



## Ejemplo de módulo con 1 archivo

```
module moda; {archivo: moda.pas}

export {sección de exportación}
  i1 = (VerOculto, PorOculto, PonOculto); {la interfaz i1}
  i2 = (MIL, Positivo);
  i3 = (MIL, Positivo, VerOculto, PorOculto, PonOculto);

import {sección de importación (viene después de exportación)}
  standardoutput;{para VerOculto}

const MIL = 1000; {sección de constantes}

type Positivo = 1..MAXINT value 1; {sección de tipos}

procedure VerOculto; {declaración de procedimientos y funciones}
procedure PonOculto(o: positivo);
function PorOculto(n: integer):integer;

end; {final de la interfaz del módulo}

{principio de la implementación}
var oculto: positivo;

procedure PonOculto;
begin
  oculto := o
end;

procedure VerOculto; {aquí, sin parámetros}
begin
  write(oculto:0)
end;

function PorOculto;{aquí, sin parámetros}
begin
  PorOculto := oculto * n
end;

to begin do begin {lo que se hace antes de empezar}
  writeln('Inicio de 'moda');
  oculto := 1
end;

to end do {lo que se hace antes de descargar el módulo}
  write ('Fin del 'moda');

end. {el punto termina el módulo}
```

### Observaciones

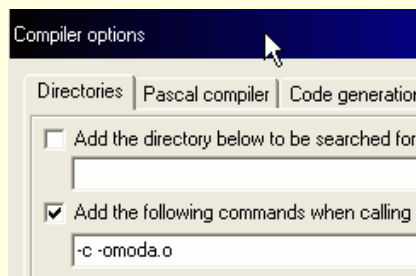
- La *variable oculto* es invisible porque
  - se declara en la sección de implementación y
  - no se exporta.
- La *definición de procedimientos y funciones* está en la implementación.
- La sección *to begin do* se ejecuta cuando se carga el módulo. Consta de una sentencia, pero si tiene que haber varias acciones se hace compuesta. Se usa para inicializar variables, como en este caso y para realizar las acciones iniciales que puedan ser necesarias como preparar archivos...
- La sección *to end do* se ejecuta cuando se descarga el módulo y se usa para realizar las operaciones que puedan ser necesarias como cerrar archivos...

### Compilación

La *compilación* del módulo de archivo único desde la línea de órdenes

```
gpc --extended-pascal -Wall -c moda.pas
```

Desde el *entorno* hay que configurar Options, Compiler options como en la figura

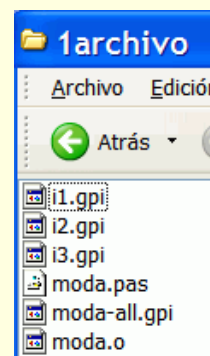


```
-c -omoda.o {entre -o y moda.o no hay espacio}
```

La primera directiva es para evitar que se genere un ejecutable. La segunda es para evitar que se guarde con el nombre moda.exe que tiene por defecto el entorno y le dice que el nombre del archivo objeto sea moda.o y que esté en el mismo directorio que los archivos de las tres interfaces.

Esta compilación genera los siguientes *archivos*

- Cuatro archivos con sendas *interfaces*
  - i1.gpi
  - i2.gpi
  - i3.gpi
  - moda-all.gpi
- El archivo *objeto* moda.o



## Uso del módulo

La importación de una interfaz puede ser

- *parcial*: en ese caso se indica con la palabra *only* y se pone entre paréntesis los identificadores que se importan.
- *cualificada*. Se indica con *qualified*. Significa que en caso de conflicto de nombres se podrán prefijar los identificadores del módulo con el nombre de la interfaz. (Ver ejemplo).

Veamos el programa VerModa que hace uso del módulo.

```
program VerModa (input, output);
import
  i1;
  i2 qualified only (Positivo => tPositivo);
  {de i2 sólo se importa Positivo y lo renombra a tPositivo}

type tPositivo = 0..MAXINT value 0; {choca con el importado}

var n: tPositivo;

procedure VerTPositivo;
var k: i2.tPositivo; {la cualificación evita el choque}
begin
  writeln('tPositivo n = ', n:0);
  writeln('i2.tPositivo k = ', k:0)
end;

begin
  writeln('Valor inicial de oculto: ', PorOculto(1):0);
  PonOculto (123);
  Writeln('Valor actual de oculto: ', PorOculto(1):0);
  writeln('5 * Oculto = ', PorOculto(5):0);
  VerTPositivo
end.
```

La ejecución de este programa produce el siguiente resultado

```
C:\GPAS\EXE\VerModa.exe
Inicio de moda
Valor inicial de oculto: 1
Valor actual de oculto: 123
5 * Oculto = 615
tPositivo n = 0
i2.tPositivo k = 1
```

```
C:\WINDOWS\system32\cmd.exe
C:\GPAS\EXE>VerModa
Inicio de moda
Valor inicial de oculto: 1
Valor actual de oculto: 123
5 * Oculto = 615
tPositivo n = 0
i2.tPositivo k = 1
Fin de moda
C:\GPAS\EXE>
```

La finalización del módulo se produce después de terminar el programa, por eso no se ve desde el entorno, porque antes se cierra la ventana. Si se ejecuta el desde el símbolo del sistema se verá lo que muestra la imagen de la derecha.

