

1. Przetestuj pracę przykładowego stylu book.xml konwertującego dokument book.xml do poprawnego dokumentu HTML. Skorzystaj w tym celu z procesora <http://www.purplegene.com/static/transform.html>
Uwaga! Strona wykazuje błędy, jeżeli pierwszy wiersz we wklejonych dokumentach jest pusty!
2. Dodaj do stylu reguły konwertujące główny tytuł książki, tytuły podrozdziałów (sekcji) i element `<important>`. Tytuł książki powinien być nagłówkiem pierwszego stopnia, tytuł sekcji nagłówkiem trzeciego stopnia a tekst oznaczony przez znacznik `<important>` powinien być wyświetlany czcionką pogrubioną.
3. Przetestuj możliwość dołączenia stylu transformującego do dokumentu czytanego bezpośrednio przez przeglądarkę:

```
<?xml-stylesheet type="text/xsl" href="book.xml"?>
```

4. Zmień formatowanie akapitów tekstowych z atrybutem `type` ustawionym na `quote` tak, aby prezentowane były one wewnątrz znacznika `<blockquote>`. Uwaga! Zdefiniuj dwie reguły: jedną dla akapitów tekstowych z atrybutem `type` równym `quote` i jedną dla akapitów tekstowych, których atrybut `type` jest różny od `quote`.
5. Wprowadź numerację akapitów tekstowych: przed każdym akapitem powinien się pojawić jego numer w sekcji. Wykorzystaj znacznik `<xsl:number/>`.
6. Dodaj hierarchiczną numerację rozdziałów i sekcji, czyli np. wszystkie sekcje w rozdziale 2, mają numery 2.1, 2.2 itd. Wykorzystaj znacznik `<xsl:number/>` z odpowiednimi wartościami atrybutów `count` i `level`.
7. Dodaj do stylu reguły generujące dynamicznie spis treści książki (tytuły rozdziałów i podrozdziałów) na początku pliku wynikowego. Do formatowania spisu wykorzystaj listy numerowane HTML. Przykładowy styl generujący spis treści jest w pliku `toc.xml`. Uwaga! Aby oddzielić przetwarzanie samej „książki” od jej spisu treści, wykorzystaj tzw. tryby dla znaczników `<xsl:applytemplates>` i `<xsl:template>` (atrybut `mode`). Ćwiczenie wykonaj w dwóch podejściach:
 - a. Spis treści zawiera tylko tytuły rozdziałów.
 - b. Spis treści zawiera tytuły rozdziałów i podrozdziałów.
8. Zmień prezentację tytułów rozdziałów w spisie na duże litery. Wykorzystaj funkcje `XPATCH: translate()` i `text()` oraz znacznik `<xsl:value-of>`.
9. Zdefiniuj reguły dla prezentacji elementu `<important>`, który powinien zmieniać krój czcionki na pochyloną, a w przypadku zagnieżdżenia anulować pochYLENIE. Wykonaj to ćwiczenie w dwóch podejściach:
 - a. Arkusz XSL uwzględnia tylko znaczniki `<important>` zagnieżdżone w `<para>` (1 poziom zagnieżdżenia) i znaczniki `<important>` zagnieżdżone w `<important>` (2 poziom zagnieżdżenia).
 - b. Arkusz XSL uwzględnia dowolnie zagnieżdżone znaczniki `<important>` przełączając na przemian tekst pochylony i prosty. HINT: wykorzystaj tryby pracy `<xsl:applytemplates>` i `<xsl:template>` (atrybut `mode`).
10. Przygotuj reguły generujące spis treści w sposób proceduralny z wykorzystaniem pętli `<xsl:foreach>`. Prócz `<xsl:for-each>` wykorzystaj jeszcze funkcję `text()` i `<xsl:value-of>`. Ćwiczenie wykonaj w dwóch podejściach:
 - a. Spis treści zawiera tylko tytuły rozdziałów.

b. Spis treści zawiera tytuły rozdziałów i podrozdziałów.

11. Dodaj do stylu głównego generację spisu treści elementem `<toc/>`. Od tego momentu spis treści nie powinien być automatycznie generowany na początku dokumentu HTML, tylko w miejscu, którym w dokumencie XML pojawia się element `<toc/>`.
12. **DLA CHĘTNYCH.** Uzupełnij formatowanie spisu treści książki, tak aby tytuły rozdziałów i sekcji stały się odnośnikami do odpowiednich miejsc w tekście. **HINT.** Wykorzystaj:
- odnośniki wewnątrz strony `<a name="label">` i `<a href="#label">` w generowanym dokumencie
 - wstawianie wyników wyrażeń XPATH wewnątrz wartości atrybutów generowanego dokumentu, żeby poprawnie generować wartości atrybutów `name` i `href` znacznika `<a>`
 - funkcję `text()`
13. **DLA CHĘTNYCH.** Dodaj do książki możliwość obsługi elementu `<xref>` reprezentującego odwołania do innych fragmentów książki. Element `<xref>` pozwala na wstawienie odnośnika do wybranego rozdziału na podstawie jego `id`. Jeżeli element `<xref>` zawiera tekst, to jest to tekst, który jest łączem. Jeżeli nie zawiera tekstu, to tekst łączy to właściwy tytuł rozdziału. Oto przykłady użycia tego elementu:

```
<chapter id="wstep">
<title>Wstęp</title>
...
<chapter>
...
<chapter>
<para>Zobacz <xref target="wstep">rozdział poprzedni</xref>
lub rozdział pod tytułem <xref target="wstep"/>.
<para>
</chapter>
```

HINT. Wykorzystaj:

- odnośniki wewnątrz strony `<a name="label">` i `<a href="#label">` w generowanym dokumencie
- test, czy element zawiera podwężły dla odpowiedniej obsługi znacznika dwóch wariantów `<xref>`
- wstawianie wyników wyrażeń XPATH wewnątrz wartości atrybutów generowanego dokumentu, żeby poprawnie generować wartości atrybutów `name` i `href` znacznika `<a>`
- znacznik `<xsl:variable>` - może się przydać przy szukaniu właściwego tytułu rozdziału dla drugiego wariantu znacznika `<xref>`.
- zapytanie XPATH zależne od wartości zmiennej - może się przydać przy szukaniu właściwego tytułu rozdziału dla pierwszego wariantu znacznika `<xref>`.