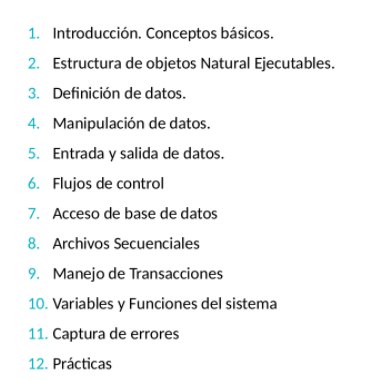
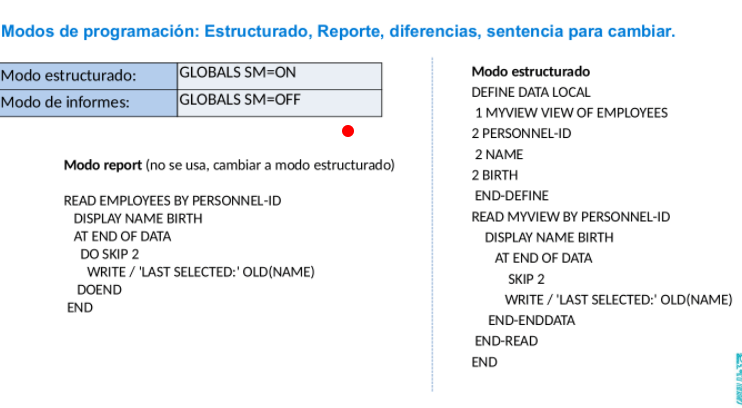
Tema vistos en el curso:

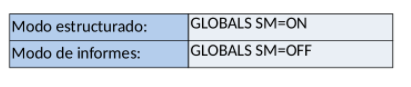


Utilizado para, entre otras cosas, extraer datos de las bases de **Adabast**

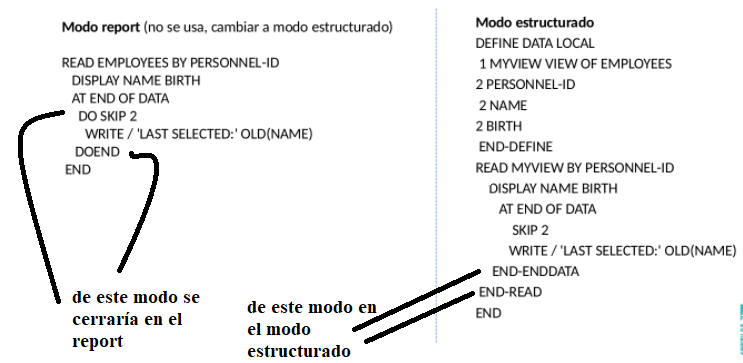
En Natural –One se manejan dos formas de trabajo: Modo estructurado y Modo Report (informes) Siendo que se trabaja más con el primero, se requiere conocer sentencias para poder trabajar con el segundo, o sino no podría compilarse en caso de tener uno.



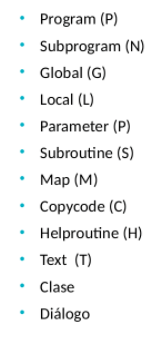
En el mainframe hay una sentencia para cambiar de uno de estos modos al otro; en ***Natural –one*** se hace desde las propiedades, desde donde sí se está generando un objeto, una local, un mapa, un subprograma, en las propiedades se elige de qué forma se va a trabajar: si bien acá por default tenemos en modo estructurado; pero si necesitáramos compilar un programa en modo *Report* que sea complejo, la sentencia se puede modificar y evitar que de error de compilación.



Desde estas dos sentencias es que se puede trabajar con uno o con otro.



Se puede reconocer también en qué modo se encuentra según de qué manera se cierren las sentencias.

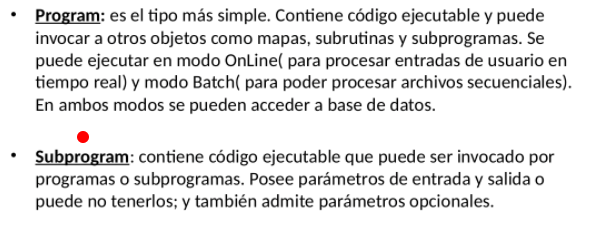


Tipos de objetos de **Natural**:

Estas son herramientas para que, llegado el caso, se pueda modificar o trabajar en el host, siempre que no se esté utilizando el N-O.

Las Global es un área de datos que se comparte con todos los programas del entorno. Las locales pueden ser internas o externas. Los parámetros son Locales que se usan en los subprogramas, se comparten en distintos módulos. Las subrutinas hay externas e internas; como subprogramas que no tiene una interfaz, solo módulos que se comparten en el programa, no devuelven datos. Los Copycode son sentencias que ya no se usan tanto, programas o validaciones que se usaban para todos los programas por igual, s e compartía, pero esto ya no es bueno dentro de ARBA, ya no era efectivo.

La helproutine son subrutinas internas dentro de los mapas, que se utilizan para hacer validaciones dentro de los campos. Dialogo es la comunicación entre los usuarios.

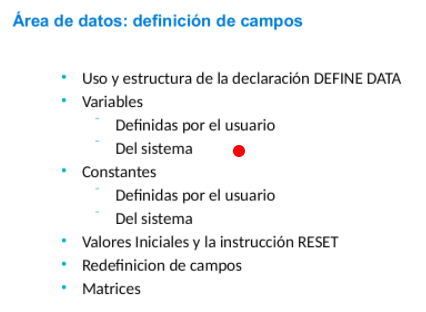


La Global, como con los copycode, son objetos que ya no se usan tanto, ya que había muchos programas que cambiaban mucho las interfaces, y cambiar algo en la global significaba catalogar todos los programas, cosa que resultaba engorrosa.

Acceso a base de datos:

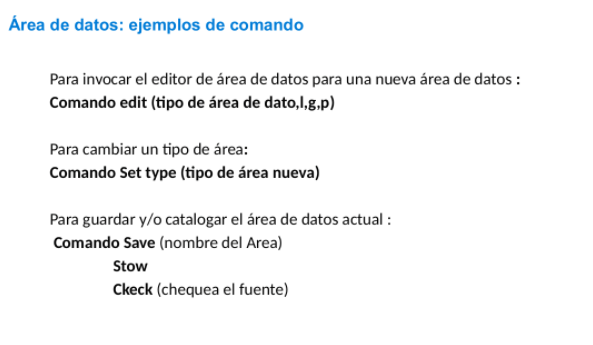
¿El programa cómo accede a una base de dos? El Natural cómo trabaja? en adabast se generan DDM, Para cada archivo físico de una base de datos, se pueden definir una o más DDM, y por cada DDM se pueden definir una o más vista de datos. En Natural se generan vistas desde seguridad a través de las DDM (que son la parte lógica) a través de las cuales se hacen los accesos a las bases. Allí se invoca un programa o una base de datos.

Las bases de Datos en Natural tienen la particularidad de que tiene campos Múltiples y Periódicos. Estos múltiples funcionan como tablas dentro del mismo registro, así como arreglos dentro de un registro. Los campos periódicos se manejan como si fueran tablas, con uno o mas campos,



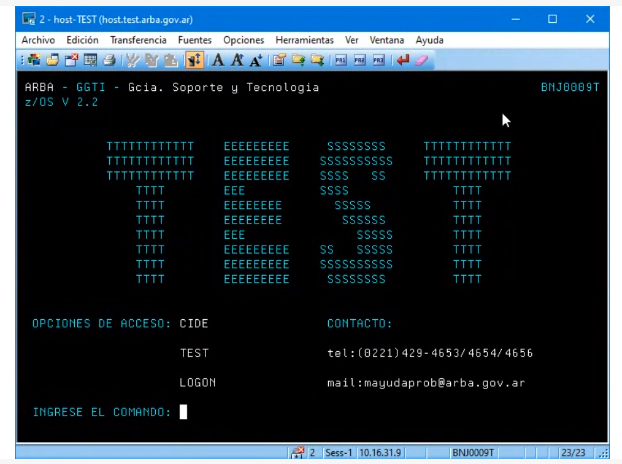
La Definición de variables: En esta se puede ver que depende de algo que se está definiendo desde la necesidad del usuario de una variable a implementar en el programa, o bien un tipo de dato que ofrezca el sistema. Que para inicializar una se utiliza: RESET, y el INIT. Dependiendo de si queremos que al inicializar la variable esta deba tener o no un valor determinado.

Comandos:

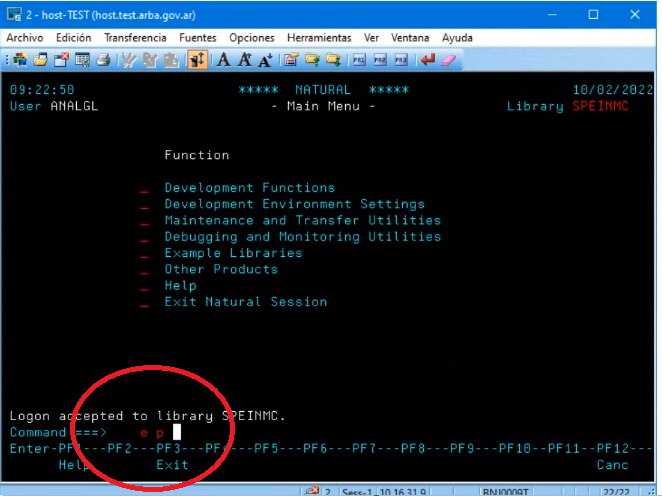


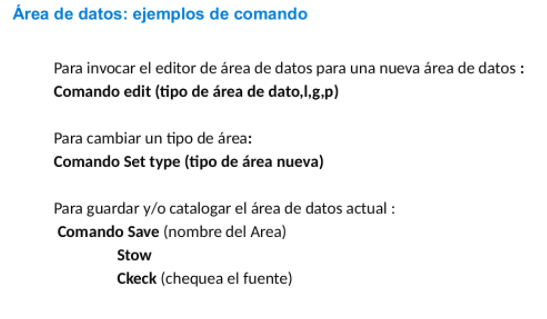
Siempre se trabaja sobre una biblioteca en el Natural, dependiendo qué tipo de trabajo se esté encarando; se asocia de forma directa a una biblioteca en particular.

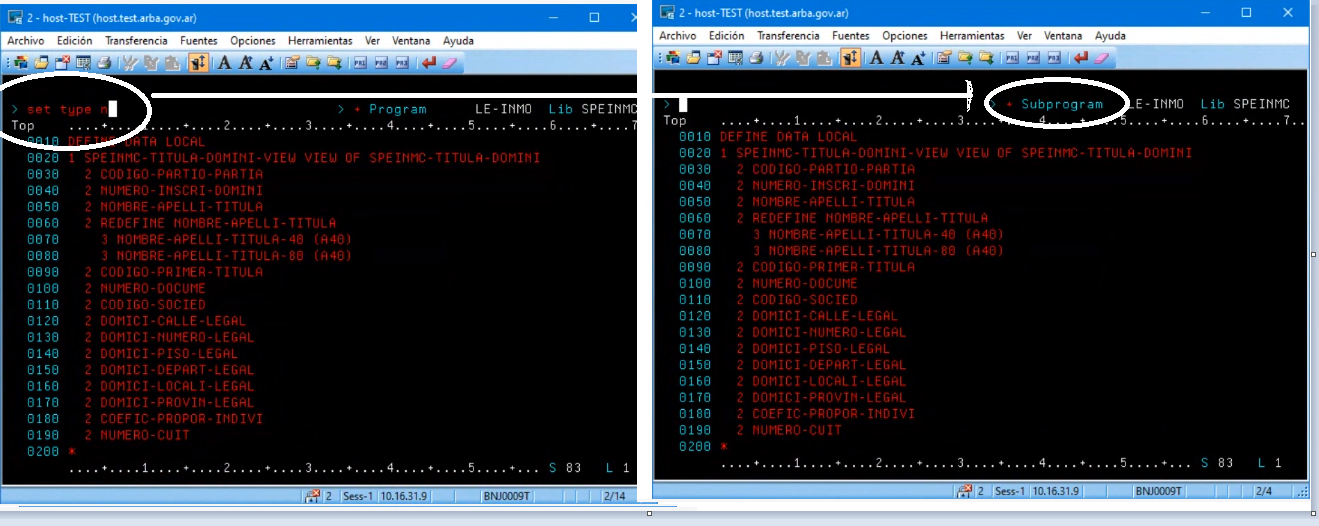
La pantalla de entrada a este sistema se vería de este modo

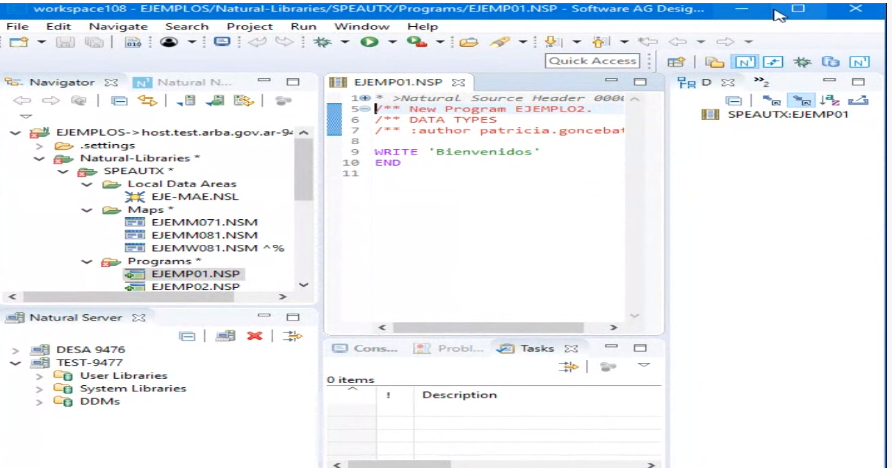


Para ingresar un programa (tras colocar usuario y contraseña) se escribe E P



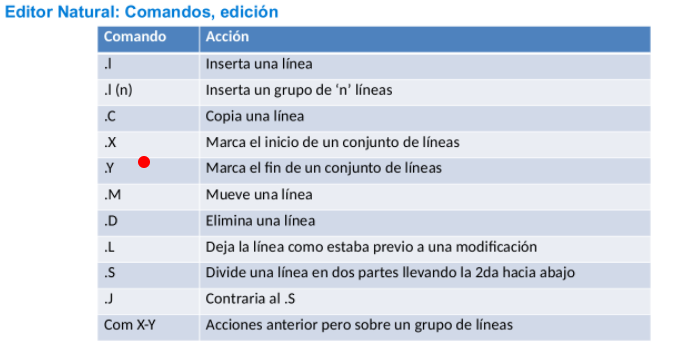


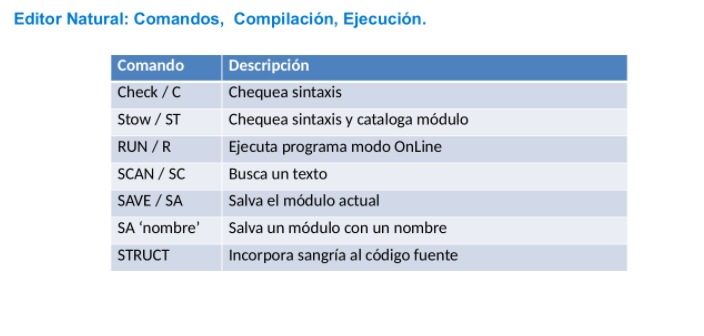




Este por ejemplo e sun comando que puede, junto con los otros : Save, Stow, Check dentro d eNatural. En Natural –One se ve el siguiente perfin para hacer este tipo de operaciones:

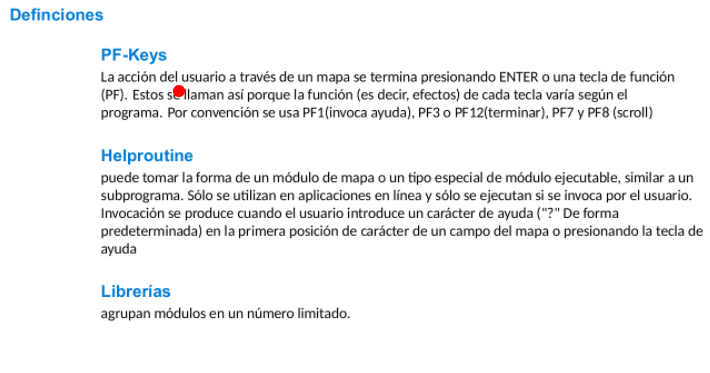
Estos son los comandos en Natural, dentro del mainframe.



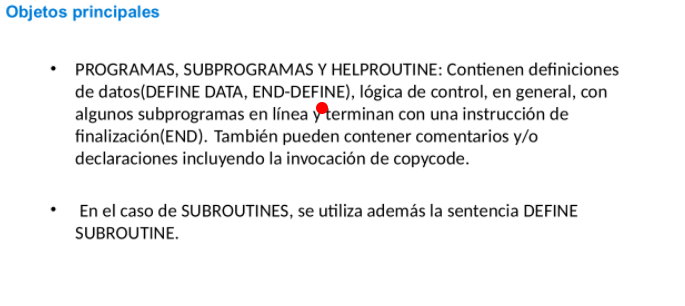


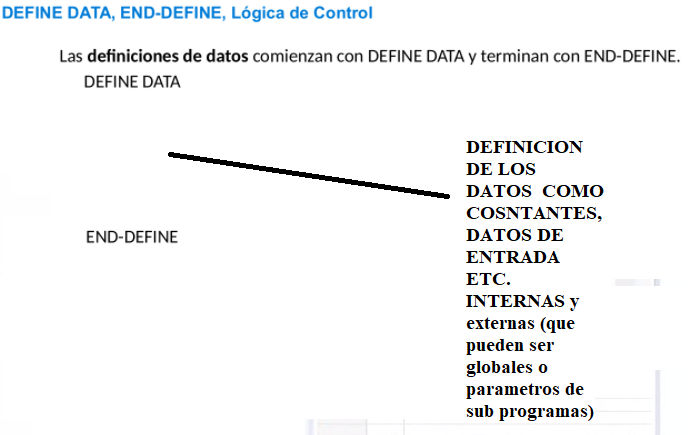
Estas formas, estos comandos del Natural, de trabajo también se van a ver en N-O.

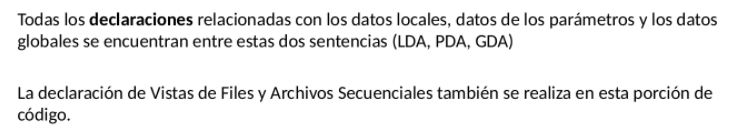
Otro de los objetos que vamos a utilizar son :



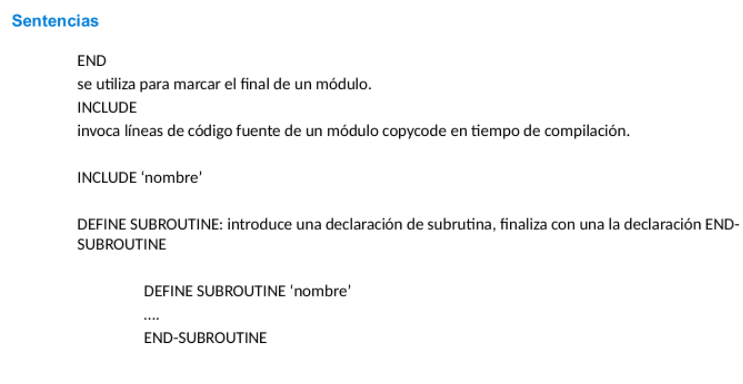
Pf-keys: Teclas que Natural usa para determinar comandos: para cada programa se activan o se desactivan. Las Helroutine están dentro de los mapas, y se utilizan para validar, lo que hacen es mostrar los posibles valores que se quieres asociar a esas Helproutines. Las Librerias, (cada uno va a tener asociada la suya según su trabajo).



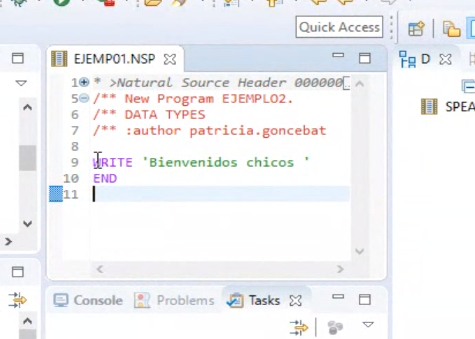




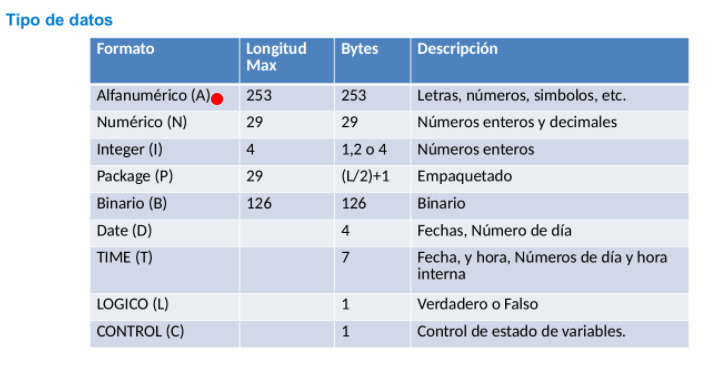
Las estructuras de control se definirán después del END-DEFINE.



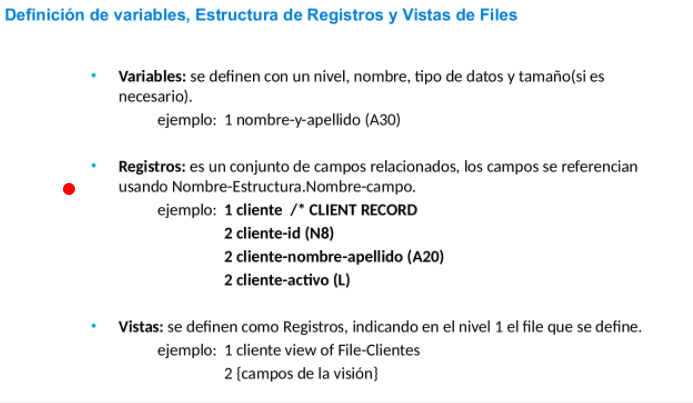
En un programa donde ya tenemos el DEFINE DATA, con una local, que lleve el END, daría un programa funcional empezado y concluido correctamente.



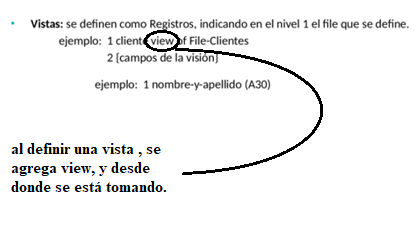
Es importante entender que sin el END el programa no va a compilar.

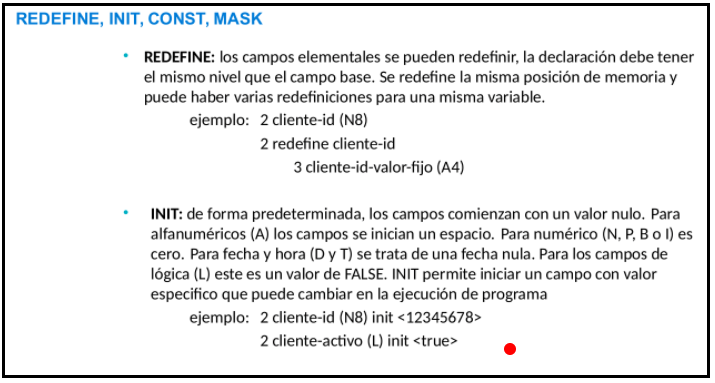
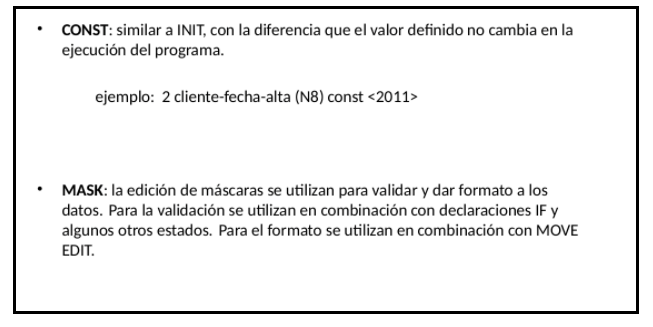


La áreas de control son generalmente las que se utilizan para los mapas, En ciertas ocasiones se quiere que estos campos sean modificables y otros no, sean solo de entrada: así el usuario no lo pueda modificar. Esta es la variable de control que se utiliza para esto.

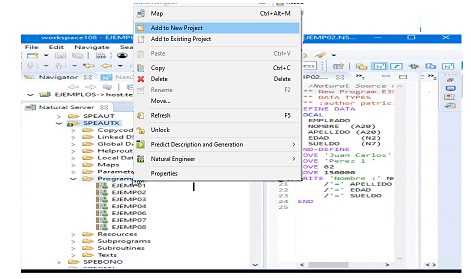
**Las variables** se definen con un tipo de nivel, 1 por ejemplo, más el nombre y el tipo de dato.

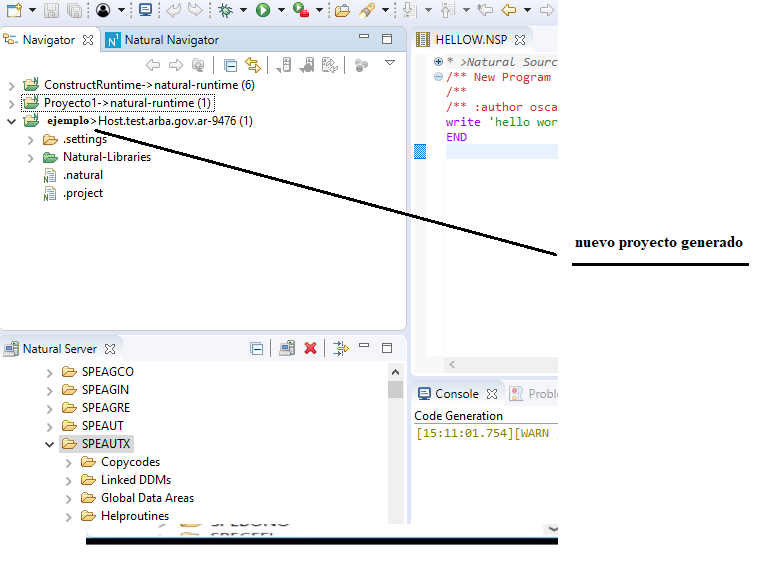
**Los registros**, campos asociados a los que se puede llegar a utilizar en un file. Se puede mencionar un con un nivel, por ejemplo 1, todo el registro, y después en cada registro un , por ejemplo, identificador diferente. (en cliente-activo (L) lo que hay es una variable lógica.

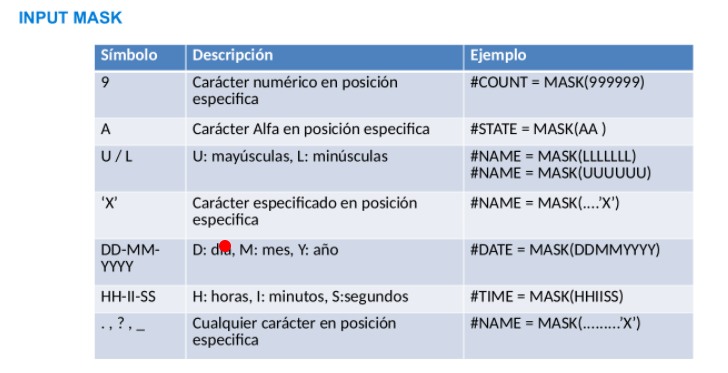




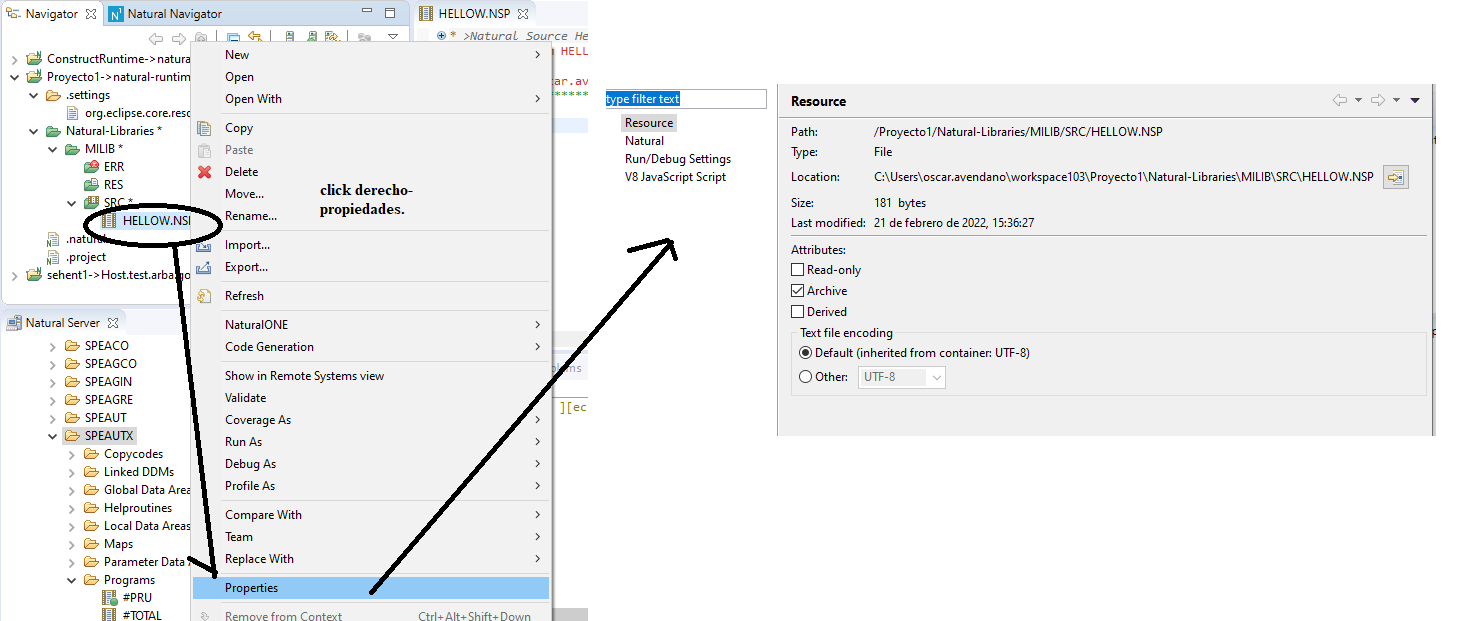
Para trabajar con un nuevo programa en Natural-one, se llega hasta el área designada para los nuevos proyectos entre las librerías y desde allí con el menú contextual de click derecho se genera un nuevo proyecto en el área local.



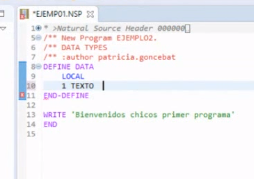




A tener en cuenta la opción de propiedades para saber, a través de **Resourse**, donde está ubicado el proyecto, y varias de sus características. Se puede ver a que servidor se está apuntando en la ejecución. Dónde está corriendo.



En el área de definición de datos en áreas locales (Cntrl + esp) para llamar a la ayuda, en la llamada a un área local, por ejemplo, se puede poner allí: texto:



Algunos datos :

RESET : reestablece valores a cero.

PERFORM : llamadoa una subrutina.

INPUT : ingreso de data por pantalla.

WRITE : egreso de info por pantalla;

FIND: accede a registros: utiliza parámetros: tipo variable WHIT condición.

NF : Distinto de: !=

CALLNAT es un llamado al subprograma.

Colocar numeración al margen: