

Tema 3. Estructuras de Control.

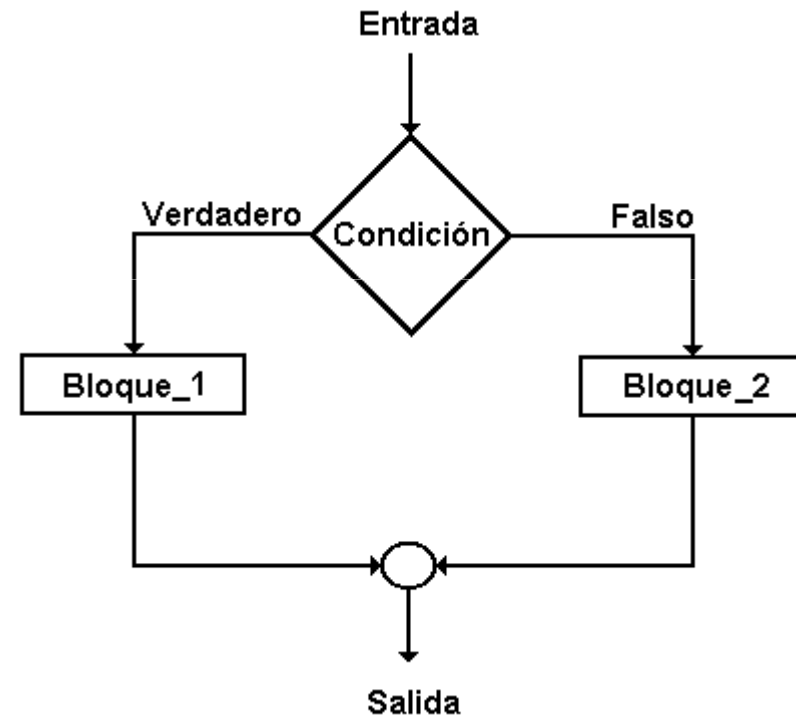
Funciones

- Decisión
 - *if ... else*
- Iteración
 - *for*
 - *while*
 - *do while*
- *break y continue*
- *Switch*
- Funciones
- Funciones predefinidas

if ... else (I)

```
if (condición)
{
  sentencias si verdadero
}
```

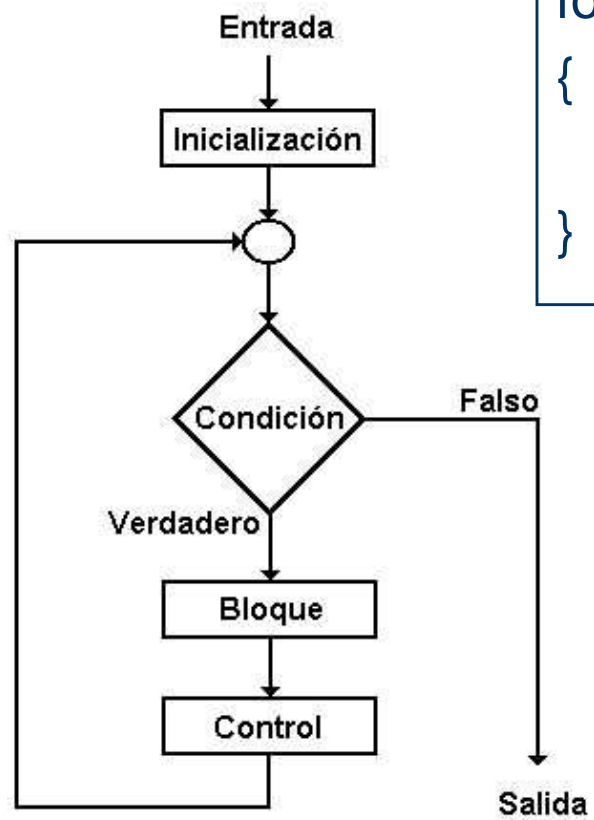
```
if (condición)
{
  sentencias si verdadero
}
else
{
  sentencias si falso
}
```



Ejemplo 1

```
<html>
<body>
<script type="text/javascript">
resp=prompt("Es de dia?(s/n):","");
if(resp=="s" | resp=="S")
    alert("Bon dia");
else
    alert ("Bona Nit");
</script>
</body>
</html>
```

for



```
for(inicialización; condición; actualización)
{
    sentencias
}
```

Ejemplo 2

```
<html>
<body>
<script type="text/javascript">
document.write("<h1>Página
    contadora</h1>");
num=prompt("Dame un número:", "");
for(i=1;i<=num;i++)
    document.write('<h3 align="center">
        <font color="blue">' + i + '</font></h3>');
</script>
</body>
</html>
```

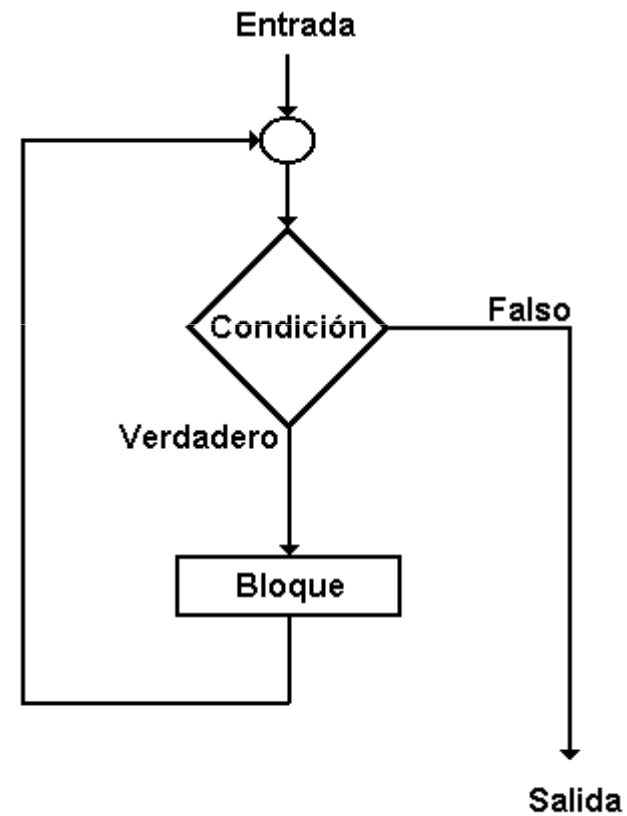
for ... in

```
for(propiedad in expresion)
{
    sentencias
}
```

```
<html>
<head>
<title>Exemple for ... in</title>
</head>
<body>
<script type="text/javascript">
for(var prop in window)
    document.write("<p>",prop);
</script>
</body>
</html>
```

while

```
while(condición)  
{  
  sentencias  
}
```

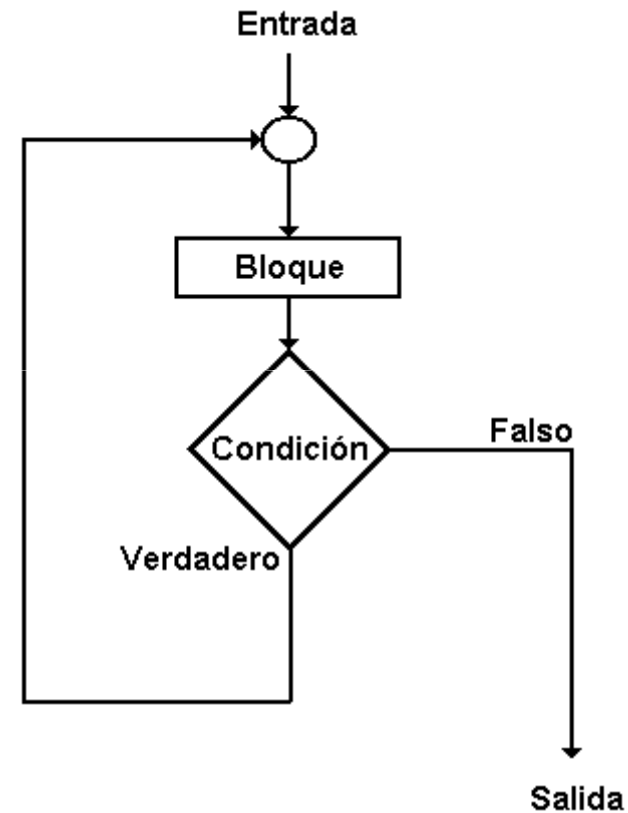


Ejemplo 3

```
<html>
<body>
<script type="text/javascript">
resp=prompt("¿Como estás?","");
document.write('<h2 align="center">Deberías contestar
               <font color="red">bien</font></h2>');
while(resp!="bien")
    resp=prompt("No te entiendo,\n¿Como estás?","");
alert("Que pases un buen dia")
</script>
</body>
</html>
```


do while

```
do  
{  
  sentencias  
}  
while(condición);
```



Ejemplo 4

```
<html>
<body>
<script type="text/javascript">
do{
    resp=prompt("¿Como estás?", "");
    confirma=confirm("Seguro que estás "+resp);
}while(!confirma);
alert("Que pases un buen dia")
</script>
</body>
</html>
```

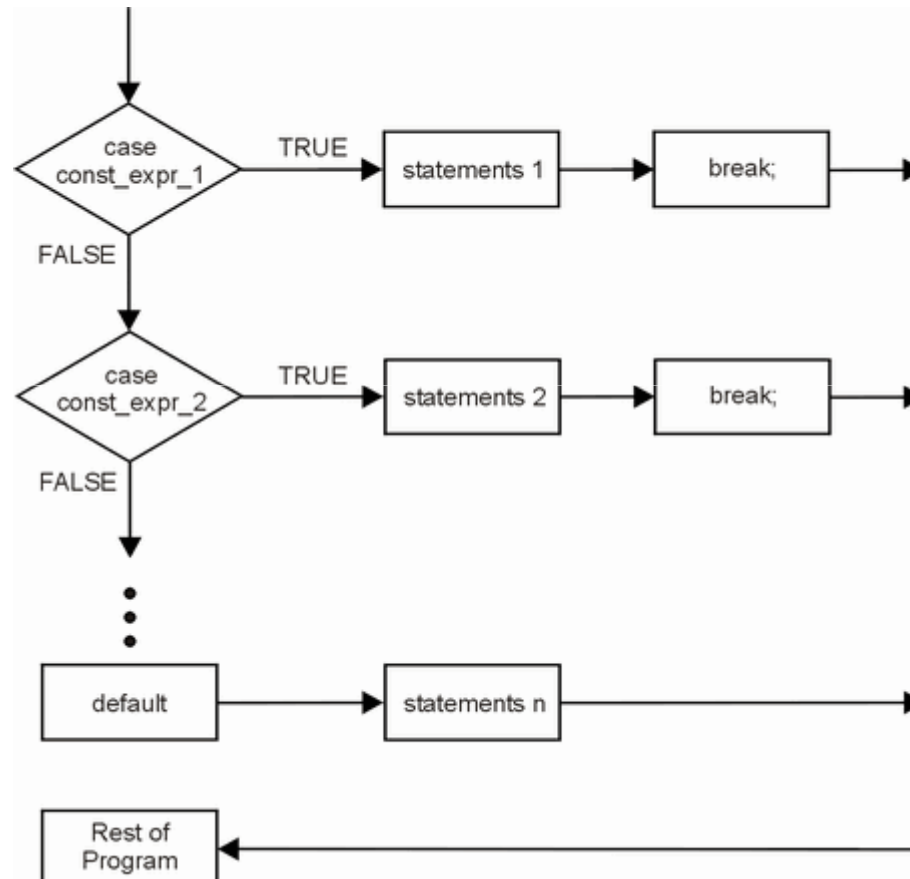
break y continue

- *break*
 - Termina con la ejecución de un bucle o una sentencia *switch*.
- *continue*
 - Provoca la siguiente iteración de un ciclo. En *while* y *dowhile* salta a la condición, en *for* al incremento.

```
<script type="text/javascript">
while(true){
    sino=confirm("¿Vas a estudiar?");
    if(sino) break;
}
alert("Que pases un buen dia")
```

switch

```
swicth(opción)
{
case opción1:
    sentencias;
case opción2:
    ...
case opciónN:
    sentecias;
default:
    sentencias;
}
```



Ejemplo 5

```
<script type="text/javascript">
document.write("<h1>Escoge una
    opción</h1><h2>Mañana<br>Tarde<br>Noche</h2>");
do{
    cita=prompt("Cuando prefieres la cita","");
    seguir=false;
    switch(cita) {
    case "mañana":
        alert('Te espero a las 9 horas');
        break;
    case "tarde":
        alert('Te espereo a las 17 horas');
        break;
    case "noche":
        alert('Lo siento, la noche es para dormir');
    default:
        alert('Por favor, escoge otra opción');
        seguir=true;
    }
}while(seguir);
alert("Que pases un buen dia");
</script>
```

Funciones

```
function nom_funcion(argumentos)
{
    sentencias
}
```

- Los parámetros se pasan por valor
- La función puede devolver un valor mediante la sentencia *return*
- Una función puede realizar llamadas a sí misma o a otras funciones definidas en el mismo documento, invocando el nombre de la función como una sentencia de su definición.
- Se pueden definir funciones con un número variable de argumentos. Para acceder a dichos parámetros dentro de la función se utiliza el vector *arguments*.

Ejemplo 6

```
<html>
<head>
<script type="text/javascript">
function tablaMult(num)
{
for(i=1;i<=10;i++)
document.write('<h3>'+num+' x '+i+' = '+num*i+'</h3>');
}
</script>
</head>
<body>
<h1>Tablas de Multiplicar</h1>
<script type="text/javascript">
do{
    var num=parseInt(prompt("¿Que tabla quieres calcular?",""));
    tablaMult(num);
    var seguir=confirm("¿Otra tabla?");
}while(seguir);
</script>
</body>
</html>
```

Funciones predefinidas

- *parseInt(cadena [, base])*
 - Convierte en un número entero la cadena que recibe, en la base indicada (por defecto base 10). Si fracasa en la conversión, devolverá el valor *NaN*.
- *parseFloat(cadena)*
 - Convierte en un número real la cadena que recibe, devolviendo *NaN* si fracasa en el intento.
- *isNaN(valor)*
 - Devuelve true si el argumento es *NaN*. Es decir, no es un número.
- *Number(objeto)*
 - Convierten a número el objeto que se le pasa como argumento
- *String(objeto)*
 - Convierten a cadena el objeto que se le pasa como argumento

```
n1=prompt("Numero 1","");
n2=prompt("Numero 2","");
alert(n1+n2);
n1=parseInt(prompt("Numero 1",""));
n2=parseInt(prompt("Numero 2",""));
alert(n1+n2);
```


Ejemplo 7

```
<html>
<body>
<script language="JavaScript" type="text/javascript">
var numero=new String("12");
alert(numero+" "+typeof numero);
var n=3+Number(numero);
alert(n);
var x="12.4.3";
document.write("<p>",parseInt(x));
//12
document.write("<p>",parseFloat(x));
//12.4
document.write("<p>",Number(x));
//NaN. Si esta ben escrit, Number pot servir per transformar tant a int
      com a float
</script>
</body>
</html>
```