

## Clase sincrónica N°5 - Almacenamiento de datos y Tipos de archivos

### Zoneado o display (EBCDIC)

El byte se divide en zona y dígito propiamente dicho



#### Ejemplo:

almacenamiento del número 25679

almacenamiento número > 0

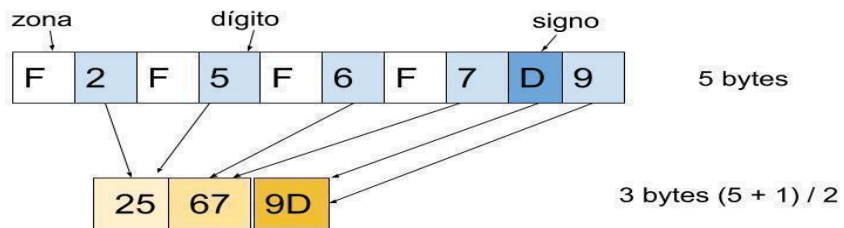


representación del número: -25679

almacenamiento número < 0

## Empaquetados o packed

Reduce la cantidad de espacio necesario para almacenar números. Se utiliza para importes o cualquier número de muchos dígitos.



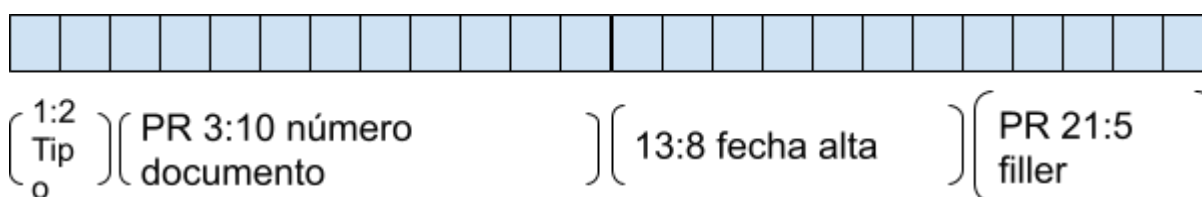
Con el mismo ejemplo:  
almacenamiento del número - 25679

almacenamiento packed número < 0

## ALMACENAMIENTO DE UN REGISTRO DE LARGO FIJO EN MEMORIA DE PROGRAMA.

**LAYOUT REGISTRO** : largo de registro 25 bytes.

|                 |           |
|-----------------|-----------|
| -Tipo documento | PIC XX    |
| - nro documento | PIC 9(10) |
| - fecha de alta | PIC 9(08) |
| - filler        | PIC X(05) |



## ¿Qué es un archivo?

Un archivo o fichero de datos es un conjunto o colección de registros que contienen los datos referidos a una estructura definida previamente y que son almacenados en el Mainframe con un nombre determinado; el cual responde a la naturaleza de los datos que contiene (**ejemplo: archivo de personas**).

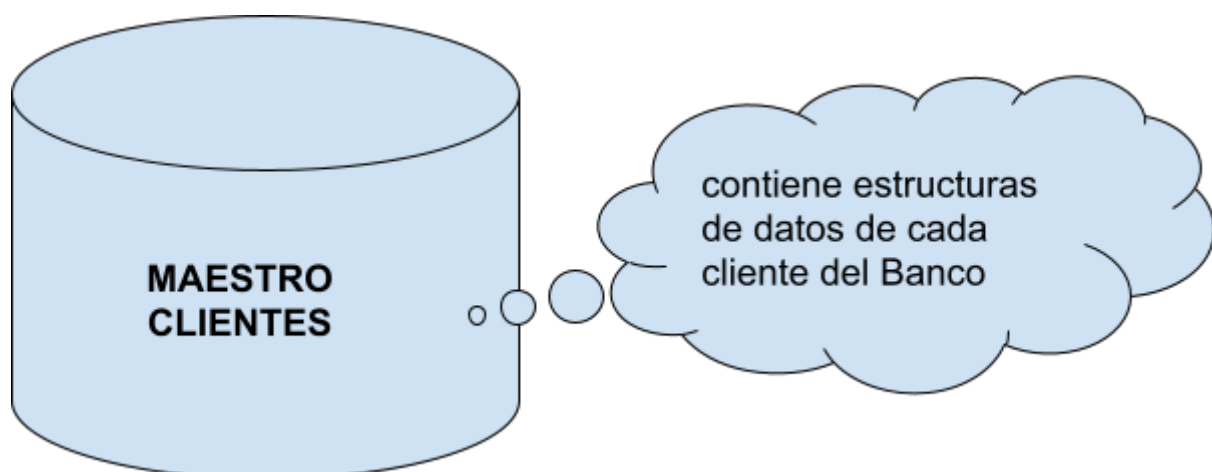
## CONCEPTO DE ARCHIVO – REGISTRO - CAMPO

**Archivo** = conjunto de registros. Un archivo se almacena en disco; contiene un agrupamiento o lote de datos que definen una clase específica o estructura de datos de la misma naturaleza. Ejemplo:

### ARCHIVO MAESTRO DE CLIENTES

Almacena los datos mínimos indispensables para la identificación exacta y precisa de todos los Clientes de una Entidad financiera o Banco. Los datos se agrupan en forma estructurada e independiente para cada Cliente.

El archivo representa el “lote de datos” que ingresan o que resultan de la salida de procesos batch que cumplen funciones específicas, codificadas en los programas COBOL del usuario.



**Registro** = conjunto de campos que contienen la información mínima indispensable para la identificación de una clase o estructura de datos de la misma naturaleza.

Contiene los datos mínimos indispensables para la identificación de cada Cliente del Total de la cartera de clientes de una Entidad financiera o Banco.

Ejemplo:

## ESTRUCTURA (LAYOUT) DE DATOS DEL MAESTRO DE CLIENTES

| NRO | NOMBRE                  | TIPO         | BYTES |
|-----|-------------------------|--------------|-------|
| 1   | CL-TIPO-DOCUMENTO       | ALFABÉTICO   | 2     |
| 2   | CL-NRO-DOCUMENTO        | NUMÉRICO     | 11    |
| 3   | CL-NRO-CLIENTE          | NUMÉRICO     | 5     |
| 4   | CL-NOMBRE-CLIENTE       | ALFABÉTICO   | 30    |
| 5   | CL-APELLIDO-CLIENTE     | ALFABÉTICO   | 30    |
| 6   | CL-DOMICILIO-LEGAL      | ALFANUMÉRICO | 30    |
| 7   | CL-CIUDAD-LEGAL         | ALFANUMÉRICO | 30    |
| 8   | CL-JURISDICCIÓN         | ALFANUMÉRICO | 30    |
| 9   | CL-PAIS-LEGAL           | ALFABÉTICO   | 30    |
| 10  | CL-CÓDIGO-POSTAL-LEGAL  | ALFANUMÉRICO | 8     |
| 11  | CL-DOMICILIO-REAL       | ALFANUMÉRICO | 30    |
| 12  | CL-CIUDAD-REAL          | ALFANUMÉRICO | 30    |
| 13  | CL-PROVINCIA            | ALFANUMÉRICO | 30    |
| 14  | CL-CÓDIGO-POSTAL-REAL   | ALFANUMÉRICO | 8     |
| 15  | CL-PAIS-REAL            | ALFABÉTICO   | 30    |
| 16  | CL-NACIONALIDAD         | ALFABÉTICO   | 30    |
| 17  | CL-MARCA-CIUDADANÍA     | ALFABÉTICO   | 2     |
| 18  | CL-FECHA-CIUDADANÍA     | DATE         | 8     |
| 19  | CL-FECHA-DE-ALTA        | DATE         | 8     |
| 20  | CL-FECHA-DE-BAJA        | DATE         | 8     |
| 21  | CL-ESTADO-CIVIL         | ALFABÉTICO   | 2     |
| 22  | CL-TIPO-INSCRIP-AFIP    | ALFANUMÉRICO | 2     |
| 23  | CL-SEXO                 | ALFABÉTICO   | 2     |
| 24  | CL-SEGURO-VIDA          | ALFANUMÉRICO | 15    |
| 25  | CL-TIPO-PERSONA         | ALFABÉTICO   | 2     |
| 26  | CL-CORREO-ELECTRÓNICO   | ALFANUMÉRICO | 60    |
| 27  | CL-FECHA-NACIMIENTO     | DATE         | 8     |
| 28  | CL-MOTIVO-BAJA          | ALFABÉTICO   | 60    |
|     |                         |              |       |
|     | LARGO REGISTRO: (BYTES) |              | 541   |

**Campo** = unidad mínima de datos dentro de un registro. Ejemplo:

Cada uno de los **28 campos** enumerados en el ejemplo del layout de registro.



## CONSTANTES FIGURATIVAS

Son constantes predefinidas de COBOL que pueden usarse en cualquier parte de la WORKING STORAGE SECTION o de la PROCEDURE DIVISION para darle valor a un campo. Son palabras reservadas que tienen un valor constante:

| ZERO – ZEROS - ZEROES     |  | Asigna valor 0                                         |
|---------------------------|--|--------------------------------------------------------|
| SPACE - SPACES            |  | Asigna espacios a un campo no numérico                 |
| HIGH -VALUE / HIGH-VALUES |  | Asigna a un campo el valor más alto que puede contener |
| LOW -VALUE / LOW-VALUES   |  | Asigna a un campo el valor más bajo que puede contener |
| ALL "literal"             |  | Representa una repetición del literal especificado     |

## TIPOS DE DATOS

Pueden usarse en cualquier parte de la WORKING STORAGE SECTION y son invocados desde la PROCEDURE DIVISION según lógica del código fuente:

| Tipo de carácter                        | Símbolo | Uso                                                                                                              | Ejemplo         |
|-----------------------------------------|---------|------------------------------------------------------------------------------------------------------------------|-----------------|
| Caracteres para la definición del campo | 9       | Campo numérico. Se pueden hacer operaciones. Máximo desde 18 posiciones hasta 32 bytes (según cada instalación). | 999 ó 9(03)     |
|                                         | X       | Campo alfanumérico                                                                                               | XXXX ó<br>X(04) |
|                                         | A       | Campo alfabético                                                                                                 | AA ó<br>A(02)   |

|                                                                                                                                                                             |      |                                                                                                |                        |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|------------------------------------------------------------------------------------------------|------------------------|
| Caracteres especiales para campos numéricos                                                                                                                                 | V    | Representa la posición de un punto decimal implícito. No forma parte de la longitud del campo. | 999V99 ó<br>9(03)V(02) |
|                                                                                                                                                                             | S    | Inclusión de signo operacional. Se utiliza para operaciones aritméticas.                       | S999 ó<br>S9(03)       |
| Caracteres de edición<br><br>(Cambio en el formato en que se muestran los datos. Fundamental para una clara presentación de los datos por pantalla o en un informe impreso) | \$   | Signo de pesos                                                                                 | \$\$\$9,99             |
|                                                                                                                                                                             | Z    | Sirve para reemplazar ceros no significativos por blancos.                                     | ZZZ.ZZZ.Z99,99         |
|                                                                                                                                                                             | *    | Para proteger las cantidades escritas en los cheques por asteriscos. *                         | \$\$*9,99              |
|                                                                                                                                                                             | ., , | Punto, coma decimal                                                                            | 999999,99              |
|                                                                                                                                                                             | + -  | Signos más o menos. Inserta el signo delante o detrás del número en la impresión               | -999999                |
|                                                                                                                                                                             | DB   | Débito. Sólo se incluye si el valor del campo es negativo.                                     | \$999.999,99DB         |
|                                                                                                                                                                             | CR   | Crédito. Sólo se incluye si el valor del campo es positivo.                                    | \$999.999,99CR         |
|                                                                                                                                                                             | B    | Inserción de espacios en blanco en las posiciones designadas.                                  | 99B99B9999             |
|                                                                                                                                                                             | 0    | Inserción de ceros en las posiciones designadas.                                               | 999909999              |
|                                                                                                                                                                             | /    | Inserción de carácter en las posiciones designadas.                                            | 99/99/9999             |
|                                                                                                                                                                             |      |                                                                                                |                        |



## Tipos de archivos de datos

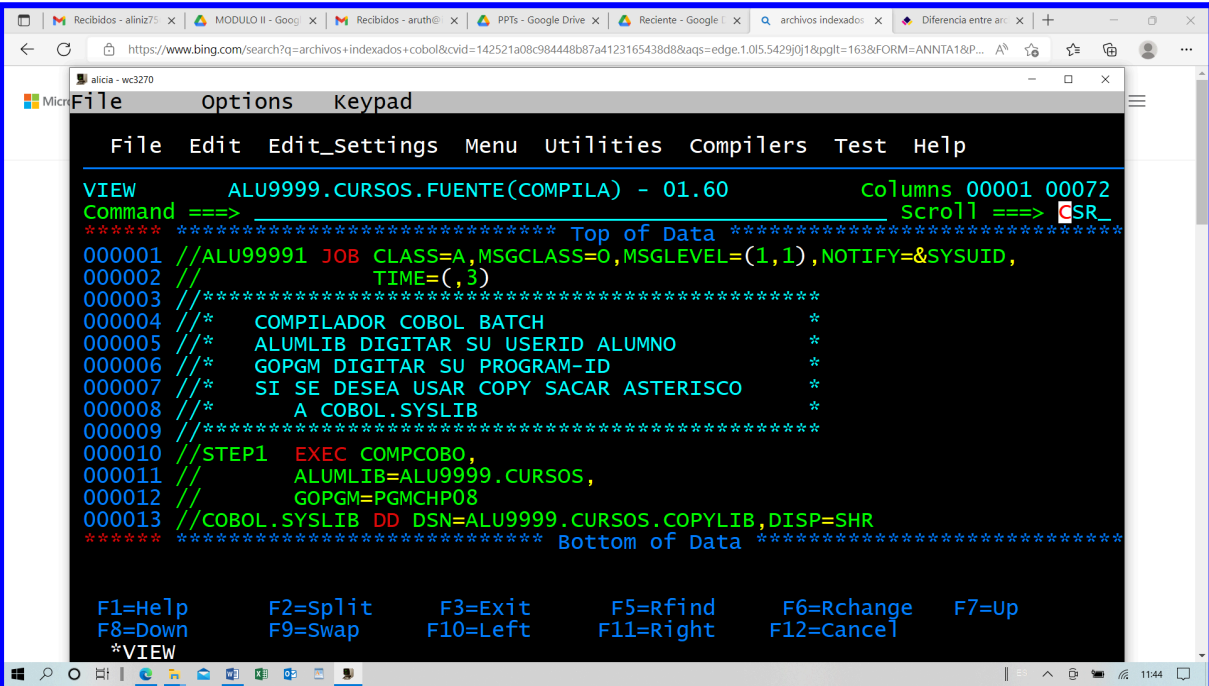
A continuación, se detallan las estructuras básicas de archivos de datos:

### ARCHIVOS QSAM, SECUENCIALES o PLANOS

Los archivos planos se llaman a todos aquellos **archivos de datos que contienen rutas de registro sin relaciones estructurales**. Para interpretar este tipo de archivo se requiere conocimiento de elementos adicionales como, por ejemplo, las propiedades de formato o estructura de datos del archivo.

Ejemplos:

1. **Archivo PDS** (particionado o biblioteca de datos) en la figura siguiente se muestra un miembro dentro de la biblioteca

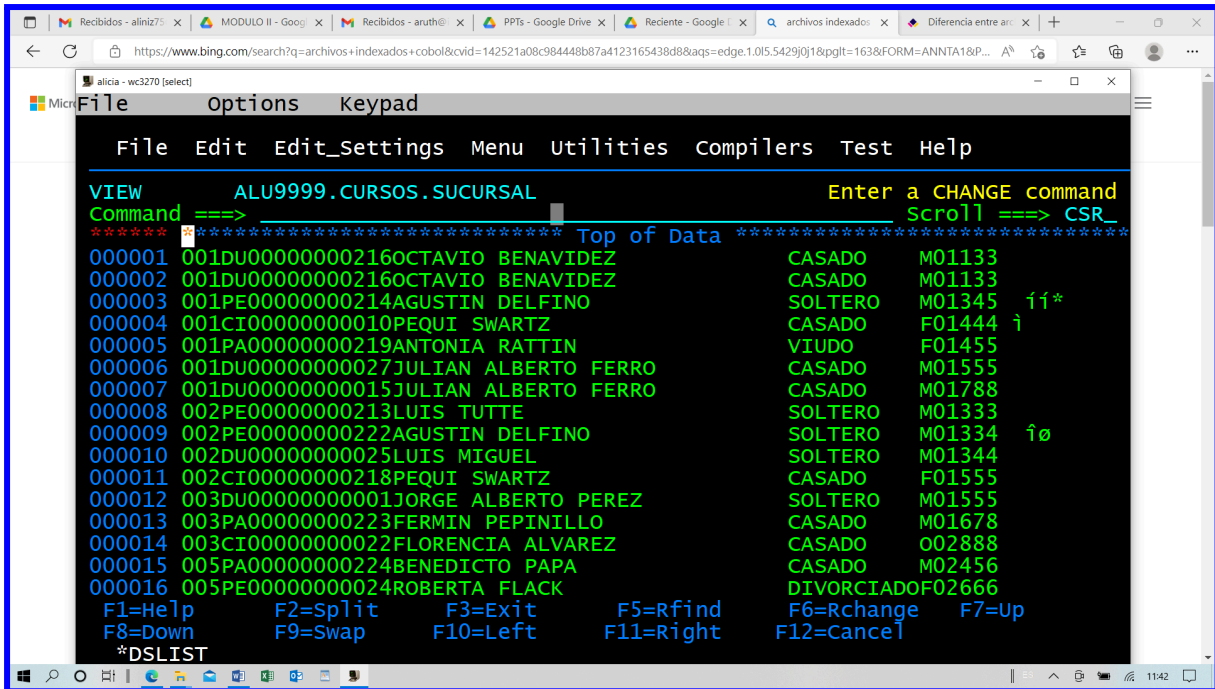


```

File Edit Edit_Settings Menu Utilities Compilers Test Help
VIEW ALU9999.CURSOS.FUENTE(COMPILA) - 01.60 columns 00001 00072
Command ==> Scroll ==> CSR_
***** Top of Data *****
000001 //ALU99991 JOB CLASS=A,MSGCLASS=0,MSGLEVEL=(1,1),NOTIFY=&SYSUID,
000002 // TIME=(,3)
000003 //*****
000004 //* COMPILADOR COBOL BATCH *
000005 //* ALUMLIB DIGITAR SU USERID ALUMNO *
000006 //* GOPGM DIGITAR SU PROGRAM-ID *
000007 //* SI SE DESEA USAR COPY SACAR ASTERISCO *
000008 //* A COBOL.SYSLIB *
000009 //*****
000010 //STEP1 EXEC COMPCOBO,
000011 // ALUMLIB=ALU9999.CURSOS,
000012 // GOPGM=PGMCHP08
000013 //COBOL.SYSLIB DD DSN=ALU9999.CURSOS.COPYLIB,DISP=SHR
***** Bottom of Data *****

F1=Help F2=Split F3=Exit F5=Rfind F6=Rchange F7=Up
F8=Down F9=Swap F10=Left F11=Right F12=Cancel
*VIEW
  
```

2. **Archivo QSAM o PLANO.** en la figura siguiente se muestra el contenido del archivo de datos **ALU9999.CURSOS.SUCURSAL**



```

VIEW      ALU9999.CURSOS.SUCURSAL      Enter a CHANGE command
Command ==>                               Scroll ==> CSR_
***** Top of Data *****
000001 001DU00000000216OCTAVIO BENAVIDEZ      CASADO      M01133
000002 001DU00000000216OCTAVIO BENAVIDEZ      CASADO      M01133
000003 001PE00000000214AGUSTIN DELFINO        SOLTERO      M01345  ff*
000004 001CI00000000010PEQUI SWARTZ           CASADO      F01444  i
000005 001PA00000000219ANTONIA RATTIN          VIUDO        F01455
000006 001DU00000000027JULIAN ALBERTO FERRO    CASADO      M01555
000007 001DU00000000015JULIAN ALBERTO FERRO    CASADO      M01788
000008 002PE00000000213LUIS TUTTE                 SOLTERO      M01333
000009 002PE000000000222AGUSTIN DELFINO        SOLTERO      M01334  i0
000010 002DU00000000025LUIS MIGUEL              SOLTERO      M01344
000011 002CI000000000218PEQUI SWARTZ           CASADO      F01555
000012 003DU00000000001JORGE ALBERTO PEREZ    SOLTERO      M01555
000013 003PA000000000223FERMIN PEPINILLO          CASADO      M01678
000014 003CI00000000022FLORENCIA ALVAREZ        CASADO      O02888
000015 005PA000000000224BENEDICTO PAPA         CASADO      M02456
000016 005PE00000000024ROBERTA FLACK          DIVORCIADO  F02666
F1=Help      F2=Split      F3=Exit      F5=Rfind      F6=Rchange  F7=Up
F8=Down      F9=Swap      F10=Left   F11=Right   F12=Cancel
*DSLIST
  
```

## DIFERENCIA ENTRE ARCHIVOS SECUENCIALES, INDEXADOS Y RELATIVOS EN COBOL

Los archivos son la colección de registros relacionados con una entidad en particular. Mediante el manejo de archivos, podemos almacenar estos registros en un orden organizado. Estos registros se almacenan en cinta magnética o en un disco duro. Los archivos se clasifican además en 3 tipos:

- Organización secuencial de archivos.
- Organización relativa de archivos.
- Organización de archivos indexados.

### Ventajas del manejo de archivos:

- Tienen almacenamiento ilimitado y, por lo tanto, almacenan un gran volumen de datos.

- Almacenan los datos de forma permanente en el dispositivo.
- Reducen la reedición de datos.

La **organización de archivos secuenciales** almacena los datos en orden secuencial. Podemos acceder a los datos de forma secuencial y los datos se pueden almacenar solo al final del archivo; o sea a continuación del último registro de datos.

La **organización de archivos indexados** almacena el registro secuencialmente dependiendo del valor de RECORD-KEY (generalmente en orden ascendente). Una CLAVE DE REGISTRO en un archivo indexado es una variable que debe ser parte del registro / datos. En el caso de archivos indexados, se crean dos tipos de archivos:

**Archivo de datos:** consta de los registros en orden secuencial.

**Archivo de índice:** Consiste en RECORD-KEY y la dirección de RECORD-KEY en el archivo de datos.

Se puede acceder al archivo indexado de forma secuencial al igual que la organización de archivos secuenciales, así como de forma aleatoria solo si se conoce la CLAVE DE REGISTRO.

La **organización de archivos relativa** almacena el registro sobre la base de su dirección relativa.

Cada registro se identifica por su Número de registro relativo, un Número de registro relativo es la posición del registro desde el principio del archivo. Se puede acceder a estos registros de forma secuencial al igual que la organización de archivos secuenciales, así como de forma aleatoria, para acceder a los archivos de forma aleatoria, el usuario debe especificar el número de registro relativo.

---

### Diferencia entre archivos secuenciales, indexados y relativos:

| ARCHIVOS SECUENCIALES                                                                                                           | ARCHIVOS INDEXADOS                                                                                                                                     | ARCHIVOS RELATIVOS                                                                                                |
|---------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------|
| Solo se puede acceder a estos archivos de forma secuencial.                                                                     | Se puede acceder a estos archivos de forma secuencial y aleatoria con la ayuda de la clave de registro.                                                | Se puede acceder a estos archivos de forma secuencial y aleatoria con la ayuda de su número de registro relativo. |
| Los registros se almacenan secuencialmente.                                                                                     | Los registros se almacenan en función del valor de la CLAVE DE REGISTRO que es parte de los datos.                                                     | Los registros se almacenan por su dirección relativa.                                                             |
| Los registros no se pueden eliminar y solo se pueden almacenar al final del archivo.                                            | Es posible almacenar los registros en medio del archivo.                                                                                               | Los registros se pueden insertar en cualquier posición.                                                           |
| Ocupa menos espacio ya que los registros se almacenan en orden continuo.                                                        | Ocupa más espacio.                                                                                                                                     | Ocupa más espacio.                                                                                                |
| Proporciona un acceso lento, ya que para acceder a cualquier registro se debe acceder primero a todos los registros anteriores. | También proporciona un acceso lento (pero es rápido en comparación con el acceso secuencial) ya que lleva tiempo buscar el índice.                     | Proporciona un acceso rápido, ya que proporciona la clave de registro en comparación con los otros dos.           |
| En la organización de archivos secuenciales, los registros se leen y escriben en orden secuencial.                              | En la organización de archivos indexados, los registros se escriben en orden secuencial, pero se pueden leer tanto en orden secuencial como aleatorio. | En la organización relativa de archivos, los registros se pueden escribir y leer en orden secuencial y aleatorio. |
| No es necesario declarar ninguna CLAVE para almacenar y acceder a los registros.                                                | Se pueden crear una o más LLAVES para almacenar y acceder a los registros.                                                                             | Solo se declara una CLAVE única para almacenar y acceder a los registros.                                         |