

Practica de Trigger y Stored Procedure

Utilizando la base de datos que se encuentra en el modelo al pie:

- 1) Realizar un Stored Procedure/Función denominada iniciales que reciba como parámetro un varchar y devuelva las iniciales. Ejemplo si se aplica a "Maradona Diego Armando" debe devolver "MDA".
- 2) Realizar un Procedimiento que reciba un Cuit y devuelva un atributo que determine si es valido o no.

```
werror = "".  
IF LENGTH(cuit_a_validar) <> 11  
  THEN werror = "Cuit con Menos de 11 Digos".  
  ELSE DO: wcontrol = 0.  
    FOR wi = 1 TO 10 DO  
      BEGIN  
        wcontrol = wcontrol +  
          INTEGER(SUBSTRING(cuit_a_validar,wi,1)) *  
          INTEGER(SUBSTRING("5432765432",wi,1)).  
      END.  
  
      IF (( 11 - (wcontrol MODULO 11)) MODULO 11 ) <>  
        INTEGER(SUBSTRING(cuit_a_validar,11,1))  
        THEN werror = "Digito Invalido".  
    END.  
IF werror = "" THEN msg "El cuit es valido"
```

Prototipo en Excel: Ver anexo Ejercicio 4: guia stored procedure – anexo ejercicio 4.xls

- 3) Agregue una nueva tabla denominada DEPARTAMENTOS que contendra los departamentos de la empresa con los siguientes datos:

ID_DPTO.	NOMBRE	ID_DPTO_PADRE
01	GERENCIA GRAL.	NULL.
10	GERENCIA ADMINISTRATIVA	01
15	GERENCIA COMERCIAL	01
30	TESORERIA	10
40	CONTADURIA	10
50	VENTAS	15
60	AREA NUEVOS PRODUCTOS	15
80	VENTAS NACIONALES	50
90	VENTAS INTERNACIONALES	50

Genere un Stored Procedure que reciba como parámetro un ID_DPTO y genere una fila por cada departamento dependiente de este con las columnas IID_DPTO, NOMBRE y DEPENDE

Barrido por niveles

C <-- Inicio busqueda (gralmente el nodo raiz)

```
          / P <-- C  
          |  
          | Tratar P  
          |  
mientras ( C tenga filas ) < Por cada hijo de P (i) < C <-- Hijo i (P)  
          |  
          |  
          \
```

```
create table departamentos  
( id smallint,  
  nombre varchar(40),  
  id_dpto_padre smallint  
);
```

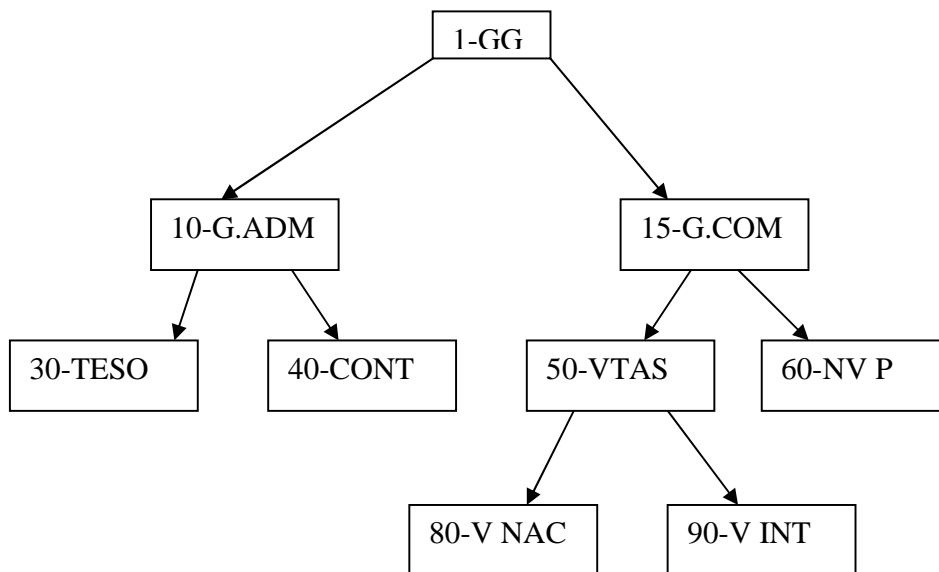
```

insert into departamentos values ( 1 , 'GERENCIA GRAL', NULL);
insert into departamentos values ( 10 , 'GERENCIA ADMINISTRATIVA', 1);
insert into departamentos values ( 15, 'GERENCIA COMERCIAL', 1);
insert into departamentos values ( 30, 'TESORERIA', 10);
insert into departamentos values ( 40, 'CONTADURIA', 10);
insert into departamentos values ( 50, 'VENTAS', 15);
insert into departamentos values ( 60, 'AREA NUEVOS PRODUCTOS', 15);
insert into departamentos values ( 80, 'VENTAS NACIONALES', 50);
insert into departamentos values ( 90, 'VENTAS INTERNACIONALES', 50);

```

Estrategia que

Armar una cola con una tabla temporal



4) Realizar un Stored Procedure que reciba como parámetros una tabla, un atributo y devuelva a que tipos de datos base pertenece y si es numérico el dominio del tipo base independientemente de las restricciones impuestas.

5) Genere un Stored Procedure que retorne una fila por cada tabla con 2 atributos, nombre de tabla y cantidad de filas.

El Procedimiento debe ser totalmente dinámico y no depender de su modificación para realizar el agregado o eliminación de tablas nuevas o borradas.

6) Genere un Stored Procedure que retorne una fila por cada tabla con 2 atributos, nombre de tabla y cantidad de filas.

El Procedimiento debe ser totalmente dinámico y no depender de su modificación para realizar el agregado o eliminación de tablas nuevas o borradas.

7) Agregue el atributo cantidad_items a la tabla Facturas y genere un trigger que mantenga la redundancia respecto de los items que posee cada factura.

8) Genere un trigger que no permita ingresar Productos de Familia A y C simultáneamente en la misma factura.

9) Genere un trigger que convierta el nombre de clientes a mayúsculas.

Modelo de Facturación Simple

