

E-legion academy API

Общая информация:

На любой запрос, требующий авторизации, сервер может вернуть статус 401 Unauthorized. Это значит, что в заголовке не был указан token, или он недействителен.

Токен имеет неограниченное время жизни. Он инвалидируется только в случае запроса `signin`.

На любой запрос можно получить статус 400 Bad Request. Это значит, что запрос имеет некорректный формат. Например, необходимые параметры отсутствуют или имеют недопустимые значения.

Параметры запроса передаются одним из четырех способов:

- В пути запроса. Например, `data/{id}`. Такие параметры имеют тип `path`.
- В параметрах запроса. Например, `data?param1=value1¶m2=value2`. Такие параметры имеют тип `query`.
- В заголовке запроса. Такие параметры имеют тип `header`. Например:

```
POST /data HTTP/1.1
Param1: value1
```
- В теле запроса. Такие параметры передаются в формате JSON и имеют тип `body`. В скобках рядом с ним указан тип данных. Пример JSON:

```
{
  "param1": "value1"
  "param2": 342
}
```

Если запрос содержит JSON, то у него должен быть проставлен соответствующий `Content-Type` в заголовке.

Изображения для новых публикаций должны иметь формат `jpg` и отправлены в виде `base64` строки.

Все параметры обязательны если иное не указано явно. Логин для авторизации `"user"`, пароль `"qwerty"`. Без кавычек. Адрес сервера `http://localhost:8080`.

POST signin

Авторизует пользователя и выдает токен.

Request:

name	description	parameter type
login	Логин пользователя	body (String)
password	Пароль пользователя	body (String)

Response:

Запрос выполнен успешно:

Status: 200

Header:

Body:

```
{
  "token": "string"
}
```

Неверный логин и пароль:

Status: 422

Header:

Body:

POST signout

Деавторизует пользователя и инвалидирует токен.

Request:

name	description	parameter type
token	Токен текущего пользователя	header

Response:

Запрос выполнен успешно:

Status: 200

Header:

Body:

GET checkToken

Проверяет валидность токены.

Request:

name	description	parameter type
token	Токен текущего пользователя	header

Response:

Токен валиден:

Status: 200

Header:

Body:

GET users/me

Возвращает информацию о текущем пользователе.

Request:

name	description	parameter type
token	Токен текущего пользователя	header

Response:

Запрос выполнен успешно:

Status: 200

Header:

Body:

```
{
  "id": "string",
  "username": "string",
  "fullName": "string",
  "avatar": "URL string",
  "currentUserFollowsThisUser": bool,
  "currentUserIsFollowedByThisUser": bool,
  "followsCount": number,
  "followedByCount": number
}
```

GET users/{userID}

Возвращает информацию о пользователе с запрошенным ID.

Request:

name	description	parameter type
token	Токен текущего пользователя	header
userID	ID нужного пользователя	path

Response:

Запрос выполнен успешно:

Status: 200

Header:

Body:

```
{
  "id": "string",
  "username": "string",
  "fullName": "string",
  "avatar": "URL string",
  "currentUserFollowsThisUser": bool,
  "currentUserIsFollowedByThisUser": bool,
  "followsCount": number,
  "followedByCount": number
}
```

Пользователь с запрошенным ID не найден:

Status: 404

Header:

Body:

POST users/follow

Подписывает текущего пользователя на пользователя с запрошенным ID.

Request:

name	description	parameter type
token	Токен текущего пользователя	header
userID	ID нужного пользователя	body (String)

Response:

Пользователь успешно подписан или уже является подписчиком:

Status: 200

Header:

Body:

```
{
  "id": "string",
  "username": "string",
  "fullName": "string",
  "avatar": "URL string",
  "currentUserFollowsThisUser": bool,
  "currentUserIsFollowedByThisUser": bool,
  "followsCount": number,
  "followedByCount": number
}
```

Пользователь с запрошенным ID не найден:

Status: 404

Header:

Body:

Попытка подписаться на самого себя:

Status: 406

Header:

Body:

POST users/unfollow

Отписывает текущего пользователя от пользователя с запрошенным ID.

Request:

name	description	parameter type
token	Токен текущего пользователя	header
userID	ID нужного пользователя	body (String)

Response:

Пользователь успешно отписан или уже не является подписчиком:

Status: 200

Header:

Body:

```
{
  "id": "string",
  "username": "string",
  "fullName": "string",
  "avatar": "URL string",
  "currentUserFollowsThisUser": bool,
  "currentUserIsFollowedByThisUser": bool,
  "followsCount": number,
  "followedByCount": number
}
```

Пользователь с запрошенным ID не найден:

Status: 404

Header:

Body:

Попытка отписаться от самого себя:

Status: 406

Header:

Body:

GET users/{userID}/followers

Возвращает подписчиков пользователя с запрошенным ID.

Request:

name	description	parameter type
token	Токен текущего пользователя	header
userID	ID нужного пользователя	path

Response:

Запрос выполнен успешно:

Status: 200

Header:

Body:

```
[{
  "id": "string",
  "username": "string",
  "fullName": "string",
  "avatar": "URL string",
  "currentUserFollowsThisUser": bool,
  "currentUserIsFollowedByThisUser": bool,
  "followsCount": number,
  "followedByCount": number
},
...
]
```

Пользователь с запрошенным ID не найден:

Status: 404

Header:

Body:

GET users/{userID}/following

Возвращает подписки пользователя с запрошенным ID.

Request:

name	description	parameter type
token	Токен текущего пользователя	header
userID	ID нужного пользователя	path

Response:

Запрос выполнен успешно:

Status: 200

Header:

Body:

```
[{
  "id": "string",
  "username": "string",
  "fullName": "string",
  "avatar": "URL string",
  "currentUserFollowsThisUser": bool,
  "currentUserIsFollowedByThisUser": bool,
  "followsCount": number,
  "followedByCount": number
},
...
]
```

Пользователь с запрошенным ID не найден:

Status: 404

Header:

Body:

GET users/{userID}/posts

Возвращает публикации пользователя с запрошенным ID.

Request:

name	description	parameter type
token	Токен текущего пользователя	header
userID	ID нужного пользователя	path

Response:

Запрос выполнен успешно:

Status: 200

Header:

Body:

```
[{
  "id": "string",
  "description": "string",
  "image": "URL string",
  "createdTime": "yyyy-MM-dd'T'HH:mm:ssZZZZZ"
  "currentUserLikesThisPost": bool,
  "likedByCount": number,
  "author": "string",
  "authorUsername": "string",
  "authorAvatar": "URL string"
},
...
]
```

Пользователь с запрошенным ID не найден:

Status: 404

Header:

Body:

GET posts/feed

Возвращает публикации пользователей, на которых подписан текущий пользователь.

Request:

name	description	parameter type
token	Токен текущего пользователя	header

Response:

Запрос выполнен успешно:

Status: 200

Header:

Body:

```
[{
  "id": "string",
  "description": "string",
  "image": "URL string",
  "createdTime": "yyyy-MM-dd'T'HH:mm:ssZZZZZ"
  "currentUserLikesThisPost": bool,
  "likedByCount": number,
  "author": "string",
  "authorUsername": "string",
  "authorAvatar": "URL string"
},
...
]
```

GET posts/{postID}

Возвращает публикации пользователей, на которых подписан текущий пользователь.

Request:

name	description	parameter type
token	Токен текущего пользователя	header
postID	ID нужной публикации	path

Response:

Запрос выполнен успешно:

Status: 200

Header:

Body:

```
{
  "id": "string",
  "description": "string",
  "image": "URL string",
  "createdTime": "yyyy-MM-dd'T'HH:mm:ssZZZZZ"
  "currentUserLikesThisPost": bool,
  "likedByCount": number,
  "author": "string",
  "authorUsername": "string",
  "authorAvatar": "URL string"
}
```

Публикация с запрошенным ID не найдена:

Status: 404

Header:

Body:

POST posts/like

Ставит лайк от текущего пользователя на публикации с запрошенным ID.

Request:

name	description	parameter type
token	Токен текущего пользователя	header
postID	ID нужной публикации	body (String)

Response:

Лайк поставлен успешно или был поставлен ранее:

Status: 200

Header:

Body:

```
{
  "id": "string",
  "description": "string",
  "image": "URL string",
  "createdTime": "yyyy-MM-dd'T'HH:mm:ssZZZZZ"
  "currentUserLikesThisPost": bool,
  "likedByCount": number,
  "author": "string",
  "authorUsername": "string",
  "authorAvatar": "URL string"
}
```

Публикация с запрошенным ID не найдена:

Status: 404

Header:

Body:

POST posts/unlike

Удаляет лайк от текущего пользователя на публикации с запрошенным ID.

Request:

name	description	parameter type
token	Токен текущего пользователя	header
postId	ID нужной публикации	body (String)

Response:

Лайк удален успешно или его не было:

Status: 200

Header:

Body:

```
{
  "id": "string",
  "description": "string",
  "image": "URL string",
  "createdTime": "yyyy-MM-dd'T'HH:mm:ssZZZZZ"
  "currentUserLikesThisPost": bool,
  "likedByCount": number,
  "author": "string",
  "authorUsername": "string",
  "authorAvatar": "URL string"
}
```

Публикация с запрошенным ID не найдена:

Status: 404

Header:

Body:

GET posts/{postID}/likes

Возвращает пользователей, поставивших лайк на публикацию с запрошенным ID.

Request:

name	description	parameter type
token	Токен текущего пользователя	header
postID	ID нужной публикации	path

Response:

Запрос выполнен успешно:

Status: 200

Header:

Body:

```
[{
  "id": "string",
  "username": "string",
  "fullName": "string",
  "avatar": "URL string",
  "currentUserFollowsThisUser": bool,
  "currentUserIsFollowedByThisUser": bool,
  "followsCount": number,
  "followedByCount": number
},
...
]
```

Публикация с запрошенным ID не найдена:

Status: 404

Header:

Body:

POST posts/create

Создает новую публикацию.

Request:

name	description	parameter type
token	Токен текущего пользователя	header
image	Изображение для публикации в формате base64	body (String)
description	Описание публикации	body (String)

Response:

Запрос выполнен успешно:

Status: 200

Header:

Body:

```
{
  "id": "string",
  "description": "string",
  "image": "URL string",
  "createdTime": "yyyy-MM-dd'T'HH:mm:ssZZZZZ"
  "currentUserLikesThisPost": bool,
  "likedByCount": number,
  "author": "string",
  "authorUsername": "string",
  "authorAvatar": "URL string"
}
```