

# Adatbázisok 1.

## XML lekérdezőnyelvek

### – 2. rész

XPath

XQuery

# Utak, melyek akárhol kezdődhetnek

- Ha az út a dokumentum pontból indul, akkor a `//X` útkifejezés megtalálja az összes `X` gyerekelemet a beágyazottság bármely szintjén, és visszaadja abban a sorrendben, ahogy az a dokumentumban megjelenik,
- illetve ha ez a útkifejezés folytatódik: `//X/...`, akkor bármely olyan részelemből fog indulni, aminek a tagje `X`.

## Példa: //ár

<teázók>

<teázó név = "JoeTeázója">

<ár melyikTea = "Brisk">2.50</ár>

<ár melyikTea = "Pyramid">3.00</ár>

</teázó> ...

<tea név = "Brisk" árulja = "JoeTeázója

SueTeázója ... "> ...

</teázók>

Ezek az ár elemek és bármely  
másik ár eleme a  
dokumentumnak.

# Jolly-joker: \*

- A csillag (\*) tetszőleges tag nevet helyettesít.
- **Példa:** /\*/\*/ár az összes „dédunoka” (harmadik szinten lévő) ár elemet adja vissza.

## Példa: /teázók/\*

Ez a teázó elem és esetleg más teázó elemek, plusz ez a tea elem és esetleg más tea elemek.

<teázók>

<teázó név = "JoeTeázója">

<ár melyikTea = "Brisk">2.50</ár>

<ár melyikTea = "Pyramid">3.00</ár>

</teázó> ...

<tea név = "Brisk" árulja = "JoeTeázója  
SueTeázója ... "> ...

</teázók>

# Szűrési feltétel (*selection conditions*)

- A tageket feltételek [...] követhetik.
- Ilyenkor csak azok az útkifejezésre illeszkedő utak kerülnek be az eredménybe, melyek a feltételt is kielégítik.

# A tagek atomi értékű „törzsének” kiválasztása

- Szeretnénk tageknek a „törzsét” vagy „belsejét” visszakapni, ha az atomi értékű
- Ehhez a `text()` függvényt kell használni
- A hivatalos XPath specifikációban `[text()]` kifejezést kell írni, a `text()` függvény igazgal tér vissza, ha a tétel atomi érték.

## Példa: /teázók/teázó/ár[text()]

<teázók>

<teázó név = "JoeTeázója">

<ár melyikTea = "Brisk">2.50</ár>

<ár melyikTea = "Pyramid">3.00</ár>

</teázó> ...

<tea név = "Brisk" árulja = "JoeTeázója

SueTeázója ... "> ...

</teázók>

ár elemek belsejéből kiválasztja a számokat, melyeket más teázók ár elemeinek belsejei követhetnek.



## Példa: szűrési feltétel

- /teázók/teázó/ár[. < 2.75]

<teázók>

<teázó név = "JoeTeázója">

<ár melyikTea = "Brisk">2.50</ ár>

<ár melyikTea = "Pyramid">3.00</ár>

Az aktuális  
elem.

Csak ez az ár elem kerül be az  
eredménybe a láthatók közül.

## Példa: szűrés attribútummal

- /teázók/teázó/ár[@melyikTea = "Pyramid"]

<teázók>

<teázó név = "JoeTeázója">

<ár melyikTea = "Brisk">2.50</ ár>

<ár melyikTea = "Pyramid">3.00</ár>

# Tengely (*axis*)

- Általánosan: az útkifejezésekben minden egyes lépésnél *tengelyekkel* (*axes*) adhatjuk meg a következő lépésnél feldolgozandó pontok listáját.
- A default tengely a *child::* – ami az összes gyermeket veszi az aktuális pontoknak.

# Példa: tengelyek

- /teázók/tea valójában a /child::teázók/child::tea rövidítése.
- @ pedig az **attribute::** tengely rövidítése.
  - Így a, /teázók/tea[@név = "Brisk" ] jelentése  
/teázók/tea[attribute::név = "Brisk"]

# További tengelyek

- Néhány további hasznos tengely:
  1. **parent::** = az aktuális pont(ok) szülője (szülei).
  2. **descendant-or-self::** = az aktuális pont(ok) és az összes leszármazott.
    - a // ennek a rövidítése
  3. **ancestor::**, **ancestor-or-self** éít.
  4. **self** (ennek rövidítése: .)

# Gyakorló XPath példák

- Legyen adva az alábbi XML:

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

```
<könyvesbolt>
```

```
<könyv kategória="főzés">
```

```
<cím nyelv="hu">Leonardo lakomái - Az olasz konyha titkos története</cím>
```

```
<szerző>Dave Dewitt</szerző>
```

```
<év>2009</év>
```

```
<ár>3000</ár>
```

```
</könyv>
```

```
<könyv kategória="gyerek">
```

```
<cím nyelv="hu">Harry Potter</cím>
```

```
<szerző>J K. Rowling</szerző>
```

```
<év>1999</év>
```

```
<ár>3300</ár>
```

```
</könyv>
```

...

# Gyakorló

## XPath

### példák

Folytatás

...

```
<könyv kategória="web">
  <cím nyelv="en">XQuery: The XML Query Language</cím>
  <szerző>Michael Brundage</szerző>
  <szerző>Paul Peterson</szerző>
  <év>2004</év>
  <ár>22000</ár>
</könyv>
```

```
<könyv kategória="web">
  <cím nyelv="hu">Az XML-kézikönyv</cím>
  <szerző>Neil Bradley</szerző>
  <év>2000</év>
  <ár>4000</ár>
</könyv>
```

```
<könyv kategória="valami">
  <cím nyelv="hu">Akármilyen</cím>
  <szerző>
    <név>X Y</név>
    <cím>Valahol</cím>
  </szerző>
  <év>2000</év>
  <ár>4000</ár>
</könyv>
```

```
</könyvesbolt>
```

# Gyakorló XPath példák

- Nézzünk egy-két példa kérdést!

/könyvesbolt/könyv/cím

Eredmény:

<cím nyelv="hu">Leonardo lakomái - Az olasz konyha titkos története</cím>

<cím nyelv="hu">Harry Potter</cím>

<cím nyelv="en">XQuery: The XML Query Language</cím>

<cím nyelv="hu">Az XML-kézikönyv</cím>

<cím nyelv="hu">Akármilyen</cím>



# Gyakorló XPath példák

```
/könyvesbolt/könyv/cím/@nyelv
```

Eredmény:

nyelv=hu

nyelv=hu

nyelv=en

nyelv=hu

nyelv=hu

# Gyakorló XPath példák

```
/könyvesbolt/könyv[ár>3500]/szerző
```

Eredmény:

```
<szerző>Michael Brundage</szerző>
```

```
<szerző>Paul Peterson</szerző>
```

```
<szerző>Neil Bradley</szerző>
```

```
<szerző>
```

```
  <név>X Y</név>
```

```
  <cím>Valahol</cím>
```

```
</szerző>
```

# Gyakorló XPath példák

`/könyvesbolt/könyv/ár[.>3500]`

Eredmény:

`<ár>22000</ár>`

`<ár>4000</ár>`

`<ár>4000</ár>`

# Gyakorló XPath példák

```
//cím
```

Eredmény:

```
<cím nyelv="hu">Leonardo lakomái - Az olasz konyha  
titkos története</cím>
```

```
<cím nyelv="hu">Harry Potter</cím>
```

```
<cím nyelv="en">XQuery: The XML Query Language</cím>
```

```
<cím nyelv="hu">Az XML-kézikönyv</cím>
```

```
<cím nyelv="hu">Akármilyen</cím>
```

```
<cím>Valahol</cím>
```

# Gyakorló XPath példák

```
/*/*/cím[@nyelv='hu']
```

Eredmény:

```
<cím nyelv="hu">Leonardo lakomái - Az olasz konyha  
titkos története</cím>
```

```
<cím nyelv="hu">Harry Potter</cím>
```

```
<cím nyelv="hu">Az XML-kézikönyv</cím>
```

```
<cím nyelv="hu">Akármilyen</cím>
```

# Gyakorló XPath példák

```
/könyvesbolt/könyv/cím[.='Az XML-kézikönyv']
```

Eredmény:

```
<cím nyelv="hu">Az XML-kézikönyv</cím>
```

# Gyakorló XPath példák

```
/könyvesbolt/könyv/cím[.='Az XML-kézikönyv']/parent::könyv
```

Eredmény:

```
<könyv kategória="web">  
  <cím nyelv="hu">Az XML-kézikönyv</cím>  
  <szerző>Neil Bradley</szerző>  
  <év>2000</év>  
  <ár>4000</ár>  
</könyv>
```