|  |  |
| --- | --- |
| Symbol projektu:  DocxDocument. ModelGen | |
| Nazwa projektu:  **Generator modelu obiektowego dokumentu** | |
| Tytuł dokumentu: | **ModelDocumentation** |
| Nr wersji: | **2.0** |
| Data utworzenia: | **30.05.2023** |
| Data ostatniej aktualizacji: | **11.06.2024** |
| Osoba odpowiedzialna: | **Jarosław Kuchta (JK)** |
| Autorzy: | **Jarosław Kuchta (JK)** |

Historia dokumentu

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Wersja | Opis modyfikacji | Autor | Data otwarcia | Data zamknięcia |
| 1.0 | Pierwsza wersja | JK | 30.05.2023 | 27.10.2023 |
| 2.0 | Wersja z bazą danych | JK | 2.06.2024 |  |

# Wprowadzenie

Projekt ModelDocumentation służy do pobierania dokumentacji modelu obiektowego w standardzie OpenXml i przygotowania jej do dalszego wykorzystania. Są trzy źródła dokumentacji:

* deklaracje schematu OpenXml zapisane w plikach z rozszerzeniem XSD,
* specyfikacja standardu ECMA-376 opublikowana w kilku plikach PDF i przekonwertowana do formatu DOCX,
* pliki źródłowe pakietu bibliotek DocumentFormat.OpenXml.

Model obiektowy z bibliotek DocumentFormat.OpenXml jest słabo zdokumentowany. Większość komentarzy dokumentacyjnych zawartych w kodzie źródłowym jest trywialna. Najbardziej bogata dokumentacja jest w zawarta w standardzie ECMA, zwłaszcza w pliku „Ecma Office Open XML Part 1 - Fundamentals And Markup Language Reference.pdf”. Celem projektu jest powiązanie modelu obiektowego DocumentFormat.OpenXml z dokumentacją pobraną z pliku ECMA.

Projekt składa się z dwóch modułów:

* ModelDocApp – aplikacji konsolowej do parsowania plików i tworzenia bazy danych dokumentacji modelu OpenXml,
* ModelDoc – modułu dostępowego do bazy danych, który jest wykorzystywany przez aplikację ModelDocApp i może być wykorzystywany w innych aplikacjach.

Projekt działa w oparciu o bazę danych MS Access o nazwie Models.accdb położonej w ścieżce „d:\VS\Docs\DocumentModel”. Ten sposób magazynowania danych został wybrany ze względu na dużą łatwość edycji rekordów bazy danych. Do odwzorowania modelu klas encyjnych w model relacyjnej bazy danych jest wykorzystywany framework Microsoft.EntityFramework.Core.

# Model encji

Model encji dokumentacji składa się z następujących klas encyjnych:

* Namespace – przestrzeń nazw,
* SchemaElement – element ze schematu OpenXml,
* SchemaAttribute – atrybut ze schematu OpenXml,
* ComplexType – typ złożony ze schematu OpenXml,
* SimpleType – typ prosty ze schematu OpenXml,
* StandardElement – element ze standardu ECMA,
* StandardAttribute – atrybut ze standardu ECMA,
* Type – typ (klasa, typ wyliczeniowy) z biblioteki DocumentFormat.OpenXml,
* Property – właściwość klasy z biblioteki DocumentFormat.OpenXml.
* EnumValue – wartość typu wyliczeniowego z biblioteki DocumentFormat.OpenXml

Każda encja jest przechowywana w bazie danych w odpowiedniej tabeli, której nazwa jest nazwą encji w liczbie mnogiej (np. encja Namespace jest przechowywana w tabeli Namespaces).

## Przestrzenie nazw

*Przestrzeń nazw* (Namespace) organizuje nazwy typów. W jednej przestrzeni nazw wszystkie typy muszą mieć unikatowe nazwy. Przestrzenie zostały zaimportowane z kodu źródłowego biblioteki DocumentFormat.OpenXML. Każda przestrzeń nazw ma *nazwę złożoną* z kilku *nazw prostych* oddzielonych kropkami. Nazwa prosta jest *identyfikatorem* składającym się z ciągu liter (łacińskich) i cyfr i rozpoczynającym się od litery. Wszystkie przestrzenie nazw mają nazwy złożone rozpoczynające się od „DocumentFormat.OpenXml”.

Oprócz nazwy przestrzeń nazw ma *prefiks*, który jest krótkim (maks. 10 znaków) ciągiem liter i cyfr, pod którym przestrzeń nazw jest deklarowana w pliku XML dyrektywą xmlns. Ten prefiks jest dodawany przed nazwą elementu XML (oddzielony dwukropkiem). Nie wszystkie przestrzenie nazw mają prefiksy.

Przykład

<w:document   
...  
xmlns:w="http://schemas.openxmlformats.org/wordprocessingml/2006/main" xmlns:w14="http://schemas.microsoft.com/office/word/2010/wordml"   
...>

<w:body>

<w:p w14:paraId="1C1F454E" w14:textId="107F22F8" w:rsidR="007D6C6C" w:rsidRPr="00C0491F" w:rsidRDefault="00732D28" w:rsidP="00F043F7">

<w:bookmarkStart w:id="0" w:name="\_GoBack"/>

<w:bookmarkEnd w:id="0"/>

<w:r>

<w:rPr>

<w:noProof/>

<w14:ligatures w14:val="standardContextual"/>

</w:rPr>

Przestrzeni nazw jest aktualnie 131. Najważniejsze z nich (w kontekście dokumentów programu Word) to:

* XmlSchema – deklaruje podstawowe typy danych atrybutów,
* DocumentFormat.OpenXml – deklaruje podstawowe typy danych modelu OpenXml,
* DocumentFormat.OpenXml.Wordprocessing (w) – deklaruje elementy główne programu Word (w wersji 2007),
* DocumentFormat.OpenXml.Office2010.Word (w14) – deklaruje elementy programu Word dodane w wersji 2010,
* DocumentFormat.OpenXml.Math (m) – deklaruje elementy równań matematycznych,
* DocumentFormat.OpenXml.Drawing (a) – deklaruje elementy rysunkowe.

Encja Namespace ma następujące właściwości:

* ID: int – automatycznie nadawany identyfikator wewnętrzny,
* FullName: string(255) – złożona nazwa przestrzeni nazw,
* Prefix: string(10)? – opcjonalny prefiks,
* Description: text? – opcjonalny opis.

## Elementy schematu OpenXml

Dokumenty pakietu Office (a w szczególności dokumenty programu Word) są w swojej istocie pakietami ZIP składającymi się z plików XML.

Schemat standardu OpenXml zapisany w plikach z rozszerzeniem XSD zawiera definicje elementów i atrybutów plików XML. Elementy są definiowane głównie jako typy złożone (o nazwach zaczynających się od „CT\_”), a atrybuty jako typy proste (o nazwach zaczynających się od „ST\_”).

Przykład

  <**xsd:complexType** name="CT\_Color">

    <**xsd:attribute** name="val" type="ST\_HexColor" use="required"/>

    <**xsd:attribute** name="themeColor" type="ST\_ThemeColor" use="optional"/>

    <**xsd:attribute** name="themeTint" type="ST\_UcharHexNumber" use="optional"/>

    <**xsd:attribute** name="themeShade" type="ST\_UcharHexNumber" use="optional"/>

  </**xsd:complexType**>

Schematy OpenXml są rejestrowane w encji Schemes. Jej właściwości są następujące:

* ID: int – automatycznie nadawany identyfikator wewnętrzny,
* FullName: string(255) – złożona nazwa przestrzeni nazw,
* Prefix: string(10)? – opcjonalny prefiks,
* Description: text? – opcjonalny opis.

W kolejnych rozdziałach i podrozdziałach głównej części specyfikacji opisano elementy OpenXml. Dla każdego z nich podano jaki znacznik XML reprezentuje ten element, jakie inne elementy ten element może zawierać i jakie może mieć atrybuty XML. Atrybuty mają określone typy wartości. Mogą to być typy podstawowe XML Schema (jak np. xsd:string) lub typy proste OpenXml. Każdy z typów prostych jest opisany w osobnym podrozdziale specyfikacji ECMA.

Encja Element ma następujące właściwości:

* ID: int – automatycznie nadawany identyfikator wewnętrzny,
* SpecNum: varchar(255) – numer podrozdziału ze specyfikacji ECMA,
* Ns: int – identyfikator przestrzeni nazw,
* Tag: varchar(255) – znacznik elementu OpenXml,
* FullName: varchar(255) – złożona nazwa przestrzeni nazw,
* Description: varchar(MAX)? – opcjonalny opis elementu.
* Attributes: Collection<Attribute> – kolekcja atrybutów OpenXml,
* ChildElement: Collection<ChildElement> – kolekcja relacji wiążących element OpenXml z podrzędnymi elementami OpenXml.

Encja Attribute ma następujące właściwości:

* ID: int – automatycznie nadawany identyfikator wewnętrzny,
* ElementID: int – identyfikator encji Element, do której należy
* Ns: int – identyfikator przestrzeni nazw,
* Tag: varchar(255) – znacznik atrybutu OpenXml,
* FullName: varchar(255) – złożona nazwa przestrzeni nazw,
* Description: varchar(MAX)? – opcjonalny opis elementu.
* SimpleTypeID: int – identyfikator typu prostego wartości atrybutu.

Encja SimpleType ma następujące właściwości:

* ID: int – automatycznie nadawany identyfikator wewnętrzny,
* SpecNum: varchar(255)? – opcjonalny numer podrozdziału ze specyfikacji ECMA,
* Ns: int – identyfikator przestrzeni nazw,
* TypeName: varchar(255) – nazwa typu prostego OpenXml,
* FullName: varchar(255) – złożona nazwa przestrzeni nazw,
* Description: varchar(MAX)? – opcjonalny opis elementu.
* Values: Collection<SimpleTypeValue> – kolekcja wartości typu prostego.