**VARIABLES - TIPOS Y ALCANCES**

**Módulo**: archivo .prg contenedor de procedimientos y/o funciones. Debe tener al menos uno.

**Procedimiento/Función**: conjunto de líneas de código fuente que resuelve una funcionalidad determinada. Puede haber más de uno de ellos en un mismo módulo.

Las variables que se resuelven en tiempo de compilación son más rápidas que las resueltas en tiempo de ejecución, y por lo tanto, preferibles.

| **Tipo de Variable** | **Se declara en** | **Alcance Visibilidad** | **Se resuelve en** | **Es aplicable macroeval. (&)** | **Vida desde** | **Vida hasta** | **Notas** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **GLOBAL** | Módulo | Total, a nivel de aplicación, no importa en qué módulo se la declare. | Compilación | NO | Inicio aplicación | Cierre aplicación | En el módulo declarante se utiliza la sintaxis:  GLOBAL <varName> [:= <xValue> ]  En el resto de los módulos de la aplicación:  GLOBAL EXTERNAL <extVarName> |
| **STATIC** | Módulo | En el módulo (todos los proc./funciones que contenga). NO en otros módulos | Compilación | NO | Inicio aplicación | Cierre aplicación | Aunque no son visibles en otros módulos, **retienen su valor** a lo largo de toda la vida de la aplicación. |
| **STATIC** | Proc./Function | En la rutina que fue declarada. NO en otras rutinas o módulos | Compilación | NO | Inicio aplicación | Cierre aplicación | Aunque no son visibles en otros proc./funciones del mismo u otros módulos, **retienen su valor** a lo largo de toda la vida de la aplicación.  Las líneas que preceden a la sentencia STATIC no deben ser ejecutables, admitiéndose sólo declaraciones (FUNCTION, PROCEDURE, FIELD, LOCAL, MEMVAR o PARAMETERS) |
| **LOCAL** | Proc./Function | En la rutina que fue declarada. NO en otras rutinas o módulos | Compilación | NO | Inicio rutina | Cierre rutina | Cuando se encuentra un RETURN el contenido de las variables locales de la rutina se destruye.  Las líneas que preceden a la sentencia LOCAL no deben ser ejecutables, admitiéndose sólo declaraciones (FUNCTION, PROCEDURE, FIELD, MEMVAR, PARAMETERS o STATIC).  Las variables LOCAL no son visibles en subrutinas; para hacerlas visibles, deben ser pasadas como parámetro a la subrutina. |
| **PUBLIC** | Proc./Function | Total, a nivel de aplicación, aún de subrutina a rutina (de abajo a arriba) y entre módulos. | Ejecución | SI | Que son creadas | Que son explícitamente destruídas con RELEASE | PUBLIC es una sentencia ejecutable.  Para destruir una variable PUBLIC, utilizar RELEASE.  **Retienen su valor** desde que son creadas hasta que son destruidas. No requieren ser re-declaradas. |
| **PRIVATE** | Proc./Function | Desde la rutina que las creó y se extiende a las subrutinas invocadas después de su creación (de arriba a abajo). | Ejecución | SI | Que son creadas | RETURN de la rutina que las creó, o destrucción con RELEASE | PRIVATE es una sentencia ejecutable.  Las variables PRIVATE son visibles desde el momento de su creación hasta que la rutina declarante retorna, o hasta que la variable es destruida con RELEASE. La visibilidad se extiende a todas las subrutinas invocadas después de la creación de la variable. |