

XML въпроси за изпит

1. В CSS3, фиксираната (fixed) схема за позициониране:
 - a. **предизвиква повторение на кутията на всяка страница при paged media**
 - b. се характеризира със задаване на явно отместване (explicit offset) спрямо съдържащия блок
 - c. предефинира top и left позициите на кутията
 - d. се контролира от браузера с цел по-бързо показване на съдържанието
2. creator, title, publisher, contributor, date и format са име на елементи, дефинирани от:
 - a. RDFS
 - b. **DC**
 - c. FOAF
 - d. OWL
3. При избиране на елемент, наречен MyElem и имащ атрибут Attr със стойност title, ние трябва да използваме:
 - a. **select="MyElem[@Attr='title']"**
 - b. select="MyElem{ @Attr='title'}"
 - c. select="MyElem(@Attr='title')"
 - d. select="MyElem[Attr='title']"
4. Какво не е възможно да бъде описано в DTD:
 - a. къде типът на елементите може да се среща
 - b. какви имена могат да бъдат използвани за тип на елементите
 - c. йерархия на документа и грануларност
 - d. имена и типове на атрибутите на елементите минимален и максимален брой срещания на елемент
 - e. **минимален и максимален брой срещания на елемент**
5. XSL елементът Apply-Templates се използва вътре в един шаблон (template) за извикване на други шаблони. Той:
 - a. активира нерекурсивно обработката на всички наследници на елемента, за който се отнася
 - b. **активира рекурсивно обработката на всички наследници на елемента, за който се отнася**
6. Целта на валидацията на XML документ от XML парсера е да се провери дали XML документът е добре структуриран (well-formed).
 - a. Верно
 - b. **Неверно**
7. rdf:Seq задава:
 - a. **група на подредени ресурси или литерали, с вероятно дублиране**
 - b. група на неподредени ресурси или литерали, без дублиране
 - c. група на подредени ресурси или литерали, без дублиране
 - d. група на неподредени ресурси или литерали, с вероятно дублиране
 - e. група на алтернативни ресурси или литерали
8. В XSchema, локалните типове:
 - a. винаги са директни наследници на корена

- b. могат да бъдат директни наследници на корена
 - c. могат да бъдат или да не бъдат директни наследници на корена
 - d. не могат да бъдат директни наследници на корена**
9. Кое е името на подразбиращото се (default) пространство от имена в декларацията подолу:
- ```
<?xml version="1.0" encoding="ISO-8859-1"?>
<pers:person xmlns:pers="http://sernaferna.com/pers"
 xmlns:html="http://www.w3.org/1999/xhtml">
```
- a. и pers, и html
  - b. Html
  - c. Pers
  - d. зависи от XML парсера
  - e. няма подразбиращото се (default) пространство от имена**
10. DocumentType::Node Interface се използва за получаване на информация за документ, описан в DTD.
- a. DOM 1.0 разрешава редактиране на този възел
  - b. DOM 1.0 не разрешава редактиране на този възел**
11. Могат ли екземплярите на класовете да бъдат част от онтологията, описваща тези класове?
- a. Да**
  - b. Не
12. Описанието
- ```
ex:adrian foaf:knows ex:ger  
ex:adrian foaf:age "41"^^xs:int.
```
- представя RDF Triples в:
- a. RDF/XML формат.
 - b. N3 формат.**
 - c. Turtle формат.
 - d. RDFa формат.
13. XML Schema разрешава да се прави разлика между уникална стойност (unique) и ключ (key).
- a. Верно**
 - b. неверно
14. CDATA съдържанията са:
- a. или парсвани, или игнорирани от XML парсерите в зависимост от CDATA директивата
 - b. игнорирани от XML парсерите**
 - c. парсвани от XML парсерите
15. Даден CSS стил може да бъде inlined (а не embedded) в документ посредством:
- a. елемента INLINE
 - b. атрибута INLINE
 - c. атрибута STYLE**
 - d. елемента STYLE

16. Параметрично Entity:
- a. се дефинира в XML документа и може да се използва в който и да е XML документ
 - b. се дефинира в маркъп декларациите в DTD, но може да се използва в който и да е XML документ
 - c. се дефинира в маркъп декларациите в DTD и може да се използва само в DTD дефиниции**
17. Кой от следните изрази е валиден пример за дефиниране на избор (choice) от елементи в DTD:
- a. $(A \sim B \sim C)$
 - b. $(A | B | C)$**
 - c. $(A \text{ or } B \text{ or } C)$
 - d. (A, B, C)
18. В RDF, група от алтернативни стойности се задава чрез:
- a. `<rdf:Opt>`
 - b. `<rdf:Switch>`
 - c. `<rdf:Choice>`
 - d. `<rdf:Case>`
 - e. `<rdf:Alt>`**
19. В XSchema, референцията (напр.) може да реферира:
- a. само към глобални типове**
 - b. както към локални, така и към глобални типове
 - c. само към локални типове
 - d. нито към локални, нито към глобални типове
20. В XSchema, референцията към тип може да реферира:
- a. както към локални, така и към глобални типове
 - b. само към глобални типове
 - c. само към локални типове
 - d. нито към локални, нито към глобални типове
21. В OWL, също както и в ООП, не може да съществуват два класа с общи екземпляри.
- a. Верно
 - b. Неверно**
22. Binary entity може да се използва само като атрибут от тип ENTITY.
- a. Неверно
 - b. Верно**
23. В RDFS, Property е подмножество на RDFS Resources и има за домейн (rdfs:domain):
- a. типа на стойностите на това Property
 - b. класа rdfs:Resource, на който принадлежат всички екземпляри на ресурса
 - c. множество от отделни стойности на това Property
 - d. класа, асоцииран с това Property**
24. RDF графът от фигурата задава:
- a. Лекцията е посетена от Джон, Мери или Крис.

- b. Лекцията е посетена от Джон, Мери и Крис - в този ред.
 - c. Лекцията е посетена от Джон, Мери и Крис.**
 - d. Лекцията е посетена от Джон, Мери, Крис и от никой друг.
25. XSL кодът, показан на фигурата, ще даде като резултат:
- a. изходните XML елементи ще бъдат с имена, еднакви със съдържанието на елементите от сорс дървото, и със съдържание "My own contents!"**
 - b. същите елементи в изходния XML файл както в сорс XML файла, но с добавено в края съдържание "My own contents!"
 - c. същите елементи в изходния XML файл както в сорс XML файла, и с коментар "My own contents!" за всеки един елемент
26. XML пространствата от имена се използват за разграничаване:
- a. нито на XML елементи, нито на XML атрибути
 - b. и на XML елементи, и на XML атрибути**
 - c. само на XML елементи
 - d. само на XML атрибути
27. Регулярният израз `[^0-9]x` дефинира XSchema стрингови стойности, които представляват:
- a. който и да е не-цифров символ, следван от символа x**
 - b. символът ^, последван от произволна цифра следвана от символа x
 - c. x пъти повторение на който и да е цифров символ
 - d. x пъти повторение на който и да е не-цифров символ
28. Кое от твърденията е истина:
- a. само DOM Element обектите имат атрибути**
 - b. само DOM Node обектите имат атрибути
 - c. и DOM Element, и DOM Node обектите имат атрибути
29. Всяко IDREF, декларирано в DTD, трябва да:
- a. избере дадено определено ID**
 - b. избере кое да е ID
 - c. избере множество ID-та
30. Осъществяването на embedding (а не inlining) на Style Sheet според CSS правилата е възможно посредством:
- a. елемента EMBED
 - b. елемента STYLE**
 - c. елемента LINK
 - d. атрибута STYLE
31. Всеки DOM възел (Node) може да има деца.
- a. Верно
 - b. зависи от децата
 - c. неверно**
32. Кой от елементите ISBN по-долу не се описва от дефиницията `<element name="ISBN" type="cat:ISBNTypе"/>`, където ISBNTypе е:
- a. 1-23-579321-3
 - b. 4-256-76435-4

- c. 63247-84365-12345
 - d. 12345-54321-98765
 - e. **6-32-47843-5**
33. Уникална идентификация на пространство от имена се реализира с:
- a. URN
 - b. **URL**
 - c. URI (и URL, и URN)
34. Person, title, familyName, knows, age, Document и Organization са име на елементи, дефинирани от:
- a. OWL
 - b. **FOAF**
 - c. DC
 - d. RDFS
35. Квалифицирано XML име (наречено QName) е:
- a. **име от вида namespace_prefix:local_name**
 - b. всяко име на XML елемент
 - c. всяко валидно XML име
 - d. всяко XML име на пространство от имена
36. За постигане на по-малък, ефикасен и бърз код с използване на StAX, се препоръчва:
- a. StAX Direct Mapping API
 - b. iterator API
 - c. **cursor API**
 - d. StAX Events API
37. В XSchema, типът на всеки елемент на еквивалентен клас трябва да бъде:
- a. същият като типа на елемента-екземпляр
 - b. различен от типа на елемента-екземпляр
 - c. дериват на типа на елемента-екземпляр
 - d. **същият като типа на елемента-екземпляр, или негов дериват**
38. Ако сме дефинирали XSL променлива като <xsl:variable name="price">low</xsl:variable>, то тя може да се използва в XSL елемент като:
- a. <xsl:value-of select="{ \$price }"/>
 - b. <xsl:value-of select="@price"/>
 - c. **<xsl:value-of select="\$price"/>**
 - d. <xsl:value-of select="price"/>
39. Една XLink дъга (arc), която има локален стартов ресурс и отдалечен краен ресурс, се нарича:
- a. **Outbound**
 - b. сочеща трети (third-party) ресурс
 - c. inbound
40. Редът <!ATTLIST point honorific (Mr|Ms|Mrs|Rev|Dr) ... > е валиден DTD пример за:
- a. **name group**
 - b. CDATA
 - c. NMTOKENS

- d. notation
41. RDF Literals могат да бъдат зададени само като обект в RDF тройка.
- a. Неверно
 - b. Верно**
42. Annotation Property в OWL се използва за добавяне на метаданни:
- a. само към свойства (properties)
 - b. към класове, екземпляри и свойства**
 - c. към класове и към екземпляри
 - d. само към класове
 - e. към класове и към свойства
 - f. към екземпляри и към свойства
 - g. само към екземпляри
43. Описанието в RDF на група, съдържаща само зададените в описанието членове, става чрез използване на:
- a. Множество
 - b. Колекция**
 - c. Алтернатива
 - d. Bag
 - e. последователност
44. В XLink, изходящите връзки (outbound links) могат да имат:
- a. не по-малко от два участващи ресурса
 - b. точно два участващи в тях ресурса**
 - c. произволен краен брой участващи в тях ресурси
45. В XML Schema, ние можем да построим производни (derived) types:
- a. само с restrictions
 - b. както с extensions, така и с restrictions**
 - c. само с extensions
46. В XLink, входящите дъги (inbound arcs) могат да имат:
- a. произволен краен брой участващи в тях ресурси**
 - b. точно два участващи в тях ресурса
47. С един DOM Element обект:
- a. не може да направите разлика между подразбираща се (default) стойност, определена в DTD, и стойността, дадена в XML файла**
 - b. може да направите разлика между подразбираща се (default) стойност, определена в DTD, и стойността, дадена в XML файла
48. Кой от следните изрази е валиден пример за дефиниране на последователност (sequence) от елементи в DTD:
- a. (A and B and C)
 - b. (A ^ B ^ C)
 - c. (A, B, C)**
 - d. (A | B | C)
49. Един XML документ може да има повече от едно вътрешно DTD.
- a. Неверно**
 - b. Верно

50. Един XML документ може да има повече от едно DTD.
- a. **Верно**
 - b. неверно
51. Могат ли екземплярите на класовете да бъдат част от онтологията, описваща тези класове?
- a. Не
 - b. **да**
52. Resource Description Framework (RDF) представя информация за ресурси, които:
- a. не могат да бъдат достъпни в Уеб
 - b. трябва да бъдат достъпни в Уеб
 - c. **могат да бъдат или да не бъдат достъпни в Уеб**
53. Разгледайте <xsl:value-of> елемента. Ако стойността на неговия select атрибут е select='.', тогава ние избираме:
- a. текстовия контекст само на елемента
 - b. същия текстов контекст, както когато използваме text() функцията
 - c. **текстовия контекст на елемента и текстовия контекст на всички наследници на елемента**
 - d. текстовия контекст на всички наследници на елемента
54. Регулярният израз [1-9]?[0-9] дефинира XSchema стрингови стойности, които са:
- a. от 1 до 9
 - b. **от 0 до 99**
 - c. от 1 до 99
 - d. от 1 до 90
55. Записът
- ```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefix contact: <http://www.w3.org/2000/10/swap/pim/contact#>.
<http://www.w3.org/People/EM/contact#me>
rdf:type contact:Person;
contact:fullName "Eric Miller";
contact:mailbox <mailto:em@w3.org>;
contact:personalTitle "Dr.".

```
- представя по-долния RDF граф в:
- a. RDF/XML
  - b. **Turtle**
  - c. RDFa
  - d. N3
56. "Формална, експлицитна спецификация на споделена концептуализация" е дефиниция за:
- a. **нещо друго, което не присъства в тези отговори**
  - b. валиден XML документ
  - c. валиден RDFS документ
  - d. валиден RDF/XML документ
  - e. валиден XML документ, представящ антология в даден жанр
57. XSchema шаблонен фасет (pattern facet) е фасет за данни от тип:

- a. **String**
  - b. Integer
  - c. Binary
  - d. decimal
58. Resource Description Framework (RDF) служи за описания, предназначени:
- a. за потребителско визуализиране на метаданни
  - b. както за обработка от софтуерни приложения, така и за потребителско визуализиране на метаданни
  - c. нито за обработка от софтуерни приложения, нито за потребителско визуализиране на метаданни
  - d. **за обработка от софтуерни приложения**
59. Атрибутите, специфицирани в DTD посредством ключовата дума IMPLIED:
- a. имат само една допустима стойност
  - b. **могат да бъдат определени (специфицирани) в XML**
  - c. трябва да бъдат определени (специфицирани) в XML
  - d. имат стойност по подразбиране, ако не са определени (специфицирани) в XML
60. Общо текстово Entity:
- a. Не може да бъде използвано рекурсивно и не може да се появи в съдържанието на елемент и/или в стойност на атрибут
  - b. Може да бъде използвано рекурсивно, но не може да се появи в съдържанието на елемент и/или в стойност на атрибут
  - c. Може да бъде използвано рекурсивно и може да се появи в съдържанието на елемент и/или в стойност на атрибут
  - d. Може да бъде използвано рекурсивно и може да се появи в съдържанието на елемент, но не и в стойност на атрибут
  - e. **Не може да бъде използвано рекурсивно, но може да се появи в съдържанието на елемент и/или в стойност на атрибут**
61. Приложения, които имат нужда от сложни структурни манипулации на много от XML елементите, трябва да използват:
- a. SAX
  - b. **DOM**
  - c. StAX API
  - d. XSLT
  - e. CSS
62. За представяне на стойности в XML файлове, ние можем да използваме XML атрибути или XML елементи. Кой от следващите отговори не е аргумент за употреба на атрибути вместо на елементи?
- a. използваме атрибут, когато той е свойство на елемента
  - b. използваме атрибути за валидация на прост (simple) тип данни
  - c. **използваме атрибути за валидация на сложна структура**
  - d. използваме атрибут, когато информацията е присъща за елемента, а не за под-елементите
63. В примера по-долу:



```
<x xmlns:n1="http://www.hayde-de.org"
 xmlns="http://www.hayde-de.org" >
 <ok a="1" b="2" />
 <hmmm a="1" n1:a="2" />
</x>
```

- a. Дефиницията на елемента <hmmm> е невалидна според спецификацията на пространствата от имена.
  - b. Дефиницията на елемента <hmmm> е валидна според спецификацията на пространствата от имена.**
64. Единственото изискване един XML документ да бъде валиден е този документ да е структуриран (формиран) съгласно правилата за XML синтаксиса.
- a. Верно
  - b. неверно**
65. При парсване на XML документи посредством StAX, можем да се придвижваме само напред в XML документа.
- a. Истина**
  - b. Неистина
66. В XSchema, за глобалните типове:
- a. minOccurs и maxOccurs може да се дефинират само за глобални типове, които се реферират**
  - b. minOccurs и maxOccurs не може да се дефинират
  - c. minOccurs и maxOccurs може да се дефинират винаги
  - d. minOccurs и maxOccurs може да се дефинират само за глобални типове, които не се реферират
67. Ако няма специфицирано кодиране за един XML документ и документът не е нито в UTF-8, нито в UTF-16, тогава резултатът е:
- a. ISO 8859-1
  - b. ERROR**
  - c. UTF-8
  - d. UTF-16
68. В SAX, приложните обекти, имащи достъп до XML сорса:
- a. са регистрирани за callback функции или от програмиста, или от парсера
  - b. не трябва да бъдат регистрирани от програмиста за callback функции, тъй като те са listeners
  - c. трябва да бъдат регистрирани (от програмиста) за callback функциите на парсера**
69. Кое от показаните на фигурата средства не е пряко свързано с изграждане на семантичния Уеб:
- a. Ontologies
  - b. Logical Support
  - c. Tools
  - d. Semantic Annotations
  - e. Applications / Services
  - f. Collaboration synchronization**

70. Изберете верното:

- a. Един URI не може да бъде нито URL, нито URN
- b. Един URL може да бъде или URI, или URN
- c. Един URI може да бъде URL и URN едновременно
- d. Един URN може да бъде или URL, или URI
- e. Един URI може да бъде или URL, или URN**

71. Атрибутите на XML са чувствителни към регистъра.

- a. Невярно
- b. вярно**

72. MEDIA атрибутите като screen, aural, braille, tty, и т.н., са част от:

- a. само от CSS2 спецификацията
- b. само от CSS3 спецификацията
- c. CSS1 и CSS2 спецификациите
- d. CSS2 и CSS3 спецификациите**
- e. само от CSS1 спецификацията

73. XSL се използва за:

- a. трансформиране на XML документ към друг текстов документ**
- b. трансформиране на XML документ към друг документ само в HTML формат
- c. трансформиране на XML документ към друг документ само в XML или HTML формат
- d. трансформиране на XML документ към друг документ само в XML формат

74. Разгледайте P:first-letter { font-size: 200% } . Това е:

- a. CSS клас
- b. псевдо-атрибут (pseudo-attribute)
- c. нито едно от по-горе посочените
- d. псевдо-елемент (pseudo-element)**

75. При използването на XPath text() функцията, ние избираме:

- a. същият текстов контекст както когато използваме <xsl:value-of select='.'> елемента
- b. текстовия контекст на елемента и текстовия контекст на всички наследници на елемента
- c. текстовия контекст само на елемента**
- d. текстовия контекст на всички наследници на елемента

76. Ако в XSchema дефинираме exact тип, то:

- a. той може да има дериватни типове, и те могат да бъдат използвани в XML документа вместо този exact тип
- b. той може да има дериватни типове, но те не могат да бъдат използвани в XML документа вместо този exact тип**
- c. той може да има дериватни типове, но те могат да бъдат използвани в XML документа вместо този exact тип само при определени условия
- d. той не може да има дериватни типове

77. Ако даден атрибут на XML елемент има само локално име (т.е. името му не е QName), то:

- a. Той не се отнася към каквото и да било пространство от имена за документа, в който е дефиниран.
  - b. Той се отнася към пространството от имена по подразбиране (default namespace).
  - c. Той не се отнася към пространството от имена на елемента, към който принадлежи.
  - d. Той се отнася неявно към пространството от имена на елемента, към който принадлежи.**
78. В XSchema, глобалните типове:
- a. могат да не бъдат директни наследници на корена
  - b. могат да бъдат или да не бъдат директни наследници на корена
  - c. винаги са директни наследници на корена**
  - d. не могат да бъдат директни наследници на корена
79. При включването на документа "xpto.xml" с използване на <xi:include href="xpto.xml" parse="text"/>, документът "xpto.xml" ще бъде включен:
- a. като XML документ, но само ако той е добре конструиран
  - b. винаги като XML документ
  - c. като XML документ, но само ако той е добре конструиран и валиден спрямо схема или DTD
  - d. като обикновен текст**
80. Кой от следните елементи не е с грешен XML синтаксис:
- a. <birth>28.04.1990, Gabrovo</ birth>
  - b. <birth>28.04.1990, Gabrovo< /birth>
  - c. <birth>28.04.1990, Gabrovo</birth >**
  - d. <birth>28.04.1990, Gabrovo</bir th>
  - e. Всичките останали отговори съдържат елемент с грешен XML синтаксис.
81. Разгледайте имената на XML елементите <xml-tag> and <tag-xml>. Те са:
- a. и двата невалидни
  - b. и двата валидни
  - c. първият е невалиден, вторият е валиден**
  - d. нито едно от по-горе посочените
  - e. първият е валиден, вторият е невалиден
82. Чрез XPointer можем да реферираме към повече от един елемента в XML документ.
- a. Неверно
  - b. Верно**
83. Методът getAttributes() на DOM интерфейса Node връща:
- a. NamedNodeMap**
  - b. NodeList
  - c. Attr
  - d. Text
84. Кой от отговорите по-долу не съдържа име на RDFS свойство:
- a. rdfs:isDefinedBy
  - b. Rdfs:range

- c. Rdfs:comment
  - d. rdfs:subRange**
  - e. Rdfs:label
  - f. rdfs:subPropertyOf
  - g. Rdfs:member
  - h. rdfs:subClassOf
85. На показаната фигура на RDF граф, T-shirt е:
- a. Обект
  - b. Предикат
  - c. Дериват
  - d. Субект**
86. Ако свойството P е дефинирано с обхват (range) C и ресурсът R е обект в тройка с предикат P, то следва, че:
- a. C rdf:type R
  - b. P rdf:type R
  - c. C rdf:type P
  - d. R rdf:type C**
  - e. R rdf:type P
87. rdf:Bag задава:
- a. група на неподредени ресурси или литерали, с вероятно дублиране**
  - b. група на подредени ресурси или литерали, с вероятно дублиране
  - c. група от ресурси или литерали, които са алтернативи
  - d. група на подредени ресурси или литерали, без дублиране
  - e. група на неподредени ресурси или литерали, без дублиране
88. Йерархичната подредба на понятия заедно с информация за допълнителни отношения като по-широко/по-тясно понятие, синоними, еквивалентност и др., задава:
- a. Таксономия
  - b. Списък от термини
  - c. Пръстен от синоними
  - d. Тезаурус**
  - e. Онтология
89. SAX Element обектите:
- a. зависи от SAX парсера
  - b. могат да разграничават атрибутите, дефинирани изрично, от тези специфицирани в DTD**
  - c. не могат да разграничават атрибути, дефинирани изрично, от тези специфицирани в DTD
90. XPath изразът //book[@pages] връща:
- a. всички book елементи, които имат атрибут pages**
  - b. първият book елемент, който има атрибут pages
  - c. първият book елемент, който има непразен атрибут pages
  - d. всички book елементи, които имат непразен атрибут pages
91. xml:id е псевдо-атрибут, който задава условие за уникална стойност, като това:

- a. е необходимо да се декларира в XML схема.
  - b. е необходимо да се декларира в DTD или XML схема.
  - c. е необходимо да се декларира в DTD.
  - d. не е необходимо да се декларира в DTD или XML схема.**
92. При използване на вътрешна и външна DTD дефиниция, вътрешната DTD може да предефинира:
- a. само ENTITY и NOTATION на външната дефиниция
  - b. само ATTLIST и NOTATION на външната дефиниция
  - c. само ENTITY и ATTLIST на външната дефиниция**
  - d. всички дефиниции на външната DTD
  - e. само ENTITY, ATTLIST и NOTATION на външната дефиниция
93. Задаването на мрежа от хора чрез FOAF твърдения става посредством релацията:
- a. foaf:linkedTo
  - b. foaf:relatedTo
  - c. нито една от изброените
  - d. Foaf:follows
  - e. Foaf:knows**
94. "other" и "none" са възможни стойности за XLink атрибутите:
- a. arcrole и role
  - b. actuate и show**
  - c. actuate и arcrole
  - d. show и role
  - e. show и arcrole
95. В CSS, задаването на елемент, чийто атрибут attr завършва с дадена стойност xpto, става чрез:
- a. [attr\$=xpto]**
  - b. [attr|=xpto]
  - c. [attr^=xpto]
  - d. [attr=xpto]
  - e. [attr~=xpto]
96. Описание (конкретизация) на твърдение с използване на RDF речника се извършва чрез:
- a. типа rdf:Consideration и свойствата rdf:subject, rdf:predicate и rdf:object
  - b. типа rdf:Reification и свойствата rdf:subject, rdf:predicate и rdf:object
  - c. използване на други средства
  - d. типа rdf:Statement и свойствата rdf:subject, rdf:predicate и rdf:object**
  - e. типа rdf:Concretisation и свойствата rdf:subject, rdf:predicate и rdf:object
97. Свойството *rdfs:isDefinedBy* е специален (под)тип на:
- a. rdfs:is
  - b. rdfs:seeAlso**
  - c. rdfs:isCreatedBy
  - d. rdfs:isDescribedBy
  - e. rdfs:isDefined

98. XLink не може да дефинира фрагментни идентификатори за URI, сочеши към възли или части от тях в XML ресурси.
- a. **Верно**
  - b. Неверно
99. При автоматичното разпознаване на анотации, отношението броя на правилно разпознатите анотации към броя на всички съществуващи правилни анотации се нарича:
- a. пълнотата на връщане (recall)
  - b. устойчивост (стабилност)
  - c. достоверност на резултата (reliability)
  - d. **точност (precision)**
100. Дефинирането на референция (Reference) към ключ (Key) в XSD може да има полета, които са:
- a. нито атрибути, нито елементи
  - b. само елементи
  - c. само атрибути
  - d. както атрибути, така и елементи, като всеки един атрибут (или елемент) трябва да съответства на атрибут (или на елемент) в ключа
  - e. **както атрибути, така и елементи, като всички полета трябва да съответстват на типа и на позицията на тези в ключа**
101. RDF поддържа описания на групи, съдържащи само определени членове, посредством:
- a. RDF вектори
  - b. **RDF колекции**
  - c. RDF масиви
  - d. RDF enumeration
  - e. нито един от изброените начини
102. В RDF, класовете могат да бъдат екземпляри на други класове.
- a. Неверно
  - b. **Верно**
103. В XML Schema дефиницията за елементно съдържание ... order="all" ... е еквивалентна на ANY в DTD.
- a. Верно
  - b. **Неверно**
104. RDF твърдението  
dbpedia:Mount\_Etna rdf:type my-pref:Mountain, my-pref:Volcano .  
е допустимо и валидно.
- a. **Верно**
  - b. Неверно
105. В CSS3, абсолютната (absolute) схема за позициониране:
- a. задава липса на движение на кутията при движение на документа за continuous media
  - b. **се характеризира със задаване на явно отместване (explicit offset) спрямо съдържащия блок**

- c. предефинира top и left позициите на кутията
  - d. предизвиква повторение на кутията на всяка страница при paged media
106. В RDF твърдението
- S rdf:type OP
- a. S задава клас (категория), а O задава екземпляр на този клас
  - b. O задава клас (категория), а S задава екземпляр на този клас**
  - c. S и O задават класове на екземпляри
  - d. S и O задават екземпляри на класове
107. Множественото наследяване (multiple inheritance) не е допустимо в RDFS.
- a. Неверно**
  - b. Верно
108. Разгледайте имената на XML празните елементи <emptyTag /> и <emptyTag/>. Те са:
- a. нито едно от останалите
  - b. и двата невалидни
  - c. първият е невалиден, вторият е валиден
  - d. първият е валиден, вторият е невалиден**
  - e. и двата валидни
109. В какъв формат е следният запис:
- ```
<http://www.w3.org/TR/rdf-syntax-grammar> <http://xml.org/1.1/title> "XML Basics" .
<http://www.w3.org/TR/rdf-syntax-grammar> <http://example.org/stuff/xml> _:bnode .
_:bnode <http://example.org/stuff/1.0/fullname> "D. D. Vass" .
_:bnode <http://example.org/stuff/1.0/homePage> <http://www.aemon.net/>.
```
- a. друг формат, различен от останалите
 - b. RDF/XML
 - c. N3
 - d. N-Triples**
 - e. Durtle
110. Описанието (конкретизацията) на RDF твърдение чрез използване на RDF речника се нарича:
- a. Reification**
 - b. Concretization
 - c. Objectification
 - d. Consideration
111. Ако имаме дефинирани две еднакви CSS свойства с различни стойности за един и същи елемент, то ще се приложи последно дефинираното свойство:
- a. Винаги
 - b. винаги, освен когато правилото за прилагане на едно от свойствата взима под внимание кой е предходният елемент
 - c. винаги, освен когато едно от свойствата е по-специфично или правилото за прилагането му взима под внимание кой е предходният елемент
 - d. винаги, освен когато едно от свойствата е по-специфично**
112. Елементите от прост тип в една XML Schema са от даден предефиниран тип и:

- a. **не могат да имат атрибути и под-елементи**
 - b. могат да имат под-елементи, но не и атрибути
 - c. могат да имат атрибути и под-елементи
 - d. могат да имат атрибути, но не и под-елементи
113. В RDFS един ресурс може да е екземпляр на няколко класа.
- a. **Верно**
 - b. Неверно
114. За разлика от SAX при използване на StAX можем да се движим както напред така и назад в XML документа.
- a. Вярно
 - b. **Невярно**
115. XML пространствата от имена се дефинират чрез атрибути и могат да бъдат специфицирани в кой да е XML елемент.
- a. **Вярно**
 - b. Невярно
116. В DOM, неопределена колекция от възли се дефинира чрез интерфейс с име:
- a. UnorderedNodeCollection
 - b. UnorderedNodeSet //не е това
 - c. NodeAttr
 - d. NodeList
 - e. **NamedNodeMap**
117. Използването на пространства от имена в XPointer изрази е:
- a. **Разрешено**
 - b. Забранено
118. Кой два от изброените по-долу атрибути НЕ МОГАТ да участват в елемент от XLink тип arc:
- a. **Role**
 - b. From
 - c. Show
 - d. **Href**
 - e. Title
 - f. Type
 - g. To
 - h. Actuate
119. Методите за работа с атрибути на DOM Element интерфейса:
- a. Разграничават стойността на атрибут дефинирана изрично в XML документа, от стойността по подразбиране, зададена в DTD, само ако Node:setAttrDistinction(Boolean param) е извикан предварително с параметър param със стойност True
 - b. **не могат да разграничават стойността на атрибут, дефиниран изрично в XML документа, от стойността по подразбиране, зададена в DTD**
 - c. могат да разграничават стойността на атрибут, дефинирана изрично в XML документа, от стойността по подразбиране, зададена в DTD

- d. зависи от настройките на DOM парсера
120. При прилагане на XSLT трансформацията
- ```
<xsl:template match="name">
 <xsl:element name="{.}">
 Very nice!
 </xsl:element>
</xsl:template>
```
- за документа
- ```
<names>  
  <name>Bob</name>  
  <name>Steve</name>  
</names>
```
- Имената на създадените елементи за резултатното дърво ще бъдат:
- a. с името "name"
 - b. с имената на елементите в изходящото дърво
 - c. със съдържанието на елементите в изходящото дърво**
 - d. със съдържанието на атрибутите в изходящото дърво
 - e. с имената на атрибутите в изходящото дърво
121. Елементите contributor, creator, publisher и rights са част от: (**има подобен въпрос, но с други възможни отговори**)
- a. DCQ пространството (Dublin Core квалификатори)
 - b. DCM пространството (Dublin Core метаданни)
 - c. DC пространството (Dublin Core ядро)**
 - d. DCT пространството (Dublin Core термини)
122. XLink използва отдалечени (remote) ресурси, за да представя:
- a. нито един от останалите отговори не е верен
 - b. само външни за документа ресурси, но не и вътрешни ресурси, достъпни по URI
 - c. само вътрешни за документа ресурси, достъпни по URI, но не и външни ресурси
 - d. Нито външни за документа ресурси, нито вътрешни ресурси, достъпни по URI
 - e. Както външни за документа ресурси, така и вътрешни ресурси, достъпни по URI**
123. Кой(кои) RDF контейнер(и) задава(т) ресурси или литерали, които могат да бъдат евентуално дублирани:
- a. rdf:Seq и rdf:Alt
 - b. Само rdf:Bag
 - c. rdf:Bag и rdf:Seq**
 - d. rdf:Bag и rdf:Alt
 - e. Само rdf:Seq
 - f. Само rdf:Alt
124. creator е:
- a. елемент-деквалификатор от DC Metadata

- b. един от основните 15 елемента от DC Metadata**
 - c. елемент, който не принадлежи на DC Metadata
 - d. елемент-квалификатор от DC Metadata
- 125. RDF реификация се задава чрез:
 - a. Типа `rdf:Reification` и свойствата `rdf:about`, `rdf:property` и `rdf:object`
 - b. Типа `rdf:Statement` и свойствата `rdf:subject`, `rdf:predicate` и `rdf:object`**
 - c. Типа `rdf:Description` и свойствата `rdf:subject`, `rdf:predicate` и `rdf:object`
 - d. Типа `rdf:Reification` и свойствата `rdf:subject`, `rdf:predicate` и `rdf:object`
 - e. Типа `rdf:Statement` и свойствата `rdf:about`, `rdf:property` и `rdf:object`
 - f. Типа `rdf:Description` и свойствата `rdf:about`, `rdf:property` и `rdf:object`
- 126. Кои от изброените по-долу качества НЕ СА ПРИСЪЩИ на спецификация на онтология:
 - a. Споделена
 - b. Неявна**
 - c. Концептуална
 - d. Машинно-четима
 - e. Неформална**
- 127. В DOM, възли-деца могат да имат единствено възлите от тип:
 - a. Node, Element и Attribute //wrong
 - b. Root, Parent, и Ancestor
 - c. Document, DocumentFragment и Element**
 - d. CharacterData, Text и Comment
 - e. Notation, Entity, EntityReference
- 128. XQUERY изразът


```
For $o in catalog/cd
Let $l := $o/tracklist
Where $l/@num="1"
Return $l/track
```

 Избира:
 - a. Стойността на първия елемент track, който е под-елемент на tracklist, имащ атрибут num равен на 1 и който е наследник на елемент cd
 - b. стойността на всички елементи track, които са под-елементи на tracklist, наследник на елемент cd с атрибут равен на 1
 - c. Нито един от останалите въпроси не е верен
 - d. стойността на всички елементи track, които са под-елементи на tracklist, имащ атрибут num равен на 1 и който е наследник на елемент cd**
 - e. Стойността на първия елемент track, който е под-елемент на tracklist, наследник на елемент cd с атрибут num равен на 1
- 129. RDF литералите могат да се явяват както като субект, така и като обект в RDF тройка
 - a. Вярно
 - b. Невярно**

130. OWL клас, за който са описани както задължителните, така и достатъчните свойства за членство в класа, се нарича:
- a. disjoint клас
 - b. примитивен клас
 - c. анонимен клас
 - d. дефиниран клас**
131. В OWL, един екземпляр може да бъде:
- a. член на множество класове**
 - b. единствено член на само един клас
132. DOMException връща HIERARCHY_REQUEST_ERR при опит за:
- a. Вмъкване на невалиден възел в йерархията на DOM дървото
 - b. заявка за получаване на йерархията на възел с дълбочина, по-голяма от съществуващата на възела
 - c. Вмъкване на възел на неподходящо място в йерархията на DOM дървото**
 - d. Заявка за получаване на йерархията на елемент без наследници
 - e. Заявка за получаване на йерархията на атрибут
133. В SAX 2, управляваме събитията относно съдържанието на документа чрез:
- a. имплементиране на ContentHandler или разширяване на DefaultHandler**
 - b. имплементиране на EventHandler или разширяване на BaseHandler //wrong
 - c. имплементиране на HandlerBase или разширяване на DocumentHandler
 - d. имплементиране на DocumentHandler или разширяване на HandlerBase
 - e. имплементиране на DefaultHandler или разширяване на ContentHandler
134. Кой от следните типове не се среща в описанието на разширена връзка в XLink:
- a. Title
 - b. Locator
 - c. Arc
 - d. Edge**
 - e. resource
135. В RDF един екземпляр може да има няколко типа (т.е. да участва като субект в няколко релации rdf:type).
- a. Неверно
 - b. Верно**
136. В XML схема, уникалността на key и keyref елементите е:
- a. винаги за целия документ
 - b. само за йерархията на съответните елементи в документа-екземпляр, за които са дефинирани key и keyref**
 - c. според стойността на атрибута "context"
 - d. според стойността на атрибута "range"
137. Създаване на специфична метайнформация и схема за употреба с цел предоставяне на възможност за нови методи за достъп до информация се нарича:
- a. Семантично аотиране**
 - b. Семантично разглеждане

- c. Семантично препоръчване
 - d. Автоматично предсказване на потребителски код
 - e. Семантично визуализиране
138. В XSLT, вземането на решение кои елементи ще бъдат обработени се задава със следния XSLT елемент?
- a. <xsl:process-templates>
 - b. <xsl:value-of>
 - c. <xsl:template>
 - d. <xsl:for-each>
 - e. **<xsl:apply-templates>**
139. Литералите в RDF тройките могат да бъдат:
- a. както субект, така и предикат
 - b. **всеки от останалите отговори е грешен**
 - c. както предикат, така и обект
 - d. както субект, така и обект
 - e. както предикат, така и субект или обект
140. В XLink информацията за това как се преминава през двойка ресурси (напр. Посока на преминаване и за поведение при преминаването), се нарича:
- a. Траверс
 - b. Ребро
 - c. **Дъга**
 - d. Ресурс
141. Кое не е цел на RDFa:
- a. намаляване на разликата между интерпретацията на Уеб страницата от потребителя и програмистите
 - b. увеличаване на визуални данни в Уеб страниците в указания за машинно четене
 - c. подобряване на достъпността на Уеб страниците
 - d. **разширяване на XHTML атрибутите**
 - e. вграждане на обогатени метаданни в Уеб документите
142. Речник с документиран процес за актуализация се нарича:
- a. **Контролиран**
 - b. Документиран
 - c. Управляван
 - d. Деклариран
 - e. Менажиран
143. Всеки два OWL класа принципно могат да се припокриват, т.е да имат общи екземпляри:
- a. **Верно**
 - b. Неверно
144. Кой от отговорите по-долу не съдържа име на RDFS клас:
- a. rdfs:Resource
 - b. rdfs:Datatype
 - c. rdfs:ContainerMembershipProperty

- d. `rdfs:Class`
 - e. `rdfs:Container`
 - f. **`rdfs:Property`**
145. Всички членове на подклас OWL са членова на супер-класовете от този клас:
- a. **Вярно**
 - b. Невярно
146. Атрибутът `xml:base` задава:
- a. **база на относителни URI връзки към външни за документа ресурси**
 - b. база за задаване на други мета-атрибути
 - c. базов URI за дефиниране на пространство от имена
 - d. база за сливане на XML документи
147. В RDFS, `Property` е подмножество на `RDFS Resources` и има за обхват (`rdfs:range`)
- a. Типа на стойностите на това `Property`
 - b. **Множество от отделни стойности на това `Property`**
 - c. Класа `rdfs:Resource`, на който принадлежат всички екземпляри на ресурса
 - d. Класа, асоцииран с това `Property`
148. XML Reader в SAX 2.0 разширява стандартния Java Reader интерфейс:
- a. Вярно
 - b. **Невярно**
149. SAX служи както за четене на XML документи, така и за генериране на XML.
- a. Вярно
 - b. **Невярно**
150. Ако свойство `P` е за субекта `S`, който участва в RDF тройката `S P O`, и имаме `P rdfs:domain C`, то следва, че:
- a. **`S rdf:type C`**
 - b. `S rdf:type P`
 - c. `C rdf:type S`
 - d. `P rdf:type S`
 - e. `P rdf:type C`
151. Изберете едно:
- a. Както SAX, така и StAX използват pull парсване
 - b. **SAX използва push парсване, а StAX - парсване от тип pull**
 - c. Както SAX, така и StAX използват push парсване
 - d. SAX използва pull парсване, а StAX - парсване от тип push
152. Как можем да потиснем запазването на допълнителни (следващи един след друг) празни интервали в съдържанието на текстови елемент?
- a. посредством псевдо-атрибута `xml:lang`
 - b. посредством псевдо-атрибута `xml:interval`
 - c. посредством псевдо-атрибута `xml:blank`
 - d. **посредством използване на друг псевдо-атрибут, различен от изброените в останалите отговори на въпроса**

- e. посредством псевдо-атрибута `xml:empty_space`
 - f. запазването на допълнителни (следващи един след друг) празни интервали в съдържанието на текстови елемент не може да се потиска
153. В RDF, класовете могат да бъдат екземпляри на други класове.
- a. **Вярно**
 - b. Невярно
154. Според XML синтаксиса, за всеки непразен XML елемент, за отварящия му таг (маркер) трябва да има:
- a. Един или повече затварящи тагове
 - b. Един затварящ или незатварящ таг
 - c. **Точно един затварящ таг**
 - d. Повече от един затварящи тагове
155. DTD валидацията ограничава елементите и атрибутите, които могат да участват в документа, само по техните локални имена, ако в DTD дефиницията на елемента или атрибута той е участвал с локално име вместо с QName.
- a. **Вярно**
 - b. Невярно
156. Най-общо маркъп езиците включват:
- a. структурни (layout), функционални (action), семантични (meaning) но не и стилистични (appearance) описания
 - b. Стилистични (appearance), структурни (layout), семантични (meaning) но не и функционални (action) описания
 - c. Стилистични (appearance), структурни (layout), функционални (action) но не и семантични (meaning) описания
 - d. **Структурни (layout), функционални (action), семантични (meaning) и стилистични (appearance) описания**
157. Свойството `rdfs:member` е супер-свойство от всички свойства, които са екземпляри на:
- a. `rdfs:ContainerProperty`
 - b. `rdfs:MemberOfProperty`
 - c. **`rdfs:ContainerMembershipProperty`**
 - d. `rdfs:MembershipProperty`
 - e. `rdfs:MemberProperty`
158. Символите `<`, `>` и `'` са валидни PCDATA символи
- a. **Невярно**
 - b. Вярно
159. XML документите могат да съдържат само един елемент-корен
- a. **Верно**
 - b. Неверно
160. Изпълнението на XSLT декларациите `<xsl:value-of select="."/>` и `<xsl:value-of select="text()"/>` води:
- a. Винаги до различни резултати
 - b. **До един и същ или до различни резултати в зависимост от типа на съдържанието на текущия елемент**

- c. Винаги до един и същ резултат
161. XPath изразът `./book[author/last="пробен изпит"]` връща:
- Всички book елементи, които имат елемент author с атрибут last равен на "пробен изпит"
 - Елемент last със стойност "пробен изпит", който има баща елемент author с поделемент book - наследник на текущия елемент
 - Всички book елементи, които имат елемент author с поделемент last равен на "пробен изпит"
 - Всички last елементи със стойност "пробен изпит", които имат за баща елемент author с поделемент book
 - Елемент book - наследник на текущия елемент, който има елемент author с поделемент last равен на "пробен изпит"**
162. Изберете верния отговор:
- Всеки добре конструиран XML документ е валиден, но обратното не е задължително вярно.
 - Всеки валиден XML документ е добре конструиран, както и обратното - всеки добре конструиран XML документ е валиден.
 - Всички останали твърдения са неверни.
 - Всеки валиден XML документ е добре конструиран, но обратното не е задължително вярно.**
163. XML йерархията `<a> <c> </c> ` е:
- Коректна или некоректна в зависимост от DTD/XSchema валидацията
 - Нито един от останалите отговори не е верен
 - Некоректна**
 - Коректна
164. Следният XML документ е добре конструиран (well-formed):
- ```
<?xml version='1.0'?>
<!--http://www.w3.org is bound to both ref1 and default namespace-->
<myDoc xmlns:ref1="http://www.w3.org"
xmlns="http://www.w3.org"
xmlns:ref2="http://www.hmmm.bg">
<myElem a="1" b="2" />
<myElem a="1" ref1:a="2" />
<myElem ref1:a="1" ref2:a="2" />
```
- Истина**
  - Неистина
165. XML пространството от имена се отнася:
- Само за елемента, за който е дефинирано
  - Само за йерархията от елементи, за която е дефинирано**
  - Само за елемента, за който е дефинирано, както и за неговите атрибути
  - За всички оставащи от мястото на дефинирането му до края на документа елементи
166. Събитието processingInstruction възниква, когато SAX парсерът достигне до всяка една инструкция за обработка, включително и до XML декларацията

- a. Вярно
  - b. Невярно**
167. Отдалечените XLink ресурси винаги представляват външни за документа ресурси.
- a. Верно
  - b. Неверно**
- 168.