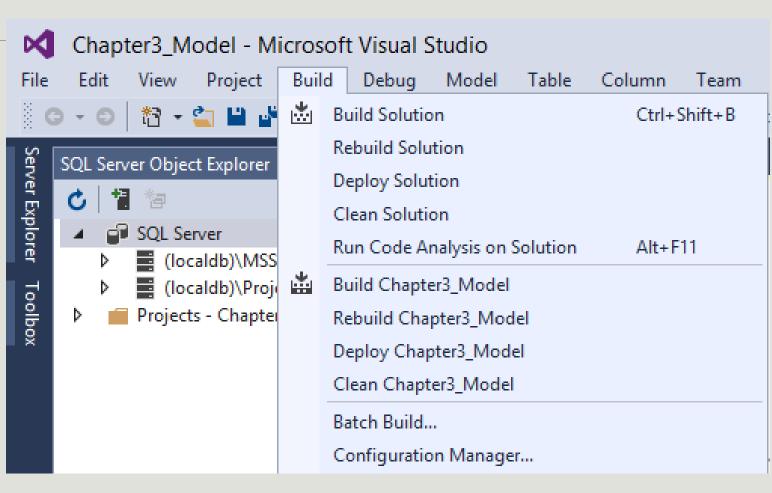
### Deploy



Deployment of your model is the final step to getting the data accessible to your users for

reporting. You have designed and built your model in Visual Studio. In order for others to

see and use it, you need to push the design and data to the Analysis Services server.



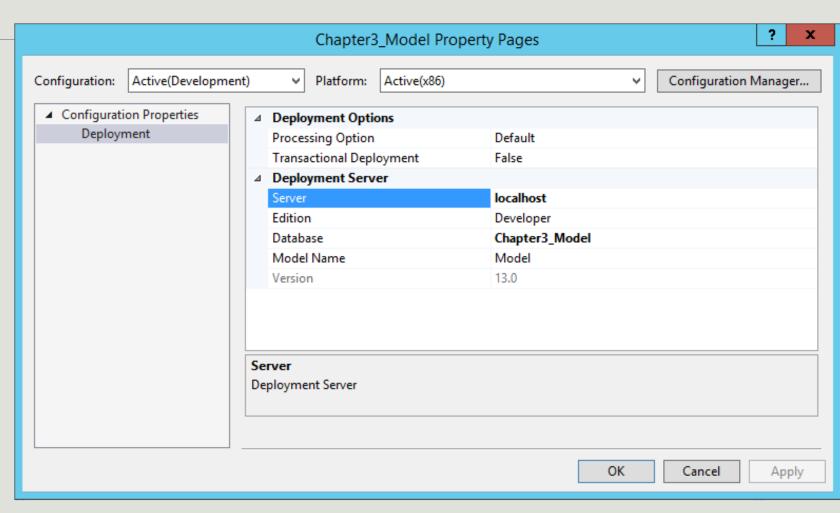
### Deploy



Deployment of your model is the final step to getting the data accessible to your users for

reporting. You have designed and built your model in Visual Studio. In order for others to

see and use it, you need to push the design and data to the Analysis Services server.



### **Browsing Your Data**



- SQL Server Management Studio
- Microsoft Excel
- Microsoft Power BI Desktop

### Perspectives

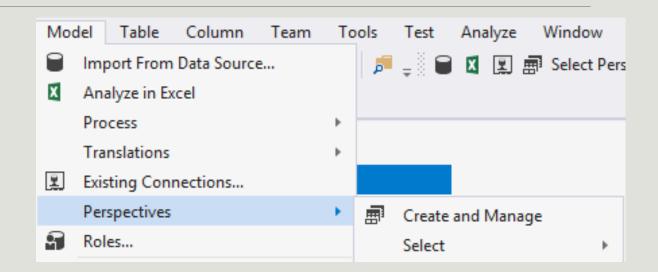


Perspectives enable you to create views of the

model that are limited in size based on your requirements. Using our example, you could

create a perspective that limits the data to being greater than 2010 and weather accidents

that occurred under rain and severe winds.



# Perspectives



e it easier to navigate large data sets.			
w Perspective			
Fields	Products	Stores	
Tables			
+ Currency			
+ Date		$\square$	
+ Product			
- Promotion			
Discount			
End Date			
Promotion			
Promotion Category			
Promotion Code			
Promotion Type		Ø	
PromotionKey *			
Start Date			
+ Sales		$\square$	
+ Store			

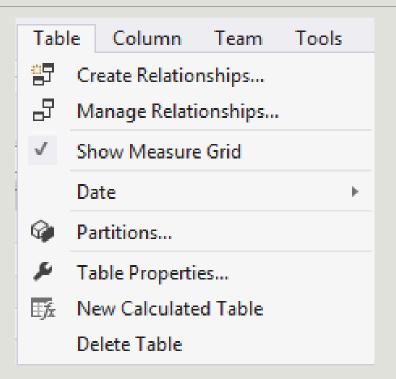
#### **Partitions**



Partitions in Analysis Services enable you to break up your data into manageable parts.

Typically, you use them when you want to limit the amount of data you need to process in

the model when the data is updated.



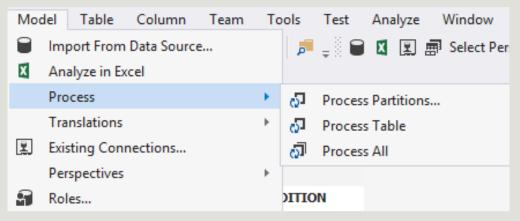
### **Partitions**



Cancel

SELECT [dbo].[CRASH\_DATA\_T].\* FROM [dbo].[CRASH\_DATA\_T]

where crash\_date >= '01/01/2012'



	Partitio	n Manager		? X
Use partitions to divide a table	into logical parts that can be processed indeper	ndently.		
Table:	CRASH_DATA_T			~
			Search Partition Names	۵
Partition Name		Last Processed		
CRASH_DATA_T		9/7/2016 6:54:53 PM		
*CRASH_DATA_GT_2015		9/12/2016 11:17:13 PM		
New Copy	Delete			
Details - CRASH_DATA_T 2   Details - CRASH_DATA_T 2	2			
Partition Name:	CRASH_DATA_GT_2015			
Connection:	SqlServer localhost Crash_Data_DB			
SQL Statement:				SQL
SELECT [dbo].[CRASH_DATA	A_T].*FROM [dbo].[CRASH_DATA_T]where cra	sh_date >= '01/01/2015'		
Validate Design			Last Processed: 9/12/2016 1	1:17:13 PM

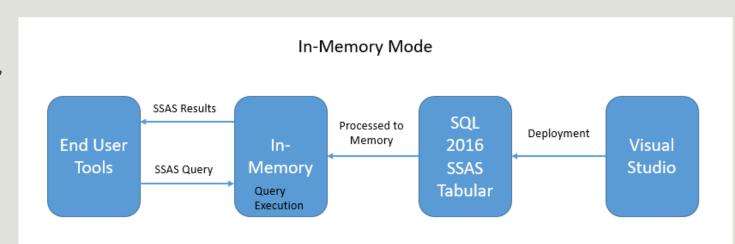
### In-Memory Vs. DirectQuery



**In-memory (Default)** Queries are answered using the data stored in cache.

First, since all data is cached in-memory, you have to ensure your server has memory, not only for the adequate dataset

Another limitation of in-memory mode is having to keep the data refreshed



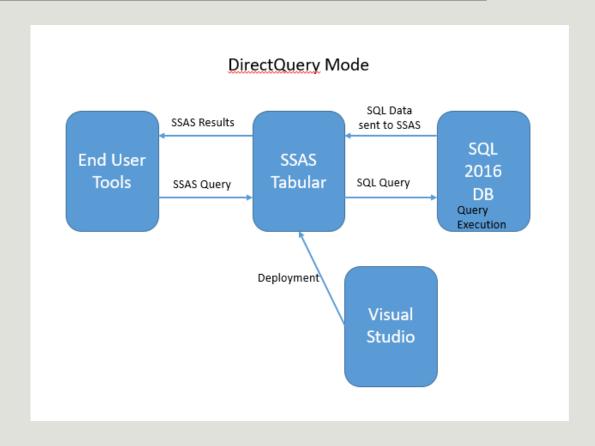
### In-Memory Vs. DirectQuery



**DirectQuery** Queries are answered by accessing the data directly from the relational database.

The default setting for your project is to have DirectQuery Mode turned off. While this setting is off, queries against the data will be directed at the in-memory VertiPaq cache.

Data is loaded into the VertiPaq cache when you process the model. One limitation to be mindful of is the amount of data you are loading and how much memory your server has available. The additional details on the benefits and limitations of DirectQuery will be provided in the later recipes.



### In-Memory Vs. DirectQuery



Feature	Restriction
Data source	Can only pull from a single relational database
Calculated tables	Calculated Tables are not supported in DirectQuery models
Query limit	By default DirectQuery is limited to 1 million rows
Stored procedures	Tables cannot be defined from stored procedures
DAX	formulas DAX formulas that cannot be converted to SQL syntax will return an error

