



Swagger UI interface showing the API endpoint `GET /reader/{id}` (reader by id). The description is "Вывод инфо о читателе по его id-номеру".

Parameters

Name	Description
id * required	Integer(int64) (path)

Responses

Code	Description	Links
200	OK	No links

Media type: `*/*`

Example Value:

```
{  "код читателя": 0,  "имя читателя": "string"}
```

Swagger UI interface showing the API endpoint `GET /reader/all` (all readers). The description is "Вывод списка всех читателей".

Parameters

No parameters

Responses

Code	Description	Links
200	OK	No links

Media type: `*/*`

Example Value:

```
{  "код читателя": 0,  "имя читателя": "string"}
```

DELETE /reader/del/{id} delete reader by id

Удаление записи о читателе по его id-номеру

Parameters

Name	Description
------	-------------

Swagger UI interface showing API documentation for a book management system. The browser address bar shows `localhost:8180/swagger-ui/index.html#/Book/deleteBook`.

DELETE /reader/del/{id} delete reader by id

Удаление записи о читателе по его id-номеру

Parameters

Name	Description
id * required Integer(\$int64) (path)	id

Responses

Code	Description	Links
200	OK	No links

Issue

POST /issue/add add issue

Добавление выдачи в список выдач

Swagger UI interface showing API documentation for a book management system. The browser address bar shows `localhost:8180/swagger-ui/index.html#/Book/deleteBook`.

DELETE /reader/del/{id} delete reader by id

Удаление записи о читателе по его id-номеру

Parameters

Name	Description
id * required Integer(\$int64) (path)	id

Responses

Code	Description	Links
200	OK	No links

Issue

POST /issue/add add issue

Добавление выдачи в список выдач

Swagger UI interface showing the POST endpoint `/issue/add` for adding a new issue.

POST `/issue/add` add issue

Добавление выдачи в список выдач

Parameters

No parameters

Request body *required* application/json

Example Value | Schema

```
{  "Код книги": 0,  "Код читателя": 0,  "Дата и время выдачи": "2024-07-12T09:44:02.605Z"}
```

Responses

Code	Description	Links
200	OK	No links

Media type: */

Controls accept header

Example Value | Schema

```
{  "Код выдачи": 0,  "Код книги": 0,  "Код читателя": 0,  "Дата выдачи": "2024-07-12T09:44:02.606Z",  "Дата возврата": "2024-07-12T09:44:02.606Z"}
```

Swagger UI interface showing the GET endpoint `/issue/{id}` for retrieving an issue by ID.

GET `/issue/{id}` issue by id

Вывод выдачи по ее ID-номеру

Parameters

Name Description

id *required* id

Integer (int64) (path)

Responses

Code	Description	Links
200	OK	No links

Media type: */

Controls Accept header

Example Value | Schema

```
{  "Код выдачи": 0,  "Код книги": 0,  "Код читателя": 0,  "Дата выдачи": "2024-07-12T09:44:02.607Z",  "Дата возврата": "2024-07-12T09:44:02.607Z"}
```

Swagger UI interface showing the API endpoint `/issue/all` (GET) and `/issue/del/{id}` (DELETE).

GET /issue/all all issues

Вывод списка всех выдач

Parameters

No parameters

Responses

Code	Description	Links
200	OK	No links

Media type: `*/*`

Example Value / Schema:

```
{  "Код выдачи": 0,  "Код книги": 0,  "Код читателя": 0,  "Дата выдачи": "2024-07-12T09:44:02.608Z",  "Дата возврата": "2024-07-12T09:44:02.608Z"}
```

DELETE /issue/del/{id} delete issue by id

Удаление записи о выдаче по ее id-номеру

Parameters

Try it out

Swagger UI interface showing the API endpoint `/issue/del/{id}` (DELETE) and `/book/add` (POST).

DELETE /issue/del/{id} delete issue by id

Удаление записи о выдаче по ее id-номеру

Parameters

Try it out

Responses

Code	Description	Links
200	OK	No links

Book

POST /book/add add book

Добавление книги в хранилище

Parameters

No parameters

Request body *required*

application/json

Try it out

Swagger UI interface showing the API endpoint **POST /book/add** (add book). The description is "Добавление книги в хранилище".

Parameters: No parameters.

Request body: Required, application/json. Example Value:

```
{  "Идентификатор": 0,  "Название книги": "string"}
```

Responses:

Code	Description	Links
200	OK	No links

Media type: */*
Controls: Accept header
Example Value: Schema

```
{  "Идентификатор": 0,  "Название книги": "string"}
```

Below this endpoint, the **GET /book** endpoint is visible, with description "Загружает список всех книг, которые есть в системе".

Swagger UI interface showing the API endpoint **GET /book** (get all books). The description is "Загружает список всех книг, которые есть в системе".

Parameters: No parameters.

Responses:

Code	Description	Links
200	OK	No links

Media type: */*
Controls: Accept header
Example Value: Schema

```
[  {    "Идентификатор": 0,    "Название книги": "string"  }]
```

Below this endpoint, the **GET /book/{id}** endpoint is visible, with description "Выбор книги по ее id-номеру".

Swagger UI interface showing the GET endpoint `/book/{id}` for retrieving a book by ID. The endpoint is described as "Выбор книги по ее id-номеру".

Parameters:

Name	Description
id * required Integer(\$int64) (path)	id

Responses:

Code	Description	Links
200	OK Media type: <code>*/*</code> Control: accept header Example Value: <pre>{ "Идентификатор": 0, "Название книги": "string"}</pre>	No links

DELETE endpoint: `/book/delete/{id}` delete book. Description: "Удаление книги из хранилища".

Swagger UI interface showing the DELETE endpoint `/book/delete/{id}` for deleting a book by ID. The endpoint is described as "Удаление книги из хранилища".

Parameters:

Name	Description
id * required Integer(\$int64) (path)	id

Responses:

Code	Description	Links
200	OK	No links

Schemas:

- Читатели** {
Код читателя Integer(\$int64)
Имя читателя string
}
- Запрос на выдачу** {
Код книги Integer(\$int64)
Код читателя Integer(\$int64)
Дата и время выдачи string(\$date-time)
}

The screenshot shows a web browser window with the Swagger UI for a local API. The browser address bar shows 'localhost:8180/swagger-ui/index.html#/Book/deleteBook'. The Swagger UI displays a 'deleteBook' endpoint with a response status of 200 OK. Below the endpoint, the 'Schemas' section lists the data models: Читатели (Readers), Запрос на выдачу (Request for issue), Выдачи (Issues), and Книги (Books).