PROIECT

Site tutorial pentru proba de bacaureat la informatica

Cuprins

1.Introducere

2.Implementare

2.1. HTML

2.2. CSS

2.3. JavaScript

3.Structura si continutul proiectului

4.Bibliografie.

1.Introducere

BAC++ isi propune sa sintetizeze teoria necesara promovarii probei scrise la informatica .

Materia este structurata pe subiectele I , II si III , pentru o mai buna organizare si pentru o intelegere mai usoara .

2. Implementare

2.1. HTML

**HyperText Markup Language** (**HTML**) este un [limbaj de marcare](http://ro.wikipedia.org/wiki/Limbaj_de_marcare) utilizat pentru crearea paginilor web ce pot fi afișate într-un [browser](http://ro.wikipedia.org/wiki/Browser) (sau navigator). Scopul HTML este mai degrabă prezentarea informațiilor – paragrafe, fonturi, tabele ș.a.m.d. – decât descrierea semanticii documentului.

HTML este o formă de marcare orientată către prezentarea documentelor text pe o singura pagină, utilizând un software de redare specializat, numit *agent utilizator HTML*, cel mai bun exemplu de astfel de software fiind *browserul web*. HTML furnizează mijloacele prin care conținutul unui document poate fi adnotat cu diverse tipuri de metadate și indicații de redare. Indicațiile de redare pot varia de la decorațiuni minore ale textului, cum ar fi specificarea faptului că un anumit cuvânt trebuie subliniat sau că o imagine trebuie introdusă, până la scripturi sofisticate, hărți de imagini și formulare. Metadatele pot include informații despre titlul și autorul documentului, informații structurale despre cum este împărțit documentul în diferite segmente, paragrafe, liste, titluri etc. și informații cruciale care permit ca documentul să poată fi legat de alte documente pentru a forma astfel [hiperlink](http://ro.wikipedia.org/wiki/Hiperlink)-uri (sau [web](http://ro.wikipedia.org/wiki/Web)-ul).

HTML este un format text proiectat pentru a putea fi citit și editat de oameni utilizând un editor de text simplu. Totuși scrierea și modificarea paginilor în acest fel solicită cunoștințe solide de HTML și este consumatoare de timp. Editoarele grafice (de tip WYSIWYG) cum ar fi [Macromedia Dreamweaver](http://ro.wikipedia.org/wiki/Macromedia_Dreamweaver), [Adobe GoLive](http://ro.wikipedia.org/w/index.php?title=Adobe_GoLive&action=edit&redlink=1) sau [Microsoft FrontPage](http://ro.wikipedia.org/wiki/Microsoft_FrontPage) permit ca paginile web sa fie tratate asemănător cu documetele Word, dar cu observația că aceste programe generează un cod HTML care este de multe ori de proastă calitate.

HTML se poate genera direct utilizând tehnologii de codare din partea serverului cum ar fi [PHP](http://ro.wikipedia.org/wiki/PHP), [JSP](http://ro.wikipedia.org/w/index.php?title=JavaServer_Pages&action=edit&redlink=1) sau [ASP](http://ro.wikipedia.org/wiki/ASP). Multe aplicații ca [sistemele de gestionare a conținutului](http://ro.wikipedia.org/w/index.php?title=Content_management_system&action=edit&redlink=1), [wiki-uri](http://ro.wikipedia.org/wiki/Wiki)și [forumuri web](http://ro.wikipedia.org/w/index.php?title=Forum_web&action=edit&redlink=1) generează pagini HTML.

HTML este de asemenea utilizat în [e-mail](http://ro.wikipedia.org/wiki/E-mail). Majoritatea aplicațiilor de e-mail folosesc un editor HTML încorporat pentru compunerea e-mail-urilor și un motor de prezentare a e-mail-urilor de acest tip. Folosirea e-mail-urilor HTML este un subiect controversat și multe [liste de mail](http://ro.wikipedia.org/w/index.php?title=Liste_de_mail&action=edit&redlink=1) le blochează intenționat.

HTML este prescurtarea de la Hyper Text Mark-up Language si este codul care sta la baza paginilor web. Paginile HTML sunt formate din etichete sau tag-uri și au extensia .html sau .htm .În marea lor majoritate aceste etichete sunt pereche, una de deschidere <eticheta> și alta de închidere </eticheta>, mai există și cazuri în care nu se închid, atunci se folosește <eticheta /> browserul interpretează aceste etichete afișând rezultatul pe ecran. HTML-ul nu este un limbaj case sensitiv (nu face deosebirea între litere mici și mari). Pagina principala a unui domeniu este fisierul index.html respectiv index.htm

Această pagină este setată a fi afișată automat la vizitarea unui domeniu. De exemplu la vizitarea domeniului www.nume.ro este afișată pagina www.nume.ro/index.html.

Unele etichete permit utilizarea de atribute care pot avea anumite valori: <eticheta atribut="valoare"> ... </eticheta>

Componenta unui document HTML este: 1. versiunea HTML a documentului 2. zona head cu etichetele <head> </head> 3. zona body cu etichetele <body> </body> sau <frameset> </frameset>

Toate paginile HTML încep și se termină cu etichetele <html> și </html>. În interiorul acestor etichete găsim perechile <head>, </head> și <body>, </body>.

**head** conține titlul paginii între etichetele <title> și </title>, descrieri de tip <meta>, stiluri pentru formatarea textului, scripturi și linkuri către fisiere externe (de exemplu scripturi, fișiere de tip CSS sau favicon).

Etichetele de tip meta conțin cuvinte cheie, descrierea paginii, date despre autor, informații utile motoarelor de căutare și au următorul format: <META NAME="nume" CONTENT="continut">

Exemplu: link către un fisier extern CSS: <link rel="stylesheet" type="text/css" href="css.css">

**body** găzduiește practic toate etichetele afișate de browser pe ecran.

Exemplu: o pagină HTML cu titlul Exemplu iar conținutul Conținut pagină

<html>

<head>

<title>Titlu</title>

</head>

<body>

Continut pagina

</body>

</html>

2.2. CSS

**CSS** *(Cascading Style Sheets)* este un standard pentru formatarea elementelor unui document [HTML](http://ro.wikipedia.org/wiki/HTML). Stilurile se pot atașa elementelor HTML prin intermediul unor fișiere externe sau în cadrul documentului, prin elementul <style> și/sau atributul style. CSS se poate utiliza și pentru formatarea elementelor [XHTML](http://ro.wikipedia.org/wiki/XHTML), [XML](http://ro.wikipedia.org/wiki/XML) și [SVGL](http://ro.wikipedia.org/w/index.php?title=SVGL&action=edit&redlink=1).

**CSS3 reprezintă un upgrade ce aduce c**â**teva atribute noi și ajută la dezvoltarea noilor concepte in webdesign.**

Unele dintre cele mai importante segmente (*module*) noi adăugate acestui standard pentru formatarea elementelor HTML aduc un plus considerabil in dezvoltarea activități webdesign.

Mai jos sunt prezente in listă cele mai importante modulele adăugate in CSS3:

* Selectors
* Box Model
* Backgrounds and Borders
* Image Values and Replaced Content
* Text Effects
* 2D/3D Transformations
* Animations
* Multiple Column Layout
* User Interface

Deși au apărut unele deficiente de compatibilitate intre browsere, majoritatea proprietăților CSS3 au fost implementate cu succes in variantele browserelor noi.

2.3. JavaScript

**JavaScript** (**JS**) este un [limbaj de programare](http://ro.wikipedia.org/wiki/Limbaj_de_programare) [orientat obiect](http://ro.wikipedia.org/wiki/Programare_orientat%C4%83_pe_obiecte) bazat pe conceptul[prototipurilor](http://ro.wikipedia.org/w/index.php?title=Prototip_(POO)&action=edit&redlink=1).[[5]](http://ro.wikipedia.org/wiki/JavaScript#cite_note-FOOTNOTEFlanaganFerguson20061-5) Este folosit mai ales pentru introducerea unor funcționalități în paginile web, codul Javascript din aceste pagini fiind rulat de către [browser](http://ro.wikipedia.org/wiki/Browser_web). Limbajul este binecunoscut pentru folosirea sa în construirea [siturilor web](http://ro.wikipedia.org/wiki/Website), dar este folosit și pentru acesul la obiecte încastrate (embedded objects) în alte aplicații. A fost dezvoltat inițial de către [Brendan Eich](http://ro.wikipedia.org/wiki/Brendan_Eich) de la [Netscape Communications Corporation](http://ro.wikipedia.org/wiki/Netscape_Communications_Corporation) sub numele de Mocha, apoi LiveScript, și denumit în final JavaScript.

În ciuda numelui și a unor similarități în sintaxă, între JavaScript și limbajul Java nu există nicio legătură. Ca și [Java](http://ro.wikipedia.org/wiki/Java), JavaScript are o sintaxă apropiată de cea a [limbajului C](http://ro.wikipedia.org/wiki/Limbajul_de_programare_C), dar are mai multe în comun cu [limbajul Self](http://ro.wikipedia.org/w/index.php?title=Limbajul_Self&action=edit&redlink=1) decât cu [Java](http://ro.wikipedia.org/wiki/Limbajul_Java).

Până la începutul lui 2005, ultima versiune existentă a fost JavaScript 1.5, care corespunde cu Ediția a 3-a a ECMA-262, [ECMAScript](http://ro.wikipedia.org/wiki/ECMAScript), cu alte cuvinte, o ediție standardizată de JavaScript. Versiunile de [Mozilla](http://ro.wikipedia.org/wiki/Mozilla) începând cu 1.8 Beta 1 au avut suport pentru [E4X](http://ro.wikipedia.org/w/index.php?title=E4X&action=edit&redlink=1), care este o extensie a limbajului care are de a face cu [XML](http://ro.wikipedia.org/wiki/XML), definit în standardul ECMA-357. Versiunea curentă de Mozilla, 1.8.1 (pe care sunt construite[Firefox](http://ro.wikipedia.org/wiki/Firefox) și [Thunderbird](http://ro.wikipedia.org/wiki/Thunderbird) versiunile 2.0) suportă JavaScript versiunea 1.7.

Cea mai des întâlnită utilizare a JavaScript este în scriptarea [paginilor web](http://ro.wikipedia.org/wiki/Pagin%C4%83_web). Programatorii web pot îngloba în paginile HTML script-uri pentru diverse activități cum ar fi verificarea datelor introduse de utilizatori sau crearea de meniuri și alte efecte animate.

Browserele rețin în memorie o reprezentare a unei pagini web sub forma unui [arbore](http://ro.wikipedia.org/wiki/Arbore) de [obiecte](http://ro.wikipedia.org/wiki/Obiect_(informatic%C4%83)) și pun la dispoziție aceste obiecte script-urilor JavaScript, care le pot citi și manipula. Arborele de obiecte poartă numele de [Document Object Model](http://ro.wikipedia.org/w/index.php?title=Document_Object_Model&action=edit&redlink=1) sau [DOM](http://ro.wikipedia.org/w/index.php?title=DOM&action=edit&redlink=1). Există un standard [W3C](http://ro.wikipedia.org/wiki/W3C) pentru DOM-ul pe care trebuie să îl pună la dispoziție un browser, ceea ce oferă premiza scrierii de script-uri [portabile](http://ro.wikipedia.org/w/index.php?title=Portabilitate&action=edit&redlink=1), care să funcționeze pe toate browserele. În practică, însă, standardul W3C pentru DOM este incomplet implementat.

**Tipuri de date**

#### Nedefinit

Valoarea de "undefined" este atribuită la toate variabilele neinițializate, și, de asemenea, într-un context boolean, valoarea undefined este considerat o valoare de fals.

var test ; // variabila declarata, dar nu este definita

#### Null

Spre deosebire de nedefinit, nul este de multe ori setat pentru a indica faptul că ceva a fost declarat, dar a fost definit pentru a fi gol. Într-un context boolean, valoarea nulă este considerată o valoare de fals în JavaScript.

#### String

Un string (sau șir) în Javascript este o secvență de caractere cuprinsă între ghilimele duble sau single.

var greeting = "Hello, world!" ;

var anotherGreeting = 'Greetings, people of Earth.' ;

Puteți accesa caractere individuale dintr-un șir utilizând charAt metoda (furnizate de String.prototype ). Aceasta este metoda preferată atunci când accesarea de caractere individuale din cadrul unui șir, pentru că, de asemenea, funcționează non-moderne browsere:

var h = greeting. charAt ( 0 ) ;

#### Boolean

JavaScript oferă un tip de date Boolean cu valorile true și false. Operatorul returnează șirul "boolean" pentru aceste tipuri de primitive.

Atunci când este utilizat într-un context logic, 0 , -0 , null , NaN , undefined , iar șir vid ( "" ) evaluează în false din cauza constrângerii automate.

Când conversia de tip este necesară, JavaScript convertește String, Number, Boolean, sau operanzilor obiect, după cum urmează: [5]

* Șir de caractere este convertit la o valoare număr. JavaScript încearcă să transforme literal șir de caractere numeric, la o valoare tip de număr. În primul rând, o valoare de matematică este derivat din literal șir de caractere numeric. Apoi, această valoare este rotunjită la cea mai apropiată valoare tip de număr.
* Dacă unul dintre operanzi este un Boolean, operand Boolean este convertit la 1 dacă este true sau la 0, dacă este false .
* Dacă un obiect este comparat cu un număr sau un șir de caractere, JavaScript încearcă să se întoarcă valoarea implicită pentru obiect. Un obiect este convertit la un șir de caractere sau o valoare numerică, folosind .valueOf() sau .toString() metode de obiect. Dacă acest lucru nu reușește, o eroare de execuție este generată.

Unii experți folosesc termenii "true" și "false" pentru a descrie modul în care valorile de diferite tipuri, se comportă atunci când a evaluat într-un context logic, în special în ceea ce privește cazurile de margine. Operatorii logici binare a returnat o valoare booleană în primele versiuni de JavaScript, dar acum se vor întoarce unul dintre operanzi loc.

Operandul stâng este returnat în cazul în care acesta poate fi evaluat ca: false , în cazul de asociere ( a && b ) , sau true , în cazul de disjuncției ( a || b ) , în caz contrar dreptul de-operand este returnat.

O expresie poate fi în mod explicit exprimate la un primitiv de booleană: dublarea logic operatorul negație ( !! ) , folosind Boolean() funcția, sau folosind operatorul condițional ( c ? t : f ) .

#### Array

Un „Array” (sau „tablou”, „matrice”) este un obiect JavaScript proiectat pentru a stoca valori de date indexate de chei întregi.

Matricea, spre deosebire de tipul de obiect de bază, se patentează cu metode si proprietati pentru a ajuta programatorul în sarcini de rutină (de exemplu, join , slice , și push ).

var myArray = [ ] ; // Creează o variabilă nouă array cu numele myArray

myArray. push ( "hello world" ) ; // Completează valoarea de index 0 cu valoarea "hello world"

Matricile au o proprietate lenght (sau “lungime”) care este întotdeauna mai mare decât indicele de cel mai mare index utilizat în matrice.

Elemente de matrice pot fi accesate folosind normale de acces obiect de proprietate notație:

myArray [ 1 ] ; // elementul 2 în myArray

myArray [ "1" ] // Cele două de mai sus sunt echivalente.

Declarația a unui tablou poate folosi fie un tablou literal sau Array constructorului:

myArray = [ 0 , 1 , , , 4 , 5 ] ; // tablou cu lungime de 6 si 2 elemente nedefinite

myArray = new Array ( 0 , 1 , 2 , 3 , 4 , 5 ) ; // tablou cu lungime de 6 si 6 elemente

myArray = new Array ( 365 ) ; // un array gol cu lungimea de 365

Se mai poate utiliza declarația obiect literal:

câine = {culoare: "maro", dimensiune: "mare"};

câine [ "culoare" ] ; // rezultatul este "maro"

câine. culoare ; // de asemenea, rezultate în "maro"

Se pot utiliza literali declarație obiect și matrice pentru a crea rapid matrice care sunt asociative, multidimensionale, sau ambele. (Tehnic, JavaScript nu acceptă tablouri multidimensionale, dar le poate imita cu tablouri-de-matrice.)

pisici = [{culoare: "maro", dimensiune: "mari"}, {culoare: "negru", dimensiune: "mici"}];

pisici [0] ["dimensiune"]; // rezultatul este "mare"

câini = {rover: {culoare: "maro", dimensiune: "mari"}, spot : {culoare: "negru", dimensiune: "mici"}};

câini ["spot"] ["dimensiune"]; // rezultatul este "mici"

câini.rover.culoare ; // results in "brown" .. caini Rover de culoare; // rezultatele în "maro"

#### Date calendaristice

Un obiect Data stochează numărul de milisecunde începând cu data de 1970-01-01 00:00:00 UT.

new Date () // creează o nouă instanță reprezentând Data curentă data / ora.

new Date ( 2010 , 2 , 1 ) // creează o nouă instanță, reprezentând data de 2010-Mar-01 00:00:00

new Date ( 2010 , 2 , 1 , 14 , 25 , 30 ) // creează o nouă instanță, reprezentând data de 2010-Mar-01 14:25:30

new Date ( "2010-3-1 14:25:30" ) // creează o nouă instanță Data dintr-un șir.

### Variabile

Variabilele în standard, JavaScript nu au tip atașat, și orice valoare poate fi stocată în orice variabilă.

Variabilele sunt declarate cu un var declarație, mai multe variabile pot fi declarate la o dată.

Un identificator trebuie să înceapă cu o literă, de subliniere (\_), sau semnul dolar ($); caractere ulterioare pot fi, de asemenea, cifre (0-9).

Deoarece JavaScript este case-sensitive, scrisori include caracterele "A" la "Z" (cu majuscule) și caracterele "A" la "Z" (cu litere mici).

Variabilele declarate în afara oricărei funcții sunt globale. În cazul în care o variabilă este declarată într-un domeniu de aplicare mai mare, ea poate fi accesată de către funcțiile apelate de domeniu.

3.Structura si continutul proiectului

4.Bibliografie

[*http://ro.wikipedia.org*](http://ro.wikipedia.org)

[*http://www.w3schools.com/*](http://www.w3schools.com/)

[*www.ebacalaureat.ro*](http://www.ebacalaureat.ro)

*“ Culegere de probleme informatica ( intensiv si neintensiv)”- Giorgie Baniel Vlad , Daniela Marcu*