



# Java Script

dr Suzana Marković, dipl.ing. el.  
suzana.markovic@bss.edu.rs

# Programski jezici

- **Kompajlerski jezici** - programski kôd konvertuje se u mašinski pre nego što se i pokrene, što znači da će se ta konverzija izvršiti samo jednom (Java);
- **Interpretatorski jezici** – konverzija kôda dešava se u momentu njegovog pokretanja i izvršava se svaki put kada se kôd pokrene (JavaScript);



# Koje su razlike između Jave i JavaScripta?

- **Programski jezik** - kreiranje pravih programa koji se izvršavaju potpuno nezavisno od WWW čitača
- **Skript jezik** – interpretira ga veb čitač

# Java i JavaScript

Prevedi se na serveru pre izvršavanja na klijentu (kompajler).	Interpretira ga WWW čitač - ne prevodi se (interpretator).
Objektno-orientisan jezik. Postoji podrška za objekte, klase, nasleđivanje...	Objektno-baziran jezik. Postoje ugrađeni objekti sa definisanim metodama i osobinama, nema klase, baziran na prototipovima, limitiran rad sa nasleđivanjem
Apleti su odvojeni od HTML dokumenata (iz kojih se pozivaju)	Integrirano u HTML dokument
Promenljive se moraju deklarirati (jaka tipizacija)	Promenljive se ne moraju deklarirati (slaba tipizacija)

# Istorijat Javascript-a

- ▶ Nastao je 1995. godine – kod je mogao da se izvršava u veb pretraživaču Netscape Navigator 2.
- ▶ Microsoft je na osnovu javne dokumentacije Netscape-a objavio skoro identičnu verziju svog jezika JScript (izvršavao se u Explorer-u 3)
- ▶ Udruženje ECMA (European Computer Manufacturers Association) je usvojilo svoj standard za ovaj jezik, pri čemu je JavaScript postao Netscape-ova implementacija, a JScript Microsoft-ova.
- ▶ Izveden je iz C jezika



# JavaScript danas

- ▶ JavaScript – multifunkcionalni programski jezik za kontrolisanje toka programa u veb pretraživaču
  - ▶ Primer: kada korisnik unosi podatke u formu, JavaScript ih proverava, pronalazi grešku i obaveštava korisnika
- ▶ Dijalekti se razlikuju u odnosu na pretraživač  
ECMA script: JScript, JavaScript, ActionScript



# JavaScript danas

- U početku male funkcije – upravljanje formama i dinamika teksta
- Danas JavaScript omogućava crtanje 3-D grafike, upravljanje hardverom, ...smatra se glavnim pokretačem veb aplikacija.
- <https://experiments.withgoogle.com/chrome?tag=Javascript>



# Html i JavaScript

- ▶ HTML ograničavajući faktor – jednostavnija dinamika strane (npr. hover efekat) ostvaruje se pomoću css-a, ali ozbiljnija dinamika nije moguća
- ▶ Dinamika veb strane može biti na strani servera – CGI script ili na strani klijenta JavaScript.





# Skriptovanje na strani servera

- To je metoda programiranja na webu koja pokreće softver na serveru, a ne pretraživač, ili služi za instaliranje plug-in-ova za kreiranje dinamičkih veb stranica.
- **Node.js** je serverska JavaScript platforma idealna za pravljenje „real-time“ veb aplikacija.

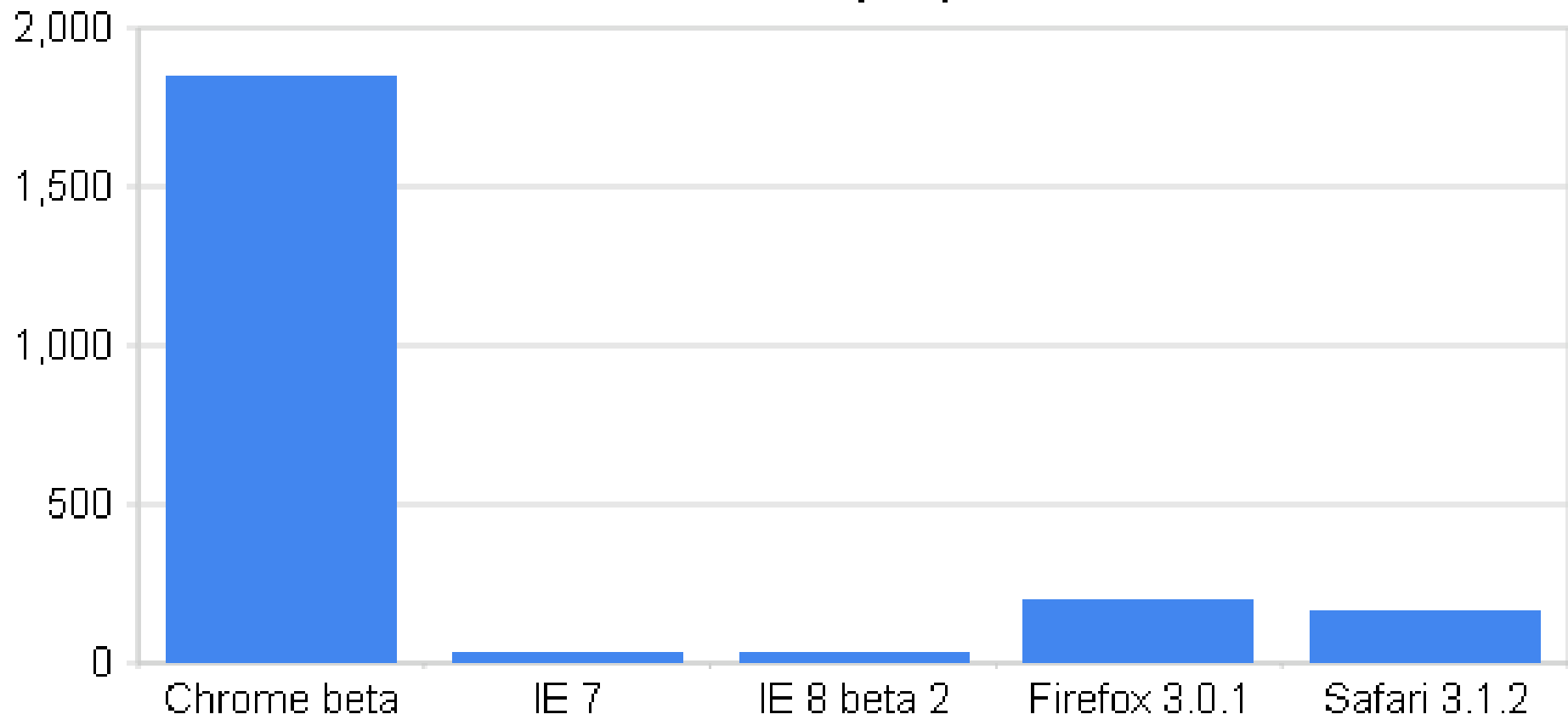


## Skriptovanje na strani klijenta

- Smešta se u HTML kod (ili kao posebna datoteka koja se poziva) i otvara u pretraživaču dajući veliku interaktivnost dokumentu.
- Npr. provera korisničke forme i otklanjanje eventualnih grešaka, pre slanja serveru.
- Omogućava promenu sadržaja veb stranice u zavisnosti od korisničkih unosa ili nekih drugih promenljivih.

# Java Script

Overall JavaScript Speed





# Osobine Javascript-a

- Javascript je platformski neutralan – kod se izvršava u okviru pretraživača bez obzira na hardversko i softversko okruženje
- Modularno programiranje – za čuvanje i izvršavanje koda koriste se posebni dokumenti sa .js ekstenzijom
- Integrisanost sa HTML-om – u okviru veb stranice moguće je kombinovati html i js kod, ali je moguće i iz js generisati html kod.



# Gde se pojavljuje i kako se implementira js

- Pojavljuje se:
  - U vidljivom delu html-a --> body tag
  - U nevidljivom delu html-a --> head tag
  - kao zaseban .js fajl
  - Script tagovi su **inline elementi** – nemaju uticaja na formatiranje strane
- Implementira se:
  - Direktnim unosom koda na stranu
  - Dodavanjem reference na eksterni fajl



# Izvršavanje koda

```
<script language="javascript">  
...javascript kod...  
</script>
```

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



# Alati za unos js koda

- Notepad
- Notepad ++
- Visual Studio
- Eclipse
- DreamViewer
- Konzola veb pretraživača (F12)

# JavaScript – korišćenje iz html – a i iz Js-a

```
<body>  
  <script>  
    document.writeln("Zdravo svete!");  
  </script>...
```

```
<script>  
  document.writeln("Zdravo svete!");  
</script>  
</head>
```

```
<script>  
  document.writeln("<h1>Zdravo svete!</h1>");  
</script>
```





# Eksterni java script

## Eksterni JS kod

```
<html>
<head>
<title>JS kod</title>
<script type="text/javascript"
src="javascriptKod.js">
</script>
</head>

<body onload="poruka()">

</body>
</html>
```

```
// JavaScript Document
function poruka()
{
    alert("Ovo je neka poruka")
}
```

# Svojstva Java Script-a

- Java script je case sensitive
- Oznaka ; na kraju naredbe je opcionalna
- Sa ; se mogu napisati višestruke JavaScript naredbe u istoj liniji
- Bez oznake ; JavaScript naredbe se moraju pisati u posebnim linijama
- Blokovi se pišu unutar zagrada {...}
- Blok predstavlja celinu – ekvivalentnu jednoj naredbi

# Komandni blokovi

- Naredbe se grupišu u blokove naredbi, koji su izdvojeni velikim zagradama:

```
{  
document.writeln("Da li ovo radi? ");  
document.writeln("Radi! <BR>");  
}
```

# Komentari u JavaScript-u

- Komentar na kraju reda – linije je `//`
- Proizvoljni komentar između znakova `/*` i `*/` – multi line comment
- Kao i u drugim programskim jezicima, komentari služe za:
  1. Opis programa običnim govornim tekstom
  2. Privremeno isključivanje delova programa tokom razvoja programa

# Komentari

```
<html>
<body>

<script type="text/javascript">
// This will write a header:
document.write("<h1>This is a header</h1>");
// This will write two paragraphs:
document.write("<p>This is a paragraph</p>");
document.write("<p>This is another paragraph</p>");
</script>

</body>
</html>
```

## **This is a header**

This is a paragraph

This is another paragraph

```
<html>
<body>

<script type="text/javascript">
/*
The code below will write
one header and two paragraphs
*/
document.write("<h1>This is a header</h1>");
document.write("<p>This is a paragraph</p>");
document.write("<p>This is another paragraph</p>");
</script>

</body>
</html>
```

## **This is a header**

This is a paragraph

This is another paragraph

# JavaScript promenljive

- ▶ JavaScript nije strogo tipizirani jezik, tako da deklaracija promenljivih nije obavezna.
- ▶ Promenljiva `x` se PRVI put u programu može navesti i kao:  
`x = 5;`
- ▶ Tip promenljive se deklarira implicitno – preko trenutno dodeljene vrednosti - script jezik
- ▶ Deklaracija promenljive `x`:  
`var x = 5;`  
`x = "Hello World";`
- ▶ Tip podatka koji sadrži promenljiva *može da se menja* u toku programa
- ▶ Ime promenljive je *case sensitive* i sastoji se od slova (a – z, A – Z), cifara (0 – 9) i znaka “\_” – underscore. Prvi znak imena promenljive mora biti slovo , “\_” ili \$.



# Definicija tipa podataka

- Ne zahteva se i nije dozvoljeno deklarirati tip podataka promenljive
- JavaScript interpreter automatski prepoznaje koji tip podataka je smešten u okviru promenljive i
- Dodeljuje tip podatka promenljivoj **dinamički**

# Osnovni – ugrađeni tipovi podataka

- Undefined – nije definisana vrednost
- Boolean – logički tip
- Number
  - Celi brojevi sa osnovom 10, 8 ili 16.
  - Racionalni brojevi - 3.14, 314E-2 ili 314e-2.
- String - proizvoljan niz karaktera između navodnika ("neki tekst") ili apostrofa ('neki tekst')
- Osim ovih osnovnih tipova postoje i objekti – ugrađeni i korisnički definisani.



# Aritmetički operatori

- ➔ Služe za obavljanje aritmetičkih operacija između promenljivih.
- ➔ U donjoj tabeli korišćena je vrednost  $y=5$ .

Operator	Description	Example	Result
+	Addition	$x=y+2$	$x=7$
-	Subtraction	$x=y-2$	$x=3$
*	Multiplication	$x=y*2$	$x=10$
/	Division	$x=y/2$	$x=2.5$
%	Modulus (division remainder)	$x=y\%2$	$x=1$
++	Increment	$x=++y$	$x=6$
--	Decrement	$x=--y$	$x=4$

**Operatori ++ i - - mogu biti prefiksni i postfiksni**

# Operatori dodele

- Služe za dodeljivanje vrednosti promenljivama.

Operator	Example	Same As	Result
=	x=y		x=5
+=	x+=y	x=x+y	x=15
-=	x-=y	x=x-y	x=5
*=	x*=y	x=x*y	x=50
/=	x/=y	x=x/y	x=2
%=	x%=y	x=x%y	x=0

# Operator + u stringovima

- Sabiranje stringova svodi se na njihovo povezivanje
- Prazan string " ".

```
txt1="What a very";  
txt2="nice day";  
txt3=txt1+txt2;
```

```
txt1="What a very";  
txt2="nice day";  
txt3=txt1+" "+txt2;
```

# Sabiranje brojeva i stringova

```
<html>
<body>

<script type="text/javascript">
x=5+5;
document.write(x);
document.write("<br />");
x="5"+"5";
document.write(x);
document.write("<br />");
x=5+"5";
document.write(x);
document.write("<br />");
x="5"+5;
document.write(x);
document.write("<br />");
</script>

<p>The common rule is: If you add a number
and a string,
the result will be a string.</p>

</body>
</html>
```

10  
55  
55  
55

The common rule is: If you add a number and a string, the result will be a string.

# Relacioni operatori

- Ovi operatori se koriste u logičkim izrazima radi utvrđivanja jednakosti ili razlike između promenljivih.
- Korišćena vrednost  $x=5$ ;

Operator	Description	Example
<code>==</code>	is equal to	<code>x==8</code> is false
<code>===</code>	is exactly equal to (value and type)	<code>x===5</code> is true <code>x==="5"</code> is false
<code>!=</code>	is not equal	<code>x!=8</code> is true
<code>&gt;</code>	is greater than	<code>x&gt;8</code> is false
<code>&lt;</code>	is less than	<code>x&lt;8</code> is true
<code>&gt;=</code>	is greater than or equal to	<code>x&gt;=8</code> is false
<code>&lt;=</code>	is less than or equal to	<code>x&lt;=8</code> is true

# Logički operatori

- Služe da odrede logiku između promenljivih
- Za  $x=6$  i  $y=3$

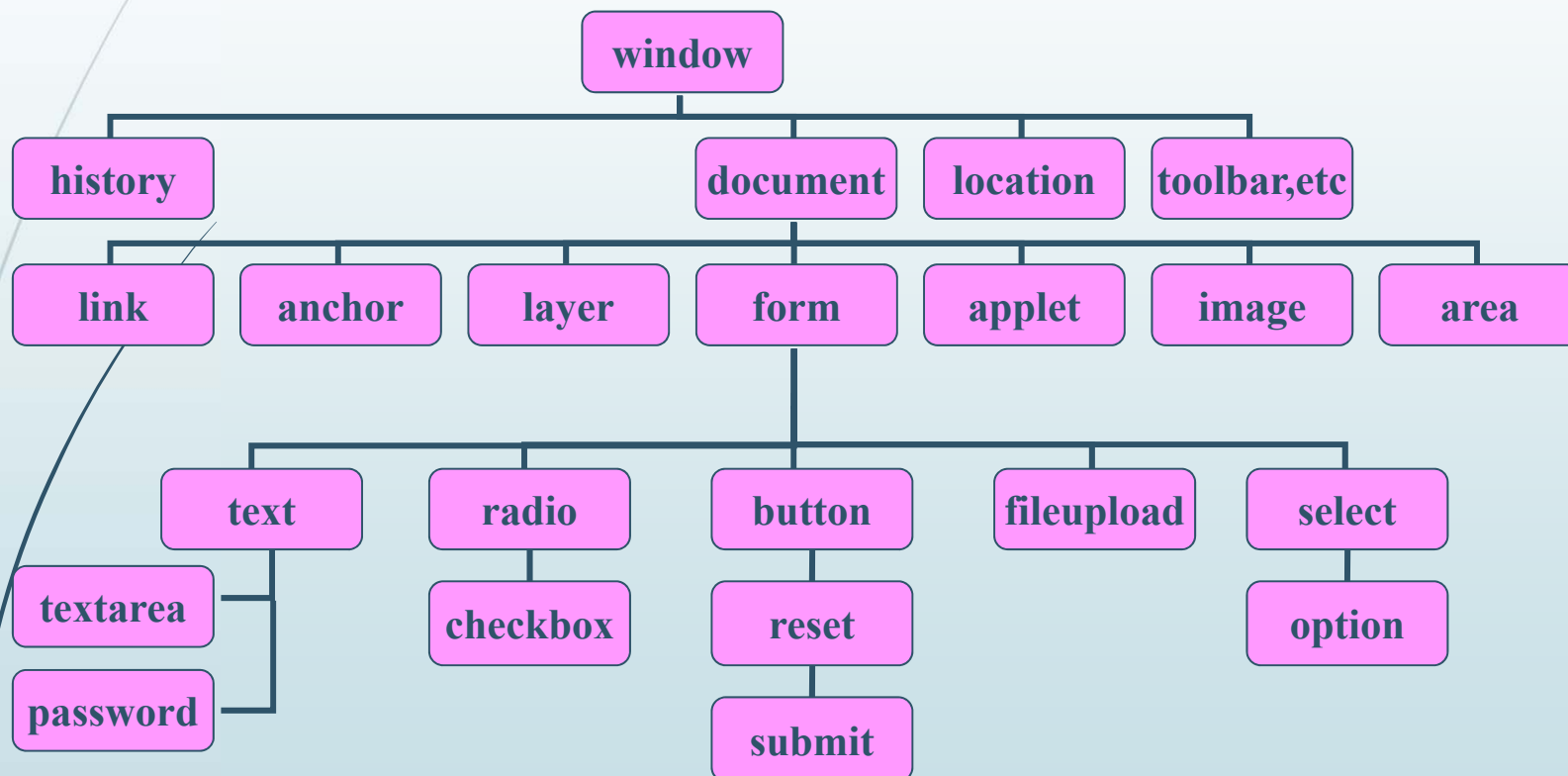
Operator	Description	Example
&&	and	$(x < 10 \ \&\& \ y > 1)$ is true
	or	$(x==5 \    \ y==5)$ is false
!	not	$!(x==y)$ is true

# Bitovski logički operatori

Operator	Description	Example
&	Logičko I	255 & 15 -- 00 001 111
	Logičko ILI	255   15 -- 11 111 111
^	Logičko Exkluzivno ILI	255 ^ 15 -- 11 110 000
<<	Pomeranje ulevo	010 << 2 -- 01000
>>	Pomeranje udesno	010000 >> 2 -- 0100
~	Logičko NE	~15 -- 11 110 000

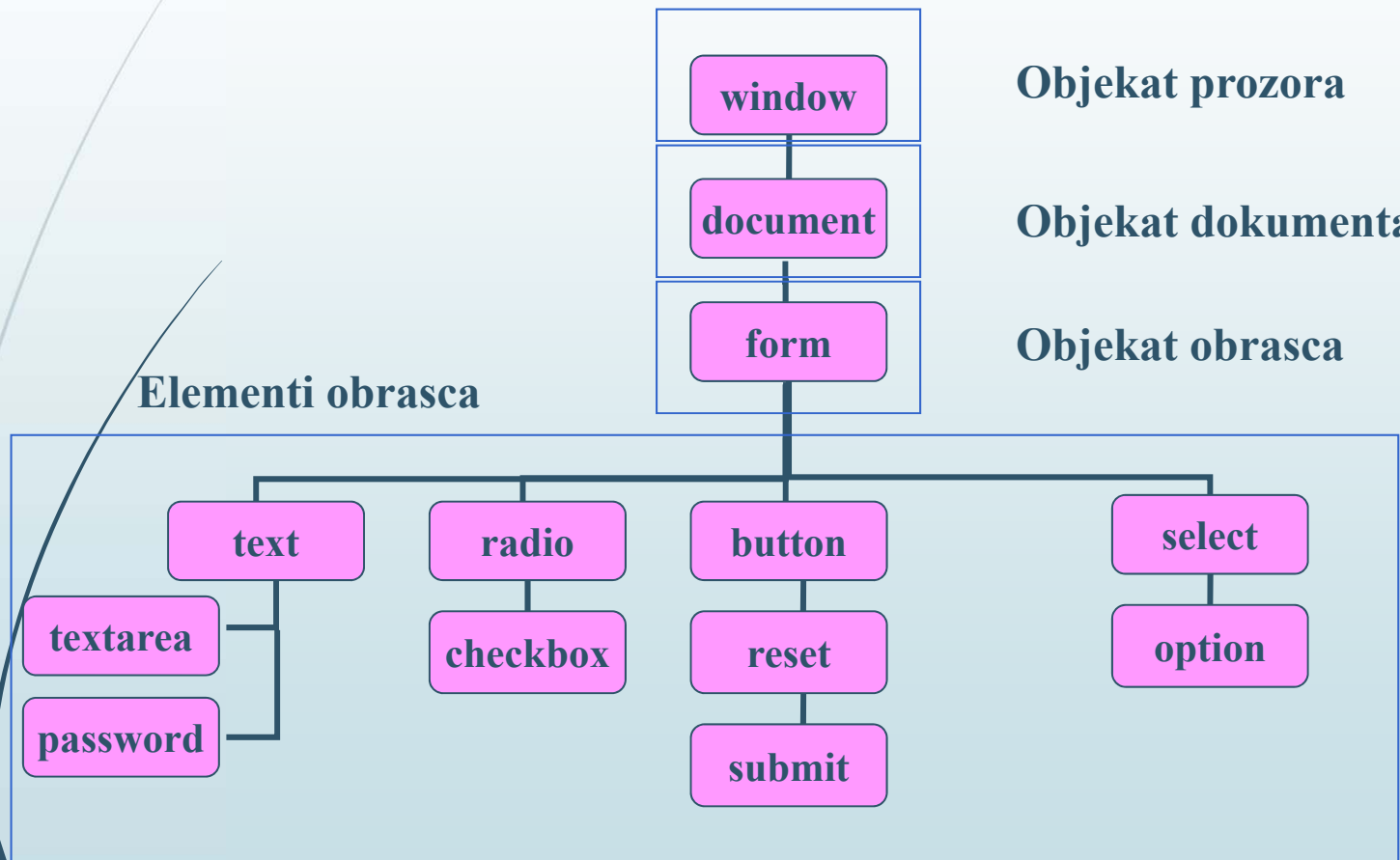
**255 --11 111 111**  
**15-- 00 001 111**

# Model objekata dokumenta



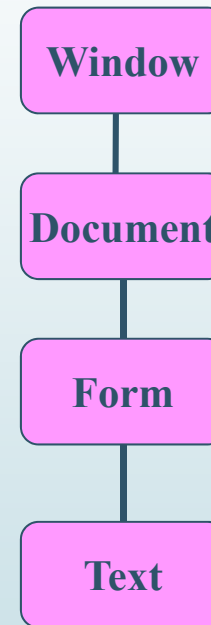


# Model objekata dokumenta



# Dodavanje elementa za unos teksta

```
<html>
<head><title>
Simple doc
</title></head>
<body>
<h1> Hello </h1>
<form>
<input type="text">
</form>
</body>
</html>
```



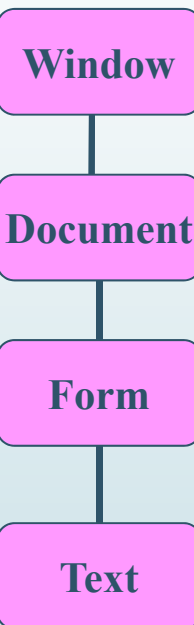
# Referenca objekta – polje za unos teksta

`window`

`window.document`

`window.document.formName`

`window.document.formName.textName`



# Definicija objekata

- ▶ U JavaScript-u objekat može biti: polje za unos teksta, dugme ili ceo HTML dokument
- ▶ Svaki od objekata je jedinstven
- ▶ Tri najvažnije odlike objekata definišu njegov izgled, ponašanje i kako ga skript kontroliše, to su:
  - ▶ **svojstva (properties)**
  - ▶ **metodi**
  - ▶ **procedure za obradu događaja**

# Svojstva

- Svojstva HTML objekta najčešće se podešavaju atributima HTML oznaka – statički metod
- Prisustvo JavaScripta često dodaje opcione attribute čija se inicijalna vrednost može modifikovati kada se dokument učita – dinamički metod
- Veb strana se tada tretira kao dokument.
- Dokument ima svojstva (properties), npr. boja njegove pozadine (bgColor).
- Pozivanje svojstva obavlja se na sledeći način:
  - document. ime svojstva - Za pristupanje svojstvima objekata u sintaksi se koristi tačka



# Skript tagovi

```
<html>  
<body bgcolor="brown">  
  <p>paragraf 1</p>  
  <script language="javascript">  
    document.bgColor="beige"  
  </script>  
</body>  
</html>
```

# Izlaz iz JavaScripta

- Ostvaruje se metodama
  - write (šalje tekst u prozor WWW čitača bez pomeranja) i
  - writeln (isto kao write(), s tim što se posle ispisa teksta kursor pomera u sledeći red objekta)
- `document.write("Test");`
- `document.writeln('Test');`

# Alert()

- ▶ JavaScript ima mogućnost da kreira male dijaloge za *ispis* ili *unos* teksta koji su potpuno nezavisni od sadržaja HTML dokumenta
- ▶ Najjednostavniji način da se prosledi kratka poruka korisniku je korišćenje *alert metoda*.
  - ▶ `alert("Kliknite na OK za nastavak.");` ili
  - ▶ `window.alert(...)`
- ▶ Window - najviši vrh hijerarhije prostora imena – globalni objekat sa metodom alert





# Interakcija sa korisnikom

- `alert()` metoda omogućava slanje poruka korisniku, ali ne i preuzimanje odgovora od njega.
- Metod `prompt()` omogućava jednostavnu interakciju sa korisnikom.
- `prompt()` kreira dijalog prozor u kome se može proslediti poruka korisniku, ali se formira i polje u koje korisnik može uneti proizvoljan tekst.

# Interakcija sa korisnikom - primer

```
<html>
<head>
  <title>Primer</title>
</head>
<body>
  <script language="JavaScript">
    document.write("<h1>Dobrodošli ");
    document.writeln(prompt("Unesite Vaše ime:",
"ime"));
    document.writeln(prompt("Unesite Vaše prezime:",
"prezime"));
    document.write(" u svet JavaScripta!</h1>");
    document.write('<h2>Čestitamo!</h2>');
  </script> </body></html>
```

# Primer

```
<html>
<body>
  <p>paragraf 1</p>
  <script language="javascript">
    // 1.script blok
    alert("prvi skript blok");
  </script>
  <p>paragraf 2</p>
  <script language="javascript">
    // 2.script blok
    document.bgColor = "brown";
    alert("drugi skript blok");
  </script>
  <p>paragraf 3</p>
</body></html>
```

// - komentar

alert() – funkcija

Poruka unutar message box-a specificira se unutar zagrada alert() funkcije i zove se funkcijski **parametar**.