

Standard:

2 pkt

- ☐ XQuery opisuje grafikę wektorową
- ☐ DOM jest uniwersalnym, zdarzeniowym standardem modelowania dokumentów XML
- ☐ XML jest kontekstowy, hierarchiczny
- ☐ DTD zawiera gramatykę języka, pozwala stworzyć przepis na konkretny język
- ☐ XML jest zależny od platformy
- ☐ XSLT umożliwia wyświetlenie zawartości XML w przeglądarce
- ☐ Wykorzystanie XSLT do tworzenia dokumentu FO pozwala na oddzielenie danych od sposobu prezentacji i wykorzystanie z różnymi zestawami danych

W XSLT value-of wyświetla pierwszą informację, określoną w atrybucie select

1 pkt

Wybierz jedną odpowiedź:

- ☐ Prawda
- ☐ Fałsz

Wskaż błędną odpowiedź. Typ złożony (complexType) jest niezbędny, gdy chcemy zdefiniować

1 pkt

- ☐ typ dla elementu z podelementami
- ☐ typ dla atrybutu
- ☐ typ dla elementu z zawartością mieszaną
- ☐ typ dla elementu z atrybutem i zawartością tekstową

Odznacz mój wybór

HTML5

1 pkt

- ☐ wymaga pluginów do wyświetlenia wideo
- ☐ umożliwia rysowanie na stronach internetowych
- ☐ umożliwia definiowanie zbioru własnych atrybutów dla kilku wybranych znaczników
- ☐ pozwala na dodawanie nowych znaczników

Odznacz mój wybór

Wskaż błędną odpowiedź

2 pkt

- ☐ w regule `h1{color:red}` `h1` jest selektorem
- ☐ selektor może zawierać tylko nazwy znaczników
- ☐ można zdefiniować klasę selektorów
- ☐ deklaracje można grupować
- ☐ reguły można grupować

Odznacz mój wybór

Jaki będzie wynik zastosowania do poniższego pliku [xml](#) pliku z transformacjami?

3 pkt

```
<Root>      <xsl:template match="/">
  <x>1</x>    <xsl:apply-templates />
  <y>4</y>    </xsl:template>
  <x>2</x>    <xsl:template match="x">
  <x>3</x>    <xsl:value-of select="." />
  <y>5</y>    </xsl:template>
  <z>        <xsl:template match="a">
    <a>a</a>  <xsl:value-of select="." />
    <b>b</b>  </xsl:template>
    <a>c</a>
  </z>
</Root>
```

Wybierz jedną odpowiedź:

- ☐ abc
- ☐ 123abc
- ☐ 14235abc
- ☐ 12345abc
- ☐ ac
- ☐ 12345ac
- ☐ 123ac

Odznacz mój wybór

Standard SVG

1 pkt

- ☐ złożone transformacje mogą być wykonywane dla dowolnych elementów rysunku
- ☐ pozwala na tworzenie na stronach internetowych tylko statycznej grafiki
- ☐ pozwala na operacje na grafice wektorowej, operacje na bitmapach nie są dostępne
- ☐ nie pozwala na interakcję z użytkownikiem

Odznacz mój wybór

Ta sama transformacja może być zastosowana do dokumentów XML:

1 pkt

Wybierz wszystkie poprawne:

- ☐ tylko wtedy gdy mają różną strukturę
- ☐ również wtedy gdy ich struktura się różni
- ☐ gdy mają tą samą lub różną strukturę
- ☐ tylko gdy mają identyczną strukturę
- ☐ tylko gdy mają korzeń o tej samej nazwie

W [XSLT](#) sortowanie elementów może się odbywać

1 pkt

Wybierz jedną odpowiedź:

- ☐ w dowolnym miejscu w szablonie
- ☐ w wywołaniu szablonu (apply-templates)
- ☐ tylko w pętli for-each
- ☐ w dowolnym miejscu pętli for-each
- ☐ w instrukcji *if*

Odznacz mój wybór

W XSLT szablon (template) musi mieć atrybut match i name

1 pkt

Wybierz jedną odpowiedź:

- ☐ Prawda
- ☐ Fałsz

- ☐ pozwala stworzyć łącze dla bitmapy
- ☐ zawiera ściśle określony zbiór znaczników
- ☐ jest metajęzykiem
- ☐ jest rozszerzalny
- ☐ znaczniki mogą mieć atrybuty

Dla poniższej definicji nieprawidłowe będzie

```
<xs:element name="r" >
  <xs:complexType mixed="true">
    <xs:sequence >
      <xs:choice>
        <xs:element name="k" type="xs:int" maxOccurs="2"/>
        <xs:element name="b" type="xs:string" />
      </xs:choice>
      <xs:choice>
        <xs:element name="a" type="xs:string" />
        <xs:element name="c" type="xs:string" />
      </xs:choice>
    </xs:sequence >
    <xs:attribute name="d" type="xs:string" use="required"/>
  </xs:complexType>
</xs:element>
```

- ☐ <r d="wartosc"> jakiś tekst<k>1</k><k>2</k><c>4</c> </r>
- ☐ <r d="wartosc"> <k>1</k><k>2</k><a>4 </r>
- ☐ <r d="wartosc"> jakiś tekst <k>1</k><k>2</k><a>4 </r>
- ☐ <r d="wartosc"> jakiś tekst<k>1</k><k>2</k><a>3<c>4</c> </r>
- ☐ <r d="wartosc"> 3<a>4 </r>

Odznacz mój wybór

W [XSLT](#) tworzenie rozgałęzień w instrukcji if można uzyskać poprzez

1 pkt

Wybierz jedną odpowiedź:

- ☐ if-else
- ☐ otherwise
- ☐ nie można
- ☐ else
- ☐ else-if

Odznacz mój wybór

Które cechy są charakterystyczne dla XML

2 pkt

- ☐ brak możliwości formalnego zdefiniowania znaczników
- ☐ przemieszanie treści ze sposobem prezentacji
- ☐ ściśle określony zbiór znaczników
- ☐ metajęzyk
- ☐ format binarny

Odznacz mój wybór

Dla dokumentu [XSLT](#) prefiksem dla przestrzeni nazw:

1 pkt

Wybierz wszystkie poprawne:

- ☐ musi być `xsl` lub `xslt`
- ☐ może być `h1h`
- ☐ musi być `xsl`
- ☐ może nie być prefiksu
- ☐ musi istnieć dowolny prefiks
- ☐ często jest `xsl`
- ☐ musi być `xslt`

Wskaż błędną odpowiedź: w standardzie [CSS](#)

2 pkt

- ☐ styl jest dziedziczony w zakresie zagnieżdżonym
- ☐ możliwa jest specyfikacja stylu w oddzielnym pliku
- ☐ musi wystąpić z każdym plikiem [HTML](#)
- ☐ możliwa jest specyfikacja stylu w nagłówku dokumentu

Odznacz mój wybór

Wybierz jedną odpowiedź:

- ☐ można dostać się tylko do węzłów położonych poniżej węzła bieżącego
- ☐ umożliwia stworzenie gramatyki języka
- ☐ @węzeł oznacza węzeł drzewa, który jest atrybutem
- ☐ stosowany jest tylko w standardzie [XSLT](#)
- ☐ \$węzeł oznacza węzeł drzewa, który jest atrybutem
- ☐ : oznacza bieżący węzeł

Odznacz mój wybór

Wskaż niepoprawną odpowiedź: XPath

2 pkt

- ☐ zapis węzeł1/węzeł2 jest równoważny zapisowi ../węzeł1/węzeł2
- ☐ zapis węzeł[\$atribut='w'] pozwoli na wybranie węzła dla którego wartość atrybutu jest równa w
- ☐ zapis /węzeł1/węzeł2/węzeł3 oznacza ścieżkę bezwzględną
- ☐ XPath traktuje XML jako drzewo elementów
- ☐ konstrukcja węzeł1//węzeł2 pozwala na wybranie elementów węzeł2, które nie są bezpośrednio dziećmi elementu węzeł1

Odznacz mój wybór

Plik Schema (xsd) jest poprawnym dokumentem [XML](#)

1 pkt

Wybierz jedną odpowiedź:

- ☐ Prawda
- ☐ Fałsz

XSLT jest językiem

2 pkt

- ☐ deklaratywnym
- ☐ sterowanym danymi
- ☐ sterowanym zdarzeniami
- ☐ obiektowym
- ☐ proceduralnym

2 pkt

Wskaż odpowiedź fałszywą: Dla poniższej definicji DTD:

<!ELEMENT nowela (tytuł, rozdział+)>

<!ELEMENT tytuł (#PCDATA)>

<!ELEMENT rozdział (tytuł | treść)+>

<!ELEMENT treść (#PCDATA)>

<!ATTLIST rozdział

wersja (szkic | edycja | druk) #REQUIRED>

- ☐ 'wersja' jest atrybutem elementu 'rozdział'
- ☐ element 'rozdział' musi mieć 'tytuł' i 'treść'
- ☐ element 'tytuł' jest łańcuchem znaków
- ☐ 'wersja' może przyjąć tylko trzy różne wartości

Odznacz mój wybór

Wybierz jedną odpowiedź:

- ☐ value-of wyświetli informacje ze wszystkich węzłów określonych w atrybucie select
- ☐ zmienne muszą zawierać zbiory węzłów i fragmenty drzewa
- ☐ Przy wywołaniu szablonu (apply-templates) konieczne jest użycie atrybutu select
- ☐ wykonanie kodu jest niedeterministyczne
- ☐ jeśli nie istnieje szablon dla danego kontekstu zastosowany zostanie szablon rodzica

Odznacz mój wybór

W XSLT szablon z atrybutem *name* wywołujemy poprzez *apply-templates*

1 pkt

Wybierz jedną odpowiedź:

- ☐ Prawda
- ☐ Fałsz

Jaki będzie wynik zastosowania do poniższego pliku [xml](#) pliku z transformacjami?

3 pkt

```
<Root>      <xsl:template match="/">
  <x>1</x>    <xsl:apply-templates select="Root/z" />
  <y>4</y>    </xsl:template>
  <x>2</x>    <xsl:template match="y">
  <x>3</x>    <xsl:value-of select="." />
  <y>5</y>    </xsl:template>
  <z>        <xsl:template match="x">
    <a>a</a>  <xsl:value-of select="." />
    <b>b</b>  </xsl:template>
    <a>c</a>
  </z>
</Root>
```

Wybierz jedną odpowiedź:

- ☐ 41
- ☐ 14235
- ☐ 14
- ☐ 45123abc
- ☐ 45123
- ☐ abc

Odznacz mój wybór

W XSLT pominięcie atrybutu *select* dla *value-of* pozwala wybrać wartości ze wszystkich dzieci bieżącego węzła **1 pkt**

Wybierz jedną odpowiedź:

- ☐ Prawda
- ☐ Fałsz

W XSLT możemy zastosować ten sam szablon do wielu różnych węzłów

1 pkt

Wybierz jedną odpowiedź:

- ☐ Prawda
- ☐ Fałsz

Jaki będzie wynik zastosowania do poniższego pliku [xml](#) pliku z transformacjami?

3 pkt

```
<Root>      <xsl:template match="/">
  <x n='a'>   <xsl:apply-templates select="Root/x" />
    1        </xsl:template>
    <z>2</z>  <xsl:template match="x">
  </x>        <xsl:apply-templates />
  <y>3</y>    </xsl:template>
  <x>4</x>    <xsl:template match="z">
  <x>5</x>      z
  <y>6</y>    </xsl:template>
  <z> 7 </z>
</Root>
```

Wybierz jedną odpowiedź:

- ☐ a1234567
- ☐ a1z45z
- ☐ 1234567
- ☐ 1z45z
- ☐ a1z45
- ☐ 1z45

Odznacz mój wybór

Dokument XML jest poprawnie uformowany jeśli:

2 pkt

- ☐ w elementach występuje co najwyżej 8 atrybutów
- ☐ zawartość pliku XML jest zgodna z definicją dokumentu (np. DTD)
- ☐ wartości atrybutów są ujęte w cudzysłowy
- ☐ istnieje tylko jeden element główny (korzeń)
- ☐ nie ma elementów pustych
- ☐ wszystkie znaczniki są pozamykane
- ☐ zakresy znaczników nie przecinają się

Wskaż błędną odpowiedź: W pliku XML

2 pkt

- ☐ znaczniki nie mogą się krzyżować
- ☐ każdy otwarty znacznik musi zostać zamknięty
- ☐ musi istnieć dokładnie jeden element główny (korzeń)
- ☐ w nazwach elementów i atrybutów nie ma znaczenia wielkość liter

Odznacz mój wybór

Dla poniższej definicji w [XML Schema](#), w pliku [XML](#) kolejność elementów może być następująca

```
<xsd:sequence maxOccurs="2">  
  <xsd:choice minOccurs="0">  
    <xsd:element name="a" type="xsd:long" />  
    <xsd:element name="b" type="xsd:string" />  
  </xsd:choice>  
  <xsd:sequence minOccurs="0" maxOccurs="unbounded">  
    <xsd:element name="c" type="xsd:int" />  
    <xsd:element name="d" type="xsd:float" />  
  </xsd:sequence>  
</xsd:sequence>
```

- ☐ a. aba
- ☐ b. bbb
- ☐ c. abacd
- ☐ d. aaa
- ☐ e. acdcdcdcd
- ☐ f. cdacdacd

Odznacz mój wybór