

Resum JavaScript

JS EN PAGINES WEB

Podem inserir el codi JavaScript en diferents punts:

- **<body>**:

Al `<body>` es posa el codi que cal que s'executi en carregar-se la pàgina

```
<body>
<script type="text/javascript">
//Codi JavaScript
</script>
</body>
```

- **<head>**:

Les funcions es posen al `<head>`

```
<head>
<script type="text/javascript">
//Codi JavaScript
</script>
</head>
```

- **Fitxer extern:**

En els fitxers externs s'hi col·loquen les funcions que es faran servir en varies pàgines web

```
<script src="nomFitxerExtern.js"></script>
```

LITERALS

Un literal és una dada que escrivim literalment. Hi han diferents literals:

- **String:** text, cadenes de caràcters. Es posa entre cometes simples o dobles. Ex: "Hola", 'Bon dia', 'A'

Caràcters especials:

| Caràcter | Descripció |
|----------|---------------------------------|
| \b | Retrocés |
| \f | From feed |
| \n | Salta línia |
| \r | Retorn de carro |
| \t | Tabulador horitzontal |
| \v | Tabulador vertical |
| ' | ' |
| " | " |
| \\ | \ |
| \XXX | Latin-1 (octal) Ex: \311 → É |
| \xXX | Latin-1 (hex.) Ex: \xC9 → É |
| \uXXXX | Caràcter UNICODE Ex: \u00C9 → É |

- **Integer:** Enters. Nombres positius o negatius, però d'unitats senceres Ex: 5, -40,0
- **Floating-Point:** Nombres molt grans, molts petits o d'unitats trencades. Ex: -3.1E129, 7.8E-45
El format permés: [digits].[digits]([E—e])([+—])digits
- **Boolean:** Dos valors possibles. Ex: **true**, **false**
- **Array:** Matriu de valors possibles. Ex: ["Joan", "Jordi", "Esther"], [5,20,77,,]
- **Object:**

Una **variable** és un contenidor de dades.

VARIABLES

Una **variable** és un contenidor de dades.

Declaració de variables

```
var bar="La ovella negra"; //Cadena (String)
var cerveses=9; // Enter (Number:Integer)
var eurosGastats=22.50; // (Number:Float)
var torrat=true; //Boolea (Boolean)
```

Noms de les variables

- No poden ser paraules reservades Ex: **var**, **for**, **if**.
- Han de començar per una lletra (A-Z) o un guió baix (-)
- No poden contenir espais

Àmbit d'una variable

- **local:** Definida en una funció. Només visible a l'interior de la funció
- **global:** Definida fora de qualsevol funció. Visible a dins i fora les funcions.

OPERADORS

Els **operadors** especifiquen l'acció a realitzar sobre les dades. Ex: sumar, comparar, assignar a una variable..

Operadors Aritmètics

| Operador | Descripció | Exemple |
|----------|---------------|---------|
| + | suma | x+5 |
| - | resta | x-5 |
| * | multiplicació | x*5 |
| / | divisó | x/5 |
| % | mòdul | x%5 |
| ++ | increment | x++ |
| -- | decrement | x-- |

Operadors Assignació

| Operador | Exemple | Equivalència |
|----------|---------|--------------|
| += | x+=5 | x=x+5 |
| -= | x-=5 | x=x-5 |
| *= | x*=5 | x=x*5 |
| /= | x/=5 | x=x/5 |
| %= | x%=5 | x=x%5 |

Operadors de comparació

| Operador | Exemple | Afirmació |
|----------|---------|---------------------------------|
| == | (x==y) | x y són iguals |
| === | (x===y) | x y són iguals en valor i tipus |
| != | (x!=y) | x y són diferents |
| > | (x>y) | x és més gran que y |
| < | (x<y) | x és més petit que y |
| >= | (x>=y) | x és més gran o igual que y |
| <= | (x<=y) | x és més petit o igual que y |

Operadors Lògics

| Operador | Exemple | Afirmació certa (true) |
|----------|----------------|---------------------------------|
| && | (x==5 && y==8) | x és 5 i y és 8 |
| | (x==5 y==8) | x és 5 o y és 8 |
| ! | !(y) | Ex: si NO y |

ESTRUCTURES DE CONTROL

Les **estructures de control** condueixen el fluxe del programa. N'hi ha de dos tipus:

- condicionals: **if**, **switch**
- iteratives: **for**, **while**, **do...while**

if

```
if (eurosGastats>=25 || cerveses>=10)
{
  document.write("Torno a casa")
}
else if (eurosGastats<=25 && cerveses>=10)
{
  document.write("M'han convidat")
}
else
{
  document.write("He lligat")
}
```

switch

```
switch (cerveses)
{
  case 0: case 1: case 2: case 3:
    document.write("Situacio sota control");
    break;
  case 4: case 5: case 6: case 7:
    document.write("Situacio complexe");
    break;
  case 8: case 9: case 10: case 11:
    document.write("Situacio perillosa");
    break;
  default:
    document.write("Situacio critica");
}
```

for

```
for (i=0;i<colegues;i++)
{
  bufar(i);//cada un dels col·legues bufa.
}
```

while

```
while (torrat)
{
  document.write("No pots agafar el cotxe");
}
```

do...while

```
do
{
  bufar();
}while (torrat)
```

FUNCIONS

Definició d'una funció

Una **funció** és una encapsulació de codi que s'utilitza repetides vegades.

```
function bufar(nivellAlcohol,nivellPermes)
{
  if (nivellAlcohol >= nivellPermes)
  {
    torrat=true;
  }

  if (nivellAlcohol < nivellPermes)
  {
    torrat=false;
  }

  return torrat;
}
```

Elements d'una funció:

- **Nom**: es farà servir per fer la crida Ex: **bufar**
- **Arguments(opcional)**: dades que se li passen a la funció Ex: **nivellAlcohol** i **nivellPermes**
- **Return**(opcional): valor que es retorna en el lloc en el qual s'ha cridat la funció Ex: **torrat**

Una funció pot ser declarada dins de una estructura condicional: **if** o **switch**

Crida d'una funció

Allà on volguem fer servir la funció fem la crida:

```
bufar(0.7,0.5) //nivellAlcohol 0.7, nivellPermes 0.5
// retorna un true.
```

Si la funció te retorn, allà on s'hagi fet la crida, s'hi posarà el valor de retorn.

Funcions predefinides

Funcions predefinides de nivell global(accessibles en qualsevol lloc)

- **eval(str)**
- **isFinite(Number)**
- **isNaN(testExpresion)**
- **parseInt(str)**
- **parseFloat(str)**
- **Number(objRef)**
- **String(objRef)**

CLASSES I OBJECTES

Una **classe** és un tipus d'objecte. Un **objecte** te propietats i mètodes.

Creació de la classe Alumne:

```
/* nom,cognom,curs i nota son
propietats de la classe Alumne */

/* escriureAlumne es un metode
de la classe Alumne*/

function Alumne(nom,cognom,curs,nota) {
  this.nom = nom;
  this.cognom = cognom;
  this.curs = curs;
  this.nota = nota;
  this.escriureAlumne = escriureAlumne;
}
```

Creació del objecte 'Pere' (que serà de la classe alumne):

```
var alumne1=new Alumne('Pere','Roure','1erESI',5);
```

Obtenir la propietat d'un objecte:

```
//Escric la nota del Pere
document.write(alumne1.nota);
//Escriuria 5
```

Canviar la propietat d'un objecte:

```
alumne1.nota=6;
```

Crear un mètode:

```
//Escric el metode escriureAlumne

function escriureAlumne(){
  document.write(this.nom);
  document.write(this.cognom);
  document.write(this.curs);
  document.write(this.nota);
}
```

Classes predefinides

Array

| <u>PROPIETATS</u> | <u>MÈTODES</u> |
|-------------------|----------------|
| constructor | join() |
| index | pop() |
| input | push() |
| length | reverse() |
| prototype | shift() |
| | slice() |
| | sort() |
| | splice() |
| | toSource() |
| | toString() |
| | unshift() |
| | valueOf() |

Boolean

| <u>PROPIETATS</u> | <u>MÈTODES</u> |
|-------------------|----------------|
| constructor | toSource() |
| prototype | toString() |
| | valueOf() |

Date

Math

String

EVENTS

Els events permeten la interacció amb l'usuari. Un event es una acció del usuari davant la qual poden realitzar-se alguns processos. Ex: canvi del valor d'un formulari, o la pulsació d'un enllaç.

Exemple d'utilització dels events:

```
<body onClick="alert ('click!!');">
```

| <u>EVENT</u> | <u>ELEMENTS HTML</u> | <u>DESCRIPCIÓ</u> |
|--------------|-------------------------|---------------------------|
| onblur | ** | Perdre el focus |
| onchange | input, select, textarea | canviar el valor |
| onclick | tots exepte: * | clickar |
| ondblclick | tots exepte: * | doble click |
| onfocus | ** | obtenir el focus |
| onkeydown | tots exepte: * | al premer una tecla |
| onkeypress | tots exepte: * | quan la tecla és premsada |
| onkeyup | tots exepte: * | en deixar una tecla |
| onload | frameset, body | al carregar |
| onmousedown | tots exepte: * | |
| onmousemove | tots exepte: * | moure el ratolí |
| onmouseout | tots exepte: * | situar ratolí fora |
| onmouseover | tots exepte: * | situar ratolí a sobre |
| onmouseup | tots exepte: * | |
| onselect | input, textarea | Seleccionar |
| onsubmit | form | Al enviar |
| onunload | frameset, body | Al sortir |

* applet, base, basefont, bdo, br, font, frame, frameset, head, html, iframe, isindex, meta, param, script, style, title
**a, area, button, input, label, select, textarea