

INFO-H-509

Exercises 1

XPath

Part 1

WELL-FORMEDNESS

XML Document

- Well-formedness – Elements:
 - Every **start tag** has a corresponding **end tag**.
`<note>...</note>`
 - Empty-elements don't have an end tag
`<note />`
 - Elements **don't overlap**
~~`<note>Don't do this.</note>`~~
~~`<note Or this> either</note>`~~
 - **Single root** element

XML Document

- Well-formedness – Attributes:
 - Every attributes appear **at most once** per element
`<note to="You" to="Him" />`
 - Attributes values are enclosed by **single or double quotes**
`<note from='Me' to="You" />`

XML Document

- Escape sequences

Character	Entity
&	&
<	<
>	>

- <!-- Comments -->

Part 2

UNICODE

Concepts

Encoding (UTF-8)

- Bytes patterns corresponding to code units
- Physical representation of code points.

E3 81 AF

Code points

- Integers in the range $0 - 10FFFF_{16}$
- Identify abstract characters

$306F_{16}$
(Hiragana letter HA)

Glyphs

- Graphical representation of abstract characters

は

Decoding UTF-8

- Byte stream (grouped by code units):

01100101 11001100 10000001

- Code points:

$$1100101_2 = 65_{16}$$

$$01100000001_2 = 301_{16}$$

- Characters:

LATIN SMALL LETTER E +
COMBINING ACUTE ACCENT

- Glyph:

é

Encoding to UTF-8

Code points	Encoding			
00 – 7F	0xxxxxxx			
80 – 7FF	110xxxxx	10xxxxxx		
800 – FFFF	1110xxxx	10xxxxxx	10xxxxxx	
10000 – 10FFFF	11110xxx	10xxxxxx	10xxxxxx	10xxxxxx

Part 3

XPATH

XPath

- Path expressions
 - Over a document, given a context
 - Composed of steps

.../axis::node-test[predicate1] [predicate2]/...

Example

```
<department head="jsmith" name="Sales">  
  <people>  
    <person id="jsmith" fullname="John smith />  
    <person id="jdoe" fullname="John Doe" />  
  </people>  
</department>
```

- The department name

/department/attribute::name

Example

```
<department head="jsmith" name="Sales">
  <people>
    <person id="jsmith" fullname="John smith />
    <person id="jdoe" fullname="John Doe" />
  </people>
</department>
```

- The department head's full name
/department/people/
person[@id=../../@head]/@fullname

Example

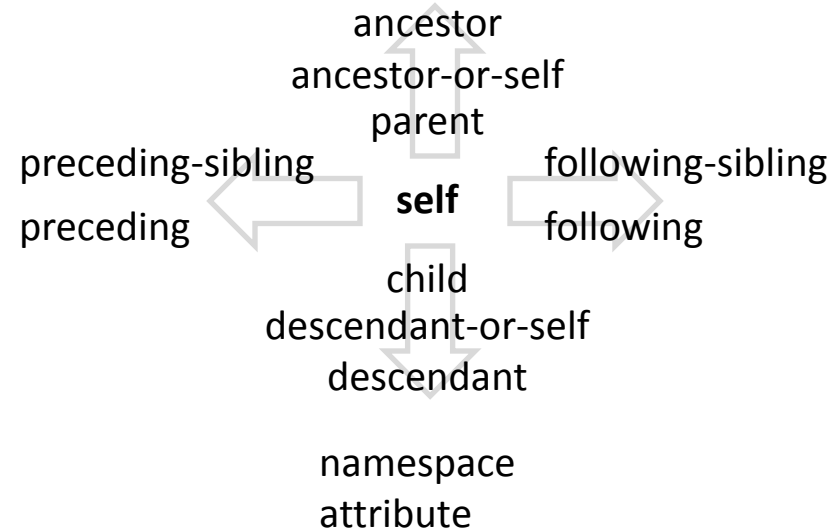
```
<department head="jsmith" name="Sales">
  <people>
    <person id="jsmith" fullname="John smith />
    <person id="jdoe" fullname="John Doe" />
  </people>
</department>
```

- The department name and its head:

```
/department/(@name,  
people/person[@id=../../@head])
```

Quick References

Axis



Node tests

node()
 text()
 comment()
 name prefix:name *:name prefix:*
 processing-instruction()

Namespace prefixes bound in documents can be different from those bound in the XPath engine

Abbreviations

child::	(nothing)
attribute::	@
/descendant-or-self::node()/	//
self::node()	.
parent::node()	..
child::node()	*
[position() = X]	[X]

Operators

/	root, step
union	sequence union
intersect	sequence intersection
except	sequence difference
= < > != <= =>	general comparison
eq lt gt ne le ge	values comparison
is << >>	node comparison
and or	logical
+ - * div mod idiv	arithmetic

Quick References

Statements

if (test-expression)
 then expression
 else expression
for \$var **in** expression
 return expression
some \$var **in** expression
 satisfies expression
Every \$var **in** expression
 satisfies expression

Selected functions

Node-set Functions

last()
position()
count(node-set)
local-name([node-set])
namespace-uri([node-set])
name([node-set])
node-name(node)
root(node)

Document Functions

doc(uri)
doc-available(uri)

String Functions

string([object])
concat(a, b, ...)
starts-with(text, pattern)
ends-with(text, pattern)
contains(text, pattern)
substring(text, start, [length])
string-length(text)
replace(text, pattern, replace)

Boolean Functions

boolean(object)
not(value)
true()
false()

Numeric Functions

number([object])
floor(number)
ceiling(number)
abs(number)

Aggregate Functions

sum(node-set)
avg(node-set)
min(node-set)
max(node-set)

DateTime Functions

dateTime(date, time)
xs:dateTime(datetime)
 Format: "2010-02-19T16:10:00"
year-from-dateTime(date)
month-from-dateTime(date)
day-from-dateTime(date)
hours-from-dateTime(date)
minutes-from-dateTime(date)
seconds-from-dateTime(date)
current-dateTime(date)