1. The example creates a view and an index on that view. Two queries are included that use the indexed view.

USE AdventureWorks;

GO

--Set the options to support indexed views.

SET NUMERIC\_ROUNDABORT OFF;

SET ANSI\_PADDING, ANSI\_WARNINGS, CONCAT\_NULL\_YIELDS\_NULL, ARITHABORT,

QUOTED\_IDENTIFIER, ANSI\_NULLS ON;

GO

--Create view with schemabinding.

IF OBJECT\_ID ('Sales.vOrders', 'view') IS NOT NULL

DROP VIEW Sales.vOrders ;

GO

CREATE VIEW Sales.vOrders

WITH SCHEMABINDING

AS

SELECT SUM(UnitPrice\*OrderQty\*(1.00-UnitPriceDiscount)) AS Revenue,

OrderDate, ProductID, COUNT\_BIG(\*) AS COUNT

FROM Sales.SalesOrderDetail AS od, Sales.SalesOrderHeader AS o

WHERE od.SalesOrderID = o.SalesOrderID

GROUP BY OrderDate, ProductID;

GO

--Create an index on the view.

CREATE UNIQUE CLUSTERED INDEX IDX\_V1

ON Sales.vOrders (OrderDate, ProductID);

GO

--This query can use the indexed view even though the view is

--not specified in the FROM clause.

SELECT SUM(UnitPrice\*OrderQty\*(1.00-UnitPriceDiscount)) AS Rev,

OrderDate, ProductID

FROM Sales.SalesOrderDetail AS od

JOIN Sales.SalesOrderHeader AS o ON od.SalesOrderID=o.SalesOrderID

AND ProductID BETWEEN 700 and 800

AND OrderDate >= CONVERT(datetime,'05/01/2002',101)

GROUP BY OrderDate, ProductID

ORDER BY Rev DESC;

GO

--This query can use the above indexed view.

SELECT OrderDate, SUM(UnitPrice\*OrderQty\*(1.00-UnitPriceDiscount)) AS Rev

FROM Sales.SalesOrderDetail AS od

JOIN Sales.SalesOrderHeader AS o ON od.SalesOrderID=o.SalesOrderID

AND DATEPART(mm,OrderDate)= 3

AND DATEPART(yy,OrderDate) = 2002

GROUP BY OrderDate

ORDER BY OrderDate ASC;

GO