



UVOD U XML



POVIJESNI RAZVOJ XML-A

- XML – eXtensible Markup Language je jezik za opisivanje podataka koji omogućuje kreiranje gotovo neograničenog broja elemenata tako da ima vrlo širok spektar primjene
- XML omogućuje strogo odvajanje strukture sadržaja od njegove prezentacije
- Razvio se iz SGML-a, standard XML 1.0 od 1998. malo se mijenjao, a služi kao temelj za mnoge druge standarde sa specijaliziranom primjenom (XHTML – jezik za opisivanje i prikaz podataka na mreži, CML – jezik za opisivanje kemijskih jednadžbi, DocBook – jezik za opisivanje tehničke dokumentacije, Graph – jezik za razmjenu grafova, GPX – jezik za razmjenu GPS koordinata, MathML – jezik za opisivanje matematičkih notacija, MML – jezik za opisivanje glazbenih notacija...)
- *Vježba: Pronaći primjere za navedene standarde*

KREIRANJE XML DOKUMENTA I SINTAKSNA PRAVILA

- XML dokument je tekstualni dokument sa ekstenzijom .xml
- Smještaj XML koda u tekstualni dokument omogućava prijenos datoteka neovisno o operacijskom sustavu i uređaju koji će datoteku prikazati
- XML element sastoji se od početne i završne oznake elementa te sadržaja među oznakama. Početna i završna oznaka sadržavaju ime elementa pisano malim slovima omeđeno sa <>. Završna oznaka ima / ispred imena.
- Prazni elementi sastoje se samo od početne oznake i / čime se prepoznaje samozatvarajući element.
- Svaki otvoreni element mora biti zatvoren.
- Gniježđenje elemenata mora biti pravilno izvedeno, odnosno on mora biti zatvoren unutar roditeljskog.
- Početne oznake elemenata XML-a mogu sadržavati i attribute koji pobliže opisuju element.

ELEMENTI XML-A, ATRIBUTI, KOMENTARI

- Atributi XML dokumenta nalaze se unutar početne oznake elementa i pišu se malim slovima. Nakon naziva atributa slijedi znak jednakosti te vrijednost atributa unutar dvostrukih navodnika. Element može biti i bez atributa ili može sadržavati jedan ili više atributa. Ako atribut sadrži više od jedne vrijednosti, one se odvajaju praznim mjestom.
- Komentari su posebna vrsta oznaka unutar XML dokumenta koji se ne procesiraju, a služe za postavljanje oznaka unutar XML koda koje dodatno opisuju kod. Počinju oznakom `<!--`, a završavaju sa `-->`

STRUKTURA XML DOKUMENTA – DEKLARACIJA, PROLOG

- Na vrhu, u prvoj liniji svakog XML dokumenta mora stajati deklaracija – instrukcija koja govori u kojem je standardu i u kojoj verziji standarda pisan ostatak dokumenta: `<?xml version='1.0' encoding='UTF-8' ?>` (preko atributa navedena je verzija dokumenta i naziv kodne stranice)
- Dio dokumenta u kojem se nalazi deklaracija dokumenta naziva se prolog koji osim deklaracije može sadržavati i definiciju tipa dokumenta: `<!DOCTYPE math PUBLIC „-//W3C//DTD MathML 2.0//EN” „http://www.w3.org/Math/DTD/mathml2/mathml2.dtd”>` te razne procesne instrukcije
- Definicija tipa dokumenta sadrži pravila po kojima dokument mora biti strukturiran kako bi bio valjan i prošao provjeru točnosti strukture
- Ključna riječ PUBLIC označava da je riječ o javnoj definiciji, SYSTEM označava privatnu definiciju tipa dokumenta (određuje se dostupnost)
- Javnoj definiciji slijedi javni identifikator tipa dokumenta
- Druga stavka označava adresu datoteke koja sadržava sva pravila (standarda) za strukturiranje datoteke
- Svaki se dokument mora zbog provjere moći povezati sa odgovarajućom dtd datotekom (potrebna internetska veza)
- Za izvođenje koda XML dokumenta nije presudna provjera valjanosti, već ispravno strukturiranje i pridržavanje sintaksnih pravila XML-a, no poželjno je da dokumenti prolaze validaciju zbog važnosti pridržavanja pravila standarda

STRUKTURA XML DOKUMENTA – KORIJENSKI ELEMENT

- Nakon prologa slijedi korijenski element XML dokumenta (kod XHTML dokumenta korijenski je html element)
- Sadržava sve ostale elemente XML dokumenta
- Primjer:

```
<?xml version='1.0' encoding='UTF-8' ?>
```

```
<ucenici>
```

```
<ime>Marko</ime>
```

```
</ucenici>
```

OPIS STRUKTURE XML DOKUMENTA

- Struktura XML dokumenta često se opisuje pomoću modela obiteljskog stabla

```
<?xml version='1.0' encoding='UTF-8' ?>
```

```
<razred>
```

```
<razrednik> Ivan Horvat </razrednik>
```

```
<ucenici>
```

```
<ucenik spol='M'>Marko Novak</ucenik>
```

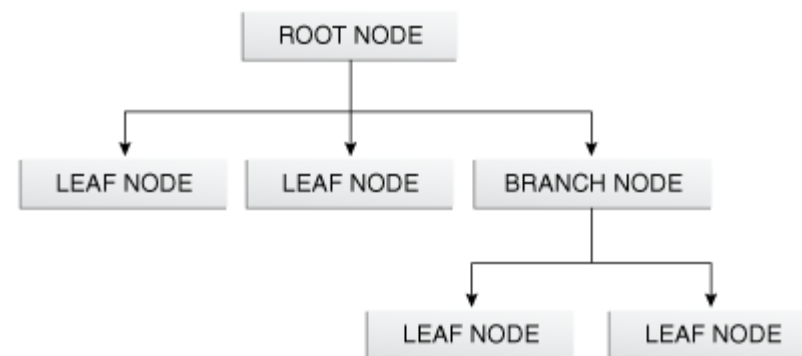
```
<ucenik spol='Ž'>Marija Herman</ucenik>
```

```
<ucenik spol='Ž'>Ana Novosel</ucenik>
```

```
</ucenici>
```

```
</razred>
```

- Sastoji se od deklaracije u prologu te korijenskog elementa razred unutar kojeg su elementi razrednik i ucenici, a u elementu ucenici nalaze se elementi ucenik



ELEMENTI POTOMCI

- Elementi dokumenta podijeljeni su po razinama pa je iz grafičkog prikaza strukture XML dokumenta lako odrediti odnose među elementima (pretke i potomke)
- Elementi učenik potomci su elementa učenici i korijenskog elementa razred.
- Element je potomak nekog drugog elementa ako je sadržan između početne i završne oznake tog elementa.
- Svi elementi unutar korijenskog elementa njegovi su potomci.

ODNOS RODITELJ-DIJETE (PARENT-CHILD) BRATSKO-SESTRINSKI ODNOSI (SIBLINGS)

- Za precizniji opis među elementima.
- Element može imati više od jednog djeteta, ali ima samo jednog roditelja.
- Elementi su u bratsko-sestrinskom odnosu ako imaju istog roditelja.

OPIS STRUKTURE XML DOKUMENTA POMOĆU MODELA STABLA

- Primjer:
- Korijenski element razred predak je svim elementima XML dokumenta, odnosno svi elementi unutar korijenskog elementa potomci su korijenskog elementa
- Element ucenici roditelj je elementima ucenik, odnosno elementi ucenik djeca su elementu ucenici
- Element razrednik i element ucenici su u bratsko sestrinskom odnosu
- Korijenski element razred je predak elementima ucenik

ČITANJE (DOHVAT) IZ XML DATOTEKE I OBRADA XML PODATAKA

```
<?xml version='1.0' encoding='UTF-8' ?>
```

```
<gradovi>
```

```
  <grad> Split </grad>
```

```
  <grad> Zadar </grad>
```

```
  <grad> Osijek </grad>
```

```
  <grad> Zagreb </grad>
```

```
  <grad> Rijeka </grad>
```

```
</gradovi>
```

FUNKCIJE ZA ČITANJE I PRIKAZ SADRŽAJA

- `function startTag($parser, $data) {echo „”;} //za izmjenu početne oznake u HTML oznaku`
- `function contents($parser, $data) {echo $data;} //za dohvaćanje podataka koji se nalaze između XML oznaka`
- `function endTag($parser, $data) {echo „
”;} //zamjenjuje završnu oznaku HTML oznakama`
- Ove funkcije pozvat ćemo pomoću posebnih PHP XML ugrađenih funkcija. Prvo je potrebno stvoriti objekt parser: `$xml_parser=xml_parser_create();`, te zatim definirati koje su funkcije za rad sa početnim i završnim oznakama, a koje za rad s podacima između njih:
- `xml_set_element_handler($xml_parser, „startTag”, „endTag”);`
- `xml_set_character_data_handler($xml_parser, „contents”);`
- Čitanje sadržaja XML datoteke u varijablu `$data`: `$file='gradovi.xml'; $fp=fopen($file, „r”); $data=fread($fp, filesize($file));`
- Glavni dio obrade XML podataka: `xml_parse($xml_parser, $data, feof($fp));`//pregledniku će biti poslan HTML kod za prikaz (`Split
.....
`)
- `xml_parse` vraća 1 ako je uspješno kreiran HTML dokument, 0 u slučaju pogreške pa je moguće napisati:
`if(!(xml_parse($xml_parser, $data, feof($fp)))){ die („Greska u liniji: „.xml_get_current_line_number($xml_parser));}`
- Skripta se završava sa: `xml_parser_free($xml_parser); fclose($fp);`//za oslobađanje memorije kojom se koristi poslužitelj kako bi stvorio XML parser i za zatvaranje konekcije prema XML datoteci