



## Modulo en 2 archivos

En este modelo hay dos archivos

- La sección que contiene la interfaz se guarda en el archivo *modulo.p*
- La sección que contiene la implementación se pone en el archivo *modulo.pas*

### Sección de interfaz

El archivo de la interfaz

```
module moda interface; {archivo1: moda.p}

export
  i1 = (VerOculto, PorOculto, PonOculto);
  i2 = (MIL, Positivo);
  i3 = (MIL, Positivo, VerOculto, PorOculto, PonOculto);

import
  standardoutput; {para VerOculto}

const MIL = 1000;

type Positivo = 1..MAXINT value 1;

procedure VerOculto;
procedure PonOculto(o: Positivo);
function PorOculto(n: integer):integer;

end.
```

La compilación de esta sección

```
gpc --extended-pascal moda.p {genera los *.gpi}
```

Esta compilación genera los archivos de interfaz (\*.gpi)

### Sección de implementación

El archivo de la implementación

```
module moda implementation; {archivo2:moda.pas}

import
  standardinput; {para finalizar moda }
  {standardoutput; ya importado en la interfaz}

var oculto: Positivo;

procedure PonOculto;
begin
  oculto := 0
end;

procedure VerOculto;
begin
  write(oculto:0)
end;

function PorOculto;
begin
  PorOculto := oculto * n
end;

to begin do begin {para hacer antes de empezar el programa}
  writeln('Inicio de moda');
  oculto := 1
end;
```

```

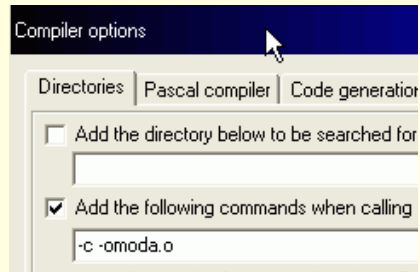
to end do begin {para ejecutar antes de descargar el módulo}
  write ('Fin de moda');
  write ('Pulse ENTER');
  readln
end;

end. {el punto termina el módulo}

```

Se compila igual que el módulo de un archivo

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



Al compilar esta sección se produce el archivo moda.o con las rutinas compiladas.

Al programa VerModa le quitamos la última sentencia readln ya que se detendrá al ejecutar el de la sección de finalización del módulo.

```

program VerModa (input, output); {archivo: vermoda.pas}
import
  i1;
  i2 qualified only (Positivo => tPositivo);
  {de i2 sólo se importa este identificador y se renombra}
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); {0}
  writeln('i2.tPositivo k = ', k:0) {1}
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.

```

Se compila el programa eliminando las opciones que se han puesto para la implementación.

```
gpc --extended-pascal -Wall vermoda.pas
```

Cuando se ejecuta resulta lo que muestra la imagen donde ya se ve desde la ventana del entorno las frases que escribe el módulo al empezar y al terminar.

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  
Fin de moda  
Pulse ENTER_
```

