

SOAP documentation

Operation: addAuthor

Description: Adds a new author

Input: 1) Name: token
 Type: string

 2) Name: author
 Type: author

Example request:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:soap="http://SOAP.app/">
  <soapenv:Header/>
  <soapenv:Body>
    <soap:addAuthor>
      <token>12345</token>
      <author>
        <!--Optional:-->
        <firstName>Joonas</firstName>
        <id>5</id>
        <!--Optional:-->
        <lastName>Lume</lastName>
      </author>
    </soap:addAuthor>
  </soapenv:Body>
</soapenv:Envelope>
```

Output: Name: author
 Type: author

Example response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:addAuthorResponse xmlns:ns2="http://SOAP.app/">
      <return>
        <firstName>Joonas</firstName>
        <id>5</id>
        <lastName>Lume</lastName>
      </return>
    </ns2:addAuthorResponse>
  </S:Body>
</S:Envelope>
```

Operation: getAllAuthors

Description: Gets the list of all authors

Input: Name: token
 Type: string

Example request:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:soap="http://SOAP.app/">
  <soapenv:Header/>
  <soapenv:Body>
    <soap:getAllAuthors>
      <token>12345</token>
    </soap:getAllAuthors>
  </soapenv:Body>
</soapenv:Envelope>
```

Output: Name: author
 Type: author

Example response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:getAllAuthorsResponse xmlns:ns2="http://SOAP.app/">
      <return>
        <firstName>Mati</firstName>
        <id>1</id>
        <lastName>Maasikas</lastName>
      </return>
      <return>
        <firstName>Mari</firstName>
        <id>2</id>
        <lastName>Mustikas</lastName>
      </return>
      <return>
        <firstName>Mart</firstName>
        <id>3</id>
        <lastName>Mardikas</lastName>
      </return>
    </ns2:getAllAuthorsResponse>
  </S:Body>
</S:Envelope>
```

Operation: getAuthor

Description: Gets an author with the specified id

Input: 1) Name: token
 Type: string

 2) Name: id
 Type: string

Example request:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:soap="http://SOAP.app/">
  <soapenv:Header/>
  <soapenv:Body>
    <soap:getAuthor>
      <token>12345</token>
      <id>1</id>
    </soap:getAuthor>
  </soapenv:Body>
</soapenv:Envelope>
```

Output: Name: author
 Type: author

Example response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:getAuthorResponse xmlns:ns2="http://SOAP.app/">
      <return>
        <firstName>Mati</firstName>
        <id>1</id>
        <lastName>Maasikas</lastName>
      </return>
    </ns2:getAuthorResponse>
  </S:Body>
</S:Envelope>
```

Operation: addBook

Description: Adds a new book

Input: 1) Name: token
 Type: string

 2) Name: book
 Type: book

Example request:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:soap="http://SOAP.app/">
  <soapenv:Header/>
  <soapenv:Body>
    <soap:addBook>
      <token>?</token>
      <book>
        <!--Zero or more repetitions:-->
        <author>
          <!--Optional:-->
          <firstName>Madis</firstName>
          <id>20</id>
          <!--Optional:-->
          <lastName>Mäe</lastName>
        </author>
        <id>10</id>
        <!--Optional:-->
        <publisher>Noone publishing</publisher>
        <!--Optional:-->
        <releaseDate>?</releaseDate>
        <!--Optional:-->
        <title>Test Book</title>
      </book>
    </soap:addBook>
  </soapenv:Body>
</soapenv:Envelope>
```

Output: Name: book
 Type: book

Example response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:addBookResponse xmlns:ns2="http://SOAP.app/">
      <return>
        <author>
          <firstName>Madis</firstName>
          <id>20</id>
          <lastName>Mäe</lastName>
        </author>
        <id>10</id>
        <publisher>Noone publishing</publisher>
        <title>Test Book</title>
      </return>
    </ns2:addBookResponse>
  </S:Body>
</S:Envelope>
```

Operation: getAllBooks

Description: Obtains all books with the specified filter

Input:	1)	Name:	token
		Type:	string
	2)	Name:	title
		Type:	string
	3)	Name:	author
		Type:	string
	4)	Name:	publisher
		Type:	string

Example request:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:soap="http://SOAP.app/">
  <soapenv:Header/>
  <soapenv:Body>
    <soap:getAllBooks>
      <token>12345</token>
      <!--Optional:-->
      <title>Programmeerimine keeles Scratch</title>
      <author>1</author>
      <!--Optional:-->
      <publisher>Koolibri</publisher>
    </soap:getAllBooks>
  </soapenv:Body>
</soapenv:Envelope>
```

Output: Name: book
 Type: book

Example response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:getAllBooksResponse xmlns:ns2="http://SOAP.app/">
      <return>
        <author>
          <firstName>Mati</firstName>
          <id>1</id>
          <lastName>Maasikas</lastName>
        </author>
        <id>1</id>
        <publisher>Koolibri</publisher>
        <releaseDate>1818-11-11T00:00:00+02:00</releaseDate>
        <title>Programmeerimine keeles Scratch</title>
      </return>
    </ns2:getAllBooksResponse>
  </S:Body>
</S:Envelope>
```

Operation: getBook

Description: Obtains a book with the specific id

Input: 1) Name: token
 Type: string

 2) Name: id
 Type: string

Example request:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:soap="http://SOAP.app/">
  <soapenv:Header/>
  <soapenv:Body>
    <soap:getBook>
      <token>12345</token>
      <id>1</id>
    </soap:getBook>
  </soapenv:Body>
</soapenv:Envelope>
```

Output: Name: book
 Type: book

Example response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:getBookResponse xmlns:ns2="http://SOAP.app/">
      <return>
        <author>
          <firstName>Mati</firstName>
          <id>1</id>
          <lastName>Maasikas</lastName>
        </author>
        <id>1</id>
        <publisher>Koolibri</publisher>
        <releaseDate>1818-11-11T00:00:00+02:00</releaseDate>
        <title>Programmeerimine keeles Scratch</title>
      </return>
    </ns2:getBookResponse>
  </S:Body>
</S:Envelope>
```


Operation: getBookWithSpecificAuthor

Description: Obtains a book with the specific author(concat first and lastname)

Input: 1) Name: token
 Type: string

 2) Name: author
 Type: string

Example request:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:soap="http://SOAP.app/">
  <soapenv:Header/>
  <soapenv:Body>
    <soap:getBooksWithSpecificAuthor>
      <token>12345</token>
      <author>Mati Maasikas</author>
    </soap:getBooksWithSpecificAuthor>
  </soapenv:Body>
</soapenv:Envelope>
```

Output: Name: book
 Type: book

Example response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:getBooksWithSpecificAuthorResponse
xmlns:ns2="http://SOAP.app/">
      <return>
        <author>
          <firstName>Mati</firstName>
          <id>1</id>
          <lastName>Maasikas</lastName>
        </author>
        <id>1</id>
        <publisher>Koolibri</publisher>
        <releaseDate>1818-11-11T00:00:00+02:00</releaseDate>
        <title>Programmeerimine keeles Scratch</title>
      </return>
    </ns2:getBooksWithSpecificAuthorResponse>
  </S:Body>
</S:Envelope>
```