

Documentation de l'api REST

Table des matières

1.	Les Services.....	2
1.1.	Ressource book	2
1.1.1.	POST post book	2
1.1.2.	GET get book	2
1.1.3.	DELETE delete book.....	2
1.1.4.	PUT put book.....	2
1.1.5.	GET get books.....	3
1.2.	Ressource library	3
1.2.1.	POST post library	3
1.2.2.	GET get library	3
1.2.3.	DELETE delete library.....	3
1.2.4.	PUT put library.....	3
1.2.5.	GET get libraries	4
1.3.	Ressource liées	4
1.3.1.	POST post biblio/id/book	4
1.3.2.	GET get biblio/1/book	4
1.3.3.	DELETE delete biblio/id/book.....	4
1.3.4.	PUT put biblio/id/book.....	4
1.3.5.	GET get biblio/1/books.....	5
2.	Les Erreurs	5
2.1.	GET Error type	5
2.2.	GET Error no collection book for null library	5
2.3.	GET Error can't update, delete, create or get book on null library	5
2.4.	GET Error Book is not existent in library	5
2.5.	OPTIONS Error status 405	5
2.6.	GET Error Cannot return, delete, update a non existent library	5
2.7.	GET Error Cannot return, delete, update a non existent book	6
2.8.	POST Error Cannot attach a book to a non existent library	6
2.9.	POST Error status 400	6

1. Les Services

1.1. Ressource book

1.1.1. POST post book

127.0.0.1:8081/tpwsrest/book?name=test&releaseDate=2017-10-09
15:00:39.019&isbn=x0001&author=toto&library.id=1

Insert a new book in data base.

PARAMS

name : nom du livre
releaseDate : date de publication
isbn : isbn du livre
author : auteur du livre
library.id : id de la bibliothèque associé

1.1.2. GET get book

127.0.0.1:8081/tpwsrest/book?id=1

Select the book with id in param.

PARAMS

id : id du book

1.1.3. DELETE delete book

127.0.0.1:8081/tpwsrest/book?id=2

Delete the book with id in param. Return string for confirm.

PARAMS

id : id du book

1.1.4. PUT put book

127.0.0.1:8081/tpwsrest/book?name=test&releaseDate=2017-10-09
15:00:39.019&isbn=x0001&author=toto&id=1

Update a book in data base. Return the book in xml Format.

PARAMS

name : nom du livre
releaseDate : date de publication

isbn : isbn du livre
author : toto
id : id du livre

1.1.5. GET get books
127.0.0.1:8081/tpwsrest/books

Select a book collection.

1.2. Ressource library

1.2.1. POST post library
127.0.0.1:8081/tpwsrest/library?name=test&address=test&yearCreated=1958

Insert a new library in data base.

PARAMS

name
test
address
test
yearCreated
1958

1.2.2. GET get library
127.0.0.1:8081/tpwsrest/library?id=2

Select the library with id in param. return the library.

PARAMS

id : id de la bibliothèque

1.2.3. DELETE delete library
127.0.0.1:8081/tpwsrest/library?id=2

delete the library with id in param. Return a string for confirm.

PARAMS

id : id de la bibliothèque

1.2.4. PUT put library
127.0.0.1:8081/tpwsrest/library?name=test&address=test&yearCreated=1958&id=1

update the library with id in param. return the library in json format.

PARAMS

name : nom de la bibliothèque

address : adresse de la bibliothèque
yearCreated : année de création
id : id de la bibliothèque

1.2.5. GET get libraries
127.0.0.1:8081/tpwsrest/libraries

Select a library collection.

1.3. Ressource liées

1.3.1. POST post biblio/id/book
127.0.0.1:8081/tpwsrest/biblio/1/book?name=test&releaseDate=2017-10-09
15:00:39.019&isbn=x0001&author=toto

Insert a new book for the library with id param in data base. Return the book in json format.

PARAMS

name : nom de la bibliothèque
releaseDate : date de publication
isbn : isbn du livre
author : auteur du livre

1.3.2. GET get biblio/1/book
http://localhost:8081/tpwsrest/biblio/1/book?book.id=1

Get the book with book.id param in the library with id param.

PARAMS

book.id : id du livre

1.3.3. DELETE delete biblio/id/book
127.0.0.1:8081/tpwsrest/biblio/1/book?book.id=3

Delete the book with book.id in the library with id param. Return a string for confirm.

PARAMS

book.id : id du book

1.3.4. PUT put biblio/id/book
127.0.0.1:8081/tpwsrest/biblio/1/book?name=test&releaseDate=2017-10-09
15:00:39.019&isbn=x0001&author=toto&book.id=1

Update the book with book.id param in the library with id param.

PARAMS

name : nom de la bibliothèque
releaseDate : date de publication
isbn : isbn du livre
author : auteur du livre
book.id : id du book

1.3.5. GET get biblio/1/books

<http://localhost:8081/tpwsrest/biblio/1/books>

Select book collection in the library with id param.

2. Les Erreurs

2.1. GET Error type

<http://localhost:8081/tpwsrest/biblio/1/books>

Return status 400 for other type than xml or json in header accept. -> Methods get, post, put for url biblio/id/book, biblio.id/books, book, books, library and libraries.

2.2. GET Error no collection book for null library

<http://localhost:8081/tpwsrest/biblio/4/books>

Return status 400. Method GET url biblio/id/books

2.3. GET Error can't update, delete, create or get book on null library

<http://localhost:8081/tpwsrest/biblio/4/book>

Return status 400. Methods GET, PUT, DELETE, POST url biblio/id/book

2.4. GET Error Book is not existent in library

<http://localhost:8081/tpwsrest/biblio/1/book/?book.id=82>

Return status 400. Methods GET, UPDATE, DELETE url biblio/id/book

2.5. OPTIONS Error status 405

<http://localhost:8081/tpwsrest/books>

No methods for url book, books, library, libraries, biblio/id/books, biblio/id/book

2.6. GET Error Cannot return, delete, update a non existent library

<http://localhost:8081/tpwsrest/library?id=84>

Return status 400. Methods GET, DELETE, PUT url library

2.7. GET Error Cannot return, delete, update a non existent book

<http://localhost:8081/tpwsrest/book?id=84>

Return status 400. Methods GET, DELETE, PUT url book

2.8. POST Error Cannot attach a book to a non existent library

<http://localhost:8081/tpwsrest/book?library.id=84>

Return status 400. Method POST url book

2.9. POST Error status 400

<http://localhost:8081/tpwsrest/book?library.id=1>

Less informations for Methods POST UPDATE url book, library, bilbio/id/book