6 Useful SQL Server Data Dictionary Queries Every DBA Should Have

Magda Lakomiec 2017-08-08 SQL · SQL Server · Data Dictionary · Queries

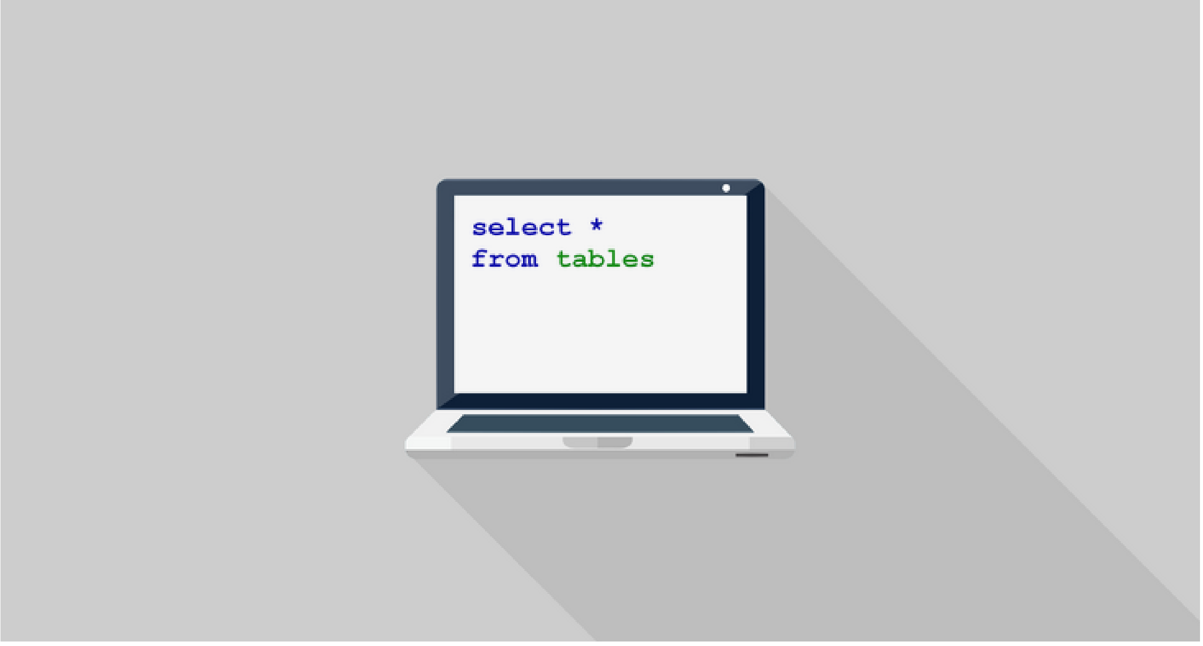


Table of Contents:

This is a list of handy SQL queries to the SQL Server data dictionary. You can also find 100+ other useful queries here.

comments and number of rows in each table.

Query

```
select schema_name(tab.schema_id) as schema_name,
       tab.name as table_name,
       tab.create_date as created,
       tab.modify_date as last_modified,
       p.rows as num_rows,
       ep.value as comments
  from sys.tables tab
       inner join (select distinct
                          p.object_id,
                          sum(p.rows) rows
                     from sys.tables t
                          inner join sys.partitions p
                              on p.object_id = t.object_id
                    group by p.object_id,
                          p.index_id) p
            on p.object_id = tab.object_id
        left join sys.extended_properties ep
            on tab.object_id = ep.major_id
           and ep.name = 'MS_Description'
           and ep.minor_id = 0
           and ep.class_desc = 'OBJECT_OR_COLUMN'
  order by schema_name,
        table_name
```

Rows

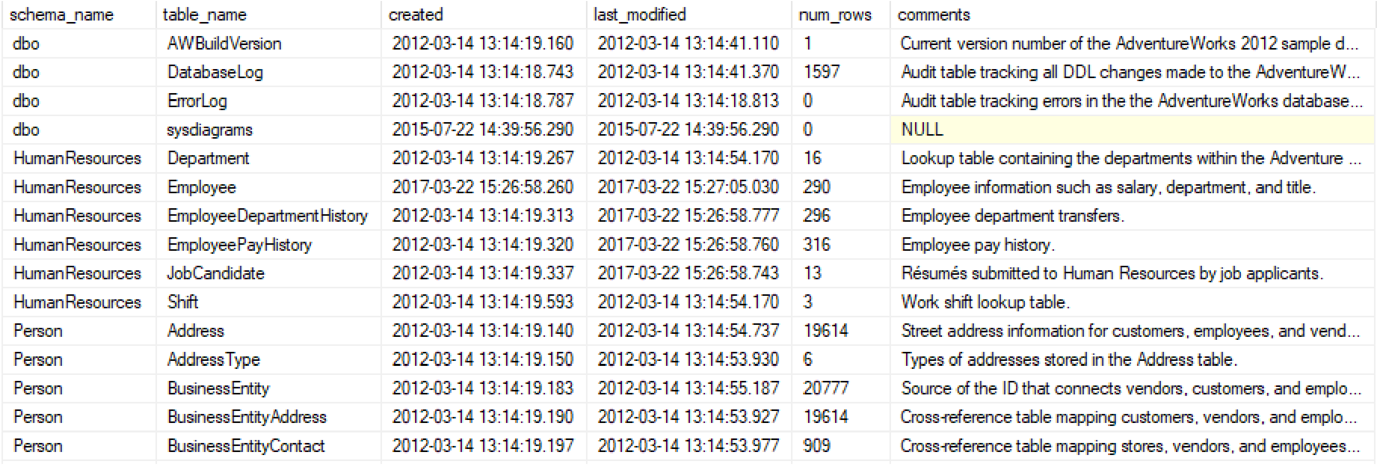
One row represents one table. All tables will be included.

Columns

Column	Meaning
SCHEMA_NAME	Schema name.
TABLE_NAME	Table name.
CREATED	Table creation date and time.
LAST_MODIFIED	Table last modification date and time.

comments	Table comments.

Sample results



schema_name	table_name	created	last_modified	num_rows	comments
dbo	AWBuildVersion	2012-03-14 13:14:19.160	2012-03-14 13:14:41.110	1	Current version number of the AdventureWorks 2012 sample d...
dbo	DatabaseLog	2012-03-14 13:14:18.743	2012-03-14 13:14:41.370	1597	Audit table tracking all DDL changes made to the AdventureW...
dbo	ErrorLog	2012-03-14 13:14:18.787	2012-03-14 13:14:18.813	0	Audit table tracking errors in the the AdventureWorks database...
dbo	sysdiagrams	2015-07-22 14:39:56.290	2015-07-22 14:39:56.290	0	NULL
HumanResources	Department	2012-03-14 13:14:19.267	2012-03-14 13:14:54.170	16	Lookup table containing the departments within the Adventure ...
HumanResources	Employee	2017-03-22 15:26:58.260	2017-03-22 15:27:05.030	290	Employee information such as salary, department, and title.
HumanResources	EmployeeDepartmentHistory	2012-03-14 13:14:19.313	2017-03-22 15:26:58.777	296	Employee department transfers.
HumanResources	EmployeePayHistory	2012-03-14 13:14:19.320	2017-03-22 15:26:58.760	316	Employee pay history.
HumanResources	JobCandidate	2012-03-14 13:14:19.337	2017-03-22 15:26:58.743	13	Résumés submitted to Human Resources by job applicants.
HumanResources	Shift	2012-03-14 13:14:19.593	2012-03-14 13:14:54.170	3	Work shift lookup table.
Person	Address	2012-03-14 13:14:19.140	2012-03-14 13:14:54.737	19614	Street address information for customers, employees, and vend...
Person	AddressType	2012-03-14 13:14:19.150	2012-03-14 13:14:53.930	6	Types of addresses stored in the Address table.
Person	BusinessEntity	2012-03-14 13:14:19.183	2012-03-14 13:14:55.187	20777	Source of the ID that connects vendors, customers, and emplo...
Person	BusinessEntityAddress	2012-03-14 13:14:19.190	2012-03-14 13:14:53.927	19614	Cross-reference table mapping customers, vendors, and emplo...
Person	BusinessEntityContact	2012-03-14 13:14:19.197	2012-03-14 13:14:53.977	909	Cross-reference table mapping stores, vendors, and employees...

2. List of views with definition and comments

This query returns list of database views with their definition SQL and a comment.

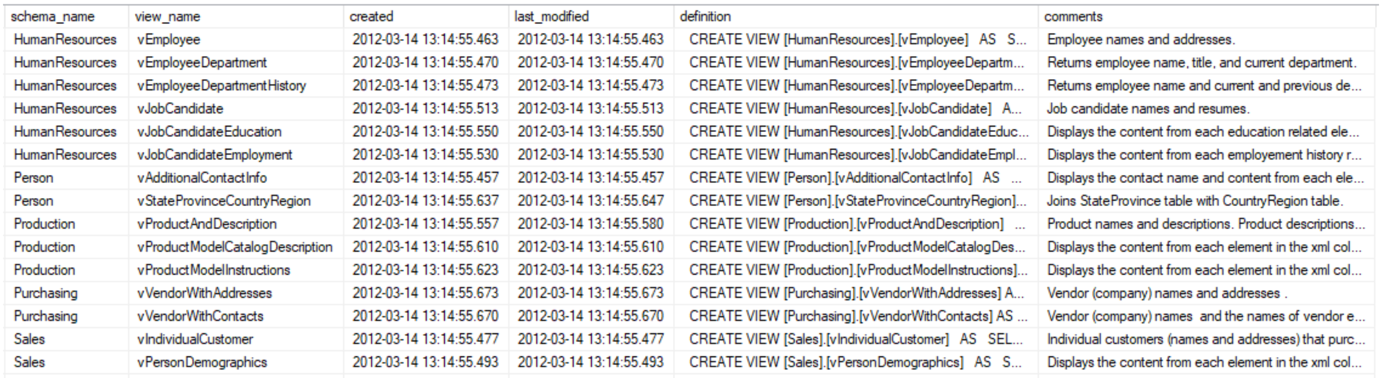
Query

```
select schema_name(v.schema_id) as schema_name,
       v.name as view_name,
       v.create_date as created,
       v.modify_date as last_modified,
       m.definition,
       ep.value as comments
  from sys.views v
       left join sys.extended_properties ep
           on v.object_id = ep.major_id
          and ep.name = 'MS_Description'
          and ep.minor_id = 0
          and ep.class_desc = 'OBJECT_OR_COLUMN'
       inner join sys.sql_modules m
           on m.object_id = v.object_id
 order by schema_name,
          view_name
```

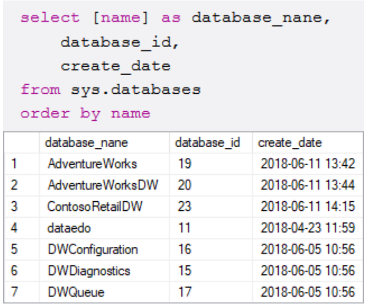
Columns

Column	Meaning
SCHEMA_NAME	Schema name.
VIEW_NAME	View name.
CREATED	View creation date and time.
LAST_MODIFIED	View last modification date and time.
DEFINITION	View definition (SQL query).
COMMENTS	View comments.

Sample results



schema_name	view_name	created	last_modified	definition	comments
HumanResources	vEmployee	2012-03-14 13:14:55.463	2012-03-14 13:14:55.463	CREATE VIEW [HumanResources].[vEmployee] AS S...	Employee names and addresses.
HumanResources	vEmployeeDepartment	2012-03-14 13:14:55.470	2012-03-14 13:14:55.470	CREATE VIEW [HumanResources].[vEmployeeDepartm...	Returns employee name, title, and current department.
HumanResources	vEmployeeDepartmentHistory	2012-03-14 13:14:55.473	2012-03-14 13:14:55.473	CREATE VIEW [HumanResources].[vEmployeeDepartm...	Returns employee name and current and previous de...
HumanResources	vJobCandidate	2012-03-14 13:14:55.513	2012-03-14 13:14:55.513	CREATE VIEW [HumanResources].[vJobCandidate] A...	Job candidate names and resumes.
HumanResources	vJobCandidateEducation	2012-03-14 13:14:55.550	2012-03-14 13:14:55.550	CREATE VIEW [HumanResources].[vJobCandidateEduc...	Displays the content from each education related ele...
HumanResources	vJobCandidateEmployment	2012-03-14 13:14:55.530	2012-03-14 13:14:55.530	CREATE VIEW [HumanResources].[vJobCandidateEmpl...	Displays the content from each employement history r...
Person	vAdditionalContactInfo	2012-03-14 13:14:55.457	2012-03-14 13:14:55.457	CREATE VIEW [Person].[vAdditionalContactInfo] AS ...	Displays the contact name and content from each ele...
Person	vStateProvinceCountryRegion	2012-03-14 13:14:55.637	2012-03-14 13:14:55.647	CREATE VIEW [Person].[vStateProvinceCountryRegion]...	Joins StateProvince table with CountryRegion table.
Production	vProductAndDescription	2012-03-14 13:14:55.557	2012-03-14 13:14:55.580	CREATE VIEW [Production].[vProductAndDescription] ...	Product names and descriptions. Product descriptions...
Production	vProductModelCatalogDescription	2012-03-14 13:14:55.610	2012-03-14 13:14:55.610	CREATE VIEW [Production].[vProductModelCatalogDes...	Displays the content from each element in the xml col...
Production	vProductModelInstructions	2012-03-14 13:14:55.623	2012-03-14 13:14:55.623	CREATE VIEW [Production].[vProductModelInstructions]...	Displays the content from each element in the xml col...
Purchasing	vVendorWithAddresses	2012-03-14 13:14:55.673	2012-03-14 13:14:55.673	CREATE VIEW [Purchasing].[vVendorWithAddresses] A...	Vendor (company) names and addresses .
Purchasing	vVendorWithContacts	2012-03-14 13:14:55.670	2012-03-14 13:14:55.670	CREATE VIEW [Purchasing].[vVendorWithContacts] AS ...	Vendor (company) names and the names of vendor e...
Sales	vIndividualCustomer	2012-03-14 13:14:55.477	2012-03-14 13:14:55.477	CREATE VIEW [Sales].[vIndividualCustomer] AS SEL...	Individual customers (names and addresses) that purc...
Sales	vPersonDemographics	2012-03-14 13:14:55.493	2012-03-14 13:14:55.493	CREATE VIEW [Sales].[vPersonDemographics] AS S...	Displays the content from each element in the xml col...



```
select [name] as database_nane,
    database_id,
    create_date
from sys.databases
order by name
```

	database_nane	database_id	create_date
1	AdventureWorks	19	2018-06-11 13:42
2	AdventureWorksDW	20	2018-06-11 13:44
3	ContosoRetailDW	23	2018-06-11 14:15
4	dataedo	11	2018-04-23 11:59
5	DWConfiguration	16	2018-06-05 10:56
6	DWDiagnostics	15	2018-06-05 10:56
7	DWQueue	17	2018-06-05 10:56

Query

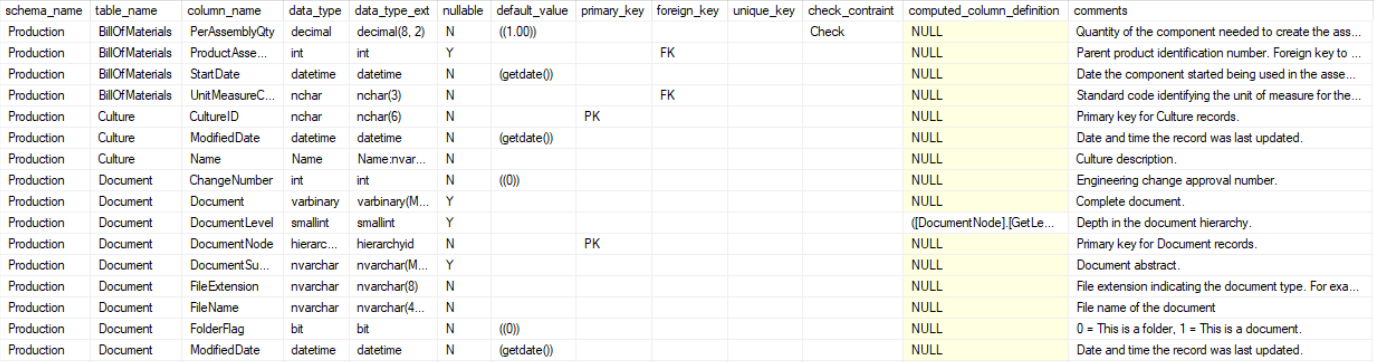
```
select schema_name(tab.schema_id) as schema_name,
       tab.name as table_name,
       col.name as column_name,
       t.name as data_type,
       t.name +
       case when t.is_user_defined = 0 then
                 isnull('(' +
                 case when t.name in ('binary', 'char', 'nchar',
                           'varchar', 'nvarchar', 'varbinary') then
                           case col.max_length
                                when -1 then 'MAX'
                                else
                                     case when t.name in ('nchar',
                                               'nvarchar') then
                                               cast(col.max_length/2
                                               as varchar(4))
                                          else cast(col.max_length
                                               as varchar(4))
                                     end
                           end
                      when t.name in ('datetime2', 'datetimeoffset',
                           'time') then
                           cast(col.scale as varchar(4))
                      when t.name in ('decimal', 'numeric') then
                            cast(col.precision as varchar(4)) + ', ' +
                            cast(col.scale as varchar(4))
                 end + ')', '')
            else ':' +
                 (select c_t.name +
                         isnull('(' +
                         case when c_t.name in ('binary', 'char',
                                   'nchar', 'varchar', 'nvarchar',
                                   'varbinary') then
                                    case c.max_length
                                         when -1 then 'MAX'
                                         else
                                              case when t.name in
                                                        ('nchar',
                                                        'nvarchar') then
                                                        cast(c.max_length/2
                                                        as varchar(4))
                              when c_t.name in ('datetime2',
                                   'datetimeoffset', 'time') then
                                   cast(c.scale as varchar(4))
                              when c_t.name in ('decimal', 'numeric') then
                                   cast(c.precision as varchar(4)) + ', '
                                   + cast(c.scale as varchar(4))
                         end + ')', '')
                    from sys.columns as c
                         inner join sys.types as c_t
                             on c.system_type_id = c_t.user_type_id
                   where c.object_id = col.object_id
                     and c.column_id = col.column_id
                     and c.user_type_id = col.user_type_id
                 )
        end as data_type_ext,
        case when col.is_nullable = 0 then 'N'
             else 'Y' end as nullable,
        case when def.definition is not null then def.definition
             else '' end as default_value,
        case when pk.column_id is not null then 'PK'
             else '' end as primary_key,
        case when fk.parent_column_id is not null then 'FK'
             else '' end as foreign_key,
        case when uk.column_id is not null then 'UK'
             else '' end as unique_key,
        case when ch.check_const is not null then ch.check_const
             else '' end as check_contraint,
        cc.definition as computed_column_definition,
        ep.value as comments
   from sys.tables as tab
        left join sys.columns as col
            on tab.object_id = col.object_id
        left join sys.types as t
            on col.user_type_id = t.user_type_id
        left join sys.default_constraints as def
            on def.object_id = col.default_object_id
        left join (
                  select index_columns.object_id,
                         index_columns.column_id
                    from sys.index_columns
                         inner join sys.indexes
                             on index_columns.object_id = indexes.object_id
                            and index_columns.index_id = indexes.index_id
                   where indexes.is_primary_key = 1
                  ) as pk
```

```
                       fc.parent_object_id
                  from sys.foreign_keys as f
                       inner join sys.foreign_key_columns as fc
                           on f.object_id = fc.constraint_object_id
                 group by fc.parent_column_id, fc.parent_object_id
                ) as fk
          on fk.parent_object_id = col.object_id
         and fk.parent_column_id = col.column_id
      left join (
                select c.parent_column_id,
                       c.parent_object_id,
                       'Check' check_const
                  from sys.check_constraints as c
                 group by c.parent_column_id,
                       c.parent_object_id
                ) as ch
          on col.column_id = ch.parent_column_id
         and col.object_id = ch.parent_object_id
      left join (
                select index_columns.object_id,
                       index_columns.column_id
                  from sys.index_columns
                       inner join sys.indexes
                           on indexes.index_id = index_columns.index_id
                          and indexes.object_id = index_columns.object_id
                  where indexes.is_unique_constraint = 1
                  group by index_columns.object_id,
                        index_columns.column_id
                ) as uk
          on col.column_id = uk.column_id
         and col.object_id = uk.object_id
      left join sys.extended_properties as ep
          on tab.object_id = ep.major_id
         and col.column_id = ep.minor_id
         and ep.name = 'MS_Description'
         and ep.class_desc = 'OBJECT_OR_COLUMN'
      left join sys.computed_columns as cc
          on tab.object_id = cc.object_id
         and col.column_id = cc.column_id
order by schema_name,
      table_name,
      column_name;
```

Columns

Column	Meaning
SCHEMA_NAME	Schema name.
TABLE_NAME	Table name.
COLUMN_NAME	Column name.
DATA_TYPE	Data type. For instance, varchar or decimal.
DATA_TYPE_EXT	Data type with information about scale/precision or string length. For instance, varchar(100) or decimal(8, 2).
NULLABLE	Nullable flag. "Y" if column is nullable, "N" if column is not nullable.
DEFAULT_VALUE	Column default value.
PRIMARY_KEY	Primary key flag. "PK" when column is part of table primary key.
FOREIGN_KEY	Foreign key flag. "FK" when column is part of foreign key.
UNIQUE_KEY	Unique key flag. "UK" when column is part of unique key.
CHECK_CONSTRAINT	Check constraint flag. "Check" when column is part of check constraint.
COMPUTED_COLUMN_DEFINITION	Computed column definition (not null only if column is computed).
COMMENTS	Column comments.

Sample results



schema_name	table_name	column_name	data_type	data_type_ext	nullable	default_value	primary_key	foreign_key	unique_key	check_contraint	computed_column_definition	comments
Production	BillOfMaterials	PerAssemblyQty	decimal	decimal(8, 2)	N	((1.00))				Check	NULL	Quantity of the component needed to create the ass...
Production	BillOfMaterials	ProductAsse...	int	int	Y			FK			NULL	Parent product identification number. Foreign key to ...
Production	BillOfMaterials	StartDate	datetime	datetime	N	(getdate())					NULL	Date the component started being used in the asse...
Production	BillOfMaterials	UnitMeasureC...	nchar	nchar(3)	N			FK			NULL	Standard code identifying the unit of measure for the...
Production	Culture	CultureID	nchar	nchar(6)	N		PK				NULL	Primary key for Culture records.
Production	Culture	ModifiedDate	datetime	datetime	N	(getdate())					NULL	Date and time the record was last updated.
Production	Culture	Name	Name	Name:nvar...	N						NULL	Culture description.
Production	Document	ChangeNumber	int	int	N	((0))					NULL	Engineering change approval number.
Production	Document	Document	varbinary	varbinary(M...	Y						NULL	Complete document.
Production	Document	DocumentLevel	smallint	smallint	Y						([DocumentNode].[GetLe...	Depth in the document hierarchy.
Production	Document	DocumentNode	hierarc...	hierarchyid	N		PK				NULL	Primary key for Document records.
Production	Document	DocumentSu...	nvarchar	nvarchar(M...	Y						NULL	Document abstract.
Production	Document	FileExtension	nvarchar	nvarchar(8)	N						NULL	File extension indicating the document type. For exa...
Production	Document	FileName	nvarchar	nvarchar(4...	N						NULL	File name of the document
Production	Document	FolderFlag	bit	bit	N	((0))					NULL	0 = This is a folder, 1 = This is a document.
Production	Document	ModifiedDate	datetime	datetime	N	(getdate())					NULL	Date and time the record was last updated.

Dataedo

Query

```
select schema_name(tab.schema_id) as table_schema_name,
       tab.name as table_name,
       col.name as column_name,
       fk.name as constraint_name,
       schema_name(tab_prim.schema_id) as primary_table_schema_name,
       tab_prim.name as primary_table_name,
       col_prim.name as primary_table_column,
       schema_name(tab.schema_id) + '.' + tab.name + '.' +
            col.name + ' = ' + schema_name(tab_prim.schema_id) + '.' +
            tab_prim.name + '.' + col_prim.name as join_condition,
       case
            when count(*) over (partition by fk.name) > 1 then 'Y'
            else 'N'
       end as complex_fk,
       fkc.constraint_column_id as fk_part
  from sys.tables as tab
       inner join sys.foreign_keys as fk
           on tab.object_id = fk.parent_object_id
       inner join sys.foreign_key_columns as fkc
           on fk.object_id = fkc.constraint_object_id
       inner join sys.columns as col
           on fkc.parent_object_id = col.object_id
          and fkc.parent_column_id = col.column_id
       inner join sys.columns as col_prim
           on fkc.referenced_object_id = col_prim.object_id
          and fkc.referenced_column_id = col_prim.column_id
       inner join sys.tables as tab_prim
           on fk.referenced_object_id = tab_prim.object_id
 order by table_schema_name,
       table_name,
       primary_table_name,
       fk_part;
```

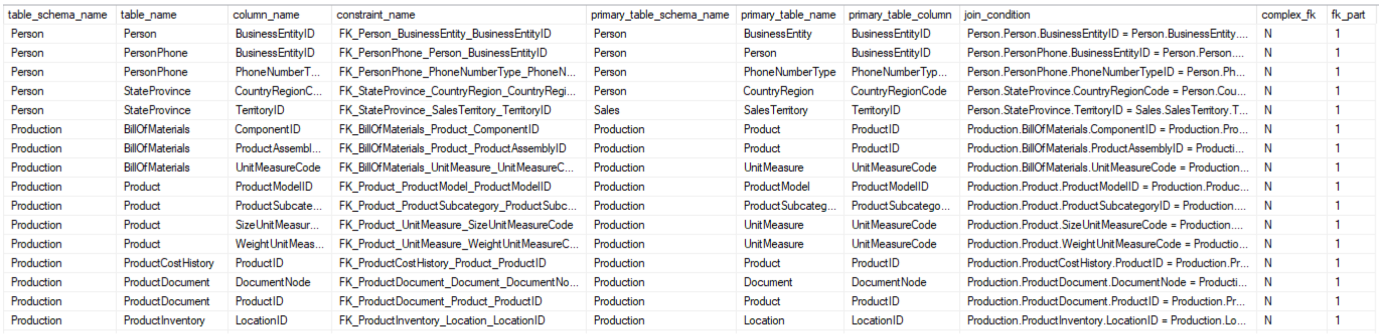
Rows

One row represents one pair of columns in foreign key.

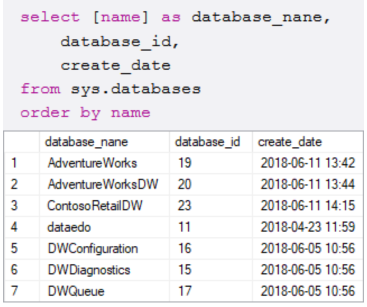
Columns

TABLE_NAME	Foreign table name.
COLUMN_NAME	Foreign table column name.
CONSTRAINT_NAME	Constraint name.
PRIMARY_TABLE_SCHEMA_NAME	Primary table schema name.
PRIMARY_TABLE_NAME	Primary table name.
PRIMARY_TABLE_COLUMN	Primary table column name.
JOIN_CONDITION	Join condition containing foreign and primary key tables and columns.
COMPLEX_FK	Complex foreign key flag. "Y" when foreign key is complex, otherwise "N".
FK_PART	Represents part number of foreign key.

Sample results



table_schema_name	table_name	column_name	constraint_name	primary_table_schema_name	primary_table_name	primary_table_column	join_condition	complex_fk	fk_part
Person	Person	BusinessEntityID	FK_Person_BusinessEntity_BusinessEntityID	Person	BusinessEntity	BusinessEntityID	Person.Person.BusinessEntityID = Person.BusinessEntity....	N	1
Person	PersonPhone	BusinessEntityID	FK_PersonPhone_Person_BusinessEntityID	Person	Person	BusinessEntityID	Person.PersonPhone.BusinessEntityID = Person.Person....	N	1
Person	PersonPhone	PhoneNumberT...	FK_PersonPhone_PhoneNumberType_PhoneN...	Person	PhoneNumberType	PhoneNumberTyp...	Person.PersonPhone.PhoneNumberTypeID = Person.Ph...	N	1
Person	StateProvince	CountryRegionC...	FK_StateProvince_CountryRegion_CountryRegi...	Person	CountryRegion	CountryRegionCode	Person.StateProvince.CountryRegionCode = Person.Cou...	N	1
Person	StateProvince	TerritoryID	FK_StateProvince_SalesTerritory_TerritoryID	Sales	SalesTerritory	TerritoryID	Person.StateProvince.TerritoryID = Sales.SalesTerritory.T...	N	1
Production	BillOfMaterials	ComponentID	FK_BillOfMaterials_Product_ComponentID	Production	Product	ProductID	Production.BillOfMaterials.ComponentID = Production.Pro...	N	1
Production	BillOfMaterials	ProductAssembl...	FK_BillOfMaterials_Product_ProductAssemblyID	Production	Product	ProductID	Production.BillOfMaterials.ProductAssemblyID = Producti...	N	1
Production	BillOfMaterials	UnitMeasureCode	FK_BillOfMaterials_UnitMeasure_UnitMeasureC...	Production	UnitMeasure	UnitMeasureCode	Production.BillOfMaterials.UnitMeasureCode = Production...	N	1
Production	Product	ProductModelID	FK_Product_ProductModel_ProductModelID	Production	ProductModel	ProductModelID	Production.Product.ProductModelID = Production.Produc...	N	1
Production	Product	ProductSubcate...	FK_Product_ProductSubcategory_ProductSubc...	Production	ProductSubcateg...	ProductSubcatego...	Production.Product.ProductSubcategoryID = Production....	N	1
Production	Product	SizeUnitMeasur...	FK_Product_UnitMeasure_SizeUnitMeasureCode	Production	UnitMeasure	UnitMeasureCode	Production.Product.SizeUnitMeasureCode = Production....	N	1
Production	Product	WeightUnitMeas...	FK_Product_UnitMeasure_WeightUnitMeasureC...	Production	UnitMeasure	UnitMeasureCode	Production.Product.WeightUnitMeasureCode = Productio...	N	1
Production	ProductCostHistory	ProductID	FK_ProductCostHistory_Product_ProductID	Production	Product	ProductID	Production.ProductCostHistory.ProductID = Production.Pr...	N	1
Production	ProductDocument	DocumentNode	FK_ProductDocument_Document_DocumentNo...	Production	Document	DocumentNode	Production.ProductDocument.DocumentNode = Producti...	N	1
Production	ProductDocument	ProductID	FK_ProductDocument_Product_ProductID	Production	Product	ProductID	Production.ProductDocument.ProductID = Production.Pr...	N	1
Production	ProductInventory	LocationID	FK_ProductInventory_Location_LocationID	Production	Location	LocationID	Production.ProductInventory.LocationID = Production.Lo...	N	1



```
select [name] as database_nane,
    database_id,
    create_date
from sys.databases
order by name
```

	database_nane	database_id	create_date
1	AdventureWorks	19	2018-06-11 13:42
2	AdventureWorksDW	20	2018-06-11 13:44
3	ContosoRetailDW	23	2018-06-11 14:15
4	dataedo	11	2018-04-23 11:59
5	DWConfiguration	16	2018-06-05 10:56
6	DWDiagnostics	15	2018-06-05 10:56
7	DWQueue	17	2018-06-05 10:56

Query

```
select schema_name(v.schema_id) as schema_name,
       v.name as view_name,
       col.name as column_name,
       t.name as data_type,
       t.name +
       case when t.is_user_defined = 0 then
                 isnull('(' +
                 case when t.name in ('binary', 'char', 'nchar',
                           'varchar', 'nvarchar', 'varbinary') then
                           case col.max_length
                                when -1 then 'MAX'
                                else
                                     case
                                         when t.name in ('nchar',
                                              'nvarchar') then
                                              cast(col.max_length/2
                                              as varchar(4))
                                         else cast(col.max_length
                                              as varchar(4))
                                     end
                           end
                      when t.name in ('datetime2',
                           'datetimeoffset', 'time') then
                            cast(col.scale as varchar(4))
                      when t.name in ('decimal', 'numeric') then
                           cast(col.precision as varchar(4)) + ', ' +
                           cast(col.scale as varchar(4))
                 end + ')', '')
            else ':' +
                 (select c_t.name +
                         isnull('(' +
                         case when c_t.name in ('binary', 'char',
                                   'nchar', 'varchar', 'nvarchar',
                                   'varbinary') then
                                   case c.max_length
                                        when -1 then 'MAX'
                                        else case when t.name in
                                                       ('nchar',
                                                        'nvarchar')
                                                  then cast(c.max_length/2
                                                       as varchar(4))
```

```
                             when c_t.name in ('datetime2',
                                  'datetimeoffset', 'time') then
                                  cast(c.scale as varchar(4))
                             when c_t.name in ('decimal', 'numeric') then
                                  cast(c.precision as varchar(4)) +
                                  ', ' + cast(c.scale as varchar(4))
                        end + ')', '')
                   from sys.columns as c
                        inner join sys.types as c_t
                            on c.system_type_id = c_t.user_type_id
                  where c.object_id = col.object_id
                    and c.column_id = col.column_id
                    and c.user_type_id = col.user_type_id
                )
      end as data_type_ext,
      case when col.is_nullable = 0 then 'N' else 'Y' end as nullable,
      ep.value as comments
 from sys.views as v
      join sys.columns as col
          on v.object_id = col.object_id
      left join sys.types as t
          on col.user_type_id = t.user_type_id
      left join sys.extended_properties as ep
          on v.object_id = ep.major_id
         and col.column_id = ep.minor_id
         and ep.name = 'MS_Description'
         and ep.class_desc = 'OBJECT_OR_COLUMN'
order by schema_name,
      view_name,
      column_name;
```

Rows

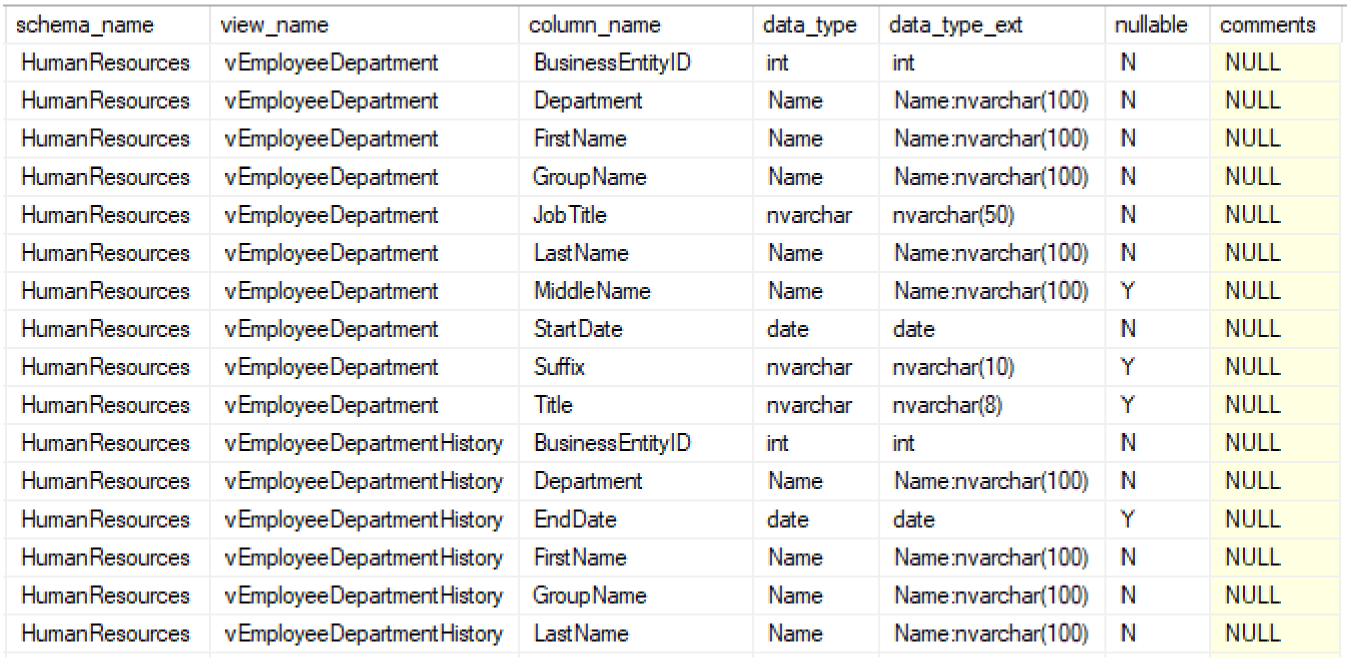
One row represents one view column.

Columns

Column	Meaning
SCHEMA_NAME	Schema name.
VIEW_NAME	View name.
COLUMN_NAME	View column name.

DATA_TYPE_EXT	Data type with information about scale/precision or string length. For instance, varchar(100) or decimal(8, 2).
NULLABLE	Nullable flag. "Y" if column is nullable, "N" if column is not nullable.
COMMENTS	Column comments.

Sample results



schema_name	view_name	column_name	data_type	data_type_ext	nullable	comments
HumanResources	vEmployeeDepartment	BusinessEntityID	int	int	N	NULL
HumanResources	vEmployeeDepartment	Department	Name	Name:nvarchar(100)	N	NULL
HumanResources	vEmployeeDepartment	FirstName	Name	Name:nvarchar(100)	N	NULL
HumanResources	vEmployeeDepartment	GroupName	Name	Name:nvarchar(100)	N	NULL
HumanResources	vEmployeeDepartment	JobTitle	nvarchar	nvarchar(50)	N	NULL
HumanResources	vEmployeeDepartment	LastName	Name	Name:nvarchar(100)	N	NULL
HumanResources	vEmployeeDepartment	MiddleName	Name	Name:nvarchar(100)	Y	NULL
HumanResources	vEmployeeDepartment	StartDate	date	date	N	NULL
HumanResources	vEmployeeDepartment	Suffix	nvarchar	nvarchar(10)	Y	NULL
HumanResources	vEmployeeDepartment	Title	nvarchar	nvarchar(8)	Y	NULL
HumanResources	vEmployeeDepartmentHistory	BusinessEntityID	int	int	N	NULL
HumanResources	vEmployeeDepartmentHistory	Department	Name	Name:nvarchar(100)	N	NULL
HumanResources	vEmployeeDepartmentHistory	EndDate	date	date	Y	NULL
HumanResources	vEmployeeDepartmentHistory	FirstName	Name	Name:nvarchar(100)	N	NULL
HumanResources	vEmployeeDepartmentHistory	GroupName	Name	Name:nvarchar(100)	N	NULL
HumanResources	vEmployeeDepartmentHistory	LastName	Name	Name:nvarchar(100)	N	NULL

6. Tables by number of columns

This query returns list of tables sorted by the number of columns they contain.

Query

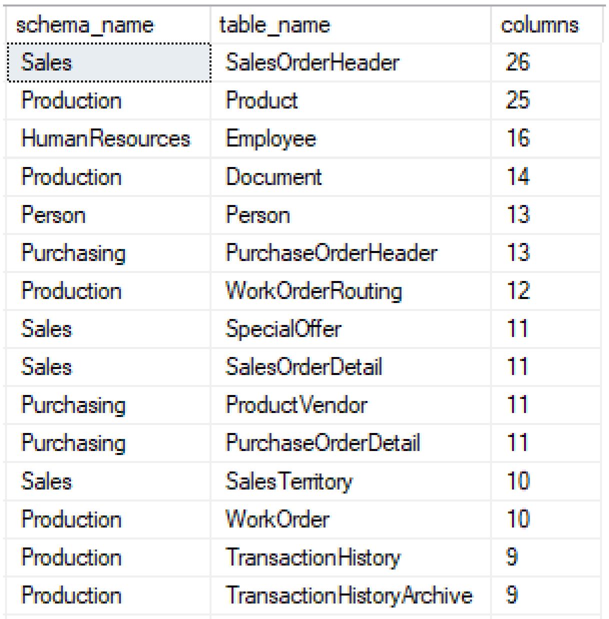
```
select schema_name(tab.schema_id) as schema_name,
       tab.name as table_name,
       count(*) as columns
  from sys.tables as tab
       inner join sys.columns as col
           on tab.object_id = col.object_id
 group by schema_name(tab.schema_id),
       tab.name
 order by count(*) desc;
```

Datædo

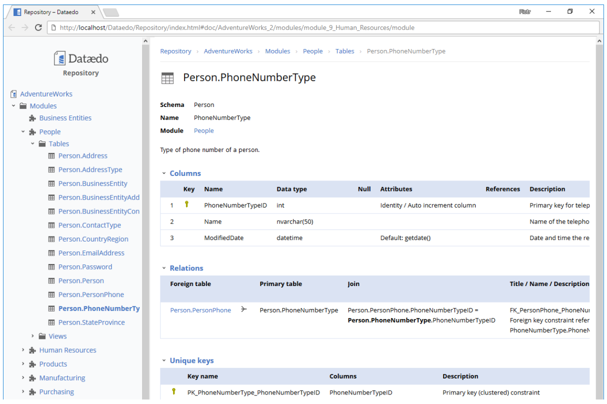
Columns

Column	Meaning
SCHEMA_NAME	Schema name.
TABLE_NAME	Table name.
COLUMNS	Number of columns table contains.

Sample results



schema_name	table_name	columns
Sales	SalesOrderHeader	26
Production	Product	25
HumanResources	Employee	16
Production	Document	14
Person	Person	13
Purchasing	PurchaseOrderHeader	13
Production	WorkOrderRouting	12
Sales	SpecialOffer	11
Sales	SalesOrderDetail	11
Purchasing	ProductVendor	11
Purchasing	PurchaseOrderDetail	11
Sales	SalesTerritory	10
Production	WorkOrder	10
Production	TransactionHistory	9
Production	TransactionHistoryArchive	9



Person.PhoneNumberType

Schema Person
Name PhoneNumberType
Module People

Type of phone number of a person.

Columns

	Key	Name	Data type	Null	Attributes	References	Description
1	🔑	PhoneNumberTypeID	int		Identity / Auto increment column		Primary key for telep
2		Name	nvarchar(50)				Name of the telepho
3		ModifiedDate	datetime		Default: getdate()		Date and time the re

Relations

Foreign table		Primary table	Join	Title / Name / Description
Person.PersonPhone	≻	Person.PhoneNumberType	Person.PersonPhone.PhoneNumberTypeID = **Person.PhoneNumberType**.PhoneNumberTypeID	FK_PersonPhone_PhoneNu Foreign key constraint refer PhoneNumberType.Phone

Unique keys

	Key name	Columns	Description
🔑	PK_PhoneNumberType_PhoneNumberTypeID	PhoneNumberTypeID	Primary key (clustered) constraint

Extract and share data dictionary from SQL Server

7 Comments **dataedo** **Disqus' Privacy Policy** 1 **Rajesh R Rajamani**

♡ **Recommend** 4 Tweet Share **Sort by Best**

Join the discussion…

Dmitriy Yudnikov • 2 months ago

Cool, thanks!!! (in bookmarks)

^ | ˅ • Reply • Share ›

Grade Rachmanda • 6 months ago

this very useful. awesome tricky of DBA.

^ | ˅ • Reply • Share ›

Jonathan Nigrine • a year ago • edited

Really useful! I created an SSRS report from this. I joined to INFORMATION_SCHEMA.COLUMNS to report and sort by ORDINAL_POSITION as well, and included data from both the table query and a similar one for views.

^ | ˅ • Reply • Share ›

Leon Carpay • a year ago

very useful

^ | ˅ • Reply • Share ›

Phil Gardocki • a year ago

Ditto, thanks

^ | ˅ • Reply • Share ›

Lucy Gray • a year ago

Thanks!

^ | ˅ • Reply • Share ›

Missy • 2 years ago

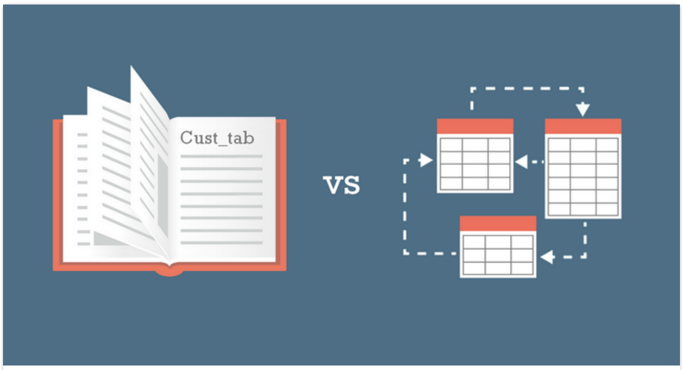
Awesome info! Thank you.

^ | ˅ • Reply • Share ›

Subscribe **Add Disqus to your siteAdd DisqusAdd** **Do Not Sell My Data**

Dataedo

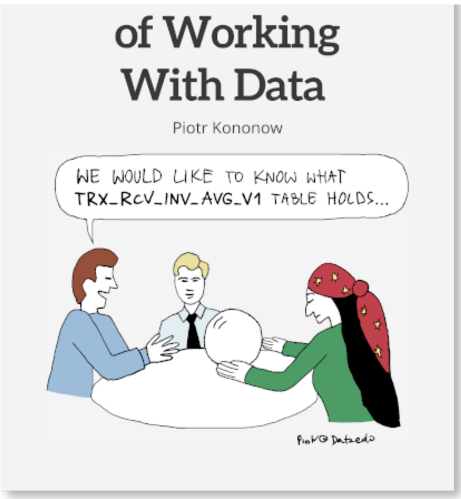
6 Useful Oracle Data Dictionary Queries Every DBA Should Have



ER Diagram vs Data Dictionary – Which is Better for Documenting Data Models

How to Document SQL Server Database in 5 Minutes with Dataedo [Free Tool]

Dataedo


of Working
With Data
Piotr Kononow
WE WOULD LIKE TO KNOW WHAT TRX_RCV_INV_AVG_V1 TABLE HOLDS...
Piotr@Dataedo