

3. HTML, CSS, XML, (DTD), (XPath), XSLT

```
<?xml version="1.0" e
<quiz>
  <question>
    Who was the forty-second
    president of the U.S.A.?
  </question>
  <answer>
    William Jefferson Clinton
  </answer>
  <!-- Note: We need to add
  more questions later.-->
</quiz>
```

XML

BODÓ ZALÁN
Matematika és Informatika Kar,
BBTE, Kolozsvár

1. HTML

HyperText Markup Language

- hypertext = hiperszöveg (hipertext)
 - **def.:** *nemlineáris, digitális közegben hálózatosan épülő szövegrendszer*
- hyperlink = hiperhivatkozás (hiperlink)
- 1945: *Vannevar Bush* (mérnök): Memex
- 1965: első szóhasználat: *Theodor H. Nelson* (szociológus, filozófus)

A HTML története

- ❑ **1960:** GML (Generalized Markup Language) - *Charles Goldfarb, Edward Mosher és Raymond Lorie* (IBM)
- ❑ **1986:** SGML – ISO szabvány (a következők W3C ajánlások)
- ❑ 1989 – **1993:** HTML – Tim Berners-Lee (CERN)

A HTML dokumentum felépítése

```
<html>
```

```
<head>
```

```
...
```

```
<title>Cím</title>
```

```
...
```

```
</head>
```

```
<body>
```

A dokumentumtörzs:

```
<h1>Alcím</h1>
```

```
<p>Egy bekezdés</p>
```

```
</body>
```

```
</html>
```

A fejléc

- Az egész dokumentumra vonatkozó beállítások:
pl. cím, stílusfájl, karakterkódolás (encoding)
stb.
- `<head> ... </head>`
- Pl.:
`<title>Honlap</title>`
`<meta http-equiv="Content-Type"`
`content="text/html; charset=iso-8859-2"/>`
`<link href="styles.css" rel="stylesheet"`
`type="text/css">`

A szövegtest/dokumentumtörzs

- ❑ `<body> ... </body>`
- ❑ Attribútumok:
 - background - háttér
 - bgcolor - háttérszín
 - text – szövegszín
 - link – link színe
 - alink – aktív link színe
 - vlink – már bejárt link színe

Címek és alcímek

- `<h1> ... </h1>`
- `<h2> ... </h2>`
- ...
- `<h6> ... </h6>`

Bekezdések

- `<p> ... </p>`
- attribútum:
 - `align: {left, center, right, justify}`
- sortörés: `
`
- sortörés tiltása: `<nobr> ... </nobr>`

Hivatkozások

□ Linkek:

` ... `

□ Könyvjelzők:

■ Könyvjelző:

` ... `

■ Hivatkozás:

` ... `

□ Pl.:

`Ugrás az oldal aljára`

`
`

A `Google` kereső.

`
`

`Írj nekem egy levelet!`

`Ez itt a dokumentum vége.`

Fontosabb karakterformázások

- ❑ ``félkövér (bold)``
- ❑ `<i>`dőlt (kurzív, italic)`</i>`
- ❑ `<u>`aláhúzott (underline)`</u>`
- ❑ `<s>`áthúzott (strikethrough)`</s>`
- ❑ `<tt>`írógép, fixpontos`</tt>`
- ❑ ``kiemelés (emphasize)``
- ❑ ``félkövér kiemelés``
- ❑ `_{`alsóindex`}`
- ❑ `^{`felsőindex`}`
- ❑ `<blink>`villogó szöveg`</blink>`

-
- Betűtípus:

 ...

Színkódok HTML-ben:

- a. Hexadecimális számokkal: #abcdef,
 ahol $a,b,c,d,e,f \in \{0,1,\dots,F\}$
- b. Az rgb jelöléssel: rgb(a,b,c), ahol $a,b,c \in \{0,1,\dots,255\}$
- c. 16 db előredefiniált szín: *aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white, yellow*

Képek

□ ``

Pl.: ``

□ map:

rect: bal-x, felső-y, jobb-x, alsó-y.
circle: közép-x, közép-y, sugár.
poly: x1, y1, x2, y2, ..., xN, yN

```
<map name="map1">  
  <area href="hiv1" alt="sz1"  
    shape="alak" coords="x1,y1,x2,y2,..." />
```

...

```
</map>
```

GIMP
www.gimp.org

□ Pl.:

```

```

```
<map name="map1">
```

```
  <area href="http://www.google.com" alt="Google"
  shape="rect" coords="90,70,530,200"/>
```

```
  <area href="http://www.microsoft.com" alt="Microsoft"
  shape="circle" coords="320,300,60"/>
```

```
  <area href="http://www.yahoo.com" alt="Microsoft"
  shape="circle" coords="410,325,30"/>
```

```
</map>
```

<http://www.gauson.com/files/2009/08/google-msn-yahoo-fish-cartoon.jpg>

Listák

- számozott lista:

```
<ol seqnum="szam" [continue]>
```

```
...
```

```
</ol>
```

- számozatlan lista:

```
<ul src="kep.jpg" dingbat="karakter">
```

```
...
```

```
</ul>
```

- leíró lista:

```
<dl>
```

```
...
```

```
</dl>
```


-
- listafejléc: `<lh> ... </lh>` (számozott és számozatlan)
 - listák sorai:
 - ` ... `
 - `<dt> ... </dt>` és `<dd> ... </dd>`

Táblázatok

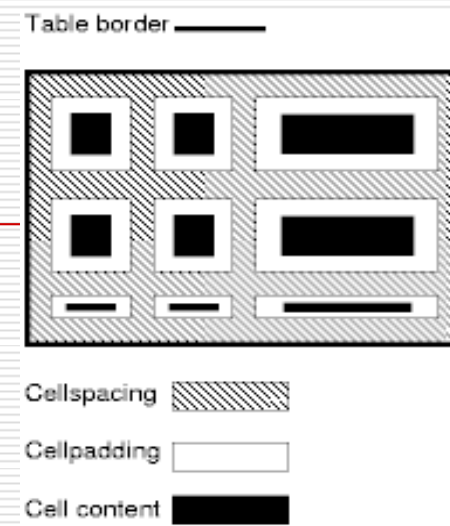
```
<table ...>  
  <caption> ... </caption>  
  <tr>  
    <th> ... </th> ...  
  </tr>  
  <tr>  
    <td> ... </td>  
  </tr>  
</table>
```

□ a table attribútumai:

- border: pixelekben
- align: {left, center, right}
- nowrap
- cellpadding: a cellák körüli keret mérete
- cellspacing: a cellák közötti távolság és a táblázat keretének mérete
- bgcolor

□ a caption attribútumai:

- align: {top, bottom, left, right}



□ a td attribútumai:

- colspan: szomszédos cellák egyesítése vízszintesen
- rowspan: szomszédos cellák egyesítése függőlegesen
- align: {left, center, right, justify, char}
- valign: {top, middle, bottom, baseline}

□ Pl.:

```
<table align="center" border="1" cellspacing="0" cellpadding="5">
  <caption align="bottom">Néhány LOST szereplő</caption>
  <tr>
    <th>#</th>
    <th>Név</th>
  </tr>
  <tr>
    <td>1.</td>
    <td>Benjamin Linus</td>
  </tr>
  <tr>
    <td>2.</td>
    <td>James Ford</td>
  </tr>
  <tr>
    <td>3.</td>
    <td>Kate Austen</td>
  </tr>
</table>
```

Keretek

pixel: pl. **200,***
százalék: pl. **10%,***

- frameset:
`<frameset rows="oszlophatarok">...`
`</frameset>`
`<frameset cols="sorhatarok">...`
`</frameset>`
- frame:
`<frame`
 `name="nev"`
 `src="objektum"`
 `scrolling="ertek"`
 `marginwidth="szam"`
 `marginheight="szam"`
`/>`

-
- ❑ `<base target="keretnev"/>`
 - be lehet állítani, hogy egy adott html kimenetét melyik keretbe akarjuk irányítani (a head részben)
 - ❑ **fontos:** a frameset meg kell, hogy előzze a body taget!
 - ❑ Pl.:
- ```
<html>
<frameset cols="10%,*">
 <frame src="menu.html" name="menu" scrolling="no"/>
 <frameset rows="10%,*">
 <frame src="felso.html" name="felso" scrolling="no"/>
 <frame src="jobb.html" name="jobboldal"/>
 </frameset>
</frameset>
<body>
</body>
</html>
```

# Űrlapok (formok)

---

```
<form
 method="mod"
 action="cimzett">
```



```
...
</form>
```

## □ Űrlapmezők:

```
<input name="név" type="típus" align="hely">
<textarea name="név" rows="magaság" cols="szélesség" value="szöveg"> ... </textarea>
<select name="név" size="sor"> ... </select>
```



---

## □ mezők típusai: (input)

- TEXT
- PASSWORD
- HIDDEN
- RADIO
- RANGE
- FILE
- SUBMIT
- RESET
- BUTTON

- 
- ❑ (value attribútum: alapértelmezett érték stb.)
  - ❑ TEXT típus: size="meret", maxlength="ertek"
  - ❑ CHECKBOX, RADIO típusok: CHECKED (RADIO – ua. legyen a neve a kül. rádiógomboknak)
  - ❑ RANGE típus: min="minertek", max="maxertek"
  - ❑ SUBMIT, RESET: value="szoveg"

megjelenítendő sorok  
száma

- ❑ textarea: hosszabb szövegek beviteléhez
- ❑ kiválasztásos menü (select):

```
<select name="nev" [size="sor"] [multiple]>
```

```
 <option>1. opcio</option>
```

```
 <option selected>2. opcio</option>
```

```
 ...
```

```
</select>
```

alapértelmezett érték

több érték  
kiválasztása  
egyszerre

---

□ Pl.:

```
<form method="get" action="#">
<table>
 <tr>
 <td align="right">Név:</td>
 <td><input name="nev" type="text"/></td>
 </tr>
 <tr>
 <td align="right">Nem:</td>
 <td><input name="nem" type="radio" checked/>Férfi</td>
 </tr>
 <tr>
 <td align="right"></td>
 <td><input name="nem" type="radio"/>Nő</td>
 </tr>
</table>
</form>
```

---

# Speciális karakterek, HTML entitások

---

- ❑ független a karakter kódolástól
- ❑ escape szekvencia és ISO-kód
- ❑ néhány példa:

á &#225;      &aacute;

Á &#193;      &Aacute;

" &#34;      &quot;

@ &#64;      @

® &#174;      &reg;

| &#166;      &brvbar;

...

<http://www.ascii-code.com/>



## 2. CSS

# Cascading Style Sheets

---

- ❑ **1998** – W3C (HTML 4.0-hoz)
- ❑ stílusok létrehozása és beillesztése weblapokba
- ❑ egységes stílus a HTML dokumentumoknak
- ❑ rövidebb, átláthatóbb HTML oldalak
- ❑ tartalom és stílus szétválasztása

---

□ a HTML head elemében:

■ belső:

```
<style type="text/css">
```

```
...
```

```
</style>
```

■ külső:

```
<link href="fnev.css" rel="stylesheet"
 type="text/css"/>
```

□ a tagen belül:

```
<elemnev style="..."> ... </elemnev>
```

```
Pl.: <p style="font-size=14pt"> ... </p>
```

---

## □ stílus prioritásai:

- inline módon megadott
- belső stíluslap
- külső stíluslap
- böngésző alapértelmezett stílusbeállításai



---

## □ szintaxis:

szelektor {tulajdonsag: ertek}

Pl.:

p {font-size: 14pt}

p {color: red; font-size: 10pt}

body {bgcolor: #336699}

...

- 
- ❑ szelektorok csoportosítása:  
elem1, elem2, ... elemn {  
    font-size: 14pt  
}
  - ❑ class-ok szerinti stílusok:  
elem.egyik {      font-size: 14pt }  
elem.masik { font-size: 20pt }
  - ❑ class-ok szerinti stílusok rögzített elem nélkül:  
.egyik {    font-size: 14pt }
  - ❑ adott attribútummal rendelkező elem stílusa:  
elem [attr=ertek] {...}
  - ❑ ID szerinti stílus:  
#id {...}

---

## □ pseudo-szelektorok:

### ■ link:

a:link {color: red}

### ■ visited:

a:visited {color: blue}

### ■ hover:

a:hover {color: green}

### ■ active:

a:active {color: yellow}

---

□ betűtípusok:

font-family, font-style, font-size, font-weight,...

□ színek:

color, background-color

háttérkép: background-image: url('valami.com/kep.jpg')

ismétlés: {background-image: url('...');  
                    background-repeat: repeat-x  
                    }

...

- 
- szövegek:  
word-spacing, letter-spacing,  
text-decoration: none | [ underline || overline || line-through || blink ]  
...
  - dobozok:  
margin-top, margin-right, margin-bottom, margin-left  
VAGY margin  
padding  
border-width  
border-color  
...

---

## □ display:

- block – sortörés az elem előtt és után
- inline – nincs sortörés
- list-item – ua. mint a block + listaelem jelölő
- none – nincs megjelenítés

## □ hosszértékek:

- em (font magassága), ex (x betű magassága), px, in (1 in = 2.54 cm), cm, mm, pt (1 pt = 1/72 in), pc (1 pc = 12 pt)

## 3. XML

# eXtensible Markup Language

---

```
<?xml version="1.0" encoding="iso-8859-1" standalone="yes"?>
<!DOCTYPE menu SYSTEM "menu.dtd">
<menu>
 <etel>
 <nev>tukortojas</nev>
 <ar>5,30</ar>
 ...
 </etel>
 <etel>
 <nev>rantotta</nev>
 <ar>7,20</ar>
 ...
 </etel>
</menu>
```

- 
- ❑ version – XML verziószáma
  - ❑ encoding – szöveg kódolása
  - ❑ standalone – a dokumentum “megállja helyét egyedül is”, csak *validálásra* szolgál
  - ❑ DOCTYPE – a dokumentum DTD-je
  - ❑ menu – legfelső szint
  - ❑ SYSTEM – hol találja meg a rendszeren
  - ❑ “[elérési útvonal]”
  - ❑ <menu> -> </menu> – az XML törzse



# Az XML története

---

- **1960:** GML (Generalized Markup Language) - *Charles Goldfarb, Edward Mosher és Raymond Lorie (IBM)*
- **1986:** SGML – ISO szabvány (a következők W3C ajánlások)
- 1989 – **1993:** HTML – Tim Berners-Lee (CERN)
- 1997 – **1998:** XML – Tim Bray (Textuality and Netscape) és mások
- **2000:** XHTML

# Mi az XML?

---

- ❑ egy bővíthető (X) jelölőnyelv (ML)
- ❑ W3C ajánlása
- ❑ metanyelv, nyelvleíró nyelv
- 1. Bővíthető – mert saját elemeket lehet értelmezni
- 2. Jelölő – a leírandó dokumentum különböző részeit különbözőképpen jelöli
- 3. Nyelv – rögzíthető a szókincs és a szabályok

# Különbségek: SGML, HTML, XML, XHTML

---

- ❑ SGML és XML – ugyanazon a szinten helyezkednek el
- ❑ az XML az SGML egy egyszerűsített változata
- ❑ *az SGML és XML leíró nyelvek*
- ❑ a HTML és XHTML konkrét nyelvek, alkalmazások
- ❑ *a HTML SGML alkalmazás*
- ❑ *az XHTML XML alkalmazás*

# Néhány XML-re alapuló nyelv

---

- ❑ **XHTML** – HTML XML-en alapuló változata
- ❑ **TEI-XML** – Text Encoding Initiative, szövegek leírása
- ❑ **Docbook** – dokumentációk, könyvek leírása
- ❑ **WML** – “WAP-os telefonok HTML-je” (Wireless Markup Language)
- ❑ **SVG** – Scalable Vector Graphics – vektorgrafikus képformátum
- ❑ **SOAP** – Simple Object Access Protocol – webes alkalmazások közötti kommunikációs protokoll

# XML felépítése

---

- ❑ egyetlen gyökérelem (tag)
- ❑ minden nyitótaghez tartozik zárótag  
Pl.: `<valami ...> ... </valami>`
- ❑ egyszerűsített formában, ha nincs szükség az egymásbaágyazottságra:  
`<valami ... />`
- ❑ a tagek szigorúan egymásba ágyazódnak  
Pl.: `<a><b></a></b>`  
**HELYTELEN!**

- 
- ❑ különbség kis- és nagybetű között (case sensitive)
  - ❑ érvényes név:
    - betűvel vagy "\_" -sal kezdődik
    - ezután tetszőleges számú betűt, számot vagy pontot, aláhúzást, kötőjelet tartalmaz
  - ❑ a tagek rendelkezhetnek attribútumokkal; ezeket ' ' vagy " " jelek határolják  
Pl.: <auto nev="Audi" szín="piros" />

# DTD

---

- Dokumentum külső file-ban megadott DTD-vel:

```
<?xml version="1.0"?>
```

```
<!DOCTYPE greeting SYSTEM "hello.dtd">
```

```
<greeting>Hello, world!</greeting>
```

- Dokumentum helyileg definiált DTD-vel:

```
<?xml version="1.0"?>
```

```
<!DOCTYPE greeting [
```

```
<!ELEMENT greeting (#PCDATA)>
```

```
<greeting>Hello, world!</greeting>
```

# Megjegyzések

---

□ a HTML-hez hasonlóan:

<!-- A megjegyzes,  
amely akar többsoros  
is lehet

-->



# Feldolgozási utasítások

---

<?cel\_alkalmazas parameterek?>

□ szabványos feldolgozó utasítások:

■ CSS:

<?xml-stylesheet type="text/css"  
href="stilus.css"?>

□ egyéni feldolgozó szoftverek:

<?sajat\_alk param1="ertek1" ?>

# XML névterek

---

- ❑ hasonló pl. a C++-os névterekhez (namespace)

- ❑ névkonfliktusok elkerülése

- ❑ szintakszis:

`xmlns:h="http://www.w3.org/TR/htm"`

- ❑ lehet az első használatkor, vagy az XML gyökérelemében deklarálni

---

□ Pl.:

```
<h:table
xmlns:h="http://www.w3.org/TR/html4/">
 <h:tr>
 <h:td>Nev</h:td>
 <h:td>Jegy</h:td>
 </h:tr>
</h:table>
```

## 4. DTD

# Document Type Definition

---

- ☐ adatok formai leírása
- ☐ adatok *validálása*
- ☐ az adatok *nyelvének reguláris grammatikával való leírása*
- ☐ XML 1.0

---

## ❑ **belső**

```
<?xml version="1.0"?>
<!DOCTYPE note [<!ELEMENT note
 (to,from,heading,body)>
<!ELEMENT to (#PCDATA)>
<!ELEMENT from (#PCDATA)>
<!ELEMENT heading (#PCDATA)>
<!ELEMENT body (#PCDATA)>
]>
<note>
<to>Jack Shepard</to>
<from>John Locke</from>
<heading>Letter</heading>
<body>I wish you had believed me.
</body>
</note>
```

## ❑ **külső**

```
<?xml version="1.0"?>
<!DOCTYPE note SYSTEM "note.dtd">
<note>
<to>Jack Shepard</to>
<from>John Locke</from>
<heading>Letter</heading>
<body>I wish you had believed me.
</body>
</note>
```

# XML dokumentumok felépítése

---

- ❑ **elemek** (elements) – pl. body
- ❑ **tagek** (tags) – pl. <body> ... </body>
- ❑ **attribútumok** (attributes) –  
Pl. 
- ❑ **entitások** (entities) - &lt; (<), &gt; (>),  
&amp; (&), &quot; ("), &apos; (')
- ❑ **PCDATA** – Parsed Character Data
- ❑ **CDATA** – Character Data

# Elemek

---

## □ Szintakszis:

```
<!ELEMENT
 elemnev
 (elemtartalom)
>
```

---

□ Üres elem:

<!ELEMENT img (EMPTY)>

□ Adatokat tartalmazó:

<!ELEMENT note (#PCDATA)>

□ Elem leszármazottakkal/gyerekekkel  
(*children*):

<!ELEMENT note (to,from,heading,body)>



- 
- ❑ Egyetlen gyerek:  
    <!ELEMENT note (message)>
  - ❑ Legalább 1 gyerek:  
    <!ELEMENT note (message+)>
  - ❑ Akármennyi gyerek (0,1,2,...):  
    <!ELEMENT note (message\*)>
  - ❑ Opcionális gyerek (0 vagy 1):  
    <!ELEMENT note (message?)>
  - ❑ Kevert tartalom:  
    <!ELEMENT note  
        (to+,from,header,message\*,#PCDATA)>

# Attribútumok

---

## □ Szintakszis:

```
<!ATTLIST
 elemnev
 attr_nev
 attr_tipus
 alap_ertek
>
```

# Attribútumtípusok

---

- ❑ CDATA - karakterek
- ❑ (tip1|tip2|...) – felsorolás típus
- ❑ ID – egyedi azonosító (ID)
- ❑ IDREF – egy másik elem ID-ja
- ❑ IDREFS – más elemek ID-jai
- ❑ NMTOKEN – egy érvényes XML név
- ❑ NMTOKENS – több érvényes XML név
- ❑ ENTITY – entitás
- ❑ ENTITIES - entitások
- ❑ NOTATION – egy jelölés neve
- ❑ xml: - előredefiniált

# Attribútumok alapértelmezett értékei

---

- ❑ #DEFAULT érték – alapértelmezett érték
- ❑ #REQUIRED - kötelező
- ❑ #IMPLIED – nem kötelező
- ❑ #FIXED érték - rögzített

# Példák

---

- ❑ Alapértelmezett érték:  
    <!ATTLIST auto szín CDATA #DEFAULT "piros">
- ❑ Nem kötelező érték:  
    <!ATTLIST elerhetoseg fax CDATA #IMPLIED>
- ❑ Kötelező érték:  
    <!ATTLIST személy szszam CDATA #REQUIRED>
- ❑ Rögzített érték:  
    <!ATTLIST tamogatas ceg CDATA #FIXED "ABC">
- ❑ Felsorolt értékek:  
    <!ATTLIST fizetes tipus (keszpenz|bkartya) "keszpenz">

# Entitások

---

## □ Szintakszis:

```
<!ENTITY
 ent_nev
 "ent_ertekek"
>
```

---

□ Belső entitások:

<!ENTITY nev "John Locke">

<!ENTITY intezmeny "BBTE">

□ Külső entitások:

<!ENTITY nev SYSTEM

"http://www.valami.com/ent.xml">

<!ENTITY intezmeny SYSTEM

"http://www.valami.com/ent.xml">

# Példa

---

```
<!DOCTYPE TVSCHEDULE [
 <!ELEMENT TVSCHEDULE (CHANNEL+)>
 <!ELEMENT CHANNEL (BANNER, DAY+)>
 <!ELEMENT BANNER (#PCDATA)>
 <!ELEMENT DAY ((DATE, HOLIDAY) | (DATE, PROGRAMSLOT+))+>
 <!ELEMENT HOLIDAY (#PCDATA)>
 <!ELEMENT DATE (#PCDATA)>
 <!ELEMENT PROGRAMSLOT (TIME, TITLE, DESCRIPTION?)>
 <!ELEMENT TIME (#PCDATA)>
 <!ELEMENT TITLE (#PCDATA)>
 <!ELEMENT DESCRIPTION (#PCDATA)>

 <!ATTLIST TVSCHEDULE NAME CDATA #REQUIRED>
 <!ATTLIST CHANNEL CHAN CDATA #REQUIRED>
 <!ATTLIST PROGRAMSLOT VTR CDATA #IMPLIED>
 <!ATTLIST TITLE RATING CDATA #IMPLIED>
 <!ATTLIST TITLE LANGUAGE CDATA #IMPLIED>
>
```



## 5. XPath

# XML Path Language

---

- ❑ **1999** – W3C
- ❑ XML lekérdező nyelv
- ❑ XSLT-hez fejlesztették

- 
- ❑ adott úton található elem lekérdezése:  
/AAA/BBB
  - ❑ relatív út:  
//BBB
  - ❑ a \* (bármely) operátor:  
/AAA/\*  
//BBB/\*/\*
  - ❑ a [] operátor:
    - AAA első BBB eleme: /AAA/BBB[1]
    - AAA utolsó BBB eleme: /AAA/BBB[last()]

---

## □ attribútumok:

### ■ attribútumok kiválasztása:

//@id

### ■ attribútumok alapján történő kiválasztás:

//BBB[@name]

//BBB[@\*]

//BBB[not(@\*)]

//BBB[@id='ertek']

## □ a VAGY művelet:

//CCC | /AAA/BBB

- 
- ❑ `count()` függvény: elemek számlálása

```
//*[count(BBB)=2]
```

```
//*[count(*)=2]
```

- ❑ más fontosabb függvények:

```
//*[name()='BBB']
```

```
//*[starts-with(name(),'B')]
```

```
//*[contains(name(),'C')]
```

```
//*[string-length(name()) = 3]
```

```
//BBB[position() mod 2 = 0]
```

```
//CCC[position() = floor(last() div 2 + 0.5) or position() =
ceiling(last() div 2 + 0.5)]
```

- 
- ❑ child: /child::AAA ua. mint /AAA
  - ❑ descendant:
    - összes leszármazott: /descendant::\*
    - CCC összes DDD nevű leszármazottja:  
//CCC/descendant::DDD
  - ❑ parent: //DDD/parent::\*
  - ❑ ancestor://FFF/ancestor::\*
  - ❑ following-sibling: következő testvér (ugyanazon a szinten)
  - ❑ preceding-sibling: előző testvér (hasonlóan)
  - ❑ following: összes utána következő, minden szinten (nem ilyen egyszerű!)
  - ❑ preceding: összes előtte levő, minden szinten (nem ilyen egyszerű!)

- 
- ❑ descendant-or-self: descendant + az illető elem
  - ❑ ancestor-or-self: ancestor + az illető elem

# 6. XSLT

## eXtensible Stylesheet Language Transformations

---

- ❑ XML dokumentumok -> XML dokumentumok
- ❑ például: XML -> HTML
- ❑ **1999**: XSLT 1.0 (W3C) – James Clark
- ❑ kétféle feldolgozás:
  - kliensoldali: pl. Internet Explorer 6.x-től kezdődően beépített
  - szerveroldali: pl. Javascript, PHP stb.
- ❑ XSLT + XPath

---

□ **Példa:**

```
<xsl:stylesheet version = '1.0'
 xmlns:xsl='http://www.w3.org/1999/XSL/Transform'>

 <xsl:template match="color">
 <TABLE>
 <TR>
 <TD>
 <xsl:attribute name="style">
 <xsl:text>color:</xsl:text>
 <xsl:value-of select="."/>
 </xsl:attribute>
 <xsl:value-of select="."/>
 </TD>
 </TR>
 </TABLE>
 </xsl:template>

</xsl:stylesheet>
```



# Alkotóelemek

---

- ❑ elemek (tagek)
  - pl. apply-templates, template stb.
- ❑ függvények
  - pl. ceiling, id, concat stb.
- ❑ tengelyek (axisok)
  - pl. child, descendant, parent stb.
- ❑ operátorok
  - pl. !=, \*, (), +, - stb.

# Bevezető

---

- ❑ az XSLT maga is egy XML dokumentum:
- ❑ felépítés:

```
<xsl:stylesheet version = '1.0'
 xmlns:xsl='http://www.w3.org/1999/XSL/Transform'>
 ...
</xsl:stylesheet>
```

# Sablonok (template-ek)

---

```
<xsl:stylesheet version = '1.0'
 xmlns:xsl='http://www.w3.org/1999/XSL/
 Transform'>
 <xsl:template match="bold">
 <p>

 <xsl:value-of select="."/>

 </p>
 </xsl:template>

 <xsl:template match="red">
 <p style="color:red">
 <xsl:value-of select="."/>
 </p>
 </xsl:template>

 <xsl:template match="italic">
 <p>
 <i>
 <xsl:value-of select="."/>
 </i>
 </p>
 </xsl:template>
</xsl:stylesheet>
```

- ❑ **<xsl:template ...> ... </...>**
- ❑ attribútumok:
  - match - illesztés (XPath)
  - name – nevesített sablon:  
meg lehet hívni a **call-template** segítségével
  - priority – prioritás  
(nagyobb = korábbi kiértékelés)
  - mode – több változata  
lehet egy adott sablonnak

---

## □ apply-template: sablon alkalmazása

- attr.: select, mode

- pl.:

```
<xsl:apply-templates select="//AAA/BBB" />
```

```
...
```

```
<xsl:template match="BBB">...</...>
```

## □ call-template: nevesített sablon meghívása

- pl.:

```
<xsl:call-template name="t1"/>
```

```
...
```

```
<xsl:template name = "t1"> ... </...>
```

- 
- ❑ **szabály:** ha nincs sablon a "/" útra, akkor az **implicit** sablon hajtódik végre: dolgozd fel az aktuális elem összes gyerekeit/leszármazottját
  - ❑ az alábbi "gyökérsablon" ekvivalens azzal, mintha az nem is lenne ott:

```
<xsl:template match="/">
 <xsl:apply-templates/>
</xsl:template>
```

- ❑ ha azt akarjuk, hogy egy adott elem gyerekeire is végrehajtódjon a sablon, akkor apply-template-tet kell alkalmazni

# Attribútumok

---

- ❑ az elemekhez hasonlóan érjük el őket (@):

Pl.: @name

data/@color

car[@checked] - elem kiválasztása,  
amely tartalmazza a checked  
attribútumot

# Tengelyek

---

- ❑ az XPath tengelyei (axisai):  
ancestor, ancestor-or-self, attribute, child, descendant,  
descendant-or-self, following, following-sibling, namespace,  
parent, preceding, preceding-sibling, self
- ❑ @ - attribute:: rövidítése
- ❑ // - descendant-or-self::
- ❑ .. - parent::
- ❑ . - self::
- ❑ **Példa:**  
...  

```
<xsl:for-each select="child::*">
 <xsl:value-of select="./@id"/>
 <xsl:text> </xsl:text>
</xsl:for-each>
```

  
...

# For, if, sort, choose

---

- `<xsl:for-each ...>...</...>`
  - attribútumai: select
  - gyerekei: sort
- `<xsl:if ...>...</...>`
  - attribútumai: test
- `<xsl:sort ...>...</...>`
  - attribútumai: select, lang, data-type, order, case-order
- `<xsl:choose ...>...</...>`
  - a C-beli switch megfelelője
  - attribútumai: nincsenek
  - gyerekei: when és otherwise (when attribútuma: test)



# Element, attribute, copy-of, value-of

---

- `<xsl:element ...>...</...>`
  - XML elem létrehozása a megfelelő paraméterekkel
  - attribútumai: name, namespace, use-attribute-sets
- `<xsl:attribute ...>...</...>`
  - attribútum létrehozása
  - attribútumai: name, namespace
- `<xsl:copy-of ...>...</...>`
  - adott részfa bemásolása
  - attribútumai: select
- `<xsl:value-of ...>...</...>`
  - szövegelem létrehozása
  - attribútumai: select