

Bases de Datos 2021-2021

MÁSTER UNIVERSITARIO EN DESARROLLO DE APLICACIONES Y SERVICIOS WEB

Más información: armando@dlsi.ua.es

Bases de Datos 2020-2021

Organización de contenidos y presencialidad

Las sesiones de Programación básica de servidor de base de datos

El software SQL Server

Sesión 1: Programación básica de servidor de base de datos (I)

Lotes y procedimientos almacenados

Programación básica

Programar lote

Procedimiento almacenado

Ejercicio puntuable P101

Organización de contenidos y presencialidad

La asignatura está dividida en dos periodos separados en el tiempo.

PARTE 1 T-SQL	PARTE 2 Hibernate
4 sesiones Transact-SQL NoSQL, mongoDB	4 sesiones ORM y Hibernate Acceso a BD Rendimiento Herramientas adicionales



Presentación del curso



[SQL](#)

Lecciones sobre SQL básico del grado; aunque se basan en MySQL, la mayoría de órdenes cumplen el estándar.

Las sesiones de Programación básica de servidor de base de datos

Estas sesiones están implementadas como una guía de recopilación de información que permita la resolución de ciertas tareas, sean cuestionarios, ensayos o entrega de archivos con la resolución de ciertos ejercicios.

El papel del profesor es el de proponer ciertos contenidos, el flujo de adquisición de los mismos, y la demostración de su comprensión mediante ejercicios no evaluables (por tanto, no requieren entrega de ningún tipo) y tareas que sí se entregan y califican.

Esta parte, que consta de 4 sesiones, se irá modificando y ampliando según se vaya constatando el desarrollo adecuado de las mismas. El objetivo de la planificación de éste y de los demás temas es destinar una sesión a conseguir los conocimientos necesarios para realizar, en la siguiente sesión, una práctica o conjunto de ejercicios diseñados para ser terminados en clase.

El software SQL Server

Lo que aquí vamos a ver no es un sistema real a pleno rendimiento. Si quisierais instalar una versión de SQL Server en casa, os sobra con la edición Express, de casi cualquier versión. La edición Express no precisa una gran cantidad de recursos de máquina y, además, es gratuita.

[Página de descargas de SQL Server](#)

Sesión 1: Programación básica de servidor de base de datos (I)

Esta sesión del curso Bases de Datos tiene un único objetivo: introducir las herramientas básicas del lenguaje Transact-SQL de SQL Server.

La forma de trabajar consiste en una breve descripción de elementos seleccionados del lenguaje y unos ejemplos o ejercicios sobre los mismos. Las descripciones incluyen enlaces a [la documentación de Microsoft](#), que es la réplica pública en Internet de los libros en pantalla que pueden instalarse en paralelo a SQL Server.

Lotes y procedimientos almacenados

El foco de esta sesión está en la presentación de herramientas de [T-SQL](#) que facilitan el proceso de los datos fuera de la programación procedimental tradicional, repartiendo la carga de trabajo entre el servidor y el cliente. Es bien sabido que el servidor de gestión de base de datos suele estar mejor preparado para trabajar con tablas que el cliente, tanto por cuestiones de eficiencia como de integridad y seguridad de los datos, al tiempo que ofrece una mayor independencia de datos. Desde este punto de vista, los procedimientos almacenados proporcionan persistencia a los procesos más o menos habituales que conforman nuestra programación, al tiempo que permiten el uso de funciones y procedimientos del sistema predefinidos y optimizados que permiten un desarrollo mucho más fácil y rápido.

Debe entenderse que este no es un curso de programación, se da por sabido todo lo fundamental y común a todos los lenguajes de este tipo, tan solo se incide en ciertos aspectos muy concretos. También es fundamental un dominio previo de SQL.

Aquí encontrarás una referencia a los elementos del lenguaje Transact-SQL que más utilizaremos: [T-SQL intro](#).




Programación básica

Para refrescar SQL, se propone un ejercicio de creación y población de tablas en SQL Server. En este ejercicio:

- | | |
|---|--|
| <ul style="list-style-type: none"> ● create database / drop database ● create table / drop table ● tipos de datos ● alter table | <ul style="list-style-type: none"> ● insert ● update ● delete ● select |
|---|--|

NOTA: toda esta secuencia de órdenes dependen del orden de ejecución por lo que, durante su desarrollo, si se producen mínimas variaciones (errores, reejecuciones, etc.) podría verse alterado el resultado final.

En cualquier caso, se trata de recuperar destreza sobre SQL Server, ese resultado final no tiene por qué ser exactamente igual, ni tan siquiera realizar las operaciones estrictamente como se proponen.

1. Crear una base de datos VENTAS (CREATE DATABASE, USE DATABASE)
2. Crear una tabla ARTÍCULO (puedes intentarlo en modo diseño pero recomendamos CREATE TABLE)
codigo (nchar(10), clave primaria),
nombre (nvarchar(50)),
marca (nvarchar(50)),
precio (numeric(6,2), no puede ser negativo)
3. Insertar (INSERT INTO; ahora SQL Server permite la forma compacta de inserción de varias filas típica de MySQL)
('0000000001', 'Lápiz Mágico', 'Rottrriing', 2.39);
('0000000002', 'Borrador Tenebroso', 'Rottrriing', 1.99);
('0000000003', 'Papel Inalterable', 'Rottrriing', NULL);
4. Consultar la tabla con resultados a cuadrícula y texto  (ver los botones de configuración en el entorno SQL Server Management Studio)
5. Añadir columnas cantidad (numeric(6,2)), unidad (nchar(10)) (ALTER TABLE; aquí el modo diseño puede resultar práctico)
6. Modificar cantidad, unidad (UPDATE)
('0000000001')>>(1,NULL); (en la fila identificada como '0000000001' actualizar las nuevas columnas)
('0000000002')>>(1,NULL);
('0000000003')>>(100,NULL);
7. Crear una tabla UNIDAD
unidad (nvarchar(20), clave primaria)
8. Insertar en UNIDAD
gramos
kilos
litros
9. Modificar ARTICULO añadiendo clave ajena(unidad)>>UNIDAD (ALTER TABLE, ADD FOREIGN KEY; ¡provocará un error! Ver 10.)
10. Realizar las siguientes modificaciones al esquema: (es una solución al error anterior, vamos a hacer coherentes clave primaria y ajena)
 - a. UNIDAD tendrá 2 columnas, unidad (CP) y descripción:
'u', 'unidades'
'g', 'gramos'
'k', 'kilogramos'
'l', 'litros'
 - b. ARTICULO.unidad debe cambiar de tipo de datos para coincidir con UNIDAD.unidad (ALTER TABLE)
 - c. ARTICULO.unidad no puede mantener nulos (nuevo error; ¿cómo lo solucionarías para poder ejecutar correctamente el ALTER TABLE?)



Si ya te has cansado... [aquí una solución](#).



Programar lote

Crear un lote que inserte 97 artículos más en la tabla ARTÍCULO, creada y modificada en los ejercicios anteriores, con datos aleatorios o semialeatorios de nombre, marca, precio y cantidad. Como se ha visto en el ejercicio anterior, la cantidad se complementa con la “unidad de medida”, gramos, litros o, si hace referencia al propio objeto, “unidades”.

- De ellos, 67 con unidades, 20 con gramos y 10 con litros.
 - No hace falta ser muy estricto con “cantidad”; el sentido común nos dice que lo que se mide en unidades debería tener una cantidad entera, mientras que con litros sí podríamos decir “litro y medio”. Sin embargo, en este ejercicio podemos almacenar “3.55 unidades”.
- Las marcas serán no más de 4 diferentes.
- Los precios irán de 0 a 999,99 y las cantidades de 0 a 99,99.

Necesitas, o podrías necesitar

- `set` (dar valor a una variable)
- `while`
- `rand()`
- `case`

[Resultado y solución](#)



Procedimiento almacenado

Un procedimiento almacenado es un código ejecutable que se guarda como objeto de la base de datos. Admite parámetros de entrada y de salida. Todo esto hace que sea reutilizable en cualquier momento mediante su invocación con la orden `execute`.

Previo

Crea una tabla `USUARIO` en la base de datos `VENTAS`. Para ello, copia, pega y ejecuta el siguiente código para generar la tabla y sus datos.

```
USE VENTAS
go
--if OBJECT_ID ( 'usuarios','U' ) is not null
--DROP table usuarios;
DROP table IF EXISTS usuarios;
go
create table usuarios (
email varchar(50) constraint pk_usuarios primary key,
passwd varchar(16),
nombre varchar(30))
go
insert into usuarios (email,passwd,nombre) values ('ptinto@ua.es','sdafsda','Pepe')
insert into usuarios (email,passwd,nombre) values
('pblanco.rosado@kraftt.au','13241','Juan')
insert into usuarios (email,passwd,nombre) values
('ddocupado@gmail.com','23fsdag','Paco')
insert into usuarios (email,passwd,nombre) values ('kkx2@gmail.com','345rf3','Pepe')
insert into usuarios (email,passwd,nombre) values
('f.lillo@agwab.com','vdrrg4','Pepe')
insert into usuarios (email,passwd,nombre) values
('jmf@hotmail.es','fsbd','Artuditu')
insert into usuarios (email,passwd,nombre) values ('im@hotmail.es','ffff','IronMan')
go
```

Procedimiento almacenado

Crea un procedimiento almacenado que acepte como parámetros los datos de un usuario y realice la inserción.

```
exec procusu 'asc@gmail.com', 'kkkk', 'Eva'
```

Necesitas conocer:

- la orden `EXECUTE`
- cómo manejar parámetros en un PA
- cómo imprimir resultados y mensajes de error
- cómo devolver valores

Descripción

Debe comprobar si es o no usuario nuevo, darlo de alta en caso afirmativo, o mostrar el mensaje de error pertinente. Como requisito, se exige codificar el tratamiento de errores mediante un bloque try-catch.

Mensajes en la salida de errores

- Si el usuario ya existe se debe mostrar en pantalla un mensaje con el siguiente texto: "Usuario ya existente. Por favor, elige otro identificador".
- Si no hubiera problema alguno, mostraría '¡Bienvenido!'.
- También podría ser un error inesperado a lo que el PA reacciona imprimiendo información del sistema sobre el mismo.

Devolución de valores (return)

- 0 si no hay errores,
- 1 si el usuario ya existía, o
- 2 en cualquier otro caso (se puede simular con una división por 0).

Hay varias posibilidades de programación

Para la comprobación:

- Se puede insertar directamente la nueva fila y capturar el error que nos va a producir la inserción de un identificador duplicado.
- Se puede consultar antes la tabla y mostrar el mensaje en caso de obtener resultado.

Para el código de error:

- Establecer un parámetro de salida.
- Aprovechar el valor devuelto al sistema por el procedimiento almacenado.



[Códigos de error MS SQL Server](#)

[Aquí la solución](#)



Ejercicio puntuable P101 (Moodle)



Ejercicio puntuable P101 plus (Moodle)



vídeos de ediciones anteriores del curso (versión 2008)



[SQL Server MS, entorno 1](#)

Vídeo descripción del interfaz de SQL Server Management Studio Express.

Primeros pasos: conectar, crear una base de datos, crear tablas, consultar, comentar, ejecución parcial.

El servidor remoto ya no existe, os conectaréis únicamente al servidor local.



[SQL Server MS, entorno 2](#)

Resultados a cuadrícula, a texto, modificar tablas, guardar resultados...



[SQL Server Management Studio Express: instalación](#)

Vídeo de instalación del cliente SQL Server 2008 Management Studio Express. Ciertamente obsoleto pero aún válido.