

## MÓDULO 2 | PROGRAMACION COBOL- Estructuras Avanzadas: REGISTROS

## 1 ESTRUCTURA DE DATOS : REGISTRO

Una declaración de registro consiste en un conjunto de declaraciones de datos, las cuales describen las características de un registro en particular.

La declaración de una estructura de datos de tipo registro dentro de la DATA DIVISION consta de un conjunto de atributos organizados jerárquicamente en forma de árbol determinada por el nivel que se asigne a cada uno de estos.

## 1.1 .DECLARACION

Cada declaración de registro está formada por un numero de nivel seguido de un nombre de datos, o de la cláusula *FILLER*, seguido de una serie de cláusulas si es que estas son requeridas.

Una descripción de registro puede tener una estructura jerarquizada compuesta por una o mas inscripciones jerarquizadas o campos que escribiremos de acuerdo con el siguiente formato:

	nombre de datos
número de nivel	FILLER

[REDEFINES]
[PICTURE]
[USAGE]
[VALUE]
[SIGN]
[JUSTIFIED RIGTH]
[BLANK WHEN ZERO]
[OCCURS]

El orden de estas cláusulas puede ser cualquiera, excepto la cláusula REDEFINES, que debe ser la primera, en el caso de especificarse. Toda inscripción debe terminar con un punto.

El nombre del registro lleva numero de nivel 01 y el resto de los campos lleva un numero de nivel de 02 a 49.

Los números de nivel no tienen que ser correlativos, pero sí deben ir en orden ascendente para mantener esta jerarquía.

El nivel de jerarquía que contenga el nombre del registro debe usarse *una sola vez*, en caso de utilizar dos niveles superiores iguales, COBOL interpretará al segundo como una redefinición del primero.

En el siguiente ejemplo se declara una estructura de datos simple que ocupa 117 bytes en memoria.



```
1 2 3 4 5 6 7
123456 8901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123...

DATA DIVISION.
WORKING-STORAGE SECTION.
05 CATOS-PERSONA PIC X(117).
```

Ejemplo 5 – Definición de un registro simple

La siguiente declaración tambien ocupa 117 bytes en memoria, pero se define como un grupo de campos bajo el nombre *DATOS-PERSONA*, que es el nivel superior del registro. La longitud de *DATOS-PERSONA* está dada por la sumatoria de todos los campos agrupados en forma jerárquica dentro de su estructura.

```
2
         1
                               3
                                                     5
                                                               6
12345678901234567890123456789012345678901234567890123456789012345678901234567890123\dots
       DATA DIVISION.
       WORKING-STORAGE SECTION.
       01 IATOS-PERSONA.
           02 NOMBRE
                           PIC X(40).
           02 DIRECCION.
            03 CALLE
                           PIC X(35).
                           PIC X(20).
            03 CIUDAD
            03 PROVINCIA PIC X(15).
           02 TELEFONO
                           PIC X(7).
```

Ejemplo 6 – Definición de un registro compuesto

## 2 Cláusula Redefines

Permite asignar a un mismo área de memoria distintas estructuras de datos, e incluso nombres, para almacenarlos. Para ello una estructura **redefine** otra, siempre y cuando compartan el mismo nivel, y este no sea 66 ni 88. Su sintaxis es la siguiente:

```
1 2 3 4 5 6 7

123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123...

05 FECHA.
10 DIA PIC 9(02).
10 MES PIC 9(02).
10 ANIO PIC 9(02).
05 FECHA-ACTUAL REDEFINES FECHA PIC X(06).
```

Ejemplo 7 – Definición REDEFINES

La redefinición puede hacerse de una estructura compleja a una más simple, o al revés. No existen limitaciones mas allá de deber compartir el mismo nivel en la declaración y la misma longitud.

Ejemplo 8 - Definición REDEFINES



Utilizando la definición de los ejemplos anteriores, veremos un ejemplo de aplicación donde al alterar uno de los campos, el cambio se ve reflejado en ambas estructuras:

```
MOVE '220806' TO FECHA-ACTUAL

FECHA-ACTUAL

2 2 0 8 0 6

DIA MES ANIO

FECHA

DISPLAY FECHA-ACTUAL -> 220806

DISPLAY FECHA -> 220806

DISPLAY MES -> 08

MOVE 9 TO MES

FECHA-ACTUAL

2 2 0 9 0 6

DIA MES ANIO

FECHA
```

Ejemplo 9 – Aplicación REDEFINES