**Примерен вход за програмата XMLParser и очакван изход:**

* -с този знак се означава input
* -с този знак се означава очакван output

!Този пример използва три файла: example1.xml, example2.xml и example3.xml, които се намират в build папката!

Welcome to XML Parser app!

Please enter a valid command or type "help" for more information.

1. Програмата работи с example1.xml:

* Open example1.xml
* File example1.xml successfully opened!
* Print
* Изпринтен файл.
* Select 3 genre
* comedy
* Set 1 category adult
* You have successfully changed the value of the attribute with id: 1.
* Xpath
* Supported XPath Queries:

1. The / operator: (<>/<>) returns a list of all elements of <> that are child elements of <>, for each <> element in the file.

2. The [] operator: (<>/<>[index]) returns an element of <> that is a child of <> for each element of <> in the file.

3. The @ operator: (<>[@<>]) returns a list of <> attributes of all <> elements in the file.

4. The = operator: (<>[<>=]/<>) returns a list of all names (contents of child element <>) of elements <> in the file that have a child element <> with text content " ".

Enter the number of the query you wish to execute:

* 3
* Enter the specific query:
* book[@category]
* [category=adult]

[category=cooking]

* Save
* Запазва промените в същия файл
* Close
* File example1.xml successfully closed!

Do you want to open another file or exit the program?

Choose between continue or exit:

* Continue
* Enter the file path:
* example2.xml
* Opening another file...

File example2.xml successfully opened!

1. Програмата работи с example2.xml:

* Help
* Извежда всички операции, които програмата поддържа.
* Children 1
* Children's attributes:

(id=1\_1)

(nationality=american)

(id=1\_2)

(nationality=bulgarian)

(vegan=true)

(id=7)

(nationality=australian)

* Child 7 2
* Child 2 of 7: job
* Delete 1\_2 vegan
* You have successfully deleted the attribute!
* Xpath
* Supported XPath Queries:

1. The / operator: (<>/<>) returns a list of all elements of <> that are child elements of <>, for each <> element in the file.

2. The [] operator: (<>/<>[index]) returns an element of <> that is a child of <> for each element of <> in the file.

3. The @ operator: (<>[@<>]) returns a list of <> attributes of all <> elements in the file.

4. The = operator: (<>[<>=]/<>) returns a list of all names (contents of child element <>) of elements <> in the file that have a child element <> with text content " ".

Enter the number of the query you wish to execute:

* 2
* Enter the specific query:
* person/address[1]
* [USA]

[Varna,Bulgaria]

[Varna,Bulgaria]

* Xpath
* Supported XPath Queries:

1. The / operator: (<>/<>) returns a list of all elements of <> that are child elements of <>, for each <> element in the file.

2. The [] operator: (<>/<>[index]) returns an element of <> that is a child of <> for each element of <> in the file.

3. The @ operator: (<>[@<>]) returns a list of <> attributes of all <> elements in the file.

4. The = operator: (<>[<>=]/<>) returns a list of all names (contents of child element <>) of elements <> in the file that have a child element <> with text content " ".

Enter the number of the query you wish to execute:

* 4
* Enter the specific query:
* person[address=”Varna,Bulgaria”]/name
* John Smith

Ivan Petrov

Emily Johnson

* Saveas newExample2.xml
* File newExample2.xml successfully saved!
* Close
* File example2.xml successfully closed!

Do you want to open another file or exit the program?

Choose between continue or exit:

* Continue
* Enter the file path:
* example3.xml
* Opening another file...

File example3.xml successfully opened!

1. Програмата работи с example3.xml:

* Text 1\_1
* name's text: Simba
* NewChild 2
* Enter the name of the new successor:
* Kg
* You have successfully added a new child!
* Xpath
* Supported XPath Queries:

1. The / operator: (<>/<>) returns a list of all elements of <> that are child elements of <>, for each <> element in the file.

2. The [] operator: (<>/<>[index]) returns an element of <> that is a child of <> for each element of <> in the file.

3. The @ operator: (<>[@<>]) returns a list of <> attributes of all <> elements in the file.

4. The = operator: (<>[<>=]/<>) returns a list of all names (contents of child element <>) of elements <> in the file that have a child element <> with text content " ".

Enter the number of the query you wish to execute:

* 1
* Enter the specific query:
* animal/name
* [Simba]

[Dumbo]

[Charlie]

* Exit
* Exiting the program...