1. \*\*Which XPath expression selects the root node in an XML document?\*\*

a) `/root`

b) **`//`**

c) `./`

d) `/`

2. \*\*What does the XPath expression `//book` do?\*\*

a) Selects all `book` elements with an attribute named `book`

b) Selects all `book` elements that are children of the current node

c) **Selects all `book` elements regardless of their position in the document**

d) Selects the first `book` element

3. \*\*Which XPath function is used to count the number of nodes in a selected node set?\*\*

a) `size()`

b) **`count()`**

c) `length()`

d) `number()`

4. \*\*If an element `<price>` is a child of an element `<book>`, what would the XPath expression `/catalog/book/price` select?\*\*

a) All `price` elements in the XML document

b) The first `price` element found inside the `book` node

c) **All `price` elements that are children of any `book` element under the root `catalog`**

d) Only the first `price` element inside the first `book` element under `catalog`

5. \*\*How would you select all `book` elements that have an attribute `lang` with a value of "en"?\*\*

a) `/book[@lang='en']`

b) `//book[lang='en']`

c) **`//book[@lang='en']`**

d) `/book/@lang='en'`

6. \*\*What does the `position()` function return in an XPath expression?\*\*

a) The position of the current node among its siblings

b) The position of the current node in the whole document

c) **The position of the current attribute node**

d) The total count of nodes in the document

7. \*\*Which XPath expression selects all text nodes directly under a given node?\*\*

a) **`node()/text()`**

b) `./text()`

c) `./node()`

d) `text()/node()`

8. \*\*What will the expression `ancestor::book` select?\*\*

a) **All ancestor nodes of type `book` of the current node**

b) All nodes of type `book` that are children of the current node

c) The first `book` node in the document

d) All `book` nodes that are siblings of the current node

9. \*\*Which XPath expression will select all elements with a `price` greater than 10?\*\*

a) `//price[text() > 10]`

b) `//price[number() > 10]`

c) **`//price[count() > 10]`**

d) `//price[. > 10]`

10. \*\*In an XML structure, which expression would select the parent node of the current element?\*\*

a) `ancestor::`

b) `../`

c) **`parent()`**

d) `preceding-sibling::`

11. \*\*What would `child::\*` select in an XPath expression?\*\*

a) All attributes of the current node

b) All sibling elements of the current node

c) **All child elements of the current node**

d) All descendants of the current node

12. \*\*Which function helps identify if a node has a particular attribute?\*\*

a) `attribute()`

b) **`has()`**

c) `exist()`

d) `boolean()`

13. \*\*Which axis is used to select nodes that appear before the context node in document order but are not ancestors?\*\*

a) **`preceding::`**

b) `following::`

c) `preceding-sibling::`

d) `following-sibling::`

14. \*\*How would you select nodes that match either `title` or `author` elements directly under the `book` node?\*\*

a) **`/book/title | /book/author`**

b) `//book/title and //book/author`

c) `/book/title or /book/author`

d) `book/title or book/author`

15. \*\*What would be the result of the following expression: `//book[count(\*) > 3]`?\*\*

a) All `book` nodes that have more than three child nodes

b) All `book` nodes that have more than three attributes

c) All `book` nodes with children named `count`

d) **An error message**