

Obiektowy model dokumentu

Bartłomiej Świercz

Katedra Mikroelektroniki i Technik Informatycznych

Łódź, 14 października 2005 roku

DOM – Document Object Model zapewnia:

- Zbiór obiektów reprezentujących dokumenty XML i HTML.
- Model łączenia obiektów.
- Standardowy interfejs umożliwiający dostęp i manipulację obiektami.

Początkowo firmy implementowały własny model dostępu do obiektów HTML w przeglądarkach. Wyjściem było zdefiniowanie standardowego modelu, który został nazwany **W3CDOM**. DOM jako API jest niezależne od platformy i języka programowania.

DOM – Document Object Model zapewnia:

- **Zbiór obiektów** reprezentujących dokumenty XML i HTML.
- Model łączenia obiektów.
- Standardowy interfejs umożliwiający dostęp i manipulację obiektami.

Początkowo firmy implementowały własny model dostępu do obiektów HTML w przeglądarkach. Wyjściem było zdefiniowanie standardowego modelu, który został nazwany **W3CDOM**. DOM jako API jest niezależne od platformy i języka programowania.

DOM – Document Object Model zapewnia:

- **Zbiór obiektów** reprezentujących dokumenty XML i HTML.
- **Model łączenia** obiektów.
- Standardowy interfejs umożliwiający dostęp i manipulację obiektami.

Początkowo firmy implementowały własny model dostępu do obiektów HTML w przeglądarkach. Wyjściem było zdefiniowanie standardowego modelu, który został nazwany **W3CDOM**. DOM jako API jest niezależne od platformy i języka programowania.

DOM – Document Object Model zapewnia:

- **Zbiór obiektów** reprezentujących dokumenty XML i HTML.
- **Model łączenia** obiektów.
- Standardowy **interfejs** umożliwiający dostęp i manipulację obiektami.

Początkowo firmy implementowały własny model dostępu do obiektów HTML w przeglądarkach. Wyjściem było zdefiniowanie standardowego modelu, który został nazwany **W3CDOM**. DOM jako API jest niezależne od platformy i języka programowania.

Istnieje kilka poziomów modelu DOM:

DOM Level 0 (nieoficjalny),

DOM Level 1,

DOM Level 2,

DOM Level 3,

Istnieje kilka poziomów modelu DOM:

DOM Level 0 (nieoficjalny), Model DOM z przeglądarki Netscape 3.0, skopiowany przez Microsoft i zaimplementowany we wszystkich przeglądarkach internetowych, umożliwiając dostęp tylko do pól formularzy. Nie jest standardem W3C.

DOM Level 1,

DOM Level 2,

DOM Level 3,

Istnieje kilka poziomów modelu DOM:

DOM Level 0 (nieoficjalny),

DOM Level 1, Poziom 2 umożliwia dostęp do treści dokumentu poprzez tworzenie, modyfikowanie oraz dołączanie elementów i atrybutów.

DOM Level 2,

DOM Level 3,

Istnieje kilka poziomów modelu DOM:

DOM Level 0 (nieoficjalny),

DOM Level 1,

DOM Level 2, Poziom dodatkowo wpradza możliwość obsługi zdarzeń i przestrzeni nazw.

DOM Level 3,

Istnieje kilka poziomów modelu DOM:

DOM Level 0 (nieoficjalny),

DOM Level 1,

DOM Level 2,

DOM Level 3, Na poziom trzeci składają się elementy:

- DOM Level 3 Core
- DOM Level 3 Load and Save
- DOM Level 3 XPath
- DOM Level 3 Views and Formatting
- DOM Level 3 Requirements
- DOM Level 3 Validation

Poniżej opisane są ogólne wymagania odnośnie DOM stawiane przez W3C:

- Model obiektu jest językowo obojętny.
- Jądro DOM powinno być w stanie przetwarzać dokumenty XML, HTML i CSS.
- Model obiektu może być użyty do wczytywania i zapisywania dokumentu.
- Konwencja nazewnictwa musi być jednakowa na wszystkich poziomach DOM.
- Model obiektu nie powinien narażać aplikacji użytkownika na błędy związane z bezpieczeństwem, walidacją i prywatnością.

Standard W3CDOM określa sposób w jaki błędy mają być przekazywane przez model obiektu:

- Informacje o błędach są przekazywane za pomocą wartości zwracanej.
- Wyjątki są generowane jedynie wtedy, kiedy wystąpi błąd (warunek) nieodwracalny.
- DOM powinien dostarczać opis do błędów.
- DOM można zapytać o stan błędu.

Dla większości języków istnieją biblioteki dostarczające obsługę modelu DOM dla dokumentów typu XML, jednak standard W3C definiuje API interfejsu DOM jedynie dla języków Java i JavaScript. Najbardziej zaawansowaną biblioteką jest Apache Xerces i MS XML.

Na wykładzie przedstawiona zostanie biblioteka DOM (xml.dom) dla języka Python.

Wada: dokument musi być w całości załadowany do pamięci.

Zaleta: swobodny dostęp do elementów dokumentu, możliwość jego modyfikacji i zapisu.