

# Relayer

1inch Fusion API

POST <https://api.1inch.dev/fusion/relayer/v2.0/{chain}/order/submit/many>

Submit a list of limit orders which resolvers will be able to fill

## Network



Ethereum



## Parameters

chain **number** \* (path)

1

## Request Body

Content type

application/json

Click in to adjust the request body:

```
[
  {
    "order": {
      "salt": "string",
      "makerAsset": "string",
      "takerAsset": "string",
      "maker": "string",
      "receiver": "0x0000000000000000000000000000000000000000",
      "makingAmount": "string",
      "takingAmount": "string",
      "makerTraits": "0"
    },
    "signature": "string",
    "extension": "0x",
    "quoteld": "string"
  }
]
```

## Request Body Schema:

- ▼ array
  - ▼ items SignedOrderInput object
    - ▼ order \* any
      - ▼ allOf[0] OrderInput object
        - salt \* string
        - makerAsset \* string
        - takerAsset \* string
        - maker \* string
        - receiver string
        - makingAmount \* string
        - takingAmount \* string
        - makerTraits string
      - signature \* string
      - extension string
      - quoteld \* string

## swagger.response

Code: 201

The gasless orders has been successfully saved

Code: 400

Input data is invalid

Schema:

- ▼ object
  - ▼ statusCode integer (Enum)
    - 400
    - message string
  - ▼ error string (Enum)
    - Bad Request

## Authorization - API KEY

API KEY

[Sign in](#) for automatic API key authentication.

Try it



NodeJS

Python

cURL

Go

```
const axios = require("axios");
```

```
async function httpCall() {
```

```
const url = "https://api.1inch.dev/fusion/relayer/v2.0/1/order/submit/many";
```

```
const config = {
  headers: undefined,
  params: {},
  paramsSerializer: {
    indexes: null,
  },
};
const body = [
  {
    order: {
      salt: "string",
      makerAsset: "string",
      takerAsset: "string",
      maker: "string",
      receiver: "0x0000000000000000000000000000000000000000000000000000000000000000"
    }
  }
];
```

## Response

Click the try-it button to test your API call and see the real-time response right here.

Previous

[Submit a limit order that resolvers will be able to fill](#)

Next

[Migration from v1.0 to v2.0](#)