

Ejercicio 1. (3 horas)

Utilizar su User ABAP correspondiente.

Crear un Módulo de Función Z que cuente con las siguientes características:

El programa se llamará **ZU60_PROG_MF**, el Módulo de Funciones lo

llamaremos **ZU60_FUNCION_MAT** y el Grupo de funciones **ZU60_GRUPO**.

Parámetros:

Parámetros de Entrada:

- VALOR1 del tipo ERFMG
- VALOR2 del tipo ERFMG
- OPERADOR del tipo CHAR1

Parámetros de Salida:

- DESCRIPCION del tipo CHAR20

Estructura de Salida: ZU60_OPERACIONES

- ITAB con los siguientes campos
 - SUMA del tipo ERFMG -> Campo referencia BSEG- ERFME
 - RESTA del tipo ERFMG
 - MULTI del tipo ERFMG
 - DIV del tipo ERFMG
 -

Structure	ZU60_OPERACIONES	New				
Short Description	Estructura para operaciones aritmeticas					
Attributes Components Entry help/check Currency/quantity fields						
1 / 4						
Component	Typing Method	Component Type	Data Ty...	Reference table	Ref. field	Short Description
SUMA	1 Types	ERFMG	QUAN	BSEG	ERFME	Quantity in Unit of Entry
RESTA	1 Types	ERFMG	QUAN	BSEG	ERFME	Quantity in Unit of Entry
MULTI	1 Types	ERFMG	QUAN	BSEG	ERFME	Quantity in Unit of Entry
DIV	1 Types	ERFMG	QUAN	BSEG	ERFME	Quantity in Unit of Entry

Parámetros de Excepción:

- DIV_CERO División por cero.
- NOT_FOUND Parámetro no encontrado.

¿Qué debe hacer el Módulo de Función?

La función que vamos a crear, deberá cumplir con las siguientes 4 especificaciones:

- Se debe tomar el símbolo del parámetro de entrada **OPERADOR**, y escribir su significado en el parámetro de salida **DESCRIPCION**.
- Se debe tomar los parámetros de entrada **VALOR1** y **VALOR2** para realizarle las 4 operaciones aritméticas, e ir guardando los resultados en la estructura de salida **ITAB**.
- Se debe validar que el **VALOR2** no tenga valor 0 (cero), porque daría un Dump al intentar dividir, por lo tanto, se debe controlar la excepción **DIV_CERO**.
- Se debe validar que el **OPERADOR** introducido de entrada sea una de las 4 operaciones aritméticas, de tratarse de alguna diferente, se debe controlar la excepción **NOT_FOUND**.

Ejemplos:


Caso 1:			Caso 2:			Caso 3:		
Si se llama a la función con estos valores:								
valor1 =	3,2		valor1 =	4,8		valor1 =	4,8	
valor2 =	2,5		valor2 =	0		valor2 =	2,5	
operador = +			operador = /			operador = >		
El resultado debería ser el siguiente:			El resultado debería ser el siguiente:			El resultado debería ser el siguiente:		
EXPORT			Ni EXPORT ni TABLAS debería			Ni EXPORT ni TABLAS debería		
Descripcion = SUMA			devolver datos			devolver datos		
TABLAS			EXCEPTION			EXCEPTION		
OPER			DIV_CERO (activo)			NOT_FOUND (activo)		
RESUL								
SUMA	5,7							
RESTA	0,7							
MULTI	8							
DIVISION	1,28							
EXCEPTION								
(en blanco)								

Contenido del Programa Principal

El Módulo de Función deberá ser llamado desde un programa principal, el cual tendrá las siguientes especificaciones:

- Se debe crear un programa principal que tenga como parámetros de selección, los 3 parámetros de entrada de la función, **VALOR1**, **VALOR2** y **OPERADOR**. El programa llamará al Módulo de Función pasándole estos 3 campos.

Prueba Módulo de Función



Datos de Entrada


Valor 1	<input type="text" value="3,2"/>
Valor 2	<input type="text" value="2,5"/>
Operador	<input type="text" value="+"/>

- Si los resultados son correctos (caso de ejemplo 1), tomar los resultados de la función y mostrarlos en un reporte con la sentencia WRITE.

Prueba Módulo de Función					
Descripción: SUMA					
Suma:	5,70	Resta:	0,70	Multi:	8,00
		División:	1,28		

- Si los resultados no son correctos, capturar la excepción correspondiente (casos 2 y 3), y mostrar el respectivo mensaje de error por pantalla.

Prueba Módulo de Función




Datos de Entrada

Valor 1	<input type="text" value="4,80"/>
Valor 2	<input type="text" value="0"/>
Operador	<input type="text" value="/"/>

Resultado:

☒ División por cero no permitida


Prueba Módulo de Función



Datos de Entrada

Valor 1	4,80
Valor 2	2,5
Operador	>

Resultado:

 Operador no encontrado

Ejercicio 2. (3 horas)

Realizar un Reporte que liste todos los vuelos dado una Aerolínea ingresada por el usuario.

El Layout del Reporte deberá ser el Siguiente:

Codigo de Aerolinea: XX

Nombre de la Aerolinea: XXXXXX

Moneda: XXX

Aerolínea Conexión Fldate Cityfrom Cityto Price

Aclaración: El Listado debe imprimirse en el programa, sin embargo la búsqueda de información se debe realizar en un modulo de función debido a que varios programadores van a querer reutilizarla. (Tablas SPFLI, SFLIGHT y SCARR)