

# Tehnologije za Web

Kaskadni stilovi (CSS)  
Klijentsko programiranje  
Serversko programiranje



## Kaskadni stilovi (CSS)



- ❑ HTML je izvorno projektovan kao strukturni jezik – nije bio pravljen da upravlja izgledom stranice, već se pretpostavljalo da:
  - ❖ autor pomoću HTML-a zadaje samo elemente stranice,
  - ❖ dok čitač Weba određuje **kako** će oni na ekranu izgledati.
- ❑ Sa podrškom za fontove, boje i druge vizuelne efekte, **HTML je postao i alatka za dizajniranje Web strana.**
- ❑ **Kaskadni stilovi su uvedeni s namerom da se jeziku HTML vrati prvobitna namena – čisto strukturiranje Web strana.**
- ❑ Uvođenjem kaskadnih stilova **odvaja se opis strukture (HTML) od opisa izgleda Web strane (CSS).**

## CSS



- ❑ **Kaskadni opisi stilova (CSS)** predstavljaju skup pravila pomoću kojih se **upravlja rasporedom i izgledom stranice**.
- ❑ Opisi stilova – CSS omogućavaju pisanje naredbi za izgled i formatiranje u zaglavlju Web stranice ili u spoljašnjoj datoteci, van HTML koda kojim se zadaje sadržaj stranice.
- ❑ CSS pravila su jednostavni tekstualni iskazi, pomoću kojih istovremeno može da se raspoređuje i formatira jedan ili više elemenata.
- ❑ **Stranica na kojoj su razdvojeni sadržaj i izgled lakše se održava i ažurira.**

3

## Specifikacije kaskadnih stilova



- ❑ Specifikacija kaskadnih stilova je veoma obimna i njoj se neprekidno dodaju nova svojstva

Verzija CSS-a	Prikaz svojstava date verzije
CSS1	Upravljanje tekстом, uključujući mogućnost zadavanja <b>fontova</b> , veličine slova, stila i razmaka. Mogućnost zadavanja <b>boje</b> pozadine i <b>pozadinske slike</b> . Podešavanje <b>margin</b> a, <b>ivica</b> i <b>razmaka</b> između teksta i objekata. Podrška za stilove za <b>liste</b> .
CSS2	Poboljšana svojstva vezana za <b>štampanje</b> . Podrška za <b>dinamičko iscertavanje</b> , utisnute fontove, <b>pozicioniranje elemenata</b> (CSSP), kaskadne stilove sa XML-om. Podrška <b>kontroli okruženja</b> – mogućnost <b>izmene oblika pokazivača miša</b> . Ograničeni postupci, npr. efekat lebdjenja kod hiperveza.
CSS3	Donosi podršku za <b>vertikalno</b> ispisani tekst i dizajn u više <b>kolona</b> . Poboljšana je podrška za pridruživanje postupaka i stilova. <b>Integrisana</b> je tehnologija grafike, boje i fontova.

4

## Metode izrade HTML dokumenata i kaskadnih stilova



\*WYSIWYG - What You See Is What You Get

[Šta želite, to ćete i videti](#)

Metod	Primer	Prednosti	Mane
<b>Ručno</b>	Pisanje koda u <b>Notepad-u</b>	Autor ima veliku kontrolu nad kodom. Moguće je odmah pristupiti greškama, novim HTML elementima i CSS svojstvima.	Metod je <b>spor</b> . Podložan je <b>greškama</b> . <b>Zahteva odlično poznavanje</b> HTML-a i CSS-a. Nema direktnu vizuelnu kontrolu.
<b>Prevođenje U HTML</b>	Snimanje kao HTML dokument iz <b>drugog programa</b> , npr. Word-a	Brz. Pojednostavljuje objavljivanje postojećih dokumenata.	Obično sadrži dosta grešaka. <b>Zahteva dodatno pročišćavanje</b> , dodavanje hiperveza.
<b>Pisanje koda u specijalnom programu za obradu koda</b>	Program <b>HomeSite</b>	Velika kontrola nad kodom. Brži od ručnog pisanja. Pruža pomoć, upozorava na greške i pomaže u pisanju HTML i CSS koda.	Spor.  Zahteva dobro poznavanje jezika HTML i CSS.
<b>Izrada strana u WYSIWYG programima</b>	Program <b>FrontPage</b>	Autor radi <b>direktno sa grafičkim i tekstualnim objektima</b> . Ne zahteva dobro poznavanje jezika HTML i CSS:	<b>Često daje loš HTML i CSS kod</b> . Fino podešavanje izgleda strane zahteva ručno uređivanje HTML koda. <b>5</b>

\*WYSIWYG - What You See Is What You Get

[Šta želite, to ćete i videti](#)



## Programi za izradu Web strana

Program	Platforma	Korisna adresa	Opis
<b>Dreamweaver</b>	Windows, Macintosh	<a href="http://www.adobe.com/products/dreamweaver/">http://www.adobe.com/products/dreamweaver/</a>	Dobar alat za vizuelnu izradu Web strana sa uskladenim WYSIWYG* i mogućnostima uređivanja koda. Pruža jaku podršku CSS i specifikaciji DHTML.
<b>HomeSite</b>	Windows	<a href="http://www.allaire.com/homesite">http://www.allaire.com/homesite</a>	Ima neverovatne mogućnosti za upravljanje kodom, lošu podršku za vizuelno uređivanje. Sestrinski proizvod <b>ColdFusion Studio</b> pruža bolju podršku za tehnologije dinamičke izrade Web lokacije.
<b>GoLive</b>	Macintosh Windows	<a href="http://www.adobe.com/products/golive">http://www.adobe.com/products/golive</a>	Veoma <b>popularan među korisnicima računara Macintosh</b> . Ima grafički interfejs potpuno prilagođen dizajneru. <b>Generisani kod sadrži greške</b> , nije popularan među autorima koji teže strogom poštovanju standarda.
<b>FrontPage</b>	Windows	<a href="http://www.microsoft.com/frontpage">http://www.microsoft.com/frontpage</a>	Popularan među autorima malih lokacija i dizajnerima kompanijskih prezentacija. Znatno je poboljšan od prve verzije, ali još ima reputaciju programa koji daje loše ili Microsoftske strane. <b>6</b>

## Programiranje i izrada Web strana



- Programiranje za Web može se podeliti u dve osnovne velike grupe: **klijentsko i serversko**.
  - ❖ **Klijentske tehnologije** su one koje se izvršavaju na računaru klijenta, tj. u čitaču, mada postoje i neke klijentske tehnologije, (JavaScript, ActiveX kontrole), koje se izvršavaju na klijentskom računaru, ali izvan čitača.
  - ❖ **Serverske aplikacije** su programi koji se izvršavaju na serveru.
- Svakoj tehnologiji pripada određeno mesto i primena.
- Njihovo kombinovanje pri izradi Internet aplikacija je najbolje rešenje.

7

## Tehnologije za klijentsko i serversko programiranje Web strana



Klijentske tehnologije	Serverske tehnologije
ActiveX kontrole	CGI skriptovi i programi
Java programi	Serverski API programi (ISAPI/NSAPI)
<b><u>Jezici za pisanje skriptova</u></b> <ul style="list-style-type: none"> <li>❖ Java Script</li> <li>❖ VB Script</li> <li>❖ Dynamic HTML</li> </ul>	<b><u>Serverski skriptovi</u></b> <ul style="list-style-type: none"> <li>❖ Active Server Pages - ASP</li> <li>❖ ColdFusion</li> <li>❖ PHP</li> </ul>
Pomoćni programi	Serverski Java programi
Dodaci	

8

# Klijentsko programiranje



## Klijentsko programiranje



- **Programi koji se izvršavaju na klijentskoj strani mnogo su brži od programa na serverskoj strani**
  - ❖ Nije potrebno prenositi podatke preko mreže da bi se prikazao rezultat akcije
- **Klijentsko programiranje – problemi**
  - ❖ **Nedostatak kontrole –**

teško je odrediti kakvi će korisnici pristupati lokaciji:

    - koji će čitač korisnici koristiti
    - koja će svojstva čitača biti uključena
    - koju vrstu procesora korisnik upotrebljava
    - postoje korisnici koji isključuju podršku za skriptove ili koriste programe koji se tek testiraju

## Klijentsko programiranje



Klijentsko programiranje nije uvek efikasno.

### Treba pretpostaviti:

da će sve ispravno raditi,  
ali napraviti i rezervnu varijantu!

11

## Active X



- ❑ *ActiveX* je *Microsoft-ova* tehnologija za izradu malih komponenti (kontrola) u okviru Web strane.
- ❑ To nije programski jezik, već mnoštvo integrisanih komponenti – tkz. objekata napisanih na različitim jezicima u Windows okruženju.
  - ❖ *ActiveX* objekti mogu biti programirani u *Javi*, *Delphi-ju*, *Visual Basic-u*, *C++ -u* itd.
- ❑ *ActiveX* objekti (kontrola) omogućavaju dodavanje neke funkcionalnosti Web stranici za rešavanje raznih zadataka:
  - ❖ prikazivanja poruka,
  - ❖ generisanja zvučnih izlaza za tekst,
  - ❖ prepoznavanja govora i sl.

12

## ActiveX



- ❑ **ActiveX** kontrole mogu biti izvršavane **samo na Windows** operativnim sistemima.
- ❑ **ActiveX** objekti nisu bezbedni, jer moraju da se nalaze na korisnikovom disku da bi ih bilo moguće koristiti - imaju pristup svim resursima sistema.
- ❑ Zlonamerni **ActiveX** objekat može uneti virus i oštetiti ili uništiti podatke na disku korisnika.

13

## ActiveX



- ❑ Ako korisnik poseti Web stranicu koja koristi ActiveX objekat, a nema instaliran taj objekat na svom disku, Web server će pokušati da ga pošalje na korisnikov računar.
  - ❖ Zavisno od korisnikovih bezbednosnih podešavanja, čitač Weba će prikazati okvir za dijalog, koji upozorava korisnika o preuzimanju ActiveX objekta.
- ❑ **Preporuka:** koristiti **samo svoje objekte**, ili one koje su napravili poznati i pouzdani programeri, one koji su preuzeti sa pouzdanih Web lokacija (npr. *Microsoft*).

14

## Java



- ❑ Tehnologija kompanije **Sun Microsystems**
- ❑ Java je razvojni jezik **nezavisan od platforme**, koji omogućava pisanje programa i njegovu primenu na bilo kom računarskom sistemu, čitaču ili operativnom sistemu koji podržava tkz. *Java Virtualnu Mašinu* (JVM)
- ❑ Java jezik je po sintaksi sličan jeziku C++.

15

## Java apleti

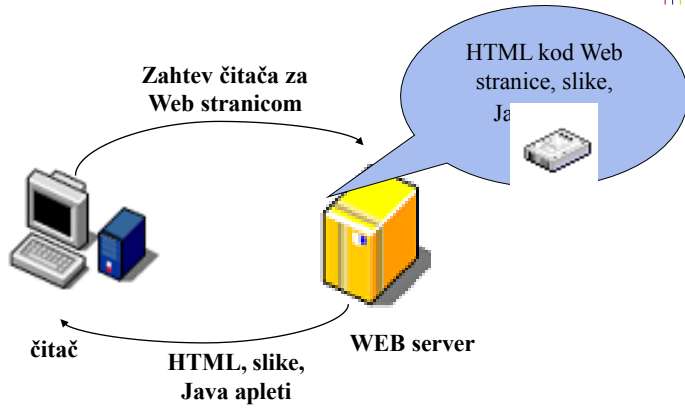


- ❑ Web strane koriste **male programe pisane na jeziku Java**, koji se nazivaju **apleti** i koji se preuzimaju sa Interneta i izvršavaju kod klijenta u samom čitaču.
- ❑ Aplet koji je napisan u jeziku Java
  - ❖ prevodi se najpre u **bajt kod**, koji je nezavisan od računara na kojem se izvršava,
  - ❖ u datoteku tipa **.class** koja se automatski preuzima u bilo koji čitač koji podržava Javu i
  - ❖ pokreće u okruženju korisnika, dakle lokalno.

16



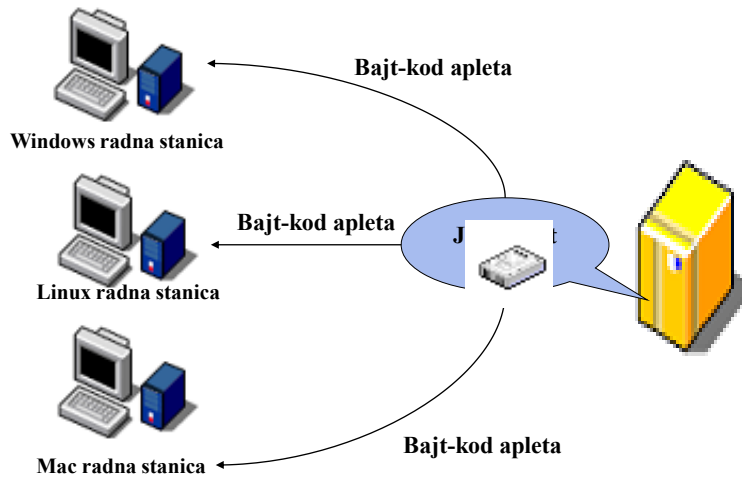
# Čitač preuzima Java aplete sa Web servera



# Java apleti



Java apleti koriste generički bajt-kod koji čitač prevodi u kod specifičan za datu mašinu – isti aplet radi na raznim tipovima mašina



## Java apleti



- ❑ Čak i na računarima sa brzim procesorom može izgledati kao da se Javin bajt kod izvršava sporije nego kod normalno prevedenih programa.
- ❑ JVM (Java Virtual Mashine) pre izvršavanja mora da interpretira bajt kod apleta.
- ❑ Zbog toga vlada mišljenje da je Java spora.
- ❑ Zapravo - njena interpretacija može biti spora.

19

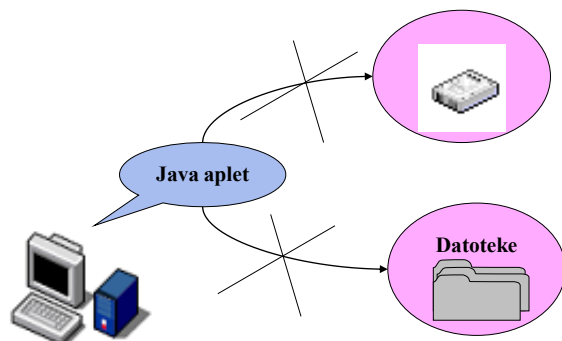
## Bezbednost Java apleta



- ❑ Java apleti nisu samostalni izvršni programi – oni se pokreću samo u okviru Web čitača.
- ❑ **Javin bezbednosni model** štiti korisnike od virusa i drugih opasnosti iz programa preuzetih s Web-a.
- ❑ Radi eliminisanja virusa i krađe podataka Java apleti:
  - ❖ ne mogu da pristupe resursima (posebno datotekama) koji se nalaze na korisnikovom disku,
  - ❖ ne mogu da pristupe svojstvima sistema,
  - ❖ ne učitavaju datoteke,
  - ❖ ne otvaraju mrežne veze ni sa jedim računarom osim sa onim sa kojeg su preuzeti.

20

## Java apleti štite rad diska i datoteka



21

## JavaScript

- ❑ **JavaScript** je skript jezik kojim Web dizajneri mogu da automatizuju postupke unutar Web stranice.
- ❑ **Skript jezik** - omogućava pisanje skripta (tj. grupe naredbi koje će Web čitač izvršavati)
  - ali se **ne koristi za pravljenje programa koji se izvršava nezavisno od čitača.**

22

## JavaScript



- ❑ **JavaScript** se ugrađuje u HTML kod Web stranice
  - ❖ nije potreban poseban alat za pisanje;
  - ❖ može se koristiti isti program za uređenje teksta koji se koristi za izradu Web stranice.
- ❑ Najpopularniji Web čitači imaju ugrađenu podršku za JavaScript – posetioци lokacije ne moraju instalirati dodatne programe na svoje računare.
- ❑ JavaScript iskazi postavljaju se između oznaka `<script>` i `</script>`, koje će ukazati čitaču Weba da izvrši JavaScript iskaze umesto da prikaže tekst na ekranu.

23

## JavaScript



- ❑ Iskazi skripta mogu se uokviriti HTML oznakama za komentar (`<!--...-->`) kako bi se sprečilo da čitači koji ne podržavaju JavaScript na ekranu prikažu iskaze skripta.
- ❑ Čitači koji ne podržavaju JavaScript ignorisaće oznake skripta i sve što se nalazi između početne i završne oznake komentara.
- ❑ Obrnuto, čitači koji podržavaju JavaScript ignorisaće oznake komentara i izvršiti funkciju skripta.

24

## Sakrivanje skripta od čitača koji ne podržavaju JavaScript



```
<script language="JavaScript">
<!--
Document.bgColor="lightyellow";
Document.fgColor="magenta";
//-->
</script>
```

25

## Pomoćni programi



- ❑ su programi koje čitač poziva u pomoć kada naiđe na tip podataka koji ne može da obradi
- ❑ Kada se objekat postavi na Web dodeljuje mu se HTTP zaglavlje sa informacijama o tipu (MIME type)
- ❑ MIME type se određuje na osnovu ekstenzije fajla (npr. .html, .gif, .doc i sl. ).
- ❑ Standardni skup MIME tipova je definisan u fajlu **`/usr/local/etc/httpd/conf/mime.types`**
- ❑ Pod pretpostavkom da se programu može pridružiti MIME tip, svaki program može biti pomoćni program za čitač Weba

26

## Pomoćni programi



- Npr.
  - Svaki dokument načinjen u programu Acrobat trebalo bi da ima pridružen MIME tip **application/pdf**.
- Kada čitač primi datoteku sa određenim MIME tipom, postupiće u skladu sa tim kako je podešen:
  - ❖ Snimanje datoteke na disk
  - ❖ Brisanje datoteke
  - ❖ Prosleđivanje datoteke drugom programu
- Pomoću MIME tipova i pomoćnog programa autor Web lokacije može da postavi npr. Word dokument koji će korisnici moći automatski da preuzmu i pregledaju, ako na računaru imaju instaliran Word

27

## Dodatni programi za čitače



- Dodaci su uvedeni da bi se rešio problem komunikacije i integracije koji je postojao kod pomoćnih programa koji nisu integrisani u čitač Weba već se izvršavaju u zasebnim prozorima i vrlo loše komuniciraju sa čitačem
- Dodaci su komponente koje se izvršavaju u okviru čitača, pa se zato mogu lako ugraditi u Web stranu
- Lako komuniciraju sa čitačem putem tehnologije kakva je JavaScript

28

## Dodatni programi za čitače




- ❑ Proširivanje čitača pomoću dodataka ima i mane –
  - ❖ korisnici moraju pronaći i preuzeti dodatke
  - ❖ instalirati ih i ponovo pokrenuti čitač.
- ❑ Za mnoge korisnike to je previše komplikovano.
- ❑ Da bi se olakšao postupak instaliranja neki popularni dodaci poput dodatka za format Flash kompanije, postali su sastavni deo instalacije samog čitača.
- ❑ Standardni dodaci – najčešće služe za reprodukovanje multimedijjskih sadržaja
- ❑ Primeri: Adobe Flash, Shockwave, Adobe Acrobat, RealVideo...

29

http://www.blic.rs/zanimljivosti.php?id=120509

Inbox (1687) - Yahoo! ... B Blic Online | Najtražnije... METAK - Engleski rečnik... Visoka poslovna škola ... Englesko-srpski rečnik ... Hipotekarna kriza - pon... My Gym Partner

Fotografisanje je obavljeno danas uoči početka razgovora delegacije Srbije koju predvodi predsednik Boris Tadić. Ansa navodi da je Šutanovac tom prilikom dobio mesto na krajnoj levoj strani, daleko od italijanskog premijera.



Microsoft  
**Silverlight**  
DOWNLOAD

Preuzmi poslednju verziju  
Microsoft Silverlight dodatka  
da pogledaš ovaj video

f t

**Akcija - LCD TV 36.499 RSD**  
Šok cena LCD TV Telefunken. Dijagonala ekrana 32inča (80 cm). Vaš CBT.  
[www.belatehnika.rs/](http://www.belatehnika.rs/)  
**Čitajte u novom broju:**

**Team Building Agencije**  
Da li su Vaši zaposleni u istom timu?

**BCGroup MegaStore Online**  
AKCIJA: ASUS K501J-SX036L za samo 32.925 dinara

Postnietni komentari (53) Pošaljite komentar

**ŠTAMPANO IZDANJE - 14. NOVEMBAR 2009. >>**

- Mnogi prinuđeni da rade prekovremeno bez naknade
- Dodik završio vrata zaposlenima u Vladi
- Dujković odlazi ukoliko mladi ne pobeđu Norvešku

**Štampano izdanje**

Najčitanije    Najviše komentara    Najviše poslanih


- 1 Berluskoni premestio Šutanovca zbog visine /video/
- 2 Dnevno krali dva automobila
- 3 Brisel: U Evropu bez viza
- 4 Policajcu sedam godina zatvora za ubistvo
- 5 Odbegli „zemunac“ zna ko je ubio Slavka Čuruviju

**Blic online facebook**

**PRIDRUŽI SE BlicOVOJ GRUPI NA FEJSBUKU**

Blic on Facebook  
Become a Fan

Blic on Facebook



# Dinamičko generisanje HTML-a



## Dinamičko generisanje HTML-a

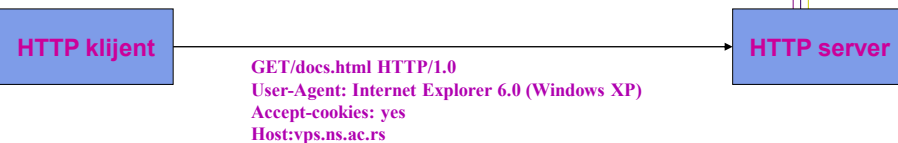


### HTTP protokol

- ❑ Web čitači su namenjeni za prikazivanje Web stranica koje im isporučuju Web serveri.
- ❑ Struktura i izgled samih stranica se opisuje jezikom HTML.
- ❑ Komunikacija između Web klijenta (tj. čitača) i Web servera odvija se po standardnom HTTP protokolu (*HyperText Transfer Protocol*).



## Slanje zahteva HTTP klijenta



**Prvi red poruke je najvažniji** - on sadrži:

- ❖ **komandu koju klijent upućuje serveru** (u ovom slučaju to je GET komanda kojom se zahteva određena datoteka sa Web servera),
- ❖ **putanju datoteke** u okviru Web sajta servera (*/docs.html*) i
- ❖ **oznaku verzije** protokola (*HTTP/1.0*).

**Naredni redovi u poruci** - dodatne informacije koje server može da iskoristi za svoje potrebe. U ovom primeru:

- ❖ polje *User-Agent* predstavlja **opis klijentskog softvera** (tip i verzija Web čitača i operativnog sistema),
- ❖ polje *Host* predstavlja **simboličku adresu klijenta**, itd.

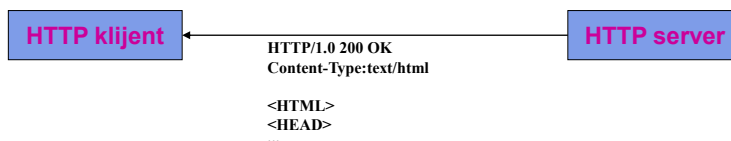
33

## Slanje odgovora HTTP klijentu

- ❑ Zadatak servera je da po prijemu zahteva HTTP klijenta odgovori na njega.
- ❑ U ovom slučaju odgovor servera treba da sadrži traženu datoteku (**docs.html**), pri čemu je format odgovora takođe definisan HTTP protokolom.

34

## Slanje odgovora HTTP klijentu

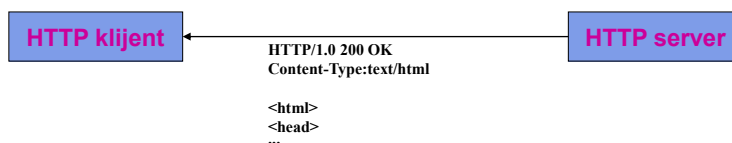


**Prvi red odgovora** sadrži:

- ❖ **oznaku protokola**, pa trocifreni broj koji predstavlja **status izvršene operacije** (200) i tekstualni **opis tog statusa** (OK).
- ❖ Konstanta **200 označava da je zahtev uspešno izvršen** i da se tražena datoteka nalazi u nastavku poruke.
- ❖ Druge konstante koje se češće sreću su:
  - ◆ 404 (tražena datoteka nije pronađena),
  - ◆ 407 (pristup datoteci nije dozvoljen), i
  - ◆ 302 (datoteka premeštena na drugo mesto).

35

## Slanje odgovora HTTP klijentu



**Naredni redovi u poruci** sadrže:

- ❖ **Content-Type - oznaka tipa sadržaja koji se vraća** (standardizovane oznake propisane u odgovarajućim RFC dokumentima (*Request For Comments* - zvanična serija dokumenata koja opisuje tehničke i organizacione detalje Interneta počevši od 1969.)
- ❖ Na primer:
  - ◆ HTML datoteke imaju oznaku **text/html**,
  - ◆ datoteke sa ASCII tekstom bez formatiranja imaju oznaku **text/plain**,
  - ◆ GIF slike **image/gif**, JPEG slike **image/jpeg**, itd.
- ❖ Jedan prazan red koji razdvaja zaglavlje odgovora od samog sadržaja datoteke.

36

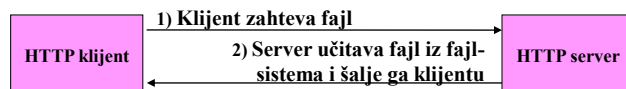
## Statički i dinamički Web sadržaji

- ❑ Jedino što klijent može da zatraži od servera je Web sadržaj - **datoteku**.
- ❑ Server ima zadatak da tu datoteku pronađe (eventualno i modifikuje!) i pošalje klijentu.
- ❑ Web sadržaji koji se smestaju na server vidljivi su klijentima kao pojedine datoteke koje:
  - ❖ mogu biti unapred pripremljene (npr. u editoru kao što je *Dreamweaver*) i smeštene u fajl-sistem Web servera.
  - ❖ ili mogu biti generisane “u letu” - po prijemu zahteva klijenta na neki poseban način;
- ❑ U tom smislu, Web sadržaje (datoteke) možemo podeliti na **statičke** i **dinamičke**.
- ❑ **Klijent ne zna da li je datoteka koju je tražio generisana statički ili dinamički.**

37

## Statički Web sadržaji

- ❑ Statički Web sadržaji su datoteke koje su unapred smeštene u odgovarajući direktorijum fajl-sistema Web servera i **spremne su za isporuku klijentima po njihovom zahtevu.**



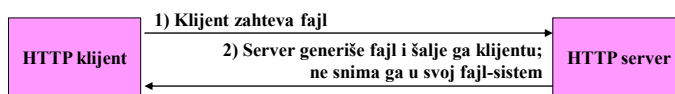
**- Isporuka statičkih sadržaja -**

38

## Dinamički Web sadržaji



- ❑ Dinamički sadržaji nisu uskladišteni unapred, već se generišu za svaki zahtev klijenta posebno.
- ❑ U ovom slučaju server neće tražiti datoteku u okviru fajl-sistema; server “zna” da je u pitanju dinamički generisana datoteka i poziva odgovarajući potprogram koji će je generisati.
- ❑ Najčešće nema potrebe ovako generisanu datoteku čuvati na serveru



- Isporuka dinamičkih sadržaja -

39

## HTTP komunikacija



Ukupna sekvenca aktivnosti klijenta i servera u HTTP komunikaciji

- klijent otvara konekciju sa serverom
- klijent šalje zahtev serveru
- server vraća odgovor
- zatvara se konekcija

40

## HTTP komunikacija



- ❑ Komunikacija između klijenta i servera zasnovana je na **zahtev/odgovor** principu.
- ❑ Svaki par zahtev/odgovor smatra se nezavisnim od ostalih.
- ❑ Recimo, u slučaju da prvi klijent pošalje zahtev serveru i dobije odgovor, zatim drugi klijent pošalje zahtev i dobije odgovor, pa potom ponovo prvi klijent pošalje novi zahtev, **nema načina da se ustanovi da je prvi klijent poslao dva zahteva** (prvi i treći).
- ❑ **Server svaki zahtev opslužuje nezavisno od ostalih zahteva.**

HTTP je **stateless** protokol: ne omogućava praćenje stanja korisničke sesije između slanja više različitih zahteva.

41

## Praćenje sesije korisnika



- ❑ HTTP protokol ne omogućava praćenje sesije korisnika.
- ❑ Kako bi se ovaj cilj ipak postigao, definisano je pomoćno rešenje - mehanizam slanja "**kolačića**" (*cookies*) između klijenta i servera
- ❑ Uveo ga je *Netscape Navigator*, a kasnije je postao standardan mehanizam za ovu namenu podržan od svih Web čitača i servera.
- ❑ **Šta je jedan cookie zapravo?**
- ❑ Možemo ga shvatiti kao string od tipično 20-30 nerazumljivih znakova koji je namenjen **za jednoznačno identifikovanje korisnika na serveru.**

42

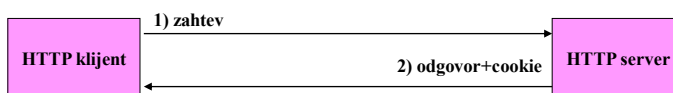
## Praćenje sesije korisnika



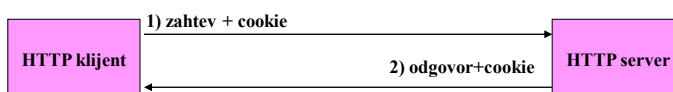
- ❑ Prilikom slanja **prvog zahteva** server će ustanoviti da mu klijent nije poslao *cookie* kao jednu stavku u zaglavlju zahteva.
- ❑ U odgovor na taj zahtev server će dodati *cookie*.
- ❑ Ukoliko je Web čitač podešen tako da radi sa *cookie-jima*, on će u svim sledećim zahtevima koje bude slao tom serveru uključiti i *cookie*, tako da će server moći da *prepozna klijenta* koga je već ranije opsluživao.

43

## Praćenje sesije korisnika



*a) slanje prvog zahteva i prijem odgovora koji uključuje cookie*



*b) svi sledeći zahtevi sadrže cookie*

44

## Svrha kolačića



- ❑ **Glavna svrha** kolačića jeste **da sačuva informacije**:
  - ❖ Oni se uglavnom koriste **za čuvanje korisničkog imena i lozinke** – korisnik ne mora da ih unosi svaki put kada pristupi lokaciji
  - ❖ Mogu se koristiti **za čuvanje svih podešavanja** koja je korisnik zadao kada je poslednji put pristupao lokaciji
- ❑ **Sa programerske tačke gledišta**:
  - ❖ Kolačići se koriste **da reše probleme vezane za upravljanje stanjem** – problem stanja je povezan sa činjenicom da se između dve posete lokaciji server ne seća korisnika

45

## Upotreba kolačića



- ❑ Nekim korisnicima se ne dopada kada neko prati njihovo ponašanje na Webu, pa će oni isključiti kolačiće
- ❑ Da bi se to izbeglo i steklo poverenje korisnika, autori Web lokacije bi trebalo da sačine:
  - ❖ poseban **dokument o privatnosti** ili
  - ❖ **izjavu o upotrebi** i tako obaveste korisnika za šta će kolačići biti upotrebljeni
  - ❖ **samo jedan kolačić** koji će istovremeno i da prati korisnika i da čuva informacije o podešavanjima koja je on zadao – jer više kolačića zahteva od korisnika niz potvrda, svaki mora biti prihvaćen
  - ❖ **ovo je alternativno rešenje** za korisnike koji ne žele da prihvate kolačiće

46

# Serversko programiranje



## Serversko programiranje



- ❑ Serversko programiranje ima mnogo oblika:
  - ❖ **CGI** programi,
  - ❖ serverski **API** programi:
    - ◆ **NSAPI** (*Netscape Server Application Programming Interface*) i
    - ◆ **ISAPI** (*Internet Server Application Programming Interface*) i
  - ❖ **Java** programi,
  - ❖ serverska okruženja za skriptove:
    - ◆ **ASP** (kompanije Microsoft), **ColdFusion** (kompanije Allaire), **PHP**
- ❑ Svaka tehnologija ima svoje prednosti i mane – zajedničko svojstvo im je: **kontrola**
- ❑ **Server je jedini deo klijent-server arhitekture nad kojim autor lokacije ima kontrolu**



## Serversko programiranje



- ❑ Serversko programiranje ne primećuje razlike koje mogu da postoje na klijentskoj strani
- ❑ Lokacija čija je interaktivnost bazirana na serveru, teoretski može da isporuči sadržaj bilo kom čitaču
- ❑ Razvoj interaktivnih elemenata, uprkos prednostima klijentske tehnologije, za veliki broj lokacija sa velikim protokom podataka oslanja se uglavnom na serverske tehnologije
- ❑ Glavna mana serverske tehnologije je: **brzina**
- ❑ Zbog kompletne interakcije koja se odvija na serveru može doći do kašnjenja zbog:
  - ❖ opterećenosti servera ili
  - ❖ zagušenosti mreže za prenos podataka

49

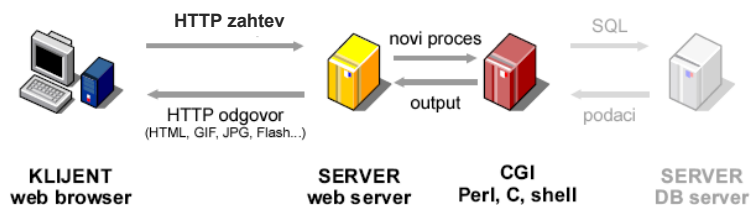
## CGI programi



- ❑ Najstarija tehnologija za serversko programiranje: **Common Gateway Interface (CGI) programi**
- ❑ CGI omogućava Web serverima da pokreću programe kao odgovor na HTTP zahteve
- ❑ CGI specificira kako proslediti argumente programu koji će se izvršiti kao deo HTTP zahteva
- ❑ CGI omogućava da se vrati rezultat u HTML ili nekom drugom formatu, u zavisnosti od parametara konkretnog zahteva

50

## CGI programi



Svaki HTTP zahtev od strane korisnika generiše novi proces na serveru.

51

## CGI programi

- ❑ CGI programi mogu da se pišu na svim jezicima koji mogu da se koriste na serveru koji izvršava program
- ❑ Najčešće se CGI programi pišu na jeziku **Perl**, zbog njegove izuzetne mogućnosti za rad sa znakovnim nizovima i zbog podrške za Web
- ❑ CGI programi mogu se pisati u bilo kom od sledećih jezika:
  - ❖ **Perl, C/C++, Java, Visual Basic**
  - ❖ **(čak i Pascal, Cobol ili Fortran)**

CGI programe treba pisati na jeziku koji je brz, prenosiv i koji odgovara serveru koji će ga izvršavati i programeru koji će ga pisati i održavati.

52

## CGI programi

- ❑ CGI program mora da kreira posebni proces za svaki zahtev posetioca, koji nestaje tek kada se u potpunosti završi transfer podataka.
- ❑ Kreiranje nove instance programa zahteva dodatno vreme:
  - ❖ OS treba da učita program,
  - ❖ alokira memorijski prostor,
  - ❖ a nakon izvršenja da dealocira i ukloni program iz memorije.
- ❑ U serverskim programima veoma **su važni**:
  - ❖ **bezbednost**
  - ❖ **performanse** – pošto server opslužuje veliki broj istovremenih zahteva

**CGI programi se lakše mogu nadgledati i održavati ukoliko se postavljaju u jedan direktorijum**

53

## Serverski API

- ❑ Serverski API programi koriste se za važne funkcije koje treba brzo da se izvršavaju
- ❑ Omogućavaju dobru integraciju sa Web serverom
- ❑ Lako pristupaju glavnim serverskim funkcijama kao što su:
  - ❖ provera identiteta, kontrola pristupa i pristup bazi podataka ili pozadinskim uslugama
- ❑ Glavna **mana serverskih API** programa jeste njihova **složenost i visoka cena**
- ❑ **Loše napisan API može da blokira ceo Web server, dok loš CGI program može samo da zauzme previše procesorskog vremena**

54

## Java Servleti

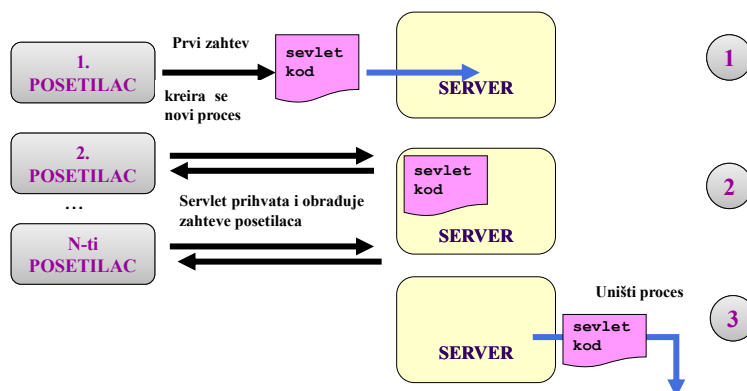
### □ Servleti –

jedna od **tehnologija za generisanje dinamičkih Web sadržaja**.

- Da bi se servleti mogli koristiti, Web server mora da ima odgovarajuću podršku za servlete.
- Pisanje servleta je **moгуće samo u programskom jeziku Java**, tako da je za njihovo izvršavanje potrebna i JVM (koju najčešće obezbeđuje Web server).
- **Servlet je Java klasa koja nasleđuje standardnu klasu *HttpServlet***.
- U pisanju servleta koriste se sledeći Java paketi klasa:
  - ❖ *javax.servlet*
  - ❖ *javax.servlet.http*

55

## Životni ciklus servleta

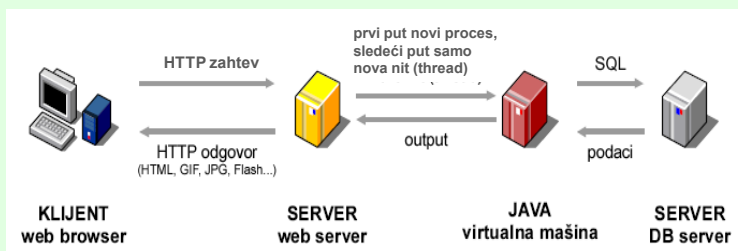


- Servlet se kreira i inicijalizira
- Servlet prihvata i obrađuje pozive od klijenta
- Servlet se uništi

56

## Prednosti Java servleta

- **Efikasnost** – prilikom prvog zahteva prema servletu, server mora pokrenuti novi proces unutar Java Virtualne Mašine, ali za svaki sledeći zahtev prema tom servletu, JVM kreira samo malu Java nit (eng. *thread*)
- **Portabilnost** – zbog prirode Java programa - *napiši jednom, pokreni bilo gde* (eng. *write once, run anywhere*), servleti su portabilni na različitim SW/HW platformama



57

## Nedostaci Java servleta

- Java servleti mogu biti **sporiji** od CGI skripta, u situacijama gde servlet poziva veliki broj korisnika
- HTML kao izlaz iz servleta je priličan problem, jer je reč o internom HTML kodu iz servleta – **teško su čitljive i složene izlazne naredbe**, koje sadrže HTML kod i programski kod ostatka aplikacije
- Vezivanje za samo **jedan programski jezik – Java** (što može biti problem samo za programere i ne smatra se nedostakom same tehnologije)

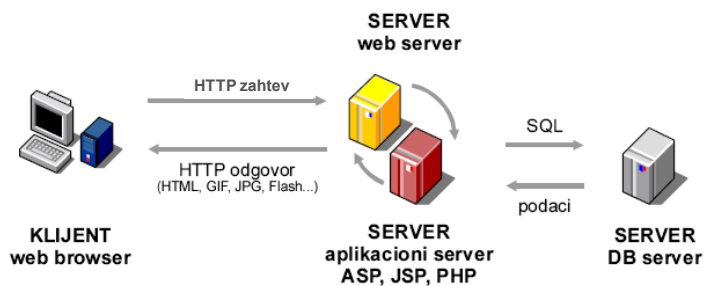
58

## Serverski skriptovi

- “serverski skriptovi” ili “serverski HTML”
- Omogućavaju pridruživanje programiranja Web stranama na relativno jednostavan i brz način
- Cilj serverskih skriptova: izrada posebnih HTML datoteka ili šablona koji sadrže **smešu skripta i HTML elemenata**, koje će server po potrebi učitati
- Postoje mnoga okruženja za serverske skriptove:
  - ❖ PHP –Personal Home Page
  - ❖ ASP - Active Server Pages
  - ❖ ASP.NET
  - ❖ JSP – Java Server Pages
  - ❖ ColdFusion ...
- Razlike između ovih okruženja su veoma male – glavna razlika jeste sintaksa

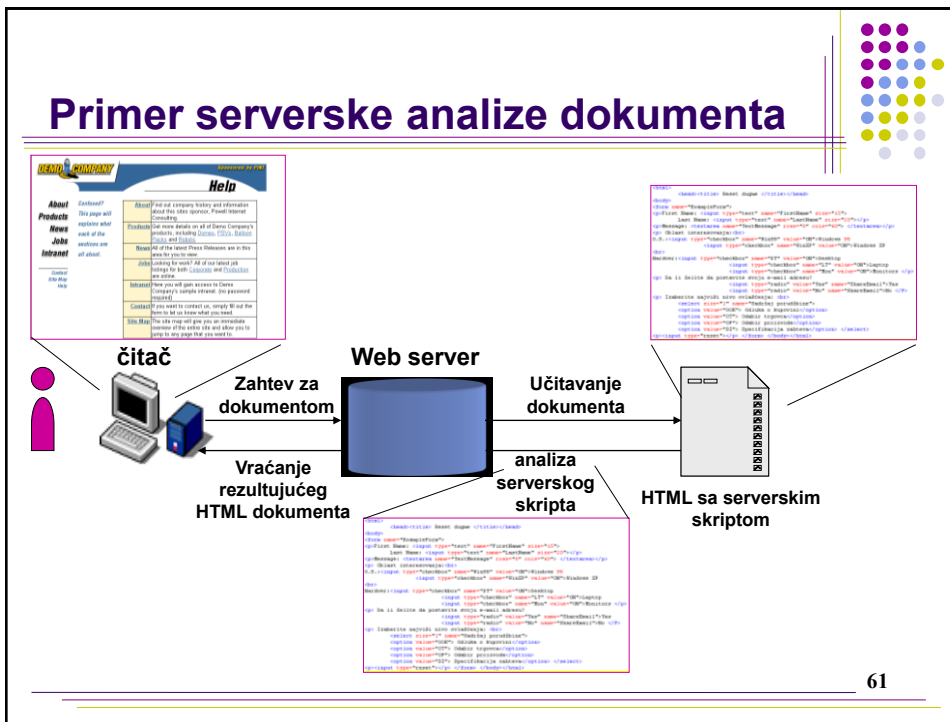
59

## Primer serverske analize HTML dokumenta



60

## Primer serverske analize dokumenta



61

## ASP

- **ASP -Active Server Pages** ("Stranice aktivne na serveru") predstavlja jedan od skriptnih jezika kompanije Microsoft za razvijanje Internet sajtova.
- Njegov glavni konkurent na tržištu Internet alata je **PHP** jezik, koji ima osobinu da je besplatan (free), dok su ASP i svi proizvodi potrebni za njegovo korišćenje komercijalni i u vlasništvu Microsoft-a.
- Microsoft konstantno održava i dodaje nove mogućnosti svom ASP paketu, pružajući svojim klijentima svu moguću tehničku podršku.

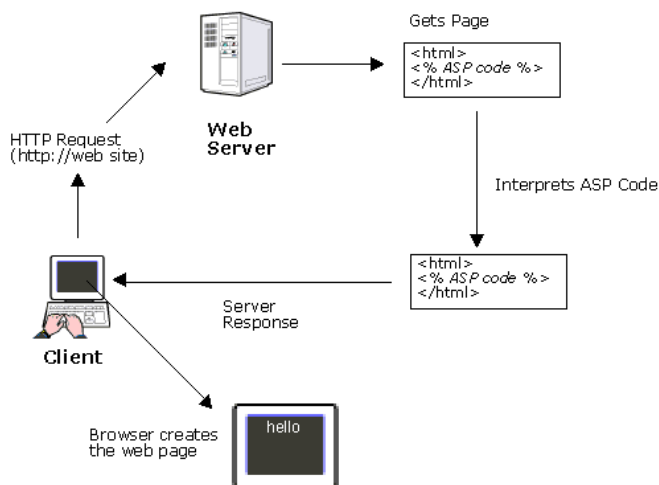
62

## ASP

- ❑ Nakon uvođenja .NET tehnologije, Microsoft je i ASP prilagodio istom, tako da je u upotrebi verzija ASP.NET 2.0, objavljena 2005. godine.
- ❑ ASP je **skriptni jezik**, kao i PHP, zahteva za svoj rad interpreter na serverskoj strani.
- ❑ ASP nije programski jezik, već tehnologija koja omogućava programiranje Web servera, dok programski jezik zavisi od izbora autora:
  - ❖ **VBScript** – najčešće korišćen
  - ❖ **JScript** – Microsoftova verzija JavaScript jezika
  - ❖ **C#** (samo za ASP.NET)
  - ❖ **kao i više od 19 jezika** razvijenih od nezavisnih firmi (samo za ASP.NET)

63

## ASP



64



# Tehnologije distribuiranih objekata



## Osnovni koncepti



### Koncept distribuiranih objekata

- ❑ Neki objekat sa svojim metodama i atributima može da “postoji” na nekom računaru a da drugi programi, odnosno objekti, koji se izvršavaju na drugim računarima mogu da mu pristupaju.
- ❑ Pristup tom objektu znači – mogućnost pozivanja njegovih metoda i pristupanja njegovim atributima.
- ❑ Pristup tom “serverskom” objektu bi, sa strane ostalih učesnika (“klijenata”) trebalo da bude što jednostavniji.
- ❑ U idealnom slučaju pristup njemu ne bi trebalo da se razlikuje od pristupa lokalnim objektima – tj. ostalim objektima koji se nalaze u sastavu programa.

## Osnovni koncepti

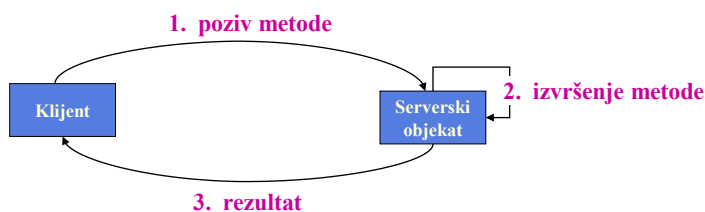


### Koncept distribuiranih objekata

- ❑ Poziv metode serverskog objekta podrazumava izvršavanje te metode na onom računaru na kome se nalazi sam objekat.
- ❑ To znači da se klijentski program izvršava na više računara – inicijalno na onom na kome je pokrenut, ali i na svim računarima na kojima se nalaze serverski objekti koje on koristi.

67

## Poziv metode serverskog objekta

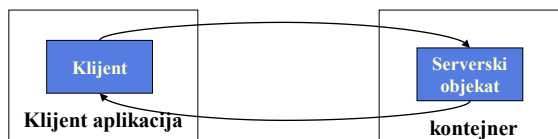


### Komunikacija dva objekta:

- ❖ klijentski objekat je deo klijentske aplikacije, a serverski objekat je deo serverske aplikacije.
- ❖ Serverska aplikacija se često naziva “kontejner” za objekte jer je njena osnovna funkcija da obezbedi mrežne i druge servise koji su neophodni za ovakav serverski objekat.

68

## Klijentski i serverski objekat u okviru svojih programa



- ❑ Posmatrano sa stanovišta autora klijentske aplikacije, i klijentski i serverski objekat su sastavni deo **jednog programa**.
- ❑ Osobina tog programa je da se izvršava na više računara u mreži.
  - ❖ Serverski program, logički posmatrano, i ne postoji;
  - ❖ Postoji samo serverski objekat koji se ponaša/koristi kao da je deo klijentskog (jedinog) programa.

69

## Tehnologije distribuiranih objekata



Najpoznatiji SW alati (tehnologije) za rad sa distribuiranim objektima su:

- ❖ **COM/DCOM** - *(Distributed) Component Object Model*
- ❖ **RMI** - *Remote Method Invocation*
- ❖ **CORBA** - *Common Object Request Broker Architecture*
- ❖ **EJB** - *Enterprise JavaBeans*

70