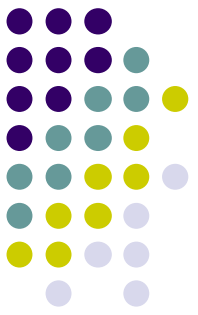


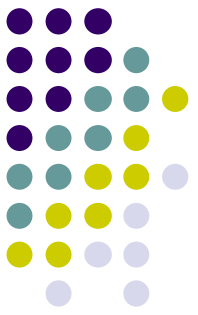
G3f: **Bezeroaren aldeko teknologiak.** **(DOM). JavaScript**

Rosa Arruabarrena, Xabier Arregi, Jose Ángel Vadillo
LSI, UPV/EHU



Motibazioa

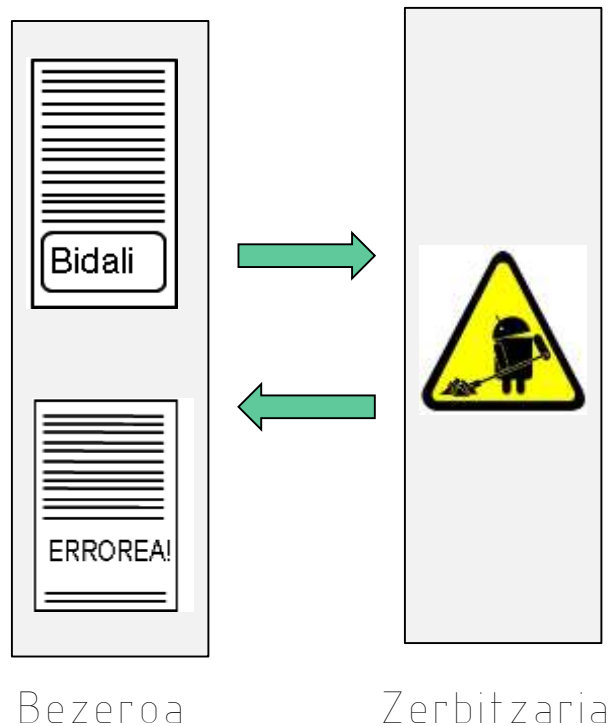
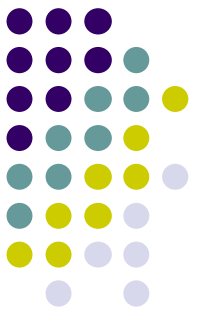
- Aplikazio baten exekuzioa zerbitzariaren eta bezeroaren artean banatzeko ideia lantzea
- JavaScript aurkeztea WS irakasgaiko beharren baitan
- DOM ereduaren erabilera jorratzea
- Web aplikazioen garapenerako JavaScript lengoaiaren erabilera berezitu batzuk azaltzea eta lantzea



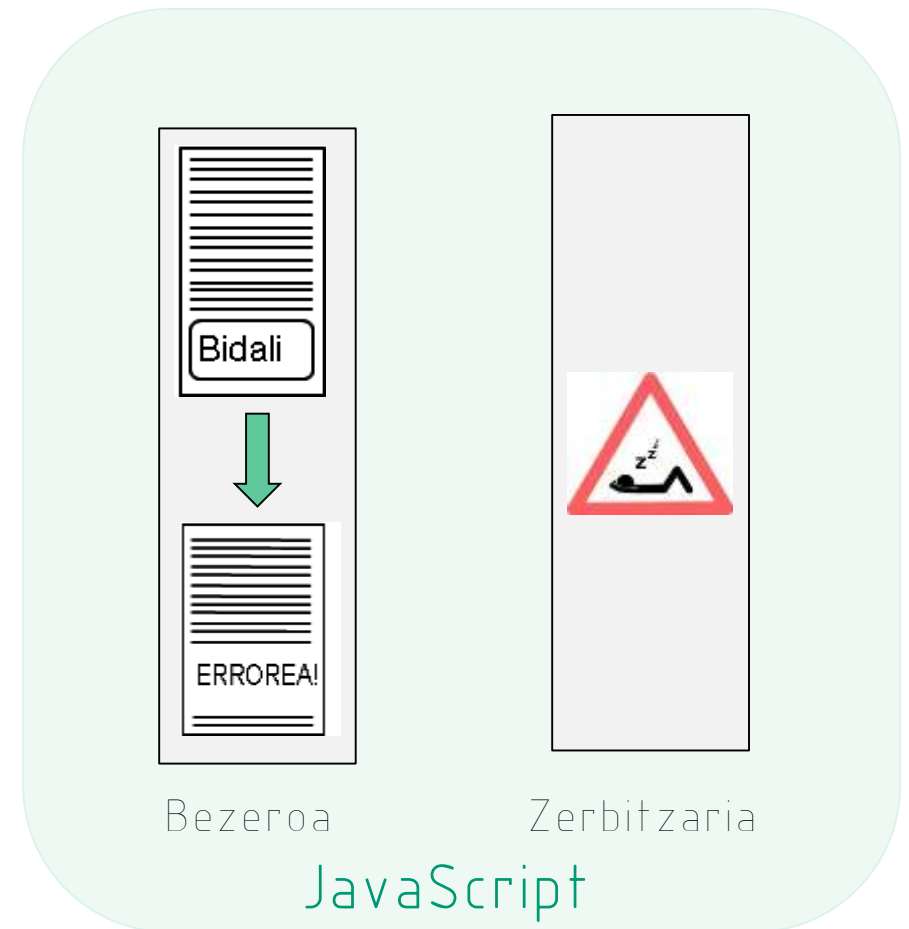
Bezeroan ala zerbitzarian

- Bezeroko script-ak
 - Nabigatzailean exekutatzen dira
 - Bezeroko baliabideak kudeatzen dira modu mugatuan
 - JavaScript, VisualBasicScript, java applets,...
- Zerbitzariko script-ak
 - Web zerbitzariak interpretatzen ditu
 - Bezerora dokumentu interpretatuak (HTML) bidaltzen dira
 - CGI, PHP, ASP, Python, Perl,...

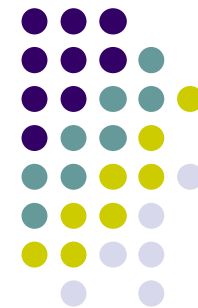
Zerbitzariaren eta bezeroaren arteko lan-banaketa



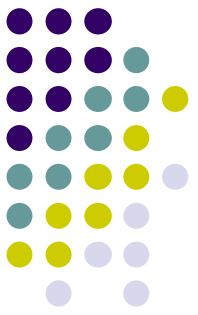
CGI, ASP
PHP



Zer da JavaScript (JS)?

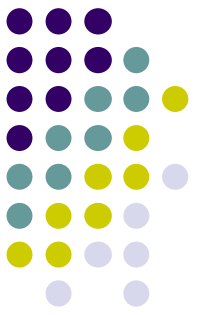


- Espektro zabaleko lengoaia da gaur egun, nahiz hasieran web orrien portaerarekin erlazionatua egon soilik; eta programatzailearen konpetentziaren arabera, programazioko kontzeptu eta teknika aurreratuenak ere pairatzen dituen.
- Programazio aurreratuen ezagutzarik gabe erabiltzen has litekeen lengoaia da, baina programatzaile adituen gaitasunak behar ditu aplikazio eta erabilpen ahaltsuenak gauzatzeko.
- Script lengoaia da; hots, euren programak interpretatuak direnak (ez konpilatuak).
- Master/Slave (Zerbitzari/bezero) web ereduari etekina ateratzen dio, web aplikazioaren exekuzioaren parte bat bezerora garraiatzea ahalbidetzen du (arakatzailera).
- JavaScript programak eraiki eta HTML orrien barne daude.
- JavaScript lengoaiatik (1) arakatzzailearen ezaugarriak atzi litezke bai eta (2) HTML orrietan dauden objektuak ere DOM-aren API-a erabiliz.



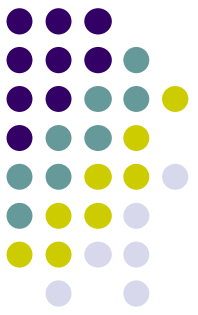
JS. Zertarako?

- Web-orrien funtzionalitatea eta interakzio-gaitasuna areagotzeko. Adibidez:
 - Formularioetan errorearen detekzioari eta tratamenduari aurre egiteko
 - Egiaztapenen eta *Warning* mezuen bidezko elkarrekintza
- HTML orrietan objektu aktiboak sartzeko:
 - kalkulagailuak, kalkulu-orriak,...
- Orriaren aurkezpenean aldaketa dinamikoak egiteko:
 - menuak, dokumentuen eguneraketa zatika...
- URLak aztertu eta arakatutako URLen historia gordetzeko
- Leihoak maneiatzeko aukera
- Irudiak eta mapak maneiatzeko
- *Cookie-ak erabiltzeko*
-



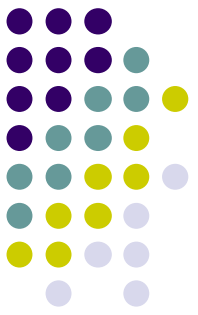
JS. Historia

- NetScape-k web arakatzailerik applet-ak exekuta ahal izan ditzatela gauzatzen du.
- NetScape-k programazio-lengoaia bat garatzen hasten da: *LiveScript*. Honek, web orrietan programa txikiak sortzeko aukera ematen zuen.
- 1995en Sun garapen proiektura gehitzen denean, *JavaScript* izena hartzen du (merkatu estrategia izanik izen aldaketa)
 - Brendan Eich-ek diseinaturia
- NetScape 2.0 JavaScript ulertzen zuen lehen nabigatzailea izan zen, eta ostean, atzetik, Microsoften beste nabigatzaile batzuk, 3.0 bertsioetik aurrera.



Java vs JavaScript

- Java eta JavaScript guztiz desberdinak diren lengoaiak dira. Besteak beste, JavaScript-ek
 - Garapena egiteko ez du KIT-ik behar
 - Ez da helburu orokorreko lengoaia
 - Motaketa ahula du
 - Erabat interpretatua da (Javan existitzen du itzulpena tarte koderak)



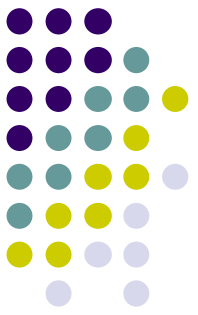
JavaScript. Ezaugarri orokorrak

- Script lengoaia interpretatua (ez konpilatua)
- Object-Based: objektuetan oinarritua
- Objektuen orientazioa: prototipatzea
- Motatze dinamikoa duena
- Lehen mailako funtzioak
- Multiparadigma: agintzailea, objektuen orientazioa, funtzionala
- Web-orrien garapenera zuzendua,
 - (batez ere) Brendan Eich-ek bezzeroaren aldean
- Azken aldian lan-ingurune eta liburutegi asko garatu dira (AJAXek izan du eragina horretan), eta “Server-Side JavaScript” (SSJS) ere izugarri hedatu da

JavaScript. Ezaugarri orokorrak (II)

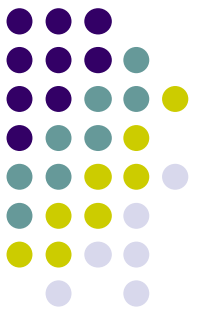


- JavaScript programak (=funtzioak)
 - web-orrietan txertatzen dira, edo
 - web-orriekin lotzen dira
- JavaScript lengoaiak erraztasunak ematen ditu arakatzaillearen propietateak eta HTML dokumentuetan dauden objektuak atzitzeko, sortzeko eta manipulatzeko



Nola sartu JavaScript

- **Non jar daitezke JavaScript script-ak HTML orrietan?**
 - Nonahi...
 - Sarri `<head>` eta `</head>` etiketen artean.



Nola sartu JavaScript (II)

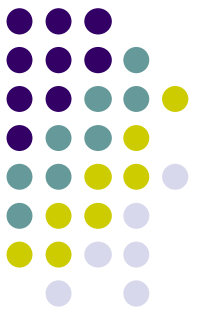
- *Script* etiketa erabiliz

- Zuzenean

```
<script language="JavaScript">  
    //hemen JavaScript-eko kodea  
</script>
```

- Script-eko kodea aparteko fitxategian egon liteke eta hura erreferentziatuz

```
<script scr="http:// .../izena.js"> </script>
```



Nola sartu JavaScript (III)

- *JavaScript* gidoiak HTML orrietan nola sartu?
 - Gertaera bati lotuta, objektu baten atributuari balioa emanez

```
<p>Botoiari eragin ordua ikusteko.</p>
```

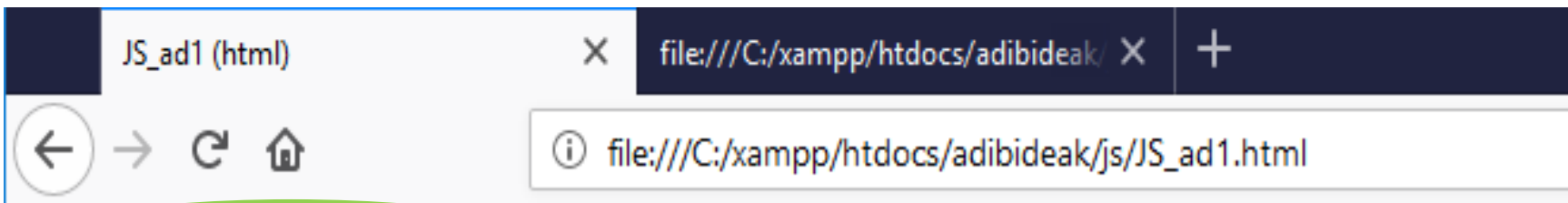
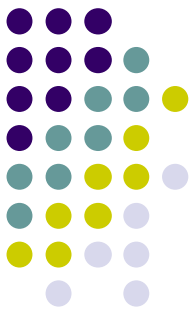
```
<button onclick="getElementById('demoa').innerHTML=  
    Date()"> Zer ordu da?
```

```
</button>
```

```
<p id="demoa"></p>
```

- `<a>` etiketaren `href` atributuari balioa emanez:

```
<a href="javascript:window.history.back()">
```



JavaScript zatian gaude!

HTML orria + JS

Scripting lengoairekin probak egiten

```
C:\xampp\htdocs\adibideak\js\JS_ad1.html - Notepad++
Archivo  Editor  Buscar  Vista  Codificación  Lenguaje  Configuración  Herramientas  Macro  Ejecutar  Plugín

JS_ad1.html x
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>JS_ad1 (html) </title>
5      <meta charset="utf-8">
6      <script language="JavaScript">
7          document.write("JavaScript zatian gaude!");
8      </script>
9  </head>
10 <body>
11   <h2>HTML orria + JS</h2>
12   Scripting lengoairekin probak egiten
13   <br/>
14 </body>
15 </html>
```



C:\xampp\htdocs\adibideak\js\JS_ad2-button-function.html - Notepad++

Archivo Editar Buscar Vista Codificación Lenguaje Configuración Herramientas Macro Ejecutar



JS_ad2-button-function.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <title>JS_ad2 (button) </title>
5     <meta charset="utf-8">
6     <script language="JavaScript">
7         function KaixoLagun() {
8             alert("Bizi al haiz?");
9         }
10    </script>
11 </head>
12 <body>
13     <h2>HTML, botoiak eta gertaera harrapaketa JS</h2>
14     Gertaera harrapaketa JS bidez
15
16     <button onclick="KaixoLagun();" > Sakatu!!! </button>
17 </body>
18 </html>
```

per Text Markup Language file

length : 376 lines : 18

Ln

JS_ad2 (button)

file:///C:/xampp/htdocs/adibide

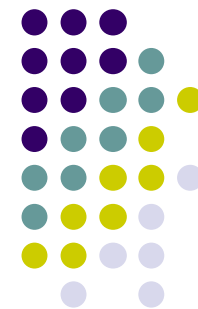
HTML, botoiak eta gertaera harrapaketa JS

Gertaera harrapaketa JS bidez

Bizi al haiz?

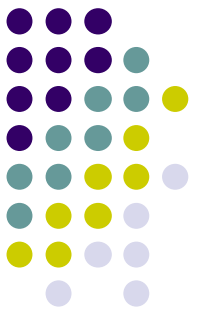
Aceptar

JavaScript programen exekuzioaz

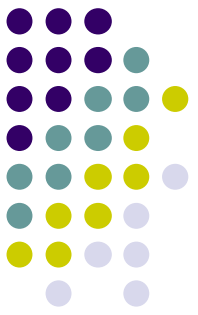


- Arakatzaileek interpretatzen dituzte.
- Besterik ezean `<script>` eta `</script>` etiketen arteko kodea:
 - Agertu ahala exekutatzen da
 - Exekuzioa HTML dokumentuaren analisiarekin batera egiten da, alegia, script-aren aurretik definitu ez diren elementuak ezin daitezke erreferentziatu.
 - Horregatik, kargatzean exekutatzen den kodeak ez luke denbora asko kontsumitu behar, script horien exekuzioa bukatu arte ezin baita dokumentua erakutsi.
 - Egokiagoa da, ahal denean, funtzioak definitzea eta funtzio horiei dei egitea gertaera-kudeatzailearen bitartez.
 - Etiketen artean dagoen kodea eta `src` atributuaren bitartez irakurtzen diren kanpo-fitxategiak, besterik ezean, berdin-berdin exekutatzen dira
 - Baina, kanpoko script-en exekuzio-modua eralda daiteke `async` eta `defer` atributuekin.

Script elementua

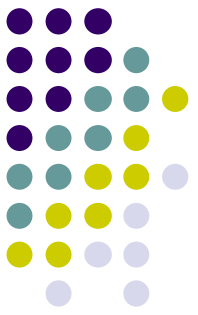


- Atributuak:
 - `type`:
 - beharrezkoa HTML 4n, ez ordea HTML5en.
 - Script-lengoaia adierazten du, MIME motaren bidez: `text/javascript`
 - `src`: script-a daukan fitxategiaren URLa adierazten du.
 - `charset`: Script-a kodetzeko erabiltzen den karaktere-multzoa.
 - `defer`: Boolearra da.
 - "defer" balioa hartzen du adierazteko script-a exekutatuko dela orriaren analisia bukatutakoan.
 - `async`: Boolearra da.
 - **Berria HTML 5en.**
 - "async" balioa hartzen du adierazteko scripta modu asinkronoan exekutatuko dela, eskuragarri dagoen bezain azkar.
 - Kanpoko scriptekin baino ezin daiteke erabili ("`src`" atributua jarri bada)



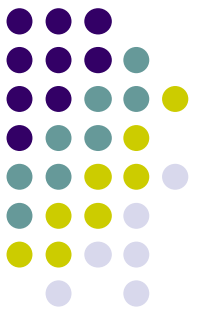
Script elementua. Oharrak

- `src` atributua jartzen bada, `<script>` elementuak hutsik egon behar du
- Kanpoko script-ak exekutatzeke moduak:
 - `async="async"` denean: script-a modu asinkronoan exekutatzen da gainontzeko orriarekiko
 - `async` jarri gabe eta `defer="defer"`: script-a exekutatzen da orriaren analisia bukatutakoan
 - `async` eta `defer` jarri gabe: script-a berehala exekutatzen da, arakatzailleak orriaren analisiarekin jarraitu baino lehen



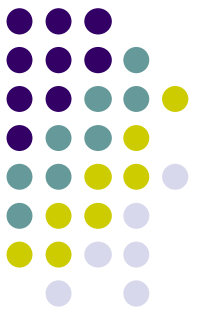
Egitura lexikala eta sintaxia

- C-ren antzekoa da eta, beraz, Javaren antza ere badu
- Banatzaileak: ; (ez da derrigorrezkoa) eta { }
- Iruzkinak: // , /* */
- Maiuskulak/minuskulak bereizi egiten dira.
- Identifikadoreak:
 - Lehenengo karakterea letra bat, _, edo \$ izan behar du.
 - Hortik aurrera digituak ere onartzen dira.
- Aldagaien asignazioa: =, -=, +=, /=, *=
- Eragile erlazionalak: <, >, ==, !=, ...
- Eragile logikoak: && , || , !



Datu-motak

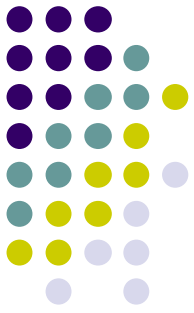
- Motatze-eredu ahula: aldagai berak mota desberdineko balioak har ditzake, eta lengoaiak berak egokitu egiten ditu balioak mota batetik bestera, komeni den bezala
- Zenbakizkoak: `var adina= 21, z=6.0;`
 - Osoak eta errealak berdin errepresentatzen dira
- Boolearrak: `erretzailea=true;`
- string edo karaktere-kateak: `izena="Pello"`
 - ' edo " karaktereak erabil daitezke string literalak mugatzeko
- Objektuak: Datu-mota generikoa. Aurrez definitutakoa izan liteke edo erabiltzaileak sor ditzake datu-motak.
 - Objektuen orientazioa: eraikitzaileak, propietateak, metodoak.
 - Metodoak: `document.write("idatziko den testua");`
 - Propietateak: `image.width` edo `image["width"]`
 - Objektuen eta array asoziatiboen arteko parekotasuna



Aldagaiak

- Aldagaiak erazagutzen dira (ez da derrigorrezkoa, baina)
 - `var` hitz erreserbatuaren bidez erazagutuz gero, aldagaiak iraunkorrak (`permanent`) *dira*.
 - Aldagai bat baino gehiago erazagut daitezke `var` gako-hitz bakarrarekin.
 - `var i, kont;`
- Erazagupenak eta hasieratzeak batera egin daitezke.
 - `var i = 0, kont = 0;`
- Aldagaien esparrua:
 - Aldagai orokorrek esparru orokorra hartzen dute (kodeko edozein tokitan definituta dago).
 - Funtzioen baitan erazagututako aldagaiek eta funtzio-parametroek esparru lokala hartzen dute (funtzioaren gorputzean baizik ez daude definituta).
 - Funtzio batean, beraz, aldagai orokorrak eta lokalak erabil daitezke.
 - *Funtzio bateko aldagaia kanpoan ere erabil daiteke, var hitz erreserbatuarekin erazagutu ez bada.*
- **this** aldagaia: *uneko objektua* adierazteko.

Aldagaiak (II)



- Aldagaiak globalak

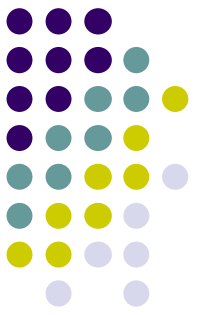
```
<script>
    var aldagaiGlobal
</script>;
```

- Eta lokak:

```
<script> function nireFun() {
    var aldagaiLokala    ...}
</script>;
```

- Aldagaien erazagupenean ez da mota esleitzen
- Array motakoko aldagaiak defini litezke, elementu bakoitza objektu bat izanik

```
var nireTau = new Array(10)
nireTau[0]=2015
```
- `this` aldagaia: uneko objektua adierazteko.

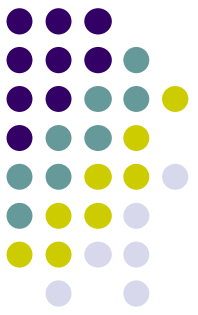


Espresioak

- Espresioen ebaluazioa egitean, JavaScript saiatzen da beharrezko diren bihurketak egiten.
- Oro har: karaktere-kateak beste edozerekin "+" eragiketaren bidez lotzen direnean, dena karaktere-kate bihurtzen da.
 - Aldiz, "-", "*" edo "/" eragiketekin konbinatuz gero, zenbaki bihurtzen ditu, ahal bada.

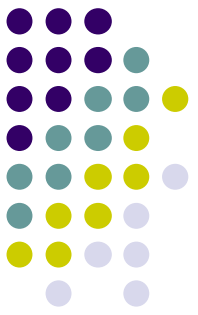
<http://www.etnassoft.com/2011/01/27/tipado-blando-en-javascript/>

- Adb.:
 - "a" * "b" espresioa ez da legezkoa
 - "3" * "4" espresioa legezkoa da!, eta 12 zenbakia da bere balioa, ez "12" string-a.



Kontrol egiturak

- if...else
- while ... break ... continue
- for
- for in
- function ...return



Kontrol egiturak: baldintzazkoak

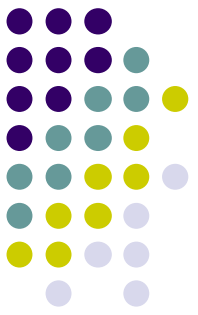
- **if...else**

```
if(baldintza){  
    aginduak}  
[ else{  
    aginduak}]
```

```
(baldintza)?  
    (agindu-bloke1):  
    (agindu-bloke2)
```

- **switch**

```
switch(n) {  
    case 1:    // 1 blokea exekutatu  
        break;  
    case 2:    // 2 blokea exekutatu  
        break;  
    ...  
    default: // blokea exekutatu  
        break;  
}
```



Kontrol egiturak: iterazioak

- while

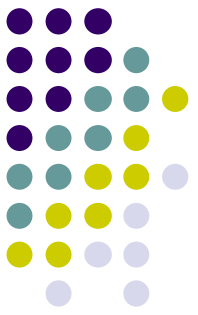
```
while (baldintza) {  
    aginduak  
}
```

- do...while

```
do {  
    aginduak  
} while (baldintza)
```

- break **eta** continue

- Begizta-etenak eragiten dituzte
- break: iterazio-agindutik irten
- continue: hurrengo begiztara pasa



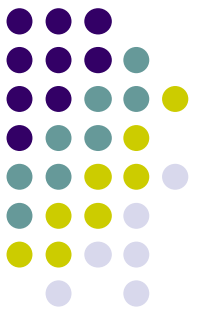
Kontrol egiturak: iterazioak

- **for**

```
for (HasierakoEspresioa; Baldintza; Inkrementua) {  
  
    aginduak  
  
}
```

- **for...in**

```
for (aldagaia in objektua) {  
  
    aginduak  
  
}
```



Kodearen egituraketa. Funtzioak

- Kodea funtzioen bidez egituratzen da.
- Funtzioetatik kanpo dagoen kodea orria kargatu ahala exekutatzen da.
- Funtzioetan argumentuak jar litezke.
- Funtzioek itzultzen duten balioaren mota ez da erazagutzen

```
function funtzio_izena (parametro_lista) {  
  ...  
  [ return dena_delakoa; ]  
}
```

- Parametroak balioz pasatzen dira

JS kodeen adibidea:

Formularioak egiaztatzeko kodea

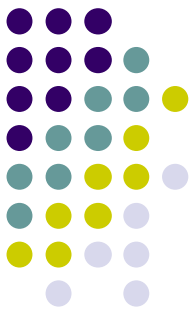


```
*C:\xampp\htdocs\adibidea\js\adFormularioaBalidatzen.html - Notepad++
Archivo  Editar  Buscar  Vista  Codificación  Lenguaje  Configuración  Herramientas  Macro  Ejecutar  Plugins  Ventana  ?
adFormularioaBalidatzen.html  new 1 x

1  <html>
2  <head>
3  <script language="javascript">
4      function egiaztatu() {
5          /*var izena = document.form1.izena.value;
6          var taldea = document.form1.taldea.value;*/
7          var izena=document.getElementById("izena").value;
8          var taldea=document.getElementById("taldea").value;
9          if (izena.length < 5 )
10             { alert("Errorea izenean"); return false; }
11          if (taldea < 1 || taldea > 34)
12             { alert("Errorea taldearen identifikazioan");
13               return false; }
14          return true;
15      }
16  </script>
17  </head>
18  <body>
19  <form action="https://www.ehu.eus/eu/" method="get" id="form1" onsubmit="return egiaztatu()">
20      Izena: <input type="text" id="izena" value=""> <br/> <br/>
21      Taldea: <input type="text" id="taldea" value="" size="3" maxlength="3"><br/> <br/>
22      <input type="submit" value="Bidali"><br/>
23  </body>
24  </html>

per Text Markup Language file      length : 803   lines : 24      Ln : 24   Col : 8   Sel : 0 | 0      Windows (CR LF)  UTF-8      INS
```

JS kodeen adibidea: Formularioak maneiatzen



```
C:\xampp\htdocs\adibideak\js\adFormularioakManeiatzen.html - Notepad++
Archivo  Editar  Buscar  Vista  Codificación  Lenguaje  Configuración  Herramientas  Macro  Ejecutar  Plugins  Ventana  ?
new 1 x aDFormularioakManeiatzen.html x
1 <html>
2 <head>
3 <title>Formularioen adibidea</title>
4 <script language="JavaScript">
5 function egiaztatu(helbidea) {
6     if (helbidea.indexOf("@") != -1) return true;
7     else { alert("Helbide zuzen bat izan behar du");
8         return false; }
9 }
10 </script>
11 </head>
12 <body>
13 <form name="nireForm1" method="POST" action="mailto:r.arruabarrena@gmail.com"
14 <enctype="text/plain" onSubmit="return egiaztatu(this.eposta.value)">
15     Bidal diezaguzu zure eposta: <input type="text" name="eposta"><BR>
16     <input type="submit" value="Bidali eposta">
17 </form>
18 </body>
19 </html>
20
```

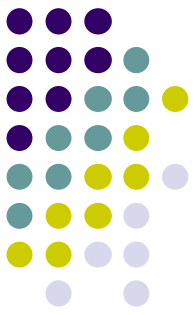
Formularioen adibidea

file:///C:/xampp/htdocs/adibideak/js/aD

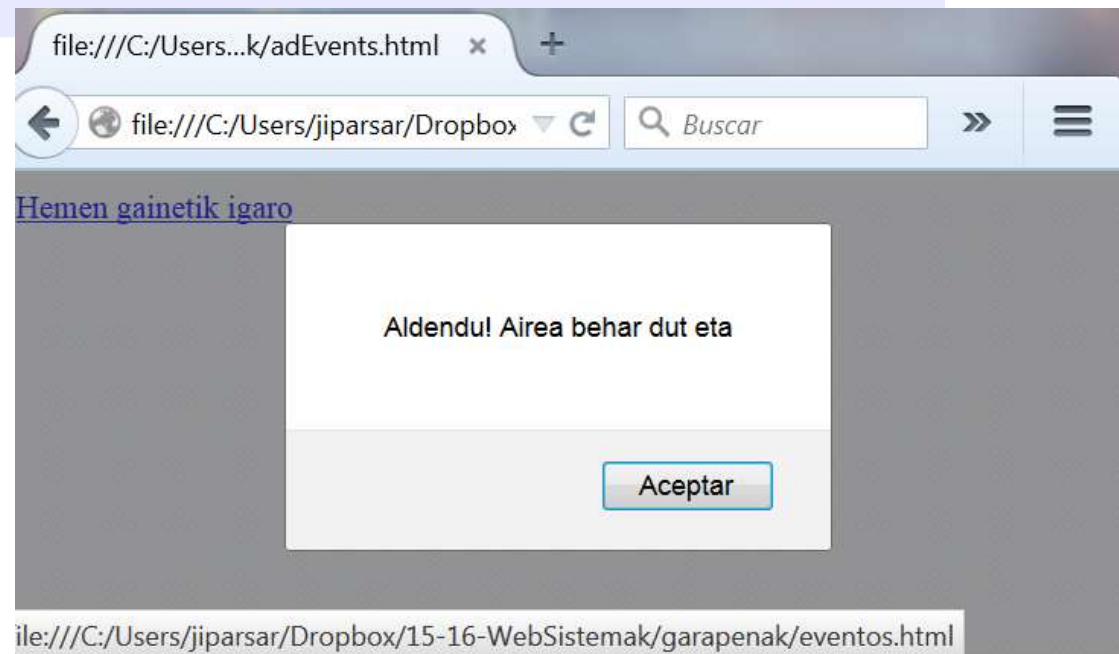
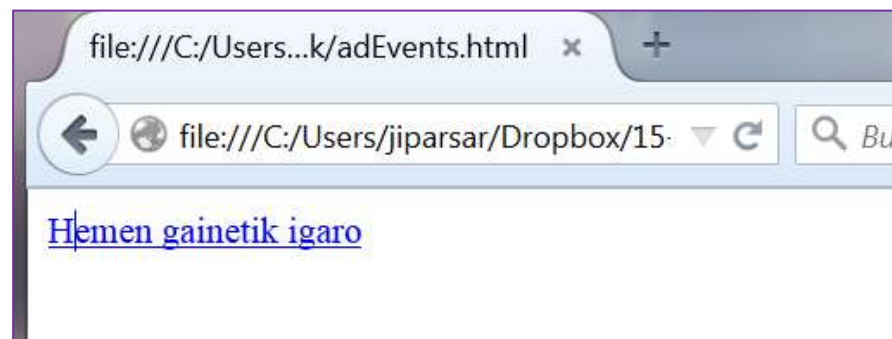
Bidal diezaguzu zure eposta: zorionak@gmail.com

Bidali eposta

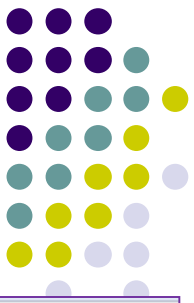
JS kodeen adibidea: Gertaeren manipulazioa



```
new 1 x adEvents.html x adLeihoaSortzen.html x
1 <html>
2 <head>
3 <script language="JavaScript">
4 <!-- JavaScript ez duten nabigatzaileentzat kodea ezkutuan uzten dute iruzkinek
5 function Alarma() { alert("Aldendu! Airea behar dut eta"); }
6 -->
7 </script>
8 </head>
9 <body>
10 <a href="http://www.w3schools.com/js/js_events.asp" onMouseOver="Alarma()" >
11 Hemen gainera igaro </a>
12 </body>
13 </html>
```



JS kodeen adibidea: Leihoen manipulazioa

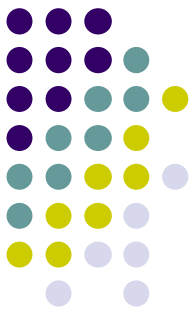


```
adLeioaSortzen.html
<html>
<head>
<script language="JavaScript">
function LeihoaIreki()
{ NireLeihoa=window.open("", "NireLeihoPropioa", "toolbar=no, directories=no, menubar=no, status=yes");
  NireLeihoa.document.write("<head><title>Leiho berri bat</title></head>");
  NireLeihoa.document.write("<center><h1><b>"
+ "Hau zuk nahi duzuna izan liteke</b></h1></center>");
}
</script>
</head>

<body>
  <form>
    <input type="button" name="Boton1" value= "Zabal nazazu!" onClick="LeihoaIreki()" >
  </form>
</body>
</html>
```



JS kodeen adibidea: Taulak eta elementuen sorrera

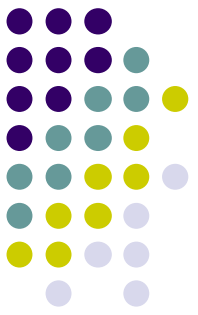


```
ikusBaliokak.js aDTableRow.html
1 <html>
2 <head>
3 <script language='javascript'>
4 function addRow(a,b) {
5     var r = document.createElement('tr');
6     var ca = document.createElement('td');
7     var cb = document.createElement('td');
8     var ta = document.createTextNode(a);
9     var tb = document.createTextNode(b);
10    var t = document.getElementById('tbl');
11    ca.appendChild(ta);
12    cb.appendChild(tb);
13    r.appendChild(ca);
14    r.appendChild(cb);
15    t.appendChild(r);
16 }
17 </script>
18 </head>
19
20 <body>
21 <form>
22     Column A: <input type='text' name='a'> <br>
23     Column B: <input type='text' name='b'> <br>
24     <input type='button' value='add Row' onClick='javascript:addRow(a.value, b.value)'/>
25 </form>
26
27 <table id='tbl'>
28     <tr><td>Column A</td><td>Column B</td></tr>
29 </table>
30
31 </body>
32 </html>
```

Hyper Text Markup Language file length : 737 lines : 32

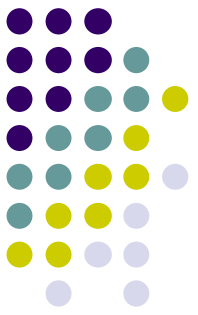
Column A: Bizkaia
Column B: Bilbo

Column A	Column B
Gipuzkoa	Donostia
Bizkaia	Bilbo



DOM eta JS (labur bilduz)

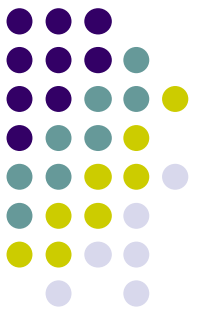
- DOM-ari esker arakatzailerik web garatzaileei web-orriak maneiatzen uzten diete JS erabiliz.
- Nagibatzailerik web-orria kargatzen duen aldi bakoitzean, HTML eta CSS kodea aztertzen du, eta dokumentuari dagokion DOM zuhaitza eraikitzen du.
- Eraikitako DOMa berez JS objektu handi bat da, orriaren inguru jakin/alda nahi genezakeen guztia duena, hala nola, etiketa bakoitza, atributu eta etiketa bakoitzaren estiloa.
- JS-etik web-orri baten DOMa atzitzeko, “document” aldagai globala erabiltzen da. Hortik, objektuaren propietateak, edo objektuko funtzioak (metodoak) erabil genitzake.



DOM eta JS (labur bilduz 2)

- **DOMa maneiatzeko** estrategia orokorrak bi pauso:
 1. DOMeko nodo bat aurkitu DOMen atzipen metodoren bat erabiliz
 - `getElementById()` // `"#idazlea"`
 - `getElementsByTagName()`
 - `querySelector()`
 - `getElementsByClassName()`
 - `querySelectorAll()` // `"p .letra"` `"ul .elikagai"`
 2. DOMeko nodoa maneiatu aldatuz haren atributuak, estiloak, innerHTML bidez, edo nodo berriak kateatuz.

Oharra: "document" ez da benetako web orria, baina nabigatzaileak uneko orriaren isla prezisoena/hurbilena izan dadila saiatzen da.



DOM eta JS (labur bilduz 3)

- **DOMa gertaeren bidez maneiatzen**

KA-k proposatzen duen estrategiak 3 pauso ditu:

1. erabaki eta aurkitu zein HTML elementuk pairatuko duen gertaera
2. gertaera gauzatzean egikaritu araziko den funtzioa definitu
3. aurreko biak batu “*addEventListener()*” erabiliz

new 2 x Arrastaka.html x domManeiuuaKA.html x KA_mousemove.js x new 3 x

1<!DOCTYPE html>

2<html>

3<head>

4<meta charset="utf-8">

5<title>DOMa maneiatzen</title>

6<style>

7body { background-color: lightblue; }

8h1 { color: white; text-align: center; }

9p { font-family: verdana; font-size: 20px; }

10

11#egilea { font-weight: bold; color: purple; }

12.elikagai { font-family: versailles; }

13</style>

14</head>

15<body>

16<h1 id="burukoa">Produktu berriak</h1>

17<h3> Ioritz Ganzara

18Hazi osasungarriak:

19<li class="elikagai">Chia

20<li class="elikagai">Kinoa

21<li class="elikagai">Liho

22

23Perretxiko osasungarriak:

24<li class="elikagai">onddo beltza

25<li class="elikagai">Amanita caesarea

26<li class="elikagai">Tripakia

27

28Perretxiko pozointsuak:

29<ul id="pozointsu">

30Amanita phalloides

31amanita muscaria

32Galanperna hiltzailea

33

34<script>

35

DOMa maneiatzen x +

file:///gorde.ehu.es/home/17\

Produktu berriak

Rosa Arruabarrena sukaldaria

Hazi osasungarriak:

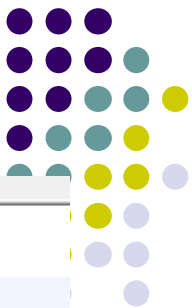
- chia
- kinoa
- liho

Perretxiko osasungarriak:

- onddo beltza
- amanita caesarea
- tripakia

Perretxiko pozointsuak:

- Amanita phalloides
- amanita muscaria
- Galanperna hiltzailea



```
new 2  Arrastaka.html  dommaneriakA.html  KA_mousemove.js  new 3
35 <script>
36
37     var egileaEl= document.getElementById("egilea");
38     console.log(egileaEl);
39     egileaEl.innerHTML="Rosa Arruabarrena";
40     egileaEl.style.color="cyan";
41     egileaEl.style.fontFamily="verdana";
42
43     var elikagaiakEl=document.querySelectorAll("ul .elikagai");
44     console.log(elikagaiakEl.length);
45     for (var i=0; i<elikagaiakEl.length ; i++){
46         var text1= elikagaiakEl[i].innerHTML;
47         elikagaiakEl[i].innerHTML=text1.toLowerCase();
48     }
49
50     var arriskuaEl=document.getElementById("pozointsu"); // gertaera pairatuko duen elementua
51     var onmouseoverArrikuspeanF=function() { // egikarituko den funtzioa gertaera
52         arriskuaEl.style.backgroundColor="red"; // gertatzerakoan
53     }
54
55     arriskuaEl.addEventListener("mouseover",onmouseoverArrikuspeanF);
56     var onmouseoutArrikuspeanF=function() { // egikarituko den funtzioa gertaera
57         arriskuaEl.style.backgroundColor="lightblue"; // gertatzerakoan
58     }
59     arriskuaEl.addEventListener("mouseout",onmouseoutArrikuspeanF);
60
61 </script>
62 </body>
63 </html>
64
```

Hyper Text Markup Language file length : 2.072 lines : 64 Ln : 57 Col : 72 Sel : 0 | 0