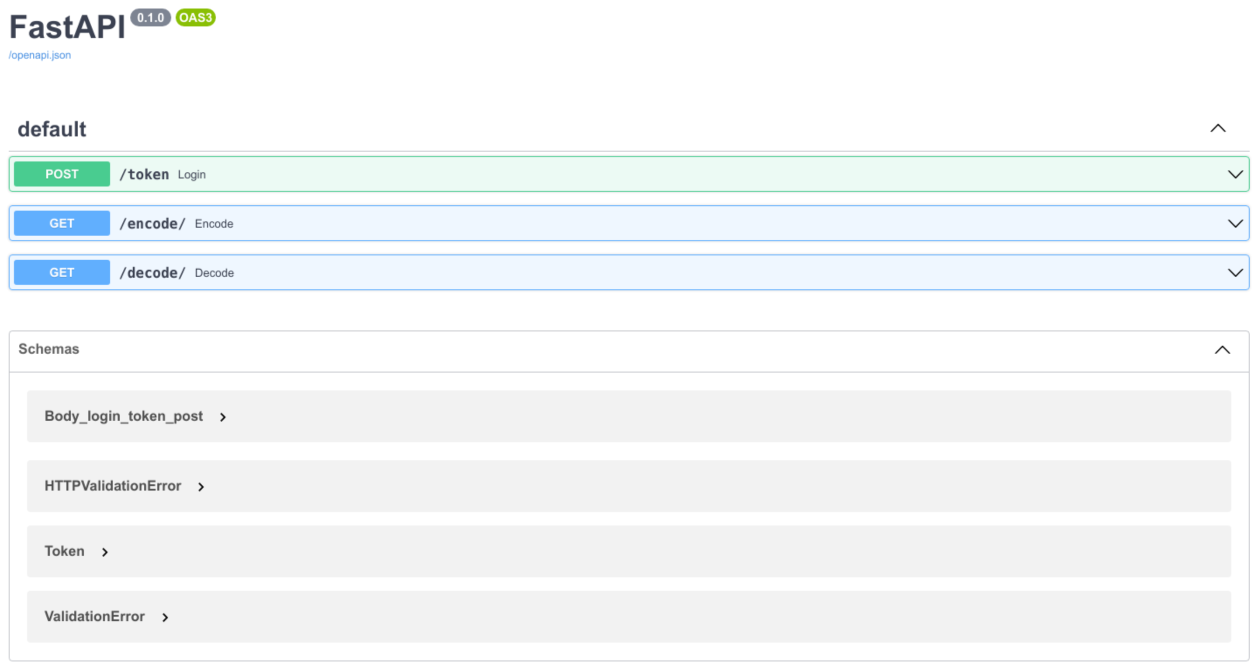
FastAPI encryptng service

I. AUTHENTICATION

After service is running, you are free to open page http://localhost:8000/docs to use smart web UI for API:



FastAPI 0.1.0 OAS3

/openapi.json

default

POST /token Login

GET /encode/ Encode

GET /decode/ Decode

Schemas

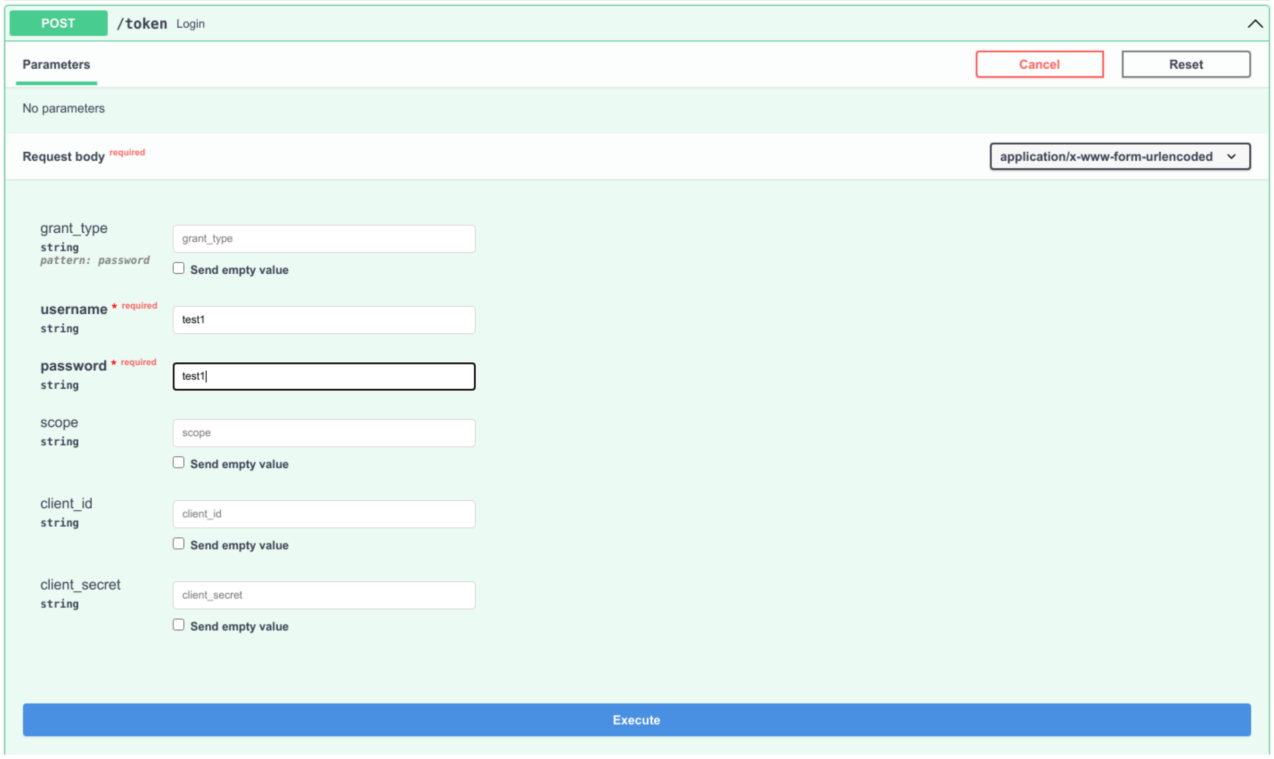
Body_login_token_post ›

HTTPValidationError ›

Token ›

ValidationError ›

First of all you must be authorizated throug /token path with user «test1» (password «test1»):



POST /token Login

Parameters

No parameters

Request body required

application/x-www-form-urlencoded

grant_type
string
pattern: password
grant_type
□ Send empty value

username * required
string
test1

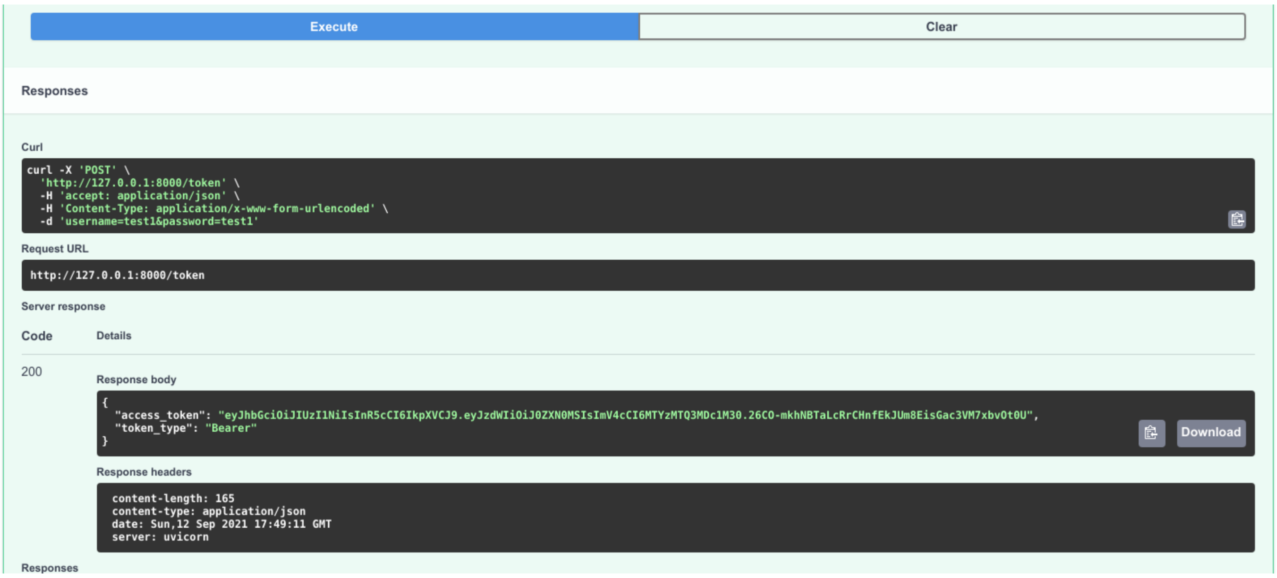
password * required
string
test1

scope
string
scope
□ Send empty value

client_id
string
client_id
□ Send empty value

client_secret
string
client_secret
□ Send empty value

After pressing «Execute» button you shoul see response from server in «Responses» area at block «200»:



Responses

Curl

```
curl -X 'POST' \
  'http://127.0.0.1:8000/token' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/x-www-form-urlencoded' \
  -d 'username=test1&password=test1'
```

Request URL

```
http://127.0.0.1:8000/token
```

Server response

Code	Details
200	Response body

```
{
  "access_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiJ0ZXN0MSIsImV4cCI6MTYzMTQ3MDc1M30.26CO-mkhNBTaLcRrCHnfEkJUm8EisGac3VM7xbvOt0U",
  "token_type": "Bearer"
}
```

Response headers

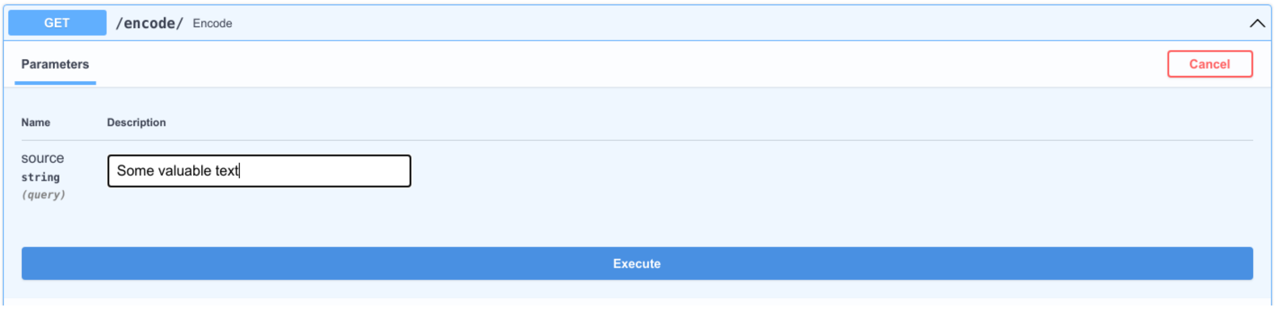
```
 content-length: 165
 content-type: application/json
 date: Sun,12 Sep 2021 17:49:11 GMT
 server: uvicorn
```

Responses

Field named «access_token» contain your special token, which used to identify you on server side. After that you are free to use /encode and /decode paths.

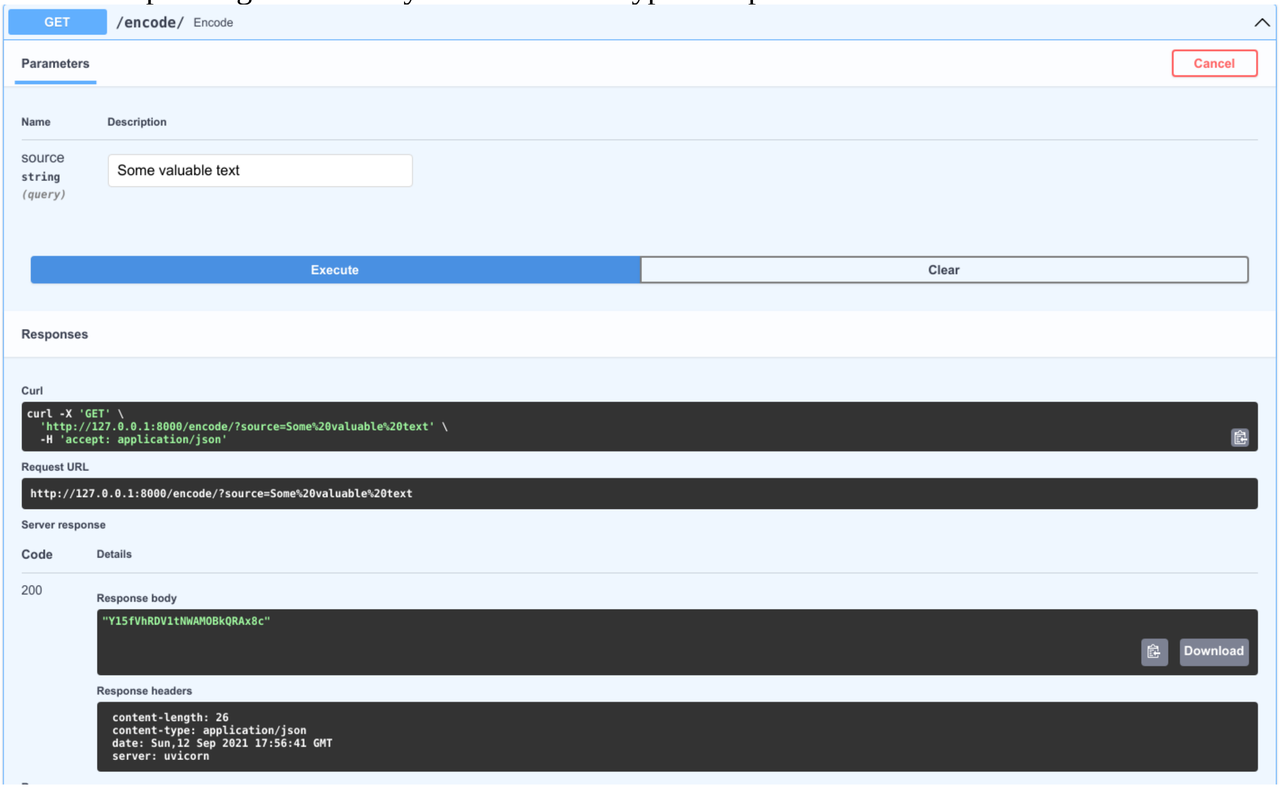
II. ENCODE

Next block, named «/encode» encodes given string using XOR cipher. Expand field, and enter some text:



Name	Description
source string *(query)*	Some valuable text

After pressing «Execute» you shall see encrypted response in block below:



Name	Description
source string *(query)*	Some valuable text

Responses

Curl

```
curl -X 'GET' \
  'http://127.0.0.1:8000/encode/?source=Some%20valuable%20text' \
  -H 'accept: application/json'
```

Request URL

```
http://127.0.0.1:8000/encode/?source=Some%20valuable%20text
```

Server response

Code Details

200

Response body

```
"Y15fVhRDV1tNWAMOBkQRAx8c"
```

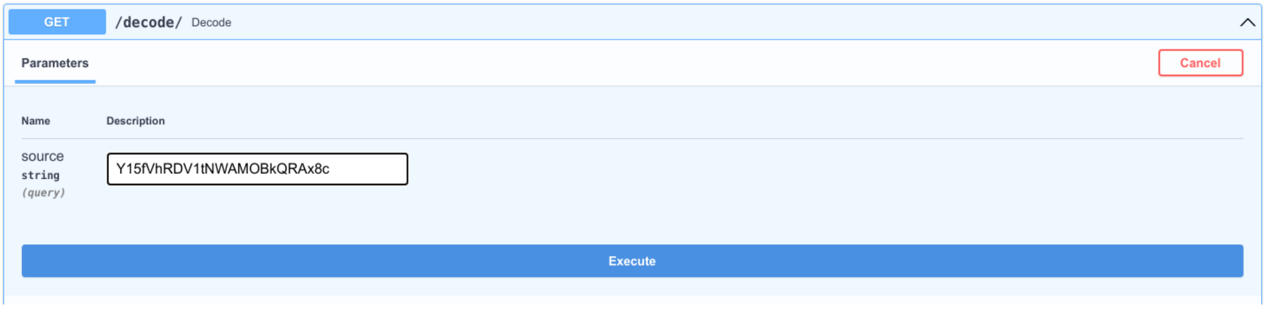
Response headers

```
content-length: 26
content-type: application/json
date: Sun,12 Sep 2021 17:56:41 GMT
server: uvicorn
```

That is your encrypted string.

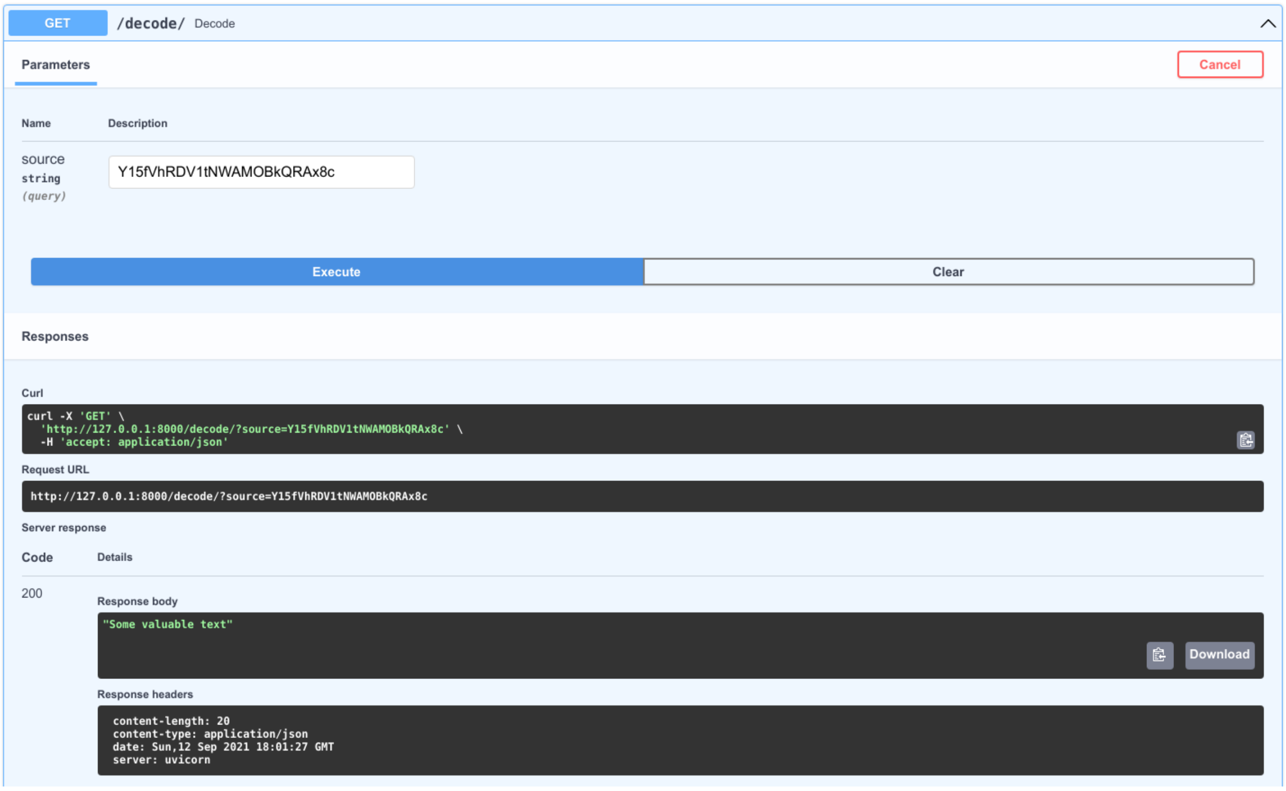
III. DECODE

To check proper work of decoding, let’s copy result from previous paragraph. Expand nex block named «/decode» and paste encoded string into field:



Name | Description
source string (query): Y15fVhRDV1tNWAMOBkQRAx8c

After pressing «Execute» button, you’ll see response with our source string:



Name | Description
source string (query): Y15fVhRDV1tNWAMOBkQRAx8c

Curl

```
curl -X 'GET' \
  'http://127.0.0.1:8000/decode/?source=Y15fVhRDV1tNWAMOBkQRAx8c' \
  -H 'accept: application/json'
```

Request URL

```
http://127.0.0.1:8000/decode/?source=Y15fVhRDV1tNWAMOBkQRAx8c
```

Server response

Code | Details
200 | Response body

```
"Some valuable text"
```

Response headers

```
content-length: 20
content-type: application/json
date: Sun,12 Sep 2021 18:01:27 GMT
server: uvicorn
```

Congrats! It’s working!