

①

~~DOM~~ DOM API

Tejlekthető a SAX-nál minél ez már több funkciót és módosítást is tud végezni.

A DOM API a memóriában teljesen felépíti a dokumentum objektumot. Metódusokkal módosíthatja a tartalmat és XML dokumentumot egy fa reprezentálja. Az API az XML fa manipulálására szolgáló elemeket tartalmaz.

~~Az alap objektum, azaz a fa~~

A teljes fa és egy-egy elem is objektumként kezelhető.

ADOM API:

- Platform és nyelv független
- Fa hierarchiát kezel
- Kódszerű az implementációja
- ~~na~~ Nagy memóriagigéző

A fa feldolgozása navigáció jellegű:

- Elemválasztáskor a nem kívánt elemek válasszhatóak ki.
- Módosításkor az aktuális elem környezetét lehet módosítani.
- Vannak nem helyben kötött műveletek. Pl: elem létrehozás és a teljes dokumentumot leíró objektumhoz kötődnek.

Interfészek:

adathozok: az elem jellemzőit írják le.

metódusok: a navigációra és a tartalom kezelésére szolgálnak.

② X PATH

Fájlok kezelési az XML dokumentumot, használják a DOM-hoz.

az XML fájlban elemek / elemjellemzők elérésére, hivatkozásokra használják.
200+ beépített függvényekkel rendelkeznek.

Előreszámítottak az XSLT-t.

Csomópont típusok:

- Gyökér
- Elem
- Elemjellemző
- Szöveg
- Stílus
- Megjegyzés
- Feldolgozási utasítás

A DOM-ban létezik ehhez nincs névbőlum kezelés, nincs CDATA kezelés, nincs DTD kezelés, minden csomópontba tartozik egy ~~név~~ névleges érték.

Elemi minta: előreszámított

Összetett minta: más minták uniójaként áll ~~elő~~ elő.

Előreszámított lépés:

navigációs tengely: megadja a keresés fő irányát

csomópont név: az érintett csomópontok halmazát

néhány lehet csomópont név, vagy típus alapján.

Logikai feltétel: az előző lépésben hivatkozott csomópontnál tartalmi név.

Tengelytípusok:

self, child, descendant, parent, ancestor, following, attribute, namespace

3.

~~XML~~ XML Schema

XML szintaktikát követi, helyesen formált XML dokumentum

DTD-vel ellentétben itt mindig külön fájlnak helyeskedik el a sémaleírás.

Használhatóak benne előre definiált és egyedi típusok.

• Keverített típusokat, köztük ~~az~~ levo specialis használatokat, valamint absztrakt típusokat is tud kezelni

Szigorú hurok, idegenhurok megköthetés

Gondos struktúra elemek

Kénterék használata támogatja

Hasznos leírás, integritási elemeket használ

Elemi típusok:

• Atom, lista, unió

Összetett típusok:

reluancia, opció lista, vegyes elem, üres elem, opció lista

④ JSON

A JSON ~~egy~~ egyszerűsített XML.

Hierarchikus adatmodell

Elemi:

- adatbázis
- kollekció
- dokumentum
- mező: skalar, tömb, dokumentum

JSON: ~~az~~ adathözjáró, strukturált adatok hierarchiái továbbítására

XML: dokumentumhözjáró

Hasonlóságok:

- egyszerű növekmény
- ember által olvasható, írható
- hierarchikus felépítés
- JS-el összeolvasható
- ~~az~~ adatok hűdése Ajax hívásokkal

Különbségek:

- nincs ~~az~~ end tag
- rövidítés
- tömbök
- minirendezési kulcsok
- ~~az~~ gyorsabbon írható, olvasható