By Sergio Diez, Mario Celada, Sara Gómez

E) Datatypes
12.Transact SQL data types summary:

Binary Strings	DESCRIPTION	EXAMPLES
binary	**binary** [(n)] Fixed-length binary data with a length of n bytes, where n is a value from 1 through 8,000. The storage size is n bytes.	Images, Word files, text files...
image	Variable-length binary data from 0 through 2^31-1 (2,147,483,647) bytes.	Since image is deprecated, we should use varbinary. Images, Word files, text files...
varbinary	**varbinary** [(n | **max**)] Variable-length binary data. n can be a value from 1 through 8,000. **max** indicates that the maximum storage size is 2^31-1 bytes. The storage size is the actual length of the data entered $+$ 2 bytes. The data that is entered can be 0 bytes in length.	**binary** data types are used to store any kind of binary data like images, word files, text files.

Other Data Types		

cursor	A data type for variables or stored procedure OUTPUT parameters that contain a reference to a cursor.	Cursors allow you to scroll through, read and eventually modify this set of results. For example: go through a database table and display the value of a record on the screen.
hierarchyid	The **hierarchyid** data type is a variable length, system data type. Use **hierarchyid** to represent position in a hierarchy.	For example is used as a data type to create tables with a hierarchical structure or to describe the hierarchical structure of stored data in another location.
sql_variant	**sql_variant** can be used in columns, parameters, variables, and the return values of user-defined functions. **sql_variant** enables these database objects to support values of other data types. A column of type **sql_variant** may contain rows of different data types. For example, a column defined as **sql_variant** can store **int**, **binary**, and **char** values.	The **sql_variant** data type allows a table column or a variable to hold values of any data type with a maximum length of 8000 bytes plus 16 bytes that holds the data type information Not admitted values: varchar(max), varbinary(max), nvarchar(max), xml, text, ntext, image, rowversion (timestamp), geography, hierarchyid, geometry, datetimeoffset, User-defined types. For example, you can have two sql_variant variables a and b with values 5 and 2; if you write a + b you need to specify if you want to sum 5 and 2 which will result in 7 or if you want to concatenate 5 and 2 that will result in 52.

rowversion	Is a data type that exposes automatically generated, unique binary numbers within a database. **rowversion** is generally used as a mechanism for version-stamping table rows. Is a read-only column that SQL Server will update for you automatically, as if by trigger. You cannot insert or update into this column. Also, you may only have one rowversion column per table. For example, if a column value is 5, and an update statement sets the value to 5, this action is considered an update even though there is no change, and the **rowversion** is incremented.	SQL Server has a built-in data type that automatically marks changes to rows. This data type is "rowversion". Formerly this data type was named "timestamp", and Microsoft recommends you start using the new name. This data type can be used for 2 main purposes: **optimistic record locking and data queue processing.**
uniqueidentifier	It is a 16-byte SQL Server binary data type that is globally unique in tables, databases, and servers. The term GUID stands for Globally Unique Identifier and is used interchangeably with UNIQUEIDENTIFIER.	It is very difficult to remember a value of uniqueidentifier. For instance, we cannot use it for a ClientID, becasuse a client could not remenber a number like 6D48C22F-3BD3-4930-8515-5478097FBEA6.

xml	Is the data type that stores XML data. You can store **xml** instances in a column, or a variable of **xml** type.xml ([XMLFRAGMENT | XMLDOCUMENT] xml_schema_collection). To create a typed **xml** column or variable, you can optionally specify the XML schema collection name. "IndividualSurveySchemaCollection"	For example: <Nombre>Jose</Nombre> </Dependientes>','04/04/1994'), ('975281','ANTONIO MENDEZ VILLA','<Dependientes>
Spatial Geometry Types	This type represents data in a Euclidean (flat) coordinate system.It is substantially a two-dimensional rendering of an object and also useful in case of represented as points on a planar, or flat-earth data.	A good example of it is (10, 2) where the first number '10' identifies that point's position on the horizontal (x) axis and the number '2' represents the point's position on the vertical axis (y).
table	**Table** is primarily used for temporarily storing a set of rows that are returned as the table-valued function result set.	Table Type Variables are types of data that are normally used in a T-SQL batch, stored procedure or user-defined function. Table type variables are created and defined equal to tables with the difference that have a defined life span.

Spacial Geography Types	This type represents data in a round-earth coordinate system. The SQL Server **geography** data type stores ellipsoidal (round-earth) data, such as GPS latitude and longitude coordinates.The coordinates are represented as latitudinal and longitudinal degrees, as on a round-earth coordinate system.	The common use case of the Geography type is to store an application's GPS data.
Numeric Data type		
bit	Integer that can be 0, 1, or NULL	0, 1
tinyint	Allows whole numbers from 0 to 255	26, 240
smallint	Allows whole numbers between -32,768 and 32,767	30, -30
int	Allows whole numbers between -2,147,483,648 and 2,147,483,647	186784, -578756786
bigint	Allows whole numbers between -9,223,372,036,854,775,808 and 9,223,372,036,854,775,807	1234567898765456, -45678986657

decimal(p,s)	Fixed precision and scale numbers.Allows numbers from -10^38 +1 to 10^38 −1. The p parameter indicates the maximum total number of digits that can be stored (both to the left and to the right of the decimal point). p must be a value from 1 to 38. Default is 18.	0,00006 , 1,6585
numeric(p,s)	Fixed precision and scale numbers.Allows numbers from -10^38 +1 to 10^38 −1. The p parameter indicates the maximum total number of digits that can be stored (both to the left and to the right of the decimal point). p must be a value from 1 to 38. Default is 18.	-0,6756778
smallmoney	Monetary data from -214,748.3648 to 214,748.3647	23456.75 56.567
money	Monetary data from -922,337,203,685,477.5808 to 922,337,203,685,477.5807	123456787654345673.00 85756645776655776.758
float(n)	Floating precision number data from -1.79E + 308 to 1.79E + 308.	0.56647645676 124.755776

real	Floating precision number data from -3.40E + 38 to 3.40E + 38	0.2345678 23.5678
String data types		
char(n)	Fixed width character string	fgh
varchar(n)	Variable width character string	hola
text	Variable width character string	esto es un texto
Date and Hour		
datetime	From January 1, 1753 to December 31, 9999 with an accuracy of 3.33 milliseconds	11/01/1785 06:56:43 08/08/8965 12:09:33
datetime2	From January 1, 0001 to December 31, 9999 with an accuracy of 100 nanoseconds	10/10/0101 15:45:00
smalldatetime	From January 1, 1900 to June 6, 2079 with an accuracy of 1 minute	11/12/2019 22:33:33

date	Store a date only. From January 1, 0001 to December 31, 9999	04/06/2345
time	Store a time only to an accuracy of 100 nanoseconds	02:22:22
datetimeoffset	The same as datetime2 with the addition of a time zone offset	15/05/0101 15:48:40
String data types unicode		
nchar	Fixed width Unicode string	fjfk
nvarchar	Variable width Unicode string	njggjk
ntext	Variable width Unicode string	gfjj vfhufg hjb