**INSTRUCCIONES REPETITIVAS**

En muchos casos, el valor de una o más variables o parámetros determinará el proceso a seguir. Cuando esto ocurre debemos usar instrucciones condicionales de tipo simple o IF cuando sólo hay una condición, alternativas (IF-THEN-ELSE) cuando hay dos posibilidades o CASE cuando hay múltiples condiciones. En este último caso se pueden anidar IF, si no son muchas, pero cuando son muchas es más sencillo utilizar el CASE. Vamos a ver estos diferentes tipos de instrucciones condicionales a través de unos ejemplos.

**SENTENCIA IF-THEN-ELSE**

La sintaxis general es:

IF exp1 THEN

…

ELSEIF exp2 THEN

…

ELSE

…

END IF

**TIENES QUE RECORDAR QUE TIENES QUE CERRAR TODOS LOS IF QUE ABRAS.**

**EJEMPLO IF**

Vamos a crear una función, dentro del esquema prueba, que devuelva MAYOR, MENOR o IGUAL dependiendo de que el número que metamos como parámetro sea mayor, menor o igual de 5.

**CREATE FUNCTION `NUM5` (X INT)**

**RETURNS VARCHAR(10)**

**BEGIN**

**DECLARE RESULTADO VARCHAR(10);**

**IF X>5 THEN**

**SET RESULTADO="MAYOR";**

**ELSE**

**IF X<5 THEN**

**SET RESULTADO="MENOR";**

**ELSE SET RESULTADO="IGUAL";**

**END IF;**

**END IF;**

**RETURN RESULTADO;**

**END**

Como puedes observar, hay 2 IF por lo que tengo que poner dos END IF de cierre.

Si en lugar de anidar los IF, usas el ELSEIF, sólo tendrías que poner un END IF.

Te pongo a continuación la misma función anterior con ELSEIF:

**CREATE FUNCTION `NUM5`(X INT) RETURNS varchar(10)**

**BEGIN**

**DECLARE RESULTADO VARCHAR(10);**

**IF X>5 THEN**

**SET RESULTADO="MAYOR";**

**ELSEIF X<5 THEN**

**SET RESULTADO="MENOR";**

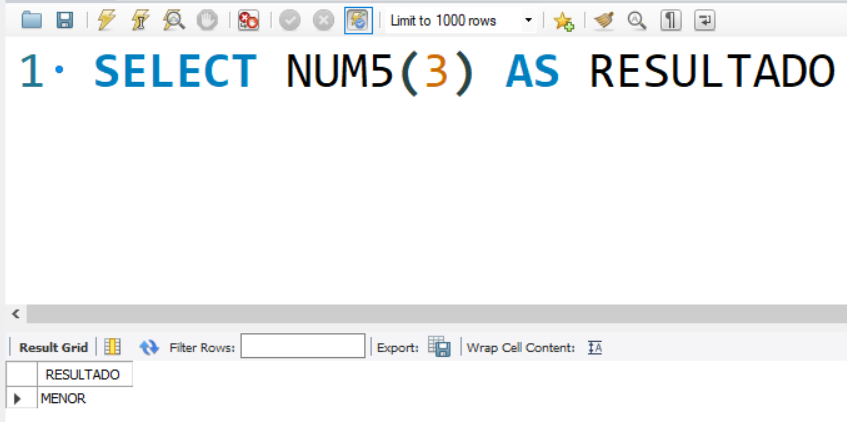
**ELSE SET RESULTADO="IGUAL";**

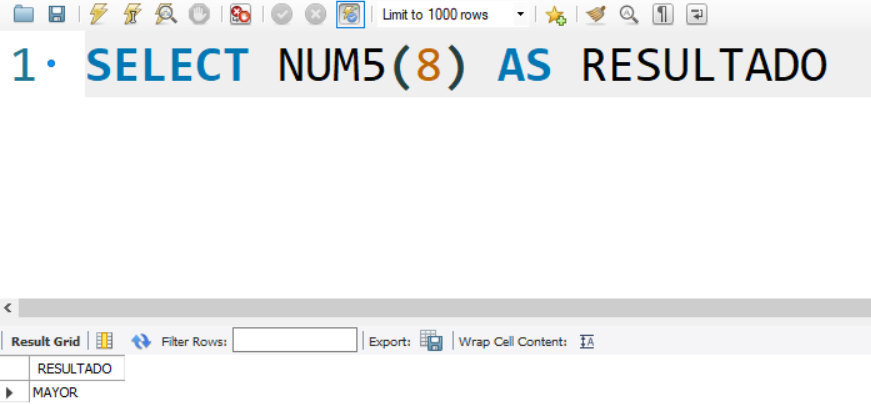
**END IF;**

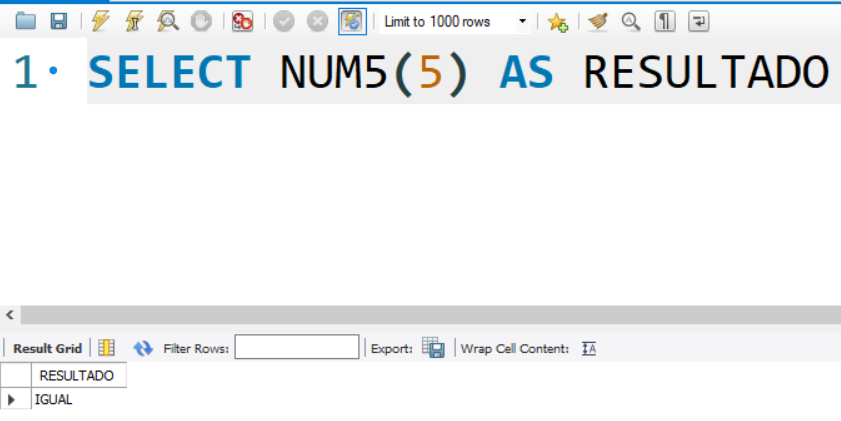
**RETURN RESULTADO;**

**END**

Si ejecutas la función, obtendrás el siguiente resultado.







Cuando hay muchas condiciones es más apropiado usar la instrucción CASE.

**SENTENCIA CASE**

CASE expresión

WHEN value THEN

Acción 1

[WHEN value THEN

Acción 2…….]

[ELSE

Acción 3]

END CASE;

Donde expresión puede tomar varios valores, dependiendo de los cuales se realiza una acción u otra.

**EJEMPLO CASE**

Vamos a ver un ejemplo para que lo entiendas. Imagina que quiero crear una función en la que meteremos un parámetro que va a ser un número entero. La función devolverá LUNES si meto un 1, MARTES si meto un 2…… DOMINGO si meto un 7 y ERROR si meto cualquier otro número.

**CREATE FUNCTION `DIASEMANA` (x INT)**

**RETURNS varchar(20)**

**BEGIN**

**DECLARE d VARCHAR(20);**

**CASE x**

**when 1 then set d="LUNES";**

**when 2 then set d="MARTES";**

**when 3 then set d="MIÉRCOLES";**

**when 4 then set d="JUEVES";**

**when 5 then set d="VIERNES";**

**when 6 then set d="SÁBADO";**

**when 7 then set d="DOMINGO";**

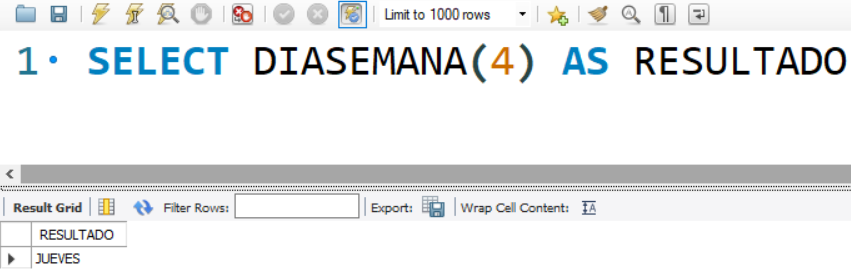
**ELSE SET d="ERROR";**

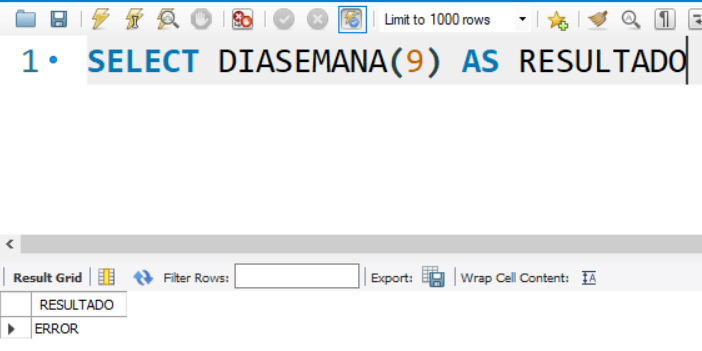
**END CASE;**

**RETURN d;**

**END**

Si la ejecutas, obtendrás lo siguiente:





**EJERCICIOS:**

1. **Crea una función que pida dos números y devuelva 1 si el primer número es divisible por el segundo y 0 en caso contrario.**
2. **A continuación, modifica el ejercicio anterior para que me devuelva primer número es divisible entre segundo número o primer número no es divisible entre segundo número.**

Ejemplo: Si meto un 4 y un 2, el mensaje que me devolverá será:

4 es divisible entre 2.

Puedes usar el CONCAT o el CONCAT\_WS para conseguir la cadena

1. **Usa el CASE para mostrar el nombre de los meses del año según un valor de entrada numérico, 1 para enero, 2 febrero….12 diciembre. Si el número es menor que 1 o mayor que 12, devolverá ERROR.**