

1)

DADOZ MIKOLÉTI
W91081
2024.08.26

A procedurális kezeléssel szemben a két legfontosabb szabvány a SAX és DOM API jelöllet. A DOM olvassa és lekezelhető mint az XML dokumentum modelljeinél is. A DOM modellben az XML dokumentumot ~~fa~~ reprezentációval írjuk le. A DOM API a fa manipulálására szolgáló elcsúsztatást tartalmaz.

A DOM API szabvány interface-ét definiál és nem ~~egy~~ osztályokat

~~Fontosabb~~ Fontosabb interfacei:

• Document (dokumentum komponens megkezelője) metódusai

createElement ^(String n) elem objektum

createTextNode ^(String szöveg) szövegkomponens

createComment (String szöveg)

createAttribute (String név)

~~getElement~~ getElementByTagName (String név) - megadott névvel rendelkező komponens önzörggijének a ~~fa~~ból

adattagok:

documentElement - gyökér elem
doctype

• Node adattagok

NodeType, nodeName, nodeValue, parentNode, childNode, siblingNode

metódusok: removeChild(*), appendChild(*), replaceChild(*), insertChild(*)

* Node elem

• Element metódusok:

getAttribute(*), setAttribute(*), removeAttribute(*), getElementByTagName(*)
String ~~name~~ name

• Character Data adattagjai: data, length

metódusai: deleteData (long k, long h), ^{insertData} insertData (long k, long h)
substringData (long k, long h)

23)

Az XML Schema előkészítő feladata hogy formalizálni tudja egy XML dokumentumot. Az XML Schema mint leírás van.
egyedi elem típusok létrehozása
az elem típusokhoz asszociáció megadása
globális elemnév ~~ter~~ használata
névszótár használata
absztrakt típusok kezelése

A Schema XML Schema leírásának .xsd

Egy elem definíciója az Element elemmel történik.

2xs: elemek, jellemzők > határán <1xs elemek

jellemzők pl: name, type, minOccurs, maxOccurs, abstract

Az öncélt típus definícióhoz nem kell többet közele egy "mixed" elemzésű jellemző ami ha "true" akkor a típus vegyes modell, azaz szöveg is szerepelhet előfordulhat egyidejűleg.

Az elem típusoknál a leírásban megadható a típus neve.
Az elemeket már meglévő meglévő típusokból lehet deriválni pl:
meglévő elem típusok megnevezésével való levezetés

A Schema definícióhoz meg lehet adni a létrehozott XML ^{elem} ~~elem~~
nevet is. Az alapértelmezett név a ^{elem} ~~elem~~
gyökérnél "target Namespace" elemjellel ^{elem} ~~elem~~
kezelhető.

Az XPATH arra használható hogy eligyünk / kiválasszunk egy bizonyos XML elemet és attribútumokat

Gizton'ja az útvonal a kifejezésben utat mutat ki az XML dokumentumok sémáira vagy sémák listájának kiválasztására

XPATH gazdag jelnyelv és kényelmesen érthető, numerikus értékek, logikai értékek.

XPATH meg határozza az XML dokumentumok név

5 XML dokumentum rendelkezik Infosettel ha jól definiált, ~~való~~ ~~specifikációs~~ megfelelő. Információelemből áll amely az XML dokumentumok részének absztrakt leírása, melyhez névvel ellátott tulajdonságok tartoznak. Legáltalában 1 elem a gyökérként funkcionáló ~~document~~ document elemet tartalmazza.

Legfontosabb ~~elemek~~ elemek:

Root Document, Element, Attribute, Character, Namespace
Xinculde rendszer célja eszközöket adni a különböző dokumentumok kezelésére

4 JSON: javascript Object Notation

Olyan univerzális adatcsere formátum amelyet minden programozási nyelv támogat.

Egy objektum " {" - el kezdődik és " } " - el zárul, kvót " " - kint és a név - értékek párok " : " - vel foglaltak

Tömb: []

Értékek " " - el közé lehet írni