

# Projektowanie języków XML 2016/2017

## Laboratorium 6

27 marca 2017

DTD nie jest wystarczająco elastycznym narzędziem w wielu zastosowaniach. Dziś skupimy się na XML Schema, o czym wspomnieliśmy już na poprzednich zajęciach.

### Zadanie wprowadzające

Będziemy potrzebować koncepcji przestrzeni nazw. Dla przypomnienia warto zajrzeć do materiałów o [przestrzeniach nazw w XML](#), a także do rekomendacji [W3C](#).

Rozszerzoną nazwą znacznika jest para (nazwa przestrzeni, nazwa lokalna znacznika). Do tej pory wykorzystywaliśmy tylko nazwę lokalną znacznika lub po prostu znacznik. Używanie całej nazwy przestrzeni może być niewygodne, więc definiuje się zazwyczaj prefiks dla danej przestrzeni nazw.

```
<Krzesło>
  <Oparcie>Proste</Oparcie>
  <Obrotowe />
</Krzesło>
```

Jeżeli wiem, w jakiej przestrzeni nazw są zdefiniowane powyższe znaczniki, możemy użyć pełnych nazw (tj. z prefiksem).

```
<mebel:Krzesło>
  <mebel:Oparcie>Proste</mebel:Oparcie>
  <mebel:Obrotowe />
</mebel:Krzesło>
```

Do indentyfikacji przestrzeni nazw używa się URI (Uniform Resource Identifier), czyli odpowiednio sformatowanej nazwy będącej identyfikatorem zasobu. Przestrzeń nazw deklaruje się z użyciem zastrzeżonego atrybutu `xmlns` (brak prefiksu – jest to tzw. przestrzeń domyślna) lub `xmlns:prefiks` (prefiksem znaczników z tej przestrzeni jest `prefiks`). Deklaracja obowiązuje tylko dla znacznika, w którym się pojawia, i w znacznikach potomnych.

Użycie przestrzeni nazw pozwala na uniknięcie kolizji nazw, np. znacznik może opisywać inną daną w zależności od kontekstu. Przestrzeń nazw można wykorzystać także do określenia struktury dokumentu pod warunkiem, że zdefiniujemy schemę (schemat) dla znaczników z takiej przestrzeni. W tym celu posłużymy się dokumentami XSD (XML Schema Definition). Pliki XSD są poprawnymi dokumentami XML, których znaczniki również należą do określonej przestrzeni nazw.

Przykład plików XSD można znaleźć na [w3schools.com](#). Przydatny będzie także [spis dozwolonych elementów plików XSD](#).

Mając daną schemę w postaci pliku XSD, walidację dokumentu XML względem schemy możemy wykonać w następujący sposób.

```
xmllint plik.xml --schema schema.xsd
```

## Zadanie 1 (1 pkt)

Na stronie laboratorium znajduje się plik `ksiazka.xsd`. Przygotuj na jego podstawie dokument XML, a następnie wykonaj walidację względem schemy.

Co mogą oznaczać znaczniki w pliku XSD?

## Zadanie 2 (1 pkt)

Strukturę plików XML definiuje się w XSD za pomocą pewnych znaczników. Głównym znacznikiem jest `schema`, w nim definiujemy opis struktury. W tym zadaniu poznamy część znaczników XSD: `element` (definicja elementów), `complexType` (definicja elementów, które mogą zawierać elementy i atrybuty) oraz `simpleType` (definicja typu i ograniczeń dla prostych informacji tekstowych, np. w atrybutach).

Typy (`complexType` i `simpleType`) mogą być definiowane w miejscu ich użycia (inline, anonimowe), a także jako typy nazwane.

Zdefiniuj dokument XSD, który opisz strukturę ogłoszenia sprzedaży samochodu. W ogłoszeniu powinny się znaleźć (w sekwencji, znacznik `sequence`) następujące rzeczy:

- marka pojazdu – jako prosty element typu `string`,
- rok produkcji – jako element typu `int` z ograniczeniem dolnym na rok,
- dane o silniku – jako element złożony z dwóch elementów: pojemności (`positiveInteger`) i typu paliwa (`string`).

Całość powinna zamknąć się w znaczniku dedykowanym ogłoszeniu i znaczniku samochodu. Wykorzystaj typy anonimowe (definicje inline).

Przygotuj dokument XML, aby zademonstrować walidację względem utworzonej schemy.

## Zadanie 3 (1 pkt)

Sekwencja (`sequence`) jest tylko jednym ze sposobów na określenie struktury elementów podrzędnych. Kolejną opcją jest `choice`, który wymaga, aby tylko jeden z podanych elementów wystąpił w plikach XML. Istnieje także opcja `all`, jest ona podobna do `sequence`, ale nie narzuca kolejności wystąpienia elementów.

Utwórz nową schemę dla ogłoszenia z poprzedniego zadania, w której nie będą występować typy anonimowe, zwiększy to czytelność dokumentu. Następnie rozszerz ją o cenę. Cena może zostać podana na jeden z dwóch sposobów: jej wysokość albo „do uzgodnienia”. Przygotuj przykładowe pliki z ogłoszeniami, które pokażą, jak działa nowy schemat.

Pamiętaj o walidacji przygotowanych dokumentów.

## Zadanie 4 (1 pkt)

Dodaj do pliku XSD z poprzedniego zadania nowy element opisujący wyposażenie samochodu. Kolejność elementów wyposażenia w ogłoszeniu nie ma znaczenia i chcemy to uwzględnić w opisie struktury. Zaproponuj kilka elementów (pustych), np. elektryczne szyby, autopilot, napęd 4x4, i dopisz je do pliku XSD. Przygotuj także nowe ogłoszenie prezentujące brak wymogu zachowania kolejności elementów wyposażenia w dokumentach XML.

Pamiętaj o walidacji przygotowanego dokumentu.

## Zadanie 5 (1 pkt)

XML Schema pozwala na narzucanie pewnych ograniczeń dla typów prostych. Na jednym z poprzednich laboratoriów pracowaliśmy z plikiem `rezerwacja.xml`. Teraz użyjemy fragmentu tego pliku (`segment.xml`) do przygotowania schematu. Dodatkowo wykorzystamy kolejny budulec struktury, jakim jest `attribute`.

Schema dla segmentu powinna opisywać następujące rzeczy:

- wszystkie elementy i atrybuty z przykładowego pliku z segmentem (odpowiednio zagnieżdżone),
- typ segmentu ograniczony do zdefiniowanej listy typów (`enumeration`),
- numer rezerwacji pasujący wyrażenia regularnego, które dopuszcza tylko duże litery alfabetu łacińskiego i cyfry, o długości od 6 do 18 (`pattern`),
- data w formacie YYYY-MM-DD (`date`),
- czas w formacie hh:mm:ss (`time`),
- miejsce (niepuste) o długości co najwyżej 50 znaków (`minLength`, `maxLength`),
- nazwa przewoźnika (niepusta) o długości co najwyżej 30 znaków,
- cena będąca nieujemną liczbą zmiennoprzecinkową z dwoma miejscami dziesiętnymi (`decimal`),
- waluta ograniczona do zdefiniowanej listy wartości.

Przeprowadź walidację dokumentu XML względem schemy. Możesz wprowadzać pewne zmiany w dokumencie, aby sprawdzić, czy dalej się waliduje.