|  | **Carrera:** Técnico Superior en Programación  **Materia:** Laboratorio de Computación III  **Tema:** Introducción a SQL Server |
| --- | --- |

**Introducción a SQL Server**

**Tipos de datos**

*Números enteros*

| **Tipo de dato** | **Descripción** | **Tamaño** |
| --- | --- | --- |
| bigint | Número entero de -2^63 (-9.223.372.036.854.775.808) a 2^63-1 (9.223.372.036.854.775.807) | 8 bytes |
| int | Número entero de -2^31 (-2.147.483.648) a 2^31-1 (2.147.483.647) | 4 bytes |
| smallint | Número entero de -2^15 (-32.768) a 2^15-1 (32.767) | 2 bytes |
| tinyint | Número entero de 0 a 255 | 1 byte |
| bit | Número entero de 0 (FALSE) a 1 (TRUE) | 1 bit |

*Números decimales y monetarios*

| **Tipo de dato** | **Descripción** | **Tamaño** |
| --- | --- | --- |
| decimal | Almacena un valor con precisión decimal (p, s) siendo *p* el parámetro que representa la cantidad de digitos del número y *s* el parámetro que representa la cantidad de digitos a la derecha de la coma.  Entonces, siendo un campo del tipo DECIMAL (10, 2). 8 digitos se utilizarán a la izquierda del separador decimal y 2 digitos a la derecha. | - |
| money | Representación monetaria de -922,337,203,685.477,5808 a 922,337,203,685.477,5807 | 8 bytes |
| smallmoney | Representación monetaria de - 214.748,3648 a 214.748,3647 | 4 bytes |
| float | Representación de datos numéricos con coma flotante de - 1,79E+308 a -2,23E-308, 0 y de 2,23E-308 a 1,79E+308 | - |

*Fecha*

| **Tipo de dato** | **Descripción** | **Tamaño** |
| --- | --- | --- |
| datetime | Fecha y hora desde el 1/1/1753 0:00:00.000 a 31/12/9999 23:59:59.997 | 8 bytes |
| smalldatetime | Fecha y hora desde el 1/1/1900 0:00:00 a 6/6/2079 23:59:59 | 4 bytes |

*Cadenas de caracteres*

| **Tipo de dato** | **Descripción** | **Tamaño** |
| --- | --- | --- |
| char (n) | Caracteres no Unicode de longitud fija. Con una longitud de *n* bytes. | - |
| varchar (n) | Caracteres no Unicode de longitud variable. Con una longitud de *n* bytes. | - |
| nchar (n) | Caracteres Unicode de longitud fija. Con una longitud de *n* bytes. | - |
| nvarchar (n) | Caracteres Unicode de longitud variable. Con una longitud de *n* bytes. | - |

**Anexo de funciones a utilizar**

**LTRIM (cadena )**

La función LTRIM( ) es una función de T-SQL que devuelve los caracteres de una cadena quitando todos los espacios en blanco de la izquierda. La función devuelve un tipo de dato varchar o nvharchar

/\* Obtiene el nombre de todos los empleados quitándole los espacios que pudieran tener a la izquierda del mismo \*/

SELECT LTRIM(nombre) FROM EMPLEADOS

**RTRIM ( cadena)**

La función RTRIM( ) es una función de T-SQL que devuelve los caracteres de una cadena quitando todos los espacios en blanco de la derecha. La función devuelve un tipo de dato varchar o nvharchar

/\* Obtiene el nombre de todos los empleados quitándole los espacios que pudieran tener a la derecha del mismo \*/

SELECT RTRIM(nombre) FROM EMPLEADOS

**GETDATE ( )**

La función GETDATE( ) es una función de T-SQL que devuelve la fecha y hora del sistema de base de datos actual con el formato datetime.

Ejemplo:

-- Obtiene la fecha del sistema

SELECT GETDATE() AS 'FECHA\_SISTEMA'

-- Obtiene los datos de todos los exámenes cuya fecha sea anterior a la de hoy.

SELECT \* FROM EXAMENES WHERE fecha\_examen < GETDATE()

**YEAR(fecha)**

La función YEAR() de T-SQL permite obtener el año a partir de una fecha que se ingresa como parámetro.

/\* Obtiene el año actual \*/

SELECT YEAR(GETDATE()) AS 'Año actual'

**MONTH(fecha)**

La función MONTH() de T-SQL permite obtener el mes a partir de una fecha que se ingresa como parámetro.

/\* Obtiene el mes de cumpleaños de cada empleado \*/

SELECT MONTH(fecha\_nacimiento) FROM EMPLEADOS

**DAY(fecha)**

La función DAY() de T-SQL permite obtener el día a partir de una fecha que se ingresa cómo parámetro.

/\* Obtiene el día de cumpleaños de cada empleado \*/

SELECT DAY(fecha\_nacimiento) FROM EMPLEADOS

**DATEADD(parte\_de\_fecha, cantidad, fecha)**

Esta función añade una *cantidad* de unidades de la *parte\_de\_fecha* que se especifique a la *fecha* que se envíe como parámetro. Su resultante será una fecha con el valor añadido.

-- Obtiene la fecha del día de mañana

SELECT DATEADD(day, 1, getdate()) as Mañana

-- Obtiene la fecha del día de ayer

SELECT DATEADD(day, -1, getdate()) as Ayer

-- Obtiene la fecha de cumpleaños número 33 de alguien nacido el 2/10/86

SELECT DATEADD(year, 33, '2/10/1986') as Vejez

**DATEDIFF(parte\_de\_fecha, fecha\_inicio, fecha\_fin)**

Es el resultado de la resta entre fecha\_inicio y fecha\_fin, expresado en cantidad de días, meses, años, etc.

*parte\_de\_fecha -* Indica qué parte de la fecha se va a utilizar para obtener el resultado de la resta entre fecha\_inicio y fecha\_fin. Puede expresarse en year (diferencia en años), month (diferencia en meses), day (diferencia en días), etc.

*fecha\_inicio -* La fecha de inicio a restar

*fecha\_fin -* La fecha de fin a restar

/\* La diferencia en días entre la fecha de nacimiento y la actual \*/

SELECT DATEDIFF(DAY, P.fecha\_nacimiento, GETDATE()) AS 'Días vividos' FROM PERSONAS P

**DATEPART(parte\_de\_fecha, fecha)**

Obtiene la parte de la fecha/tiempo del parámetro fecha que se indica.

*parte\_de\_fecha -* Indica qué parte de la fecha se va a obtener del parámetro fecha. Por ejemplo, DATEPART(HOUR, '28/09/2018 15:55') devolverá como resultado 15.

**CONVERT(tipo\_de\_dato, expresión, formato)**

*tipo\_de\_dato* - Especifica el tipo de dato a convertir.

*expresión* - Especifica el valor a convertir

*formato* - Especifica el formato de fecha y hora si corresponde

/\* Convierte un número a texto para poder concatenarlo \*/

SELECT APELLIDO + ' (' + CONVERT(VARCHAR(5), LEGAJO) + ')' AS 'APELLIDO Y LEGAJO' FROM ALUMNOS

/\* Convierte una fecha y hora a sólo fecha con formato dd/mm/yyyy - Más info de CONVERT -> <http://msdn.microsoft.com/es-ar/library/aa237895(v=SQL.80).aspx> \*/

SELECT CONVERT(NVARCHAR(20), FECHA, 103) AS FECHA FROM CALIFICACIONES

/\* Convierte una fecha y hora a sólo hora con formato hh:mi \*/

SELECT CONVERT(NVARCHAR(20), FECHA, 108) AS FECHA FROM CALIFICACIONES

**CAST (expresión AS tipo\_de\_dato)**

*expresión* - Especifica el valor a convertir

*tipo\_de\_dato* - Especifica el tipo de dato a convertir

/\* Convierte un entero a decimal \*/

SELECT CAST(LEGAJO AS DECIMAL(12, 2)) AS LEGAJO FROM ALUMNOS

**Elementos de programación**

**Condición simple**

/\* Muestra 'PLAN 2004' si el legajo es menor que 3000 y 'PLAN 2010' en caso contrario \*/

SELECT LEGAJO, CASE WHEN LEGAJO < 3000 THEN 'PLAN 2004' ELSE 'PLAN 2010' END AS 'PLAN' FROM ALUMNOS

/\*Obtiene un listado donde muestra 'Masculino' si el campo sexo contiene 'M' o 'Femenino' si contiene 'F' \*/

SELECT APELLIDO, NOMBRE, CASE WHEN SEXO = 'M' THEN 'MASCULINO' WHEN SEXO = 'F' THEN 'FEMENINO' END AS 'SEXO' FROM ALUMNOS

**Condición múltiple**

/\*Obtiene un listado donde muestra 'Masculino' si el campo sexo contiene 'M' o 'Femenino' si contiene 'F' \*/

SELECT APELLIDO, NOMBRE, CASE SEXO WHEN 'M' THEN 'MASCULINO' WHEN 'F' THEN 'FEMENINO' END AS 'SEXO'FROM ALUMNOS

**Bibliografía**

- Tipos de datos (Transact-SQL), MSDN Online, [Documento en línea], Consultado el 20 de marzo de 2012.