Documentation   **Support Center**   **Dashboard**

Introduction

Welcome to our API reference. The API reference describes each available resource individually in the Finexer API. If you want to integrate Finexer into your application, it's the right place to get started. Because the RESTful API is based on open standards, you can use any web development language to access the API.

The API is built on HTTP and is a standard **RESTful** service. It uses predictable, resource-oriented URLs, HTTP response codes to indicate errors.
All responses are returned in **JSON** with the Content-Type `application/json`.
All POST requests must have a form-encoded body with the Content-Type `application/x-www-form-urlencoded`.

Every API request has an associated request identifier. You can find this value in the response headers, under `fx-request-id`. If you need to contact us about a specific request, providing the request identifier will ensure the fastest possible resolution.

All timestamps returned by the API are represented in UTC.

> We always welcome new ideas. If you believe that our API is

API ENDPOINT

`https://api.finexer.com`

> missing an important feature, or
> that our service needs to be
> more customisable to suit your
> needs, then **drop us an email**
> or contact the **Support Center**.

Authentication

The API requires **HTTP basic authentication** for every request. To authenticate your account, you need to provide your secret API key in the request which can be found in your **account settings**.

Provide your API key as the username value in the basic authentication. You do not need to provide a password - leave it as an empty string.

If you need to authenticate via bearer authentication (e.g., manage other accounts), use the following:

```
-H "Authorization: Bearer API_KEY"
```

Because your API keys carry many privileges, do keep them secret. Don't pass them to anybody, don't share them on other sites or in client-side code.
All API requests must be made over TLS/HTTPS. Requests made over plain HTTP or without authentication will fail.

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/cus
\
    -u "API_KEY"
```

Errors

The API uses conventional HTTP

EXAMPLE RESPONSE

The API uses conventional HTTP response codes to indicate success or failure of an API request. In general, codes in the `2xx` range indicate success, codes in the `4xx` range indicate an error that resulted from the information provided (e.g., a required parameter was omitted, a payment failed, etc.), and codes in the `5xx` range indicate an error with Finexer's servers (these are rare).

```
{
  "error": {
    "type":
"request_error",
    "messag": "Unknown
currency code
supplied",
    "param": "currency"
  }
}
```

ATTRIBUTES

type	The type of the error returned.
message	A human-readable message providing details about the error.
code optional	A short code for a specific error message.
param optional	The HTTP parameter name the error relates to if the error is parameter-specific.

HTTP STATUS CODE SUMMARY

200 - OK	Request w successfu
204 - No Content	Request w successfu and no content wa sent in the response. Mostly returned i deletion requests.
400 - Bad Request	The reque was unaccepta often due missing or invalid parameter
401 - Unauthorized	No valid A key provic
402 - Request Failed	The parameter were valid the reques failed.
404 - Not Found	The requested resource doesn't ex
409 - Conflict	The reque conflicts w another request.

429 - Too Many Requests	Too many requests h the API to quickly.
5xx - Server Errors	Something went wror on our end

ERROR TYPES

auth_error	The request could not be authenticated, often due to a missing, expired or invalid API key.
request_error	The request could not be parsed, often due to a missing or an invalid parameter.
provider_error	The request was valid however a request to a 3-rd party provider failed, often due to connectivity issues with a financial institution.
server_error	Uncommon temporary problem with our servers.

Metadata

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/cus
\
    -u "API_KEY" \
    -d name="Johny" \
    -d
email="johny.hero@app.com"
\
    -d
metadata[id]="518490" \
    -d
metadata[dob]="06/09/1986"
\
    -d
metadata[state]=""
```

Metadata is additional data that can be associated with an object for later use. Anytime the API returns an object with previously added metadata to it, the object includes extra metafield with structured data.

It becomes useful if you need to add IDs from your system, phone numbers, shipping information, refund explanation or any own meaningful values.

To attach metadata to an object you can use a metadata parameter in the body of a request providing key-value (dictionary) data.

If you need to clear an attribute's value, you can pass an empty value in an argument.

> Not all objects contain this facility unless otherwise stated.
> You can specify up to 10 keys, with key names up to 24 characters long and values up to 64 characters long.

EXAMPLE RESPONSE

```
    {
      "id":
"cus_hjw5i1xO9zQK5kpS",
      "object":
"customer",
      "name": "Johny",
      "email":
"johny.hero@test.com",
      "metadata": {
        "id": "518490",
        "dob":
"06/09/1986"
      },
      ...
    }
```

Expandable

```
curl -X GET
https://api.finexer.com/pa
expand[]=data.customer
\
    -u "API_KEY"
```

Represents an attribute that can be dynamically expanded into a full API object that enables you to receive more detailed responses. This is useful when you need additional data without a need to make an extra call to the API. For example, payment object contains the customer attribute with the unique identifier of the customer object. If you need details of the customer in context of the List payments method, supply data.customer as a value of a query parameter expand[] , to bind the customer attribute with the full customer object.
Multiple attribute expansion is supported should you wish to expand multiple object attributes by supplying multiple expand[] parameters in the query.

The value for the expand parameter can be determine by the path to the attribute you need to expand returned in the response. For example, to return a customer object in context of the Get a payment method, supply just customer as a value of the expand parameter in the query.

Attributes that support the expansion are decorated with the expandable label in this documentation.

```
{
  "data": [
    {
      "id":
"pt_LiDYx1SaOBw57TOI9KPDlT
      "object":
"payment",
      "created_at":
"2020-12-11T16:25:35Z",
      "updated_at":
null,
      "amount": 712,
      "currency":
"gbp",
      "customer": {
        "id":
"cus_hjw5i1xO9zQK5kpS",
        "object":
"customer",
        "created_at":
"2020-02-10T10:42:22Z",
        "name": "Acme
Limited Altered",
        "email":
"whiliams.great@acme.com",
        {...}
      },
      "provider":
"natwest",
      {...}
    }
  ]
}
```

Pagination

Pagination

```
curl -X GET
https://api.finexer.com/cus
limit=5&offset=10 \
    -u "API_KEY"
```

Many of the API resources can contain large numbers of results when retrieving unbounded lists. All these resources share same pagination model that is returned as part of the response body for almost every "list" operation. The pagination model supports offset-based and cursor-based pagination.

By default, unbounded lists are returned in groups of 20 items. If there are more than 20 items in the list, you can use pagination on all "list" operations where pagination supported to retrieve the next set of *n* items.

QUERY PARAMETERS

limit	A limit on the number of objects to be returned, between 1 and 100.
optional, default is 20	
offset	The number of objects to skip for the first returned result.
optional, default is 0	

List object

ATTRIBUTES

object string	The name of the object: `list`	
count integer	The actual number of objects returned in `data` attribute.	
location string, nullable	The relative resource URL.	
Expand ▸ **paging** object, nullable	The object with the pagination attributes. Can be null in cases when the list object is returned as a child of another object.	
data list	The collection of objects itself.	

```
{
  "object": "list",
  "count": 5,
  "location":
"/customers",
  "paging": {
    "total": 26,
    "next":
"/customers?
limit=5&offset=5",
    "prev": null
  },
  "data": [
    {
      "id":
"cus_hjw5i1xO9zQK5kpS",
      "object":
"customer",
      "name": "Acme
Ltd",
      "email":
"whiliams.great@acme.com",
      ...
    },
    {
      "id":
"cus_vYDuVeeWto2RMQB5XoYwpl
      "object":
"customer",
      "name": "Johny",
      "email":
"johny.hero@test.com",
      ...
    },
    {...}
  ]
}
```

Vendors

Vendor object holds the contact details of a vendor. It represents a supplier that provides products or services to another entity. You can use it to group payments or consents by a vendor and/or apply the branding theme for each vendor on consent screens.

Vendor object

EXAMPLE RESPONSE

ATTRIBUTES

id string	Unique identifier for a vendor object.
object string	The name of the object: `vendor`
created_at timestamp	Time indicating when a vendor object was created.
updated_at timestamp	Time indicating when a vendor object was last time updated.
name string	The business or individual name of the vendor.

```json
{
  "id":
"vnd_PvL5q9ExrdO9g3GwSNju"
  "object": "vendor",
  "created_at": "2022-
10-02T10:00:21Z",
  "updated_at": "2020-
02-22T16:12:01Z",
  "name": "Suppliers
Limited",
  "email":
"info@suppliers.com",
  "phone":
"+4407862807022",
  "description":
"Supplier of mobile
accessories",

"default_bank_account":
null,

"default_app_template":
"ksGM2QPt9TpG",
  "address": {
    "line1": "7
riverside rd",
    "line2": null,
    "city": "London",
    "state":
"Kensington",
    "zip": "W8 7NB",
    "country": "GB"
  },
  "metadata": {}
}
```

email string	The email address belonging to the vendor.	
phone string, nullable	The phone number belonging to the vendor.	
description string, nullable	An arbitrary string attached to the vendor object.	
default_bank_account string, nullable	The default identifier of the **bank account** object for the vendor.	
default_app_template string, nullable	The default identifier of the **app template** object for the vendor. E.g. when a customer enters the hosted consent screens, an associated template theme is applied.	
Expand ▸ **address** object, nullable	See child attributes	
metadata object, { }	Set of arbitrary key/value pairs you	

can attach to the vendor. It can be useful for storing additional information about the vendor.

Create a vendor

Creates a new vendor object.

ARGUMENTS

name REQUIRED	The vendor's full name or business name.
email REQUIRED	The email address belonging to the vendor.
phone optional	The phone number belonging to the vendor.
description optional	An arbitrary string that you can attach to the vendor.
default_app_template optional	The identifier

ENDPOINT

```
POST: /vendors
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/ver
\
    -u "API_KEY" \
    -d name="Suppliers
Limited" \
    -d
email="info@suppliers.com"
\
    -d
default_app_template="ksGM
\
    -d
address.line1="7
riverside rd" \
    -d
address.city="London" \
    -d address.zip="W8
7NB" \
    -d
address.country="GB" \
    -d
metadata[affiliate]="62844
```

EXAMPLE RESPONSE

```
{
  "id":
"vnd_PvL5q9ExrdO9g3GwSNju"
  "object": "vendor",
```

of the **app template** object indicating the default branding theme for the vendor to be created.

Expand ▸	address object, nullable	See child attributes
	metadata optional	Set of arbitrary key/value pairs you can attach to the vendor.

```
  "created_at": "2020-
02-10T10:42:22Z",
  "updated_at": null,
  "name": "Suppliers
Limited",
  "email":
"info@suppliers.com",
  "phone": null,
  "description": null,

"default_bank_account":
null,

"default_app_template":
"ksGM2QPt9TpG",
  "address": {
    "line1": "7
riverside rd",
    "line2": null,
    "city": "London",
    "state": null,
    "zip": "W8 7NB",
    "country": "GB"
  },
  "metadata": {
    "affiliate":
"628448"
  }
}
```

RESPONSE

Returns the newly created **vendor** object if the request succeeded. Returns an **error** otherwise.

Get a vendor

Retrieves the details of a single existing vendor that was previously created by **Create a vendor** method. You need only supply the unique vendor `id` in the endpoint route, associated with the vendor object you want to retrieve.

RESPONSE

Returns an existing **vendor** object that matches provided vendor ID if the request succeeded. Returns an **error** otherwise.

ENDPOINT

```
GET: /vendors/{id}
```

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/ver
\
      -u "API_KEY"
```

EXAMPLE RESPONSE

```
{
  "id":
"vnd_PvL5q9ExrdO9g3GwSNju"
  "object": "vendor",
  "created_at": "2020-
02-10T10:42:22Z",
  "updated_at": null,
  "name": "Suppliers
Limited",
  "email":
"info@suppliers.com",
  "phone": null,
  "description": null,

"default_bank_account":
null,

"default_app_template":
"ksGM2QPt9TpG",
  "address": {
    "line1": "7
riverside rd",
    "line2": null,
    "city": "London",
    "state": null,
    "zip": "W8 7NB",
    "country": "GB"
  },
  "metadata": {
    "affiliate":
"628448"
  }
}
```

Update a vendor

Updates the vendor object by its `id` found on the **vendor** object. It can be used to modify attributes on the vendor, arguments of which were passed for this request.

Note if you need to clear an attribute's value, you can pass an empty value in an argument.

ARGUMENTS

name optional	The vendor's full name or business name.
email optional	The email address belonging to the vendor.
phone optional	The phone number belonging to the vendor.
description optional	An arbitrary string that you can attach to the vendor.
default_bank_account optional	The identifier of the **bank_account** object indicating the default bank account for the vendor to be updated.
default_app_template optional	The identifier of the **app template** object indicating the

ENDPOINT

```
POST: /vendors/{id}
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/ve
\
    -u "API_KEY" \
    -d name="Suppliers
Limited II" \
    -d address.city=""
\
    -d address.zip=""
\
    -d
metadata[affiliate]=""
```

EXAMPLE RESPONSE

```
{
  "id":
"vnd_PvL5q9ExrdO9g3GwSNju"
  "object": "vendor",
  "created_at": "2020-
02-10T10:42:22Z",
  "updated_at": "2020-
02-22T16:12:01Z",
  "name": "Suppliers
Limited II",
  "email":
"info@suppliers.com",
  "phone": null,
  "description": null,

"default_bank_account":
null,

"default_app_template":
"ksGM2QPt9TpG",
  "address": {
    "line1": "7
riverside rd",
    "line2": null,
    "city": null,
    "state": null,
    "zip": null,
    "country": "GB"
  },
  "metadata": {}
```

		default branding theme for the vendor to be updated.
Expand ▸	address object, nullable	See child attributes
	metadata optional	Set of arbitrary key/value pairs you can attach to the vendor.

```
          }
```

RESPONSE

Returns the **vendor** object that has been updated if the request succeeded. Returns an **error** otherwise.

Delete a vendor

Permanently deletes a vendor by its `id` found on the **vendor** object.

RESPONSE

Returns an empty response with http status code of `204` if the request succeeded. Returns an **error** otherwise.

ENDPOINT

```
DELETE: /vendors/{id}
```

EXAMPLE REQUEST

```
curl -X DELETE
https://api.finexer.com/ve
    \
    -u "API_KEY"
```

List vendors

Retrieves a **paginated** list of vendors you have previously created. The returned vendors are sorted by creation date, with the most recently created vendors

ENDPOINT

```
GET: /vendors
```

EXAMPLE REQUEST

```
curl  v GET
```

appearing first.

QUERY PARAMETERS

term optional	Only return vendors whose name contain this value.
email optional	Only return vendors whose email match the specified email address.
Expand ▸ created_at optional	Represents date time universal **ISO 8601** format. (e.g. 2019-05-25 or 2019-05-25T14:00:00)

To retrieve a complete set of paginated results, you can use **pagination** arguments too.

RESPONSE

Returns a list of existing **vendor** objects that match filter criteria if the request succeeded. Returns an **error** otherwise.

```
curl -X GET
https://api.finexer.com/ven
created_at.gte=2022-10-
01&created_at.lt=2022-
10-10 \
    -u "API_KEY"
```

EXAMPLE RESPONSE

```
{
  "object": "list",
  "count": 12,
  "location":
"/vendors",
  "paging": {
    "total": 12,
    "next": null,
    "prev": null
  },
  "data": [
    {
      "id":
"vnd_PvL5q9ExrdO9g3GwSNju"
      "object":
"vendor",
      "created_at":
"2022-10-02T10:00:21Z",
      "updated_at":
"2022-10-16T13:41:07Z",
      "name":
"Suppliers Limited",
      "email":
"info@suppliers.com",
      "phone": null,
      "description":
null,

"default_bank_account":
null,

"default_app_template":
"ksGM2QPt9TpG",
      "address": {
        "line1":
"line1",
        "line2":
"line2",
        "city":
"London",
        "state":
"Kensington",
        "zip": "WD06",
        "country": "GB"
      },
      "metadata": {}
```

```
    },
    {
      "id":
"vnd_nNkD9Vx8BeqkAsdoNody"
      "object":
"vendor",
      "created_at":
"2022-10-01T01:22:47Z",
      "updated_at":
null,
      "name":
"Verification Services
Ltd",
      "email":
"info@verifications.com",
      "phone": null,
      "description":
null,

"default_bank_account":
null,

"default_app_template":
null,
      "address": null,
      "metadata": {}
    },
    {...}
  ]
}
```

Customers

Customer object holds the contact details of a client. It is used to create, delete and update customer, create bank consents to connect banks including attach subscriptions, payment methods. API allows you to retrieve individual customers as well as a list of all your customers.

ENDPOINTS

POST: /customers

GET: /customers/{id}

POST: /customers/{id}

DELETE: /customers/{id}

GET: /customers

Customer object

EXAMPLE RESPONSE

```
{
  "id":
```

ATTRIBUTES

id string	Unique identifier for customer object.	
object string	The name of the object: customer	
created_at timestamp	Time indicating when the customer object was created.	
updated_at timestamp	Time indicating when the customer object was last time updated.	
name string	The business or person name of the customer.	
email string, nullable	The email address belonging to the customer. Can be used to send payment requests, receipts, etc.	
phone string, nullable	The phone number belonging to the customer.	
currency	3 letter	

```
  "cus_hjw5i1xO9zQK5kpS",
  "object": "customer",
  "created_at": "2020-
02-10T10:42:22Z",
  "updated_at": null,
  "name": "Acme
Limited",
  "email":
"whiliams.great@acme.com",
  "phone":
"+4407856658585",
  "currency": "gbp",
  "description":
"Frequent VM buyer",

"default_bank_account":
"ba_b7yr9Bp6dMnt5HC4Ah6bOWI
  "address": {
    "line1": "7
riverside rd",
    "line2": null,
    "city": "London",
    "state":
"Kensington",
    "zip": "W8 7NB",
    "country": "GB"
  },
  "metadata": {}
}
```

currency string	3-letter **ISO** currency code e.g., gbp, eur, usd, etc.	
description string, nullable	An arbitrary string attached to the customer.	
default_bank_account string, nullable	The default identifier of **bank account** object for this customer.	
Expand ▸ address object, nullable	See child attributes	
metadata object, { }	Set of arbitrary key/value pairs you can attach to the customer. It can be useful for storing additional information about the customer.	

Create a customer

Creates a new customer object.

ARGUMENTS

name REQUIRED	The customer's

ENDPOINT

```
POST: /customers
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/cu
```

	full name or business name.
email **REQUIRED**	The email address belonging to the customer.
currency optional	3-letter **ISO** currency code as default currency for the customer.
description optional	An arbitrary string that you can attach to the customer.
phone optional	The phone number belonging to the customer.
Expand ▸ address object, nullable	See child attributes
metadata optional	Set of arbitrary key/value pairs you can attach to the customer.

RESPONSE

Returns the newly created **customer** object if the request succeeded. Returns an **error**

```
        \
        -u "API_KEY" \
        -d name="Acme Ltd"
\
        -d
email="whiliams.great@acme
\
        -d
address.line1="7
riverside rd" \
        -d
address.city="London" \
        -d address.zip="W8
7NB" \
        -d
address.country="GB" \
        -d
metadata[affiliate]="51849(
```

EXAMPLE RESPONSE

```json
{
  "id":
"cus_hjw5i1xO9zQK5kpS",
  "object": "customer",
  "created_at": "2020-
02-10T10:42:22Z",
  "updated_at": "2020-
02-22T16:12:01Z",
  "name": "Acme
Limited",
  "email":
"whiliams.great@acme.com",
  "phone": null,
  "currency": null,
  "description": null,

"default_bank_account":
null,
  "address": {
    "line1": "7
riverside rd",
    "line2": null,
    "city": "London",
    "state": null,
    "zip": "W8 7NB",
    "country": "GB"
  },
  "metadata": {
    "affiliate":
"518490"
  }
}
```

otherwise.

Get a customer

ENDPOINT

> GET: /customers/{id}

Retrieves the details of a single existing customer that was previously created by **Create a customer** method. You need only supply the unique customer `id` in the endpoint route, associated with the customer object you want to retrieve.

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/cus
\
    -u "API_KEY"
```

RESPONSE

Returns an existing **customer** object that matches provided customer ID if the request succeeded. Returns an **error** otherwise.

EXAMPLE RESPONSE

```
{
  "id":
"cus_hjw5i1xO9zQK5kpS",
  "object": "customer",
  "created_at": "2020-
02-10T10:42:22Z",
  "updated_at": "2020-
02-22T16:12:01Z",
  "name": "Acme
Limited",
  "email":
"whiliams.great@acme.com",
  "phone":
"+4407856658585",
  "currency": "gbp",
  "description":
"Frequent VM buyer",

"default_bank_account":
null,
  "address": {
    "line1": "7
riverside rd",
    "line2": null,
    "city": "London",
    "state": null,
    "zip": "W8 7NB",
    "country": "GB"
  },
  "metadata": {
    "affiliate":
"518490"
  }
}
```

Update a customer

Updates the customer object by its
id found on the **customer** object. It
can be used to modify attributes on
the customer, arguments of which
were passed for this request.

Note if you need to clear an
attribute's value, you can pass an
empty value in an argument.

Supports mostly the same
arguments supported when
creating a customer.

ARGUMENTS

name optional	The customer's full name or business name.
currency optional	3-letter **ISO** currency code as default currency for the customer.
description optional	An arbitrary string that you can attach to the customer.
email optional	The email address belonging to the customer.
phone optional	The phone number belonging to the customer.
default_bank_account optional	The identifier of the **bank_account** object indicating the

ENDPOINT

```
POST: /customers/{id}
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/cus
\
    -u "API_KEY" \
    -d name="Acme
Limited Altered" \
    -d address.city=""
\
    -d address.zip=""
\
    -d
metadata[affiliate]=""
```

EXAMPLE RESPONSE

```
{
  "id":
"cus_hjw5i1xO9zQK5kpS",
  "object": "customer",
  "created_at": "2020-
02-10T10:42:22Z",
  "updated_at": "2020-
02-22T16:12:01Z",
  "name": "Acme Limited
Altered",
  "email":
"whiliams.great@acme.com",
  "phone":
"+4407856658585",
  "currency": "gbp",
  "description":
"Frequent VM buyer",

"default_bank_account":
null,
  "address": {
    "line1": "7
riverside rd",
    "line2": null,
    "city": null,
    "state": null,
    "zip": null,
    "country": "GB"
  },
  "metadata": {}
}
```

indicating the
default bank
account for
this customer.

Expand ▸

address object, nullable	See child attributes
metadata optional	Set of arbitrary key/value pairs you can attach to the customer.

RESPONSE

Returns the **customer** object that
has been updated if the request
succeeded. Returns an **error**
otherwise.

Delete a customer

Permanently deletes a customer by
its `id` found on the **customer**
object.

RESPONSE

Returns an empty response with
http status code of `204` if the
request succeeded. Returns an
error otherwise.

List customers

Retrieves a **paginated** list of
customers you have previously
created. The returned customers
are sorted by creation date, with

ENDPOINT

```
DELETE: /customers/{id}
```

EXAMPLE REQUEST

```
curl -X DELETE
https://api.finexer.com/cu
\
    -u "API_KEY"
```

ENDPOINT

```
GET: /customers
```

EXAMPLE REQUEST

the most recently created customers appearing first.

QUERY PARAMETERS

term optional	Only return customers whose name contain this value.
email optional	Only return customers whose email match the specified email address.
Expand ▸ **created_at** optional	Represents date time universal **ISO 8601** format. (e.g. 2019-05-25 or 2019-05-25T14:00:00)

To retrieve a complete set of paginated results, you can use **pagination** arguments too.

RESPONSE

Returns a list of existing **customer** objects that match filter criteria if the request succeeded. Returns an **error** otherwise.

```
curl -X GET
https://api.finexer.com/cu
created_at.gte=2020-01-
01&created_at.lt=2020-
03-01 \
    -u "API_KEY"
```

EXAMPLE RESPONSE

```
{
  "count": 20,
  "location":
"/customers",
  "paging": {
    "total": 26,
    "next":
"/customers?offset=20",
    "prev": null
  },
  "data": [
    {
      "id":
"cus_hjw5i1xO9zQK5kpS",
      "object":
"customer",
      "created_at":
"2020-02-10T10:42:22Z",
      "updated_at":
"2020-02-22T16:12:01Z",
      "name": "Acme
Ltd",
      "email":
"whiliams.great@acme.com",
      ...
    },
    {
      "id":
"cus_vYDuVeeWto2RMQB5",
      "object":
"customer",
      "created_at":
"2020-02-25T16:52:01Z",
      "updated_at":
"2020-02-26T14:20:56Z",
      "name": "Johny",
      "email":
"johny.hero@test.com",
      ...
    },
    {...}
  ]
}
```

Consents

Consent objects hold authorization details to access someone's bank account data that is given by an account holder. For more information on how to obtain access to bank account data, please visit our **documentation** page.

Consent object

EXAMPLE OBJECT

ATTRIBUTES

id string	Unique identifier for consent object.
object string	The name of the object: `consent`
created_at timestamp	Time indicating when the consent was created.
updated_at timestamp	Time at which the consent was last updated.
provider string, nullable expandable	The unique identifier of the supported **provider** objects the consent is created with.

```
{
  "id":
"bc_mtoK6Z9ZBtBLztQorkEm8c
  "object": "consent",
  "created_at": "2020-
01-23T10:17:03Z",
  "updated_at": "2020-
01-23T10:18:56Z",
  "provider":
"santander",
  "vendor": null",
  "customer":
"cus_hjw5i1xO9zQK5kpS",
  "consent_link": null,
  "failure_code": null,
  "failure_message":
null,
  "expiry_date": "2022-
01-22",
  "retro_date": "2020-
01-01",
  "scopes": [
    "accounts"
  ],
  "authed_at": "2020-
01-23T10:20:15Z",
  "status":
"authorized",
  "metadata": {}
}
```

vendor string, nullable `expandable`	The unique identifier of a **vendor** object the consent is attached to. E.g. an initiator of a data access request. Or, it can be the party who shares their data.
customer string, nullable `expandable`	The unique identifier of a **customer** object the consent is attached to. E.g. a party who shares their data.
consent_link string, nullable	The unique identifier of a **consent_link** object the consent is linked to.
expiry_date timestamp, nullable	Time at which the access to account data gets expired. Is set when the user authenticates at a financial institution, or when specifically provided in **consent create** method.
retro_date timestamp, nullable	The earliest time in the past from which account data can be

		pulled. Is set when the user authenticates at a financial institution, or when specifically provided in **consent create** method.
scopes	list	The list of scopes as a level of access to user data an authorized consent has. Available scopes: accounts - account name, number, sort code, IBAN, currency, etc; balance - accrued balances on accounts' statements, currency, and credit line; transactions - transactions' date, reference, amount, status, code, etc.
failure_code	string, nullable	Error code if an error raised with the consent.
failure_message	string, nullable	Detailed error message explaining reason if an error raised with the

consent.

authed_at timestamp, nullable	Time indicating when user authorized access to their bank.
status string	The status of the consent. The list of available statuses: `new` - consent was created with initial data; `pending` - consent was created with sufficient data and is waiting until the user authorizes it; `authorized` - the authorization completed, the user approved the access request and you have the access to the user's data; `failed` - authorization failed due to a 3-rd party provider; `expired` - the consent expired, no access to live data; `canceled` - the user canceled the flow, or the user or you revoked the access of a previously

		created consent.
Expand ▸		
redirect object, nullable	See child attributes	This object is returned only when consent status is `pending` .
metadata object, { }		Set of arbitrary key/value pairs you can attach to the consent. It can be useful for storing additional information about the consent.

Create a consent

Creates a new consent object. Once created, direct the user to the URL supplied in `redirect.consent_url` attribute to authorize access request.

> Note a new consent can be only created for an existing vendor, a customer, or both, the `id` of which you need to supply in the request.

ARGUMENTS

customer optional	The ID of the **customer** object a new consent will be attached to. Required

ENDPOINT

```
POST: /consents
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/co
\
    -u "API_KEY" \
    -d
provider="santander" \
    -d
customer="cus_hjw5i1xO9zQK
\
    -d
scopes[]="accounts" \
    -d
scopes[]="balance" \
    -d
metadata[track_id]="809645
```

to. Required if vendor attribute is not set.

vendor
optional

The ID of the **vendor** object a new consent will be attached to. Required if customer attribute is not set.

provider
optional

Supply the id of a supported **provider** if you already know your customer intends to share bank account data from a specific bank. Leave this empty to enable your customer to select any bank on the consent page.

consent_link
optional

The ID of an existing **consent_link** object a new consent will be linked to. If you provide the consent link ID, a new consent will be created with data saved on a consent link object, i.e. provider, expiration date. If you still supply

EXAMPLE RESPONSE

```
{
  "id":
"bc_mtoK6Z9ZBtBLztQorkEm8c
  "object": "consent",
  "created_at": "2020-
01-23T10:17:03Z",
  "updated_at": null,
  "provider":
"santander",
  "vendor": null",
  "customer":
"cus_hjw5i1xO9zQK5kpS",
  "consent_link": null,
  "failure_code": null,
  "failure_message":
null,
  "expiry_date": null,
  "retro_date": null,
  "scopes": [
    "accounts",
    "balance"
  ],
  "authed_at": null,
  "status": "pending",
  "redirect": {
    "return_url":
"https://bigtechapp.com/da
callback",
    "consent_url":
"https://finexer.com/conne
consent=bc_mtoK6Z9ZBtBLztQ
  },
  "metadata": {
    "track_id":
"8096450047"
  }
}
```

the data in this request, this will override data saved on the consent link, ID of which is passed in the request.

expiry_date
optional

Time at which the consent will expire. If you leave this empty, or provide the date beyond the maximum allowed by a financial institution, the date will be automatically set to the maximum allowed date. The date is in universal **ISO 8601** format, (e.g. 2026-01-01).

retro_date
optional

The earliest time in the past from which account data can be pulled. For example, you can set today's date and obtain only transactions dated today and in the future. If you leave this empty, the date will be automatically set to seven

days in the
past. The
date is in
universal **ISO
8601** format,
(e.g. 2019-05-
25).

return_url optional	The URL you want the user to be redirected to after they authorize the request. Leave this empty and we'll redirect the user to one of your redirect URLs saved in your app, or to our confirmation page.
scopes optional	The list of scopes as a level of access to request from user. To request certain scopes, pass each in `scopes[]` parameter. To request all scopes, either pass asterisk `*` or omit this parameter in the request.
metadata optional	Set of arbitrary key/value pairs you can attach to the consent.

Returns the newly created **consent** object if the request succeeded. Returns an **error** otherwise.

Get a consent

Retrieves a single existing consent object that was previously created. You need only supply the unique consent `id` in the endpoint route, associated with the consent object you want to retrieve.

RESPONSE

Returns an existing **consent** object that matches provided consent ID if the request succeeded. Returns an **error** otherwise.

ENDPOINT

```
GET: /consents/{id}
```

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/co
\
    -u "API_KEY"
```

EXAMPLE RESPONSE

```
{
  "id":
"bc_mtoK6Z9ZBtBLztQorkEm8c
  "object": "consent",
  "created_at": "2020-
01-23T10:17:03Z",
  "updated_at": null,
  "provider":
"santander",
  "vendor": null",
  "customer":
"cus_hjw5i1xO9zQK5kpS",
  "consent_link": null,
  "failure_code": null,
  "failure_message":
null,
  "expiry_date": null,
  "retro_date": null,
  "authed_at": null,
  "status": "pending",
   ...
}
```

Update a consent

ENDPOINT

Updates the consent object by its id found on the **consent** object. It can be used to modify attributes on the consent, arguments of which were passed for this request. For example, you can create a new consent with natwest as a provider, however, update the consent with another provider.

Note if you need to clear an attribute's value, you can pass an empty value in an argument.

> Bare in mind, updating details on consent objects is possible for objects with new or pending statuses. However, attribute metadata can be updated anytime.

ARGUMENTS

provider *optional*	The identifier of supported **provider** object.
expiry_date *optional*	Time at which the consent will expire. The date is in universal **ISO 8601** format, (e.g. 2021-02-25).
retro_date *optional*	The earliest time in the past from which account

---

POST: /consents/{id}

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/con
\
    -u "API_KEY" \
    -d
provider="natwest" \
    -d
metadata[track_id]=""
```

EXAMPLE RESPONSE

```json
{
  "id":
"bc_mtoK6Z9ZBtBLztQorkEm8c
  "object": "consent",
  "created_at": "2020-
01-23T10:17:03Z",
  "updated_at": "2020-
01-23T10:22:42Z",
  "provider":
"natwest",
  "vendor": null",
  "customer":
"cus_hjw5i1xO9zQK5kpS",
  "consent_link": null,
  "failure_code": null,
  "failure_message":
null,
  "expiry_date": null,
  "retro_date": null,
  "authed_at": null,
  "status": "pending",
  ...
}
```

data can be pulled. The date is in universal **ISO 8601** format, (e.g. 2019-05-25).

scopes optional	The list of scopes as a level of access to request from user for this consent. To request certain scopes, pass each in `scopes[]` parameter, or asterisk `*` to request all scopes.
metadata optional	Set of arbitrary key/value pairs you can attach to the consent.

RESPONSE

Returns the **consent** object that has been updated if the request succeeded. Returns an **error** otherwise.

Revoke a consent

Revokes previously granted user access. Once revoked, Finexer will no longer be able to access the user's data at the financial institution. This action can also be used to cancel a pending consent that has not yet been authorized.

RESPONSE

Returns an existing and revoked **consent** object with status `canceled` if the request succeeded. Returns an **error** otherwise.

ENDPOINT

```
POST:
/consents/{id}/revoke
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/co:
\
      -u "API_KEY"
```

EXAMPLE RESPONSE

```
{
  "id":
"bc_mtoK6Z9ZBtBLztQorkEm8c:
  "object": "consent",
  "created_at": "2020-
01-23T10:17:03Z",
  "updated_at": "2020-
01-23T10:18:56Z",
  "provider":
"santander",
  "vendor": null",
  "customer":
"cus_hjw5i1xO9zQK5kpS",
  "consent_link": null,
  "failure_code": null,
  "failure_message":
null,
  "expiry_date": "2022-
01-22",
  "retro_date": "2020-
01-01",
  "authed_at": null,
  "status": "canceled",
  "metadata": {
    "track_id":
"8096450047"
  }
}
```

List consents

Retrieves a **paginated** list of consents you have previously

ENDPOINT

```
GET: /consents
```

created. The returned consents are sorted by creation date, with the most recently created consents appearing first.

QUERY PARAMETERS

provider optional	Only return consents for the provider specified by this provider ID.
vendor optional	Only return consents for the vendor specified by this vendor ID.
customer optional	Only return consents for the customer specified by this customer ID.
consent_link optional	Only return consents that are linked to the consent link specified by this consent link ID.
status optional	Limit to records whose status match the specified status value. List of available statuses can be found at **consent**

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/co
\
    -u "API_KEY"
```

EXAMPLE RESPONSE

```
{
  "count": 2,
  "location":
"/consents",
  "paging": {
    "total": 2,
    "next": null,
    "prev": null
  },
  "data": [
    {
      "id":
"bc_mtoK6Z9ZBtBLztQorkEm8c
      "object":
"consent",
      "created_at":
"2020-01-23T10:17:03Z",
      "updated_at":
"2020-01-23T10:18:56Z",
      "provider":
"santander",
      "customer":
"cus_hjw5i1xO9zQK5kpS",
      "consent_link":
null,
      "failure_code":
null,

"failure_message":
null,
      "expiry_date":
"2022-01-22",
      "retro_date":
"2020-01-01",
      "scopes": [
        "accounts",
        "balance"
      ],
      "authed_at":
null,
      "status":
"pending",
      "redirect": {
        "return_url":
```

object.

Expand ▸

created_at

optional

Represents
date time
universal
ISO 8601
format. (e.g.
2019-05-25
or 2019-05-
25T14:00:00)

To retrieve a complete set of
paginated results, you can use
pagination arguments too.

RESPONSE

Returns a list of existing **consent**
objects that match filter criteria if
the request succeeded. Returns an
error otherwise.

```
      "https://bigtechapp.com/da
callback",
        "consent_url":
"https://finexer.com/conne
consent=bc_mtoK6Z9ZBtBLztQ
      },
      "metadata": {
        "track_id":
"8096450047"
      }
    },
    {
      "id":
"cst_gsQhLVPCPCXEEXF4F5Nv6
      "object":
"consent",
      "created_at":
"2020-02-04T11:27:12Z",
      "updated_at":
"2020-02-04T11:27:12Z",
      "provider":
"lloyds",
      "customer":
"cus_hjw5i1xO9zQK5kpS",
      "consent_link":
null,
      ...
    }
  ]
}
```

Get consent QR code

Retrieves QR code for a consent. Note codes can be returned only for consents that are pending authorization, i.e. have `redirect` attribute set on a consent object. You need only supply the unique consent `id` in the endpoint route, associated with the consent object you want to get QR code for.

RESPONSE

Returns a simple object with relevant metadata if the request succeeded. Returns an **error** otherwise.

```
GET:
/consents/{id}/qr_code
```

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/co
\
    -u "API_KEY"
```

EXAMPLE RESPONSE

```
{
  "qr_code_url":
"https://blobs.finexer.com
}
```

Consent Links

Consent links are URLs that you can send through the API, or from your dashboard to your customers through email or any other means (like copying and pasting a link) to connect banks and account(s). Accessing the secure link will take your customers straight into the consent flow.

You can also create a new consent for an existing consent link by specifying the consent link ID in **create a consent** method. Consent links support multiple consents for a single consent link object.

ENDPOINTS

POST: **/consent_links**

GET: **/consent_links/{id}**

POST: **/consent_links/{id}**

GET: **/consent_links**

POST: **/consent_links/{id}/send**

GET: **/consent_links/{id}/qr_code**

Consent link object

ATTRIBUTES

id string	Unique identifier for consent link object.
object string	The name of the object: `consent_link`
created_at timestamp	Time indicating when the consent link was created.
vendor string `expandable`	The unique identifier of a **vendor** object the consent link is issued by, or issued for.
customer string `expandable`	The unique identifier of a **customer** object the consent link is issued for.
expires_at timestamp	Time at which the consent link gets expired.
provider string, nullable `expandable`	The unique identifier of a **provider** object the consent link is issued for.
Expand ▸	
redirect	Available

```json
{
  "id": "cl_hkAx3upaLaFaRlNvcth9Vc
  "object": "consent_link",
  "created_at": "2020-
09-22T20:38:16Z",
  "vendor": null,
  "customer":
"cus_qS5hHtII3XjLV5dB92EA"
  "expires_at": "2020-
11-22T20:38:16Z",
  "provider": null,
  "redirect": null,
  "type": "single",
  "expiry_days": 7,
  "retro_days": 7,
  "scopes": [
    "accounts",
    "balance",
    "transactions"
  ],
  "sent_at": null,
  "authed_at": "2020-
09-01",
  "status":
"authorized",
  "active": true,
  "metadata": {}
}
```

	object, nullable	only if the link is not expired or connected.
type	string	Indicates how bank connections can be made against the link. `single` - only a single bank connection can be made against the link; `multiple` - multiple bank connections can be made against the link until the link expires.
expiry_days	number, nullable	The number of days in the future when consent expires from the date when consent is created through the link. For example, if `expiry_days` is set to `2`, a consent created on 2021-01-01 will expire on 2021-01-03.
retro_days	number, nullable	The number of days in the past

from the date when consent is created through the link, account data can be accessed. For example, if `retro_days` is set to `2`, a consent created on 2021-01-01 will have access to transactions dated 2020-12-30 onwards.

scopes	The list of
list	scopes as a level of access new consents created through the link will have: `accounts` - account name, number, sort code, IBAN, currency, etc; `balance` - accrued balances on accounts' statements, currency, and credit line; `transactions` - transactions' date, reference, amount, status, code, etc.

sent_at timestamp, nullable	Time indicating when the link was sent to a user.	
authed_at timestamp, nullable	Time indicating when the link was authorized. Links with `type=multiple` will contain the time from a latest created consent.	
status string	The status of the consent link. Available statuses: `new` - the link is created; `sent` - the link is sent to a customer; `authorized` - a bank connection was made against the link.	
active boolean	Boolean value indicates if the consent URL of the link is active. If `false`, no data access consent can be created.	
metadata object, { }	Set of arbitrary key/value	

key/value
pairs you
can attach
to the
consent link.

Create a consent link

Creates a new consent link object.
Once created send it to a customer
by calling **send a consent link**
method, or send the URL supplied
in `redirect.consent_url` attribute through
any other means. Note the `redirect`
attribute is populated only when the
consent link object is not expired or
connected.

ARGUMENTS

customer optional	The ID of a **customer** object the new consent link will be issued for. Required if `vendor` attribute is not set.
vendor optional	The ID of a **vendor** object the new consent link will be issued by, or issued for, if the `customer` is not set. Required if `customer` attribute is not set.

ENDPOINT

```
POST: /consent_links
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/co
\
    -u "API_KEY" \
    -d
customer="cus_qS5hHtII3XjL
```

EXAMPLE RESPONSE

```
{
  "id":
"cl_hkAx3upaLaFaRlNvcth9Vc
  "object":
"consent_link",
  "created_at": "2020-
09-22T20:38:16Z",
  "vendor": null",
  "customer":
"cus_qS5hHtII3XjLV5dB92EA"
  "expires_at": "2020-
11-22T20:38:16Z",
  "provider": null,
  "redirect": {
    "return_url":
"https://bigtechapp.com/co
callback",
    "consent_url":
"https://finexer.com/c/T9j
  },
  "type": "single",
  "expiry_days": null,
  "retro_days": null,
  "scopes": [
    "accounts",
    "balance",
    "transactions"
```

```
  ],
  "sent_at": null,
  "authed_at": null,
  "status": "new",
  "active": true,
  "metadata": {}
}
```

provider optional	The ID of a **provider** object the new consent link will be issued for. Supplying a provider will lock the consent screen to only a single bank.
expires_at optional	Time at which a new consent link will expire. Expired links won't work however you can extend the expiry date by calling the **update a consent link** method.
return_url optional	The URL you want the user to be redirected to after they connect a bank. Leave this empty, and we'll redirect the user to one of your redirect URLs saved in your app, or to our confirmation page.

type optional	The type of the link to be created that indicates how bank connections will be made against the link. Default value is `single`.
expiry_days optional	The number of days in the future when consent expires from the date when a new consent is created through the link.
retro_days optional	The number of days in the past from the date when consent is created through the link, account data can be accessed.
scopes list	The list of scopes as a level of access to request from user. To request certain scopes, pass each in `scopes[]` parameter. To request all scopes,

	either pass asterisk `*` or omit this parameter in the request.
active optional	Boolean value indicates if the consent URL of the new link will be active. Defaults to `true`.
metadata optional	Set of arbitrary key/value pairs you can attach to the consent link object.

Returns the newly created **consent_link** object if the request succeeded. Returns an **error** otherwise.

Get a consent link

ENDPOINT

GET: /consent_links/{id}

Retrieves a single existing consent link object that was previously created. You need only supply the unique consent link `id` in the endpoint route, associated with the consent link object you want to retrieve.

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/co
\
    -u "API_KEY"
```

RESPONSE

Returns an existing **consent_link** object that matches provided consent link ID if the request succeeded. Returns an **error** otherwise.

EXAMPLE RESPONSE

```
{
  "id":
"cl_hkAx3upaLaFaRlNvcth9Vc
  "object":
"consent_link",
  "created_at": "2020-
09-22T20:38:16Z",
  "vendor": null",
  "customer":
"cus_qS5hHtII3XjLV5dB92EA"
  "expires_at": "2020-
11-22T20:38:16Z",
  "provider": null,
  "redirect": {
    "return_url":
"https://bigtechapp.com/co
callback",
    "consent_url":
"https://finexer.com/c/T9j
  },
  "type": "single",
  "expiry_days": null,
  "retro_days": null,
  "scopes": [],
  "sent_at": null,
  "authed_at": null,
  "status": "new",
  "active": true,
  "metadata": {}
}
```

Update a consent link

Updates an existing consent link

ENDPOINT

POST: /consent_links/{id}

object by its `id` found on the
consent_link object. It can be
used to modify attributes on the
consent link, arguments of which
were passed for this request. For
example, you can extend an
expiration date, or remove
metadata item by specifying an
empty string as a value.

> Note you can update all
> information on consent links that
> have no associated consent
> objects. If the link contains a
> consent object, the following
> attributes cannot be updated:
> `provider` , `type` . Updated
> information applies only to new
> consents.

ARGUMENTS

provider optional	The ID of a **provider** object that will lock the consent screen to only a single bank.
expires_at optional	Time at which an existing consent link will expire. Supply `now` if you want the consent link expire right away.
return_url optional	The URL you want the user to be redirected to when they

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/co
\
    -u "API_KEY" \
    -d
expires_at="2020-12-
31T23:59:59" \
    -d
metadata[track]="5001"
```

EXAMPLE RESPONSE

```
{
  "id":
"cl_hkAx3upaLaFaRlNvcth9Vc
  "object":
"consent_link",
  "created_at": "2020-
09-22T20:38:16Z",
  "vendor": null",
  "customer":
"cus_qS5hHtII3XjLV5dB92EA"
  "expires_at": "2020-
12-31T23:59:59",
  "provider": null,
  "redirect": {
    "return_url":
"https://bigtechapp.com/co
callback",
    "consent_url":
"https://finexer.com/c/T9j
  },
  "type": "single",
  "expiry_days": null,
  "retro_days": null,
  "scopes": [
    "accounts",
    "balance",
    "transactions"
  ],
  "sent_at": null,
  "authed_at": null,
  "status": "new",
  "active": true,
  "metadata": {
    "track": "5001"
  }
}
```

	connect a bank.
type optional	The type of the link to be updated that indicates how bank connections will be made against the link.
expiry_days optional	The number of days in the future when consent expires from the date when a new consent is created through the link.
retro_days optional	The number of days in the past from the date when consent is created through the link, account data can be accessed.
scopes list	The list of available scopes as a level of access new consents created through the link will have: `accounts` - account name, number, sort code, IBAN, currency.

etc; `balance` - accrued balances on accounts' statements, currency, and credit line; `transactions` - transactions' date, reference, amount, status, code, etc.

active optional	Boolean value indicates if the consent URL of the link is active. If `false`, no data access consent can be created.
metadata optional	Set of arbitrary key/value pairs you can attach to the consent link object.

Returns the **consent_link** object that has been updated if the request succeeded. Returns an **error** otherwise.

List consent links

ENDPOINT

Retrieves a **paginated** list of consent links you have previously created. The returned consent links are sorted by creation date, with the most recently created links appearing first.

QUERY PARAMETERS

vendor optional	Only return consent links that were issued by the vendor specified by this vendor ID.
customer optional	Only return consent links that were issued for the customer specified by this customer ID.
Expand ▸ created_at optional	Represents date time universal **ISO 8601** format. (e.g. 2019-05-25 or 2019-05-25T14:00:00)

To retrieve a complete set of paginated results, you can use **pagination** arguments too.

RESPONSE

Returns a list of existing **consent_link** objects that match

GET: /consent_links

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/co
created_at.gte=2020-09-
01&created_at.lte=2021-
01-31 \
    -u "API_KEY"
```

EXAMPLE RESPONSE

```
{
  "object": "list",
  "count": 1,
  "location":
"/consent_links",
  "paging": {
    "total": 56,
    "next": null,
    "prev": null
  },
  "data": [
    {
      "id":
"cl_hkAx3upaLaFaRlNvcth9Vc
      "object":
"consent_link",
      "created_at":
"2020-09-22T20:38:16Z",
      "vendor": null",
      "customer":
"cus_qS5hHtII3XjLV5dB92EA"
      "expires_at":
"2020-11-22T20:38:16Z",
      "provider": null,
      "redirect": null,
      "type": "single",
      "expiry_days": 7,
      "retro_days": 7,
      "scopes": [
        "accounts",
        "balance",
        "transactions"
      ],
      "sent_at": null,
      "authed_at":
"2020-09-01",
      "status":
"authorized",
      "active": true,
```

```
      "metadata": {}
    }
  ]
}
```

Send a consent link

Sends a consent link to a customer to connect their bank and bank account(s). You need only supply the unique consent link id in the endpoint route, associated with the consent link object you want to send.

ARGUMENTS

email optional	The email address to send a link to. If you leave this empty, we'll send a link to the email address saved on the **customer** object the consent link was issued for.
expires_at optional	Time at which a consent link will expire. You can extend expiration date giving your

ENDPOINT

```
POST:
/consent_links/{id}/send
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/co
\
    -u "API_KEY"
```

EXAMPLE RESPONSE

```
{
  "id":
"cl_hkAx3upaLaFaRlNvcth9Vc
  "object":
"consent_link",
  "created_at": "2020-
09-22T20:38:16Z",
  "vendor": null,
  "customer":
"cus_qS5hHtII3XjLV5dB92EA"
  "expires_at": "2020-
12-31T23:59:59Z",
  "provider": null,
  "redirect": {
    "return_url":
"https://bigtechapp.com/co
callback",
    "consent_url":
"https://finexer.com/c/T9j
  },
  "type": "single",
  "expiry_days": 7,
  "retro_days": 7,
  "scopes": [
    "accounts",
    "balance",
    "transactions"
  ],
  "sent_at": "2020-09-
```

| | customer more time to connect a bank. |
| notes optional | Additional notes to be appended in the email sent to your customer. |

```
Z4T10:50:19Z ,
  "authed_at": null,
  "status": "sent",
  "active": true,
  "metadata": {}
}
```

RESPONSE

Returns the **consent_link** object that has been sent if the request succeeded. Returns an **error** otherwise.

Get consent link QR code

ENDPOINT

```
GET:
/consents/{id}/qr_code
```

Retrieves QR code for a consent link. Note codes can be returned only for links that are active, i.e. have `redirect` attribute set on a consent link object. You need only supply the unique consent link `id` in the endpoint route, associated with the consent link object you want to get QR code for.

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/co
\
    -u "API_KEY"
```

RESPONSE

Returns a simple object with relevant metadata if the request succeeded. Returns an **error** otherwise.

EXAMPLE RESPONSE

```
{
  "qr_code_url":
"https://blobs.finexer.com
}
```

Bank Accounts

Bank account object holds the bank account details of a person or business. Bank accounts can be created via API or Dashboard, or when someone connects their bank with **consents**.

ENDPOINTS

POST: /bank_accounts

GET: /bank_accounts/{id}

POST: /bank_accounts/{id}

DELETE: /bank_accounts/{id}

GET: /bank_accounts

GET: /bank_accounts/{id}/balance

Bank account object

ATTRIBUTES

EXAMPLE OBJECT

id string	Unique identifier for bank account object.

```
{
  "id":
"ba_b7yr9Bp6dMnt5HC4Ah6bOWl
  "created_at": "2021-
06-02T19:05:33Z",
  "holder_name":
"William Damian",
  "nickname": "Classic
Account",
  "currency": "gbp",
```

object string	The name of the object: `bank_account`	
created_at timestamp	Time indicating when the bank account object was created.	
holder_name string, nullable	The name of the bank account holder. This may be empty for some institutions.	
nickname string, nullable	Alias of the bank account.	
Expand ▸ **identification** object	Contains bank account identification numbers tied to a single bank account.	
currency string	3-letter **ISO** currency code e.g., usd, gbp, eur, etc.	
type string	The type of the bank account e.g., `personal`, `business`, `unknown`.	
class string, nullable	The class of the bank account e.g., `current`	

```
    "identification": {
      "sort_code":
"002020",
      "account_number":
"24009020",
      "iban":
"GB910MDDL00202024009020",
      "pan":
"***********0912"
    },
    "type": "personal",
    "class": "current",
    "provider":
"lloyds_personal",
    "consent":
"bc_czAUKXsAoVxcwajXsdmyKB
    "vendor":
"vnd_PvL5q9ExrdO9g3GwSNju"
    "customer": null,
    "fingerprint":
"Hfzqiiux7oeVVSDp8omL",
    "metadata": {}
}
```

`savings` , `emoney` , etc.	
provider string, nullable `expandable`	The unique identifier of a supported **provider** object.
consent string, nullable	The unique identifier of a **consent** object the was used to authorize access to the bank account.
vendor string `expandable`	The unique identifier of a **vendor** object the bank account belongs to.
customer string `expandable`	The unique identifier of a **customer** object the bank account belongs to.
fingerprint string	Uniquely identifies this particular bank account. You can use this value to check whether two customers are using the same bank account, for example.
metadata	Set of

object, { }	arbitrary key/value pairs you can attach to the bank account. It can be useful for storing additional information about the bank account.

Create a bank account

Creates a new bank account object bound to an existing vendor or a customer.

> Note the bank account can be only created for an existing vendor, or a customer, the `id` of which you need to supply in the request.

ARGUMENTS

vendor *optional*	The unique identifier of an existing **vendor** object. Required if `customer` attribute is not set.
customer *optional*	The unique identifier of an existing **customer** object. Required if `vendor` attribute is

ENDPOINT

```
POST: /bank_accounts
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/ba
\
    -u "API_KEY" \
    -d
vendor="vnd_PvL5q9ExrdO9g3
\
    -d
holder_name="William
Damian" \
    -d
sort_code="302030" \
    -d
account_number="40109020"
\
    -d currency="gbp"
\
    -d type="personal"
```

EXAMPLE RESPONSE

```
{
  "id":
"ba_b7yr9Bp6dMnt5HC4Ah6bOW
  "created_at": "2021-
06-02T19:05:33Z",
```

	not set.
holder_name REQUIRED	The holder name of a bank account.
currency REQUIRED	3-letter **ISO** currency code e.g., gbp, eur, usd, etc.
sort_code REQUIRED	The 6 digit number for the bank account.
account_number REQUIRED	The account number for the bank account.
iban optional	The international bank account number for the bank account.
type optional	The type of the bank account. Possible values: personal , business , unknown .
provider optional	The unique identifier of a supported **provider** object. This must be supplied if you expect this customer to accept or make payments.

```
  "holder_name":
"William Damian",
  "nickname": "Classic
Account",
  "currency": "gbp",
  "identification": {
    "sort_code":
"002020",
    "account_number":
"24009020"
  },
  "type": "personal",
  "class": "current",
  "provider":
"lloyds_personal",
  "consent": null,
  "vendor":
"vnd_PvL5q9ExrdO9g3GwSNju"
  "customer": null,
  "fingerprint":
"Hfzqiiux7oeVVSDp8omL",
  "metadata": {}
}
```

| metadata optional | Set of arbitrary key/value pairs you can attach to the bank account. |

RESPONSE

Returns the **bank account** object that has been created if the request succeeded. Returns an **error** otherwise.

Get a bank account

ENDPOINT

```
GET: /bank_accounts/{id}
```

Retrieves the details of an existing
bank account, `id` of which is
passed in the endpoint route.

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/ba
\
    -u "API_KEY"
```

RESPONSE

Returns an existing **bank account**
object that matches provided bank
account ID if the request
succeeded. Returns an **error**
otherwise.

EXAMPLE RESPONSE

```
{
  "id":
"ba_b7yr9Bp6dMnt5HC4Ah6bOW
  "created_at": "2021-
06-02T19:05:33Z",
  "holder_name":
"William Damian",
  "nickname": "Classic
Account",
  "currency": "gbp",
  "identification": {
    "sort_code":
"002020",
    "account_number":
"24009020"
  },
  "type": "personal",
  "class": "current",
  "provider":
"lloyds_personal",
  "consent": null,
  "vendor":
"vnd_PvL5q9ExrdO9g3GwSNju"
  "customer": null,
  "fingerprint":
"Hfzqiiux7oeVVSDp8omL",
  "metadata": {}
}
```

Update a bank account

Updates an existing bank account
by its `id` found on the **bank
account** object. It can be used to

ENDPOINT

```
POST: /bank_accounts/{id}
```

account object. It can be used to modify attributes on the bank account, arguments of which were passed for this request.

ARGUMENTS

holder_name optional	The holder name of the bank account to be updated.
type optional	The type of the bank account. Possible values: `personal`, `business`, `unknown`.
metadata optional	Set of arbitrary key/value pairs you can attach to the bank account.

RESPONSE

Returns the **bank account** object that has been updated if the request succeeded. Returns an **error** otherwise.

Delete a bank account

```
curl -X POST
https://api.finexer.com/bar
 \
    -u "API_KEY" \
    -d
holder_name="William
Damian Junior" \
    -d
metadata[branch]="17
High Street"
```

EXAMPLE RESPONSE

```
{
  "id":
"ba_b7yr9Bp6dMnt5HC4Ah6bOWl
  "created_at": "2021-
06-02T19:05:33Z",
  "holder_name":
"William Damian
Junior",
  "nickname": "Classic
Account",
  "currency": "gbp",
  "identification": {
    "sort_code":
"002020",
    "account_number":
"24009020"
  },
  "type": "personal",
  "class": "current",
  "provider":
"lloyds_personal",
  "consent": null,
  "vendor":
"vnd_PvL5q9ExrdO9g3GwSNju"
  "customer": null,
  "fingerprint":
"Hfzqiiux7oeVVSDp8omL",
  "metadata": {
    "branch": "17 High
Street"
  }
}
```

Permanently deletes a bank account by its `id` found on the **bank account** object.

> Note associated transactions will be unavailable immediately.
>
> Accounts that are created for one-off payments cannot be deleted nor the accounts linked to the Finexer account.

```
DELETE:
/bank_accounts/{id}
```

EXAMPLE REQUEST

```
curl -X DELETE
https://api.finexer.com/ba
\
    -u "API_KEY"
```

RESPONSE

Returns an empty response with http status code of `204` if the request succeeded. Returns an **error** otherwise.

List bank accounts

Retrieves a **paginated** list of bank accounts that are associated with a vendor ID, or a customer ID the endpoint expects. The returned bank accounts are sorted by creation date, with the most recently created bank accounts appearing first.

QUERY PARAMETERS

| vendor optional | Only return bank accounts for the vendor specified by this vendor ID. Required if `customer` |

ENDPOINT

```
GET: /bank_accounts
```

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/ba
vendor=vnd_PvL5q9ExrdO9g3G
\
    -u "API_KEY"
```

EXAMPLE RESPONSE

```
{
  "count": 2,
  "location":
"/bank_accounts",
  "paging": {
    "total": 2,
    "next": null,
```

		attribute is not set.
customer optional	Only return bank accounts for the customer specified by this customer ID. Required if `vendor` attribute is not set.	
provider optional	Only return bank accounts for the provider specified by this provider ID.	
provider optional	Only return bank accounts for the provider specified by this provider ID.	
consent optional	Only return bank accounts that were shared with the consent specified by this consent ID.	
bank_account_monitor optional	Only return bank accounts that are monitored by an existing monitor specified by this bank account monitor ID	

```
      prev : null
  },
  "data": [
    {
      "id":
"ba_b7yr9Bp6dMnt5HC4Ah6bOWI
      "created_at":
"2021-06-02T19:05:33Z",
      "holder_name":
"William Damian
Junior",
      "nickname":
"Classic Account",
      "currency":
"gbp",
      "identification":
{
        "sort_code":
"002020",

"account_number":
"24009020"
      },
      "type":
"personal",
      "class":
"current",
      "provider":
"lloyds_personal",
      "consent": null,
      "vendor":
"vnd_PvL5q9ExrdO9g3GwSNju"
      "customer": null,
      "fingerprint":
"Hfzqiiux7oeVVSDp8omL",
      "metadata": {}
    },
    {...}
  ]
}
```

Expand ▸

created_at optional	Represents date time universal **ISO 8601** format. (e.g. 2019-05-25 or 2019-05-25T14:00:00)

To retrieve a complete set of paginated results, you can use **pagination** arguments too.

RESPONSE

Returns a list of existing **bank account** objects that match filter criteria if the request succeeded. Returns an **error** otherwise.

Get a bank account balance

Retrieves the list of balances for a bank account, `id` of which is passed in the endpoint route.

ATTRIBUTES

current decimal	Current balance on an account (does not include preauthorized amounts).
available decimal	Available balance on an account.
currency string	3-letter **ISO** currency code e.g., gbp, eur, etc.

ENDPOINT

```
GET:
/bank_accounts/{id}/balan
```

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/ba
\
    -u "API_KEY"
```

EXAMPLE RESPONSE

```
{
  "object": "list",
  "location":
"/bank_accounts/ba_VKCwA4v
  "count": 1,
  "data": [
    {
      "current":
12000.00
```

synced_at *timestamp*		The last time when the balance was synced with a financial institution.
Expand ▸ overdraft *object, nullable*		Contains overdraft information.
type *string*		The type of the balance. actual - amount of money that is at the disposal of the account owner; interim - subject to further changes during the business day.

```
12000.00,
     "available":
11590.00,
     "currency":
"gbp",
     "synced_at":
"2021-08-21T09:34:43Z",
     "overdraft": {
       "limit": 4999,
       "type":
"available"
     },
     "type": "actual"
   }
  ]
}
```

RESPONSE

Returns a **list object** object with balance objects if the request succeeded. Returns an **error** otherwise.

Bank Account Sync

Bank accounts which have been imported from financial institutions (with active consent) should be synced on a regular basis so they reflect up-to-date data (e.g. transactions).

We may sync certain bank accounts automatically, however, if you'd like to know when an account was synced last time or/and run a synchronization for an account, you can use endpoints below.

Bank account sync object

```
{
  "retro_date": "2021-
01-01",
  "synced_at": "2021-
10-01T21:30:450Z",
  "status": "idle"
}
```

ATTRIBUTES

retro_date timestamp, nullable	The data on account is synced with financial institution from this period of time.
synced_at timestamp, nullable	The data on account is synced with financial institution to this period of time. Indicates when last time the account was synced.
status string	The status indicating current synchronization state on account. It can be `idle`, or `running` (account being synced at the moment).

Get sync metadata

Retrieves details of synchronization state between a bank account and a financial institution. You need only supply the unique bank account id in the endpoint route.

RESPONSE

Returns a simple object with relevant metadata if the request succeeded. Returns an **error** otherwise.

ENDPOINT

```
GET:
/bank_accounts/{id}/sync
```

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/bar
\
    -u "API_KEY"
```

EXAMPLE RESPONSE

```
{
  "retro_date": "2021-01-01",
  "synced_at": "2021-10-01T21:30:450Z",
  "status": "idle"
}
```

Sync bank account

Instructs the API to synchronize a bank account, the `id` of which is supplied in the endpoint route. Note this endpoint can be used once every 15 minutes.

QUERY PARAMETERS

retro_date *optional*	The date from which a bank account must be synced. Represents date in universal **ISO 8601** format, e.g. 2021-06-01. Note this date should not be less than the date set in `retro_date` on a **consent** object.

RESPONSE

Returns a simple object with relevant metadata if the request succeeded. Returns an **error** otherwise.

ENDPOINT

```
POST:
/bank_accounts/{id}/sync
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/ba
\
    -u "API_KEY"
```

EXAMPLE RESPONSE

```
{
   "retro_date": "2021-
01-01",
   "synced_at": "2021-
10-01T21:30:450Z",
   "status": "running"
}
```

Bank Transactions

Transaction object holds the details about money transfer from one bank account to another or withint the account. API allows you to retrieve individual transactions as well as a list of all your transactions.

Transaction object

ATTRIBUTES

id string	Unique identifier for transaction object.
object string	The name of the object: transaction
timestamp timestamp	Time indicating when the transaction was created.
amount decimal	Amount belonging to the transaction.
currency string	3-letter **ISO** currency code e.g., usd, gbp, eur, etc.
type string	Indicates whether the transaction is a credit or a debit entry.
reference	Unique

EXAMPLE RESPONSE

```
{
  "id":
"trn_rJh8U6EvN9fiyVZgZVhXY"
  "object":
"transaction",
  "timestamp": "2019-
10-02T00:00:00Z",
  "amount": -23.61,
  "currency": "gbp",
  "type": "debit",
  "reference":
"BBYY4090",
  "description": "ELITE
GUITARIST
009879102020",
  "category":
"payments",
  "balance": 6705.7,
  "merchant": "ELITE
GUITARIST",
  "status": "booked",
  "metadata": {
    "party_name":
"SpaceX",
    "party_sort_code":
"002020",
    "party_number":
"12345678",
    "otp": "true",
    "etc": "..."
  }
}
```

string, nullable	reference for the transaction.
description string	Unstructured text for the transaction.
category string	One of listed categories such as `transfers`, `payments`, `direct_debit`, `atm`, etc.
balance decimal, nullable	Cash balance of the account the transaction is applied to.
merchant string, nullable	Name of the merchant the transaction is related with.
status string	Current status of the transaction. The status `pending` indicates that a transaction has been authorized at a bank. The status `booked` indicates that a transaction has been executed.
metadata object, { }	Additional metadata supplied by bank institution, or

set of
arbitrary
key/value
pairs you
can attach
to the
transaction.

Get a transaction

Retrieves the details of a single
existing transaction. You need only
supply the unique transaction `id` in
the endpoint route, associated with
the transaction object you want to
retrieve.

RESPONSE

Returns an existing **transaction**
object that matches provided
transaction ID if the request
succeeded. Returns an **error**
otherwise.

ENDPOINT

```
GET: /transactions/{id}
```

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/tra
\
    -u "API_KEY"
```

EXAMPLE RESPONSE

```
{
  "id":
"trn_rJh8U6EvN9fiyVZgZVhXY
  "object":
"transaction",
  "timestamp": "2019-
10-02T00:00:00Z",
  "amount": -23.61,
  "currency": "gbp",
  "type": "debit",
  "reference":
"BBYY4090",
  "description": "ELITE
GUITARIST
009879102020",
  "category":
"payments",
  "balance": 6705.7,
  "merchant": "ELITE
GUITARIST",
  "status": "booked",
  "metadata": {}
}
```

Update a transaction

Updates the transaction object by its `id` found on the transaction object. It can be used to modify metadata attributes on the transaction, arguments of which were passed for this request.

Note if you need to clear an attribute's value, you can pass an empty value in an argument.

ARGUMENTS

metadata **REQUIRED**	Set of arbitrary key/value pairs you can attach to the transaction.

RESPONSE

Returns the transaction object that has been updated if the request succeeded. Returns an error otherwise.

ENDPOINT

```
POST: /transactions/{id}
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/tra
\
    -u "API_KEY" \
    -d
metadata[note]="Card
expense by Richard"
```

EXAMPLE RESPONSE

```
{
  "id":
"trn_rJh8U6EvN9fiyVZgZVhXY
  "object":
"transaction",
  "timestamp": "2019-
10-02T00:00:00Z",
  "amount": -23.61,
  "currency": "gbp",
  "type": "debit",
  "reference":
"BBYY4090",
  "description": "ELITE
GUITARIST
009879102020",
  "category":
"payments",
  "balance": 6705.7,
  "merchant": "ELITE
GUITARIST",
  "status": "booked",
  "metadata": {
    "note": "Card
expense by Richard"
  }
}
```

List transactions

Retrieves a paginated list of

ENDPOINT

```
GET:
/bank_accounts/{bank_acco
```

transactions for the specified bank account ID `bank_account_id`. The returned transactions are sorted by creation date, with the most recently created transactions appearing first.

We return the header `fx-sync` with value `on` in the response if the specified bank account is being synchronized at the moment.

QUERY PARAMETERS

currency optional	3-letter **ISO** currency code e.g., gbp, eur, usd, etc.	
term optional	Only return transactions whose description contain this value.	
Expand ▸ amount optional	Represents transaction amount, decimal number format e.g., 0.50 or -0.50.	
Expand ▸ timestamp optional	Represents date time universal **ISO 8601** format. (e.g. 2019-05-25 or 2019-05-25T14:00:00)	

To retrieve a complete set of paginated results, you can use **pagination** arguments too.

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/ban
\
    -u "API_KEY"
```

EXAMPLE RESPONSE

```
{
  "count": 2,
  "location":
"/bank_accounts/ba_og9MCsQ[
  "paging": {
    "total": 2,
    "next": null,
    "prev": null
  },
  "data": [
    [
      {
        "id":
"trn_rJh8U6EvN9fiyVZgZVhXY"
        "object":
"transaction",
        "timestamp":
"2019-10-02T00:00:00Z",
        "amount":
-23.61,
        "currency":
"gbp",
        "type":
"debit",
        "reference":
"BBYY4090",
        "description":
"ELITE GUITARIST
009879102020",
        "category":
"payments",
        "balance":
6705.7,
        "merchant":
"ELITE GUITARIST",
        "status":
"booked",
        "metadata": {
          "note": "Card
expense by Richard"
        }
      },
      [
```

RESPONSE

Returns a list of existing transaction objects that match filter criteria if the request succeeded. Returns an error otherwise.

```
      "id":
"trn_sv3X777v39OXXeUqz6flc
      "object":
"transaction",
      "timestamp":
"2019-10-03T13:20:48Z",
      "amount":
-90.77,
      "currency":
"gbp",
      "type":
"debit",
      "reference":
null,
      "description":
"JD Sports Nike Air Max
2020",
      "category":
"payments",
      "balance":
6729.31,
      "merchant": "JD
Sports Limited",
      "status":
"booked",
      "metadata": {}
    }
  ]
  ]
}
```

Payments

The payment object is used to collect a payment or send money. For more information on how to authorize a payment, please visit our documentation page.

> You can retrieve bank account details used in the payment by expanding child attributes on the payment object. Please refer to Expandable section.

ENDPOINTS

POST: /payments

GET: /payments/{id}

POST: /payments/{id}

GET: /payments

POST: /payments/{id}/receipt_send

POST: /payments/{id}/receipt_history

GET: /payments/{id}/qr_code

POST: /payments/{id}/validate

POST: /payments/{id}/refunds

GET: /payments/{id}/refunds

Payment object

ATTRIBUTES

id string	Unique identifier for payment object.
object string	The name of the object: `payment`
created_at timestamp	Time indicating when the payment was created.
updated_at timestamp, nullable	Time at which the payment was last updated.
amount decimal	A positive integer representing the payment amount, e.g., `820.16`.
currency string	3-letter **ISO** currency code e.g., gbp, eur, usd, etc., representing the payment.
statement string	The reference that will appear at the user's bank statement. Maximum of 18 characters.
description string, nullable	An arbitrary string attached to a payment for your

EXAMPLE OBJECT

```json
{
  "id":
"pt_LiDYx1SaOBw57TOI9KPDlT
  "object": "payment",
  "created_at": "2020-
12-11T16:25:35Z",
  "updated_at": null,
  "amount": 2.40,
  "currency": "gbp",
  "statement": "digit
music tech",
  "description":
"Digital music download
sale",

"target_bank_account":
"ba_VHqmNr4cHDtW4ntYKUYK9r

"source_bank_account":
"ba_gvpyO1pxWptUiXSpPuju71
  "receipt_email":
"whiliams.great@acme.com",
  "receipt_number":
"2060-8012",
  "receipt_url":
"https://finexer.com/email
  "vendor": null,
  "customer":
"cus_hjw5i1xO9zQK5kpS",
  "initiator":
"customer",
  "provider": "lloyds",
  "payment_link": null,
  "payment_source":
null,
  "completed_at":
"2020-12-11T16:28:12Z",
  "failure_code": null,
  "failure_message":
null,
  "type": "payment",
  "exec_date": null,
  "status":
"completed",
  "metadata": {}
}
```

	records.
target_bank_account *string*	The unique identifier of the **bank account** object where money are sent to.
source_bank_account *string, nullable* `expandable`	The unique identifier of the **bank account** object where money are sent from.
receipt_email *string, nullable*	The email address where an automatically-generated receipt is sent to when the payment is completed.
receipt_number *string, nullable*	The user-friendly number that can be generated when the payment is completed, or provided by you.
receipt_url *optional, nullable*	The unique URL to the payment receipt that is generated when the payment has been completed.
vendor *string, nullable* `expandable`	The unique identifier of the **vendor** object that is associated with the payment, e.g. initiated a

	payment request.
customer string, nullable expandable	The unique identifier of the **customer** object that is charged.
initiator string	The party that makes a payment i.e. payer. Can be customer as a default value, or vendor .
provider string, nullable expandable	The unique identifier of the supported **provider** object the payment is processed by.
payment_link string, nullable	The unique identifier of the **payment_link** object the payment is linked to.
payment_source string, nullable	If not null, the unique identifier of the **payment** object representing a source payment current object is linked to. If the source payment has type batch , linked payments are processed simultaneously by the source payment i.e. batch.
completed_at	Time

timestamp, nullable	indicating when the payment was executed by a bank.
failure_code string, nullable	Error code if an error raised with the payment.
failure_message string, nullable	Detailed error message explaining reason if an error raised with the payment.
type string	The type of the payment. Can be `payment` as a default type, or `batch`.
beta exec_date timestamp	The date on which the payment is scheduled to be executed. The date is in universal **ISO 8601** format, (e.g. 2025-10-10)
status string	The status of the payment. Available statuses: `pending` - payment was created and is waiting until the user authorizes it; `blocked` - payment was blocked due to suspicious activity; `authorized` - authorization completed, the

user approved the payment; `completed` - payment was processed by a financial institution; `failed` - payment failed to complete due to an occurred error; `canceled` - payment was canceled by the user or rejected by a bank.

Expand ▸		
	redirect	See child attributes
	object, nullable	This object is returned only when the payment status is `pending`.
	metadata	Set of arbitrary key/value pairs you can attach to the payment. It can be useful for storing additional information about the payment.
	object, { }	

Create a payment

Creates a new payment object. Once created, direct the user to the URL supplied in `redirect.consent_url` attribute to authorize the payment.

You can supply full bank account details for each new payment, or

ENDPOINT

```
POST: /payments
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/pay
\
```

use the `id` of an existing bank account bound to a customer object, or saved in your Finexer account. If none is supplied, we'll use a bank account saved in your Finexer account. You cannot reuse the id of full bank account details supplied in another payment.

> Note the `redirect` attribute is populated only when the payment object is in the `pending` state.

ARGUMENTS

amount REQUIRED	A positive integer representing the payment amount.
currency REQUIRED	3-letter **ISO** currency code e.g., gbp, eur, usd, etc., representing the payment.
statement REQUIRED	The reference that will appear at the user's bank statement. Must contain only letters, numbers, and be between 5 and 18 characters long. For the payment batch, existing payments with the same reference in `pending` state will be linked to the new payment

```
    -u "API_KEY" \
    -d amount=2.40 \
    -d currency="gbp"
\
    -d
statement="digit music
tech" \
    -d
description="Digital
music download sale" \
    -d
receipt_email="whiliams.gr
\
    -d
target_bank_account[holder
Limited" \
    -d
target_bank_account[sort_c
\
    -d
target_bank_account[accoun
\
    -d
provider="natwest" \
    -d
customer="cus_hjw5i1xO9zQK
\
    -d
metadata[order]="16102020A
```

EXAMPLE RESPONSE

```
{
  "id":
"pt_LiDYx1SaOBw57TOI9KPDlT
  "object": "payment",
  "created_at": "2020-
12-11T16:25:35Z",
  "updated_at": null,
  "amount": 2.40,
  "currency": "gbp",
  "statement": "digit
music tech",
  "description":
"Digital music download
sale",

"target_bank_account":
"ba_VHqmNr4cHDtW4ntYKUYK9r
  "receipt_email":
"whiliams.great@acme.com",
  "receipt_number":
null,
  "receipt_url": null,
  "vondor": null
```

	batch.
description *optional*	An arbitrary string attached to a payment for your records.
provider *optional*	Supply the id of a supported **provider** if you already know your customer intends to make a payment from a specific bank. Leave this empty to enable your customer to select any bank on the payment page.
vendor *optional*	The ID of the **vendor** object that will be associated with the new payment.
customer *optional*	The ID of the **customer** (payer) who will authorize a new payment.
initiator *optional*	Specify `customer` (default) if the payment will be made by a customer. Specify `vendor`, if the payment will be made by a vendor.

```
  "vendor" : null,
  "customer":
"cus_hjw5i1xO9zQK5kpS",
  "initiator":
"customer",
  "provider":
"natwest",
  "payment_link": null,
  "payment_source":
null,
  "completed_at": null,
  "failure_code": null,
  "failure_message":
null,
  "type": "payment",
  "exec_date": null,
  "status": "pending",
  "redirect": {
    "return_url":
"https://bigtechapp.com/pay
callback",
    "consent_url":
"https://finexer.com/conne
consent=pc_9XS1ZsVuOxSVaaT
  },
  "metadata": {
    "order":
"16102020ABC"
  }
}
```

payment_link optional		The ID of an existing **payment_link** object the new payment will be attributed to. If you provide the payment link ID, data for attributes marked as **REQUIRED** can be omitted and a new payment will be created with data saved on a payment link object. If you still supply the data in this request, this will override data saved on the payment link, ID of which is passed in the request.
payment_source optional		The ID of an existing **payment** object with the type `batch` the new payment will be part of and processed by.
type optional		The type of the new payment. Can be either `payment` or `batch`. For payment batches, `amount`, `currency` and `statement` parameters are optional.

beta	exec_date optional	The date on which the new payment will be executed by a bank. The date is in universal **ISO 8601** format, (e.g. 2025-10-10)
	Expand ▸ target_bank_account optional	The unique identifier of an existing **bank account** object where funds will be transfered to. This can be a bank account of one of your customers, or your own saved in your Finexer account. The bank account should not belong to the same customer passed in the ⌷customer⌷ parameter. Or, leave it empty and we'll use one of your bank accounts saved in your Finexer account. Or, pass bank account details via child attributes. See child attributes
	Expand ▸ source_bank_account optional	The unique identifier of an existing **bank account**

object where money will be sent from. This can be a bank account of one of your customers, or your own saved in your Finexer account. The bank account should belong to the same customer passed in the `customer` parameter. Or, pass bank account details via child attributes. See child attributes

receipt_email optional	The email address where an automatically-generated receipt will be sent to when the payment is completed.
receipt_number optional	The user-friendly number that will be attached to a payment receipt when the payment is completed. Maximum of 24 characters. Leave it empty, and we'll generate an easy-readable number.
return_url	The URL you

	optional	want the user to be redirected to after they authorize the payment. Leave this empty, and we'll redirect the user to one of your redirect URLs saved in your app, or to our confirmation page.
metadata	optional	Set of arbitrary key/value pairs you can attach to the payment.

Returns the newly created **payment** object if the request succeeded. Returns an **error** otherwise.

Get a payment

Retrieves a single existing payment object that was previously created. You need only supply the unique payment `id` in the endpoint route, associated with the payment object you want to retrieve.

Returns an existing **payment**

ENDPOINT

```
GET: /payments/{id}
```

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/pa
\
    -u "API_KEY"
```

EXAMPLE RESPONSE

object that matches provided payment ID if the request succeeded. Returns an **error** otherwise.

```
{
  "id":
"pt_LiDYx1SaOBw57TOI9KPDlT
  "object": "payment",
  "created_at": "2020-
12-11T16:25:35Z",
  "updated_at": null,
  "amount": 2.40,
  "currency": "gbp",
  "statement": "digit
music tech",
  "description":
"Digital music download
sale",

"target_bank_account":
"ba_VHqmNr4cHDtW4ntYKUYK9r
"source_bank_account":
"ba_gvpyO1pxWptUiXSpPuju71
  "receipt_email":
"whiliams.great@acme.com",
  "receipt_number":
null,
  "receipt_url": null,
  "vendor": null,
  "customer":
"cus_hjw5i1xO9zQK5kpS",
  "initiator":
"customer",
  "provider":
"natwest",
  "payment_link":
"pl_y48R75HYhHMFKsBm8ltbNF
  "payment_source":
null,
  "completed_at": null,
  "failure_code": null,
  "failure_message":
null,
  "type": "payment",
  "exec_date": null,
  "status": "pending",
  "redirect": {
    "return_url":
"https://bigtechapp.com/pa
callback",
    "consent_url":
"https://finexer.com/conne
consent=pc_9XS1ZsVuOxSVaaT
  },
  "metadata": {
    "order":
"16102020ABC"
  }
}
```

Update a payment

Updates the payment object by its `id` found on the **payment** object. It can be used to modify attributes on the payment, arguments of which were passed for this request. For example, you can create a new payment with `natwest` as a provider and update the payment with another provider.

Note if you need to clear an attribute's value, you can pass an empty value in an argument.

> Bare in mind, updating details on payment objects is possible for objects with `pending` status. However, attribute `metadata` can be updated anytime.

ARGUMENTS

amount optional	A positive integer representing the payment amount.
provider optional	The identifier of supported **provider** object.
payment_source optional	The ID of an existing **payment** object with the type `batch`.
description optional	An arbitrary string

ENDPOINT

> **POST: /payments/{id}**

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/pa
\
    -u "API_KEY" \
    -d
metadata[oto]="true"
```

EXAMPLE RESPONSE

```
{
  "id":
"pt_LiDYx1SaOBw57TOI9KPDlT
  "object": "payment",
  "created_at": "2020-
12-11T16:25:35Z",
  "updated_at": null,
  "amount": 2.40,
  "currency": "gbp",
  "statement": "digit
music tech",
  "description":
"Digital music download
sale",

"target_bank_account":
"ba_VHqmNr4cHDtW4ntYKUYK9r

"source_bank_account":
"ba_gvpyO1pxWptUiXSpPuju71
  "receipt_email":
"whiliams.great@acme.com",
  "receipt_number":
"1612-6020",
  "receipt_url":
"https://finexer.com/email
  "vendor": null,
  "customer":
"cus_hjw5i1xO9zQK5kpS",
  "initiator":
"customer",
  "provider":
"natwest",
  "payment_link": null,
```

```
    "payment_source":
null,
    "completed_at":
"2020-12-11T16:28:12Z",
    "failure_code": null,
    "failure_message":
null,
    "type": "payment",
    "exec_date": null,
    "status":
"completed",
    "metadata": {
      "oto": "true",
      "order":
"16102020ABC"
    }
}
```

exec_date optional	The date on which the existing payment will be executed by a bank. The date is in universal **ISO 8601** format, (e.g. 2025-10-10)
metadata optional	Set of arbitrary key/value pairs you can attach to the payment.

RESPONSE

Returns the **payment** object that has been updated if the request succeeded. Returns an **error** otherwise.

List payments

Retrieves a **paginated** list of payments you have previously created. The returned payments are sorted by creation date, with the most recently created payments appearing first.

QUERY PARAMETERS

provider optional	Only return payments

ENDPOINT

```
GET: /payments
```

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/pay
created_at.gte=2020-12-
01&created_at.lte=2020-
12-31 \
    -u "API_KEY"
```

	for the provider specified by this provider ID.
vendor optional	Only return payments for the vendor specified by this vendor ID.
customer optional	Only return payments for the customer specified by this customer ID.
payment_link optional	Only return payments that are linked to the payment link specified by this payment link ID.
payment_source optional	Only return payments that are linked to the payment source specified by this source payment ID. For example, return the list of payments linked to a single batch, or list of refunds made against the payment

EXAMPLE RESPONSE

```json
{
  "count": 2,
  "location":
"/payments",
  "paging": {
    "total": 2,
    "next": null,
    "prev": null
  },
  "data": [
    {
      "id":
"pt_LiDYx1SaOBw57TOI9KPDlT:
      "object":
"payment",
      "created_at":
"2020-12-11T16:25:35Z",
      "updated_at":
null,
      "amount": 2.40,
      "currency":
"gbp",
      "statement":
"digit music tech",
      "description":
"Digital music download
sale",

"target_bank_account":
"ba_VHqmNr4cHDtW4ntYKUYK9r:

"source_bank_account":
"ba_gvpyO1pxWptUiXSpPuju71q
      "receipt_email":
"whiliams.great@acme.com",
      "receipt_number":
"2060-8012",
      "receipt_url":
"https://finexer.com/email:
      "vendor": null,
      "customer":
"cus_hjw5i1xO9zQK5kpS",
      "initiator":
"customer",
      "provider":
"natwest",
      "payment_link":
"pl_y48R75HYhHMFKsBm8ltbNF(
      "payment_source":
null,
      "completed_at":
"2020-12-11T16:28:12Z",
      "failure_code":
null,
```

		source.
status optional		Limit to records whose status match the specified status value. List of available statuses can be found at **payment** object.
currency optional		Limit to records whose currency match the specified 3-letter **ISO** currency code e.g., gbp, eur, usd, etc.
Expand ▸ amount optional		A positive integer representing the payment amount, e.g., 820.16
Expand ▸ created_at optional		Represents date time universal **ISO 8601** format. (e.g. 2019-05-25 or 2019-05-25T14:00:00)

To retrieve a complete set of paginated results, you can use **pagination** arguments too.

RESPONSE

```
    "failure_message":
null,
    "type":
"payment",
    "exec_date":
null,
    "status":
"completed",
    "metadata": {
        "oto": "true",
        "order":
"16102020ABC"
    }
  },
  {...}
  ]
}
```

Returns a list of existing **payment** objects that match filter criteria if the request succeeded. Returns an **error** otherwise.

Send a receipt

Sends a payment receipt to the user when a payment is in completed status.

```
POST:
/payments/{id}/receipt_se
```

ARGUMENTS

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/pa
\
    -u "API_KEY"
```

receipt_email optional	The email address to send the receipt to. If you leave it empty, we'll send the receipt to the email address saved on the **payment** object.

EXAMPLE RESPONSE

```
{
  "sent_at": "2020-12-
22T12:15:56Z",
  "receipt_email":
"whiliams.great@acme.com",
  "receipt_number":
"2060-8012",
  "receipt_url":
"https://finexer.com/email:
}
```

RESPONSE

Returns a simple object that contains relevant metadata if the request succeeded. Returns an **error** otherwise.

List receipt history

```
GET:
/payments/{id}/receipt_hi
```

Retrieves a list of sent payment receipts that have been previously sent by you, or automatically. The returned receipts are sorted by creation date, with the most recently sent receipts appearing first.

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/pay
\
    -u "API_KEY"
```

RESPONSE

Returns a list of simple objects with relevant metadata if the request succeeded. Returns an **error** otherwise.

EXAMPLE RESPONSE

```
{
  "count": 1,
  "location":
"/payments/pt_LiDYx1SaOBw5
  "data": [
    {
      "sent_at": "2020-
12-22T12:15:56Z",
      "receipt_email":
"whiliams.great@acme.com",
      "receipt_number":
"2060-8012",
      "receipt_url":
"https://finexer.com/email
    },
    {...}
  ]
}
```

Get payment QR code

Retrieves QR code for a payment. Note codes can be returned only for payments that are pending authorization, i.e. have `redirect` attribute set on a payment object. You need only supply the unique payment `id` in the endpoint route, associated with the payment object you want to get QR code for.

RESPONSE

Returns a simple object with relevant metadata if the request succeeded. Returns an **error** otherwise.

Validate a payment   beta

Performs preflight validation checks on the payment, `id` of which is supplied in the endpoint route. This endpoint returns no content in cases when validation checks have been passed.

> Note the validation checks on this endpoint are identical to the checks enforced when a user is sent to the consent screen, the URL of which is found on the `redirect.consent_url` attribute on the **payment** object. This allows you to verify whether the request meets all requirements before sending the user to the customer payment flow.

ARGUMENTS

ENDPOINT

```
GET:
/payments/{id}/qr_code
```

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/pay
\
    -u "API_KEY"
```

EXAMPLE RESPONSE

```
{
  "qr_code_url":
"https://blobs.finexer.com
}
```

ENDPOINT

```
POST:
/payments/{id}/validate
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/pay
\
    -u "API_KEY" \
    -d
provider="natwest" \
    -d
user_ip="52.128.62.82"
\
    -d
user_agent="Mozilla/5.0
(Windows NT 10.0;
Win64; x64)
AppleWebKit/537.36
(KHTML, like Gecko)
Chrome/90.0.4430.93
Safari/537.36"
```

ARGUMENTS

user_ip string, **REQUIRED**	A valid IP address of the user that intends to initiate a payment.
user_agent string, **REQUIRED**	A valid user agent (e.g. web browser) of the user that intends to initiate a payment.
amount decimal, optional	A positive integer representing the payment amount.
statement string, optional	Reference that appears on the bank statements.
provider string, optional	An identifier of the supported **provider** object.
read_source boolean, optional	Retrieve source bank account for the purposes of making refunds. [**more info**].
live_check boolean, optional	Performs a live request to the associated provider when set to `true`.

RESPONSE

Returns the status code 204 if the request succeeded. Returns a validation **error** otherwise.

Create a refund

Creates a new payment object with type refund . Once created, direct the user (e.g. vendor who previously received funds) to the URL supplied in redirect.consent_url attribute to authorize the refund.

The refund must be made to the same bank account with the same payment reference. If the bank account wasn't captured during the original payment, you should supply bank account details in the request. For more information on how to capture the source bank account, please visit our **documentation** page.

ARGUMENTS

amount	A positive
REQUIRED	integer
	representing
	the refund
	amount. It
	can be less
	or equal to
	the amount of
	the source
	payment.

Expand ▸

target_bank_account	The unique
optional	identifier of
	an existing
	bank
	account
	object where

ENDPOINT

```
POST:
/payments/{id}/refunds
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/pa
\
    -u "API_KEY" \
    -d amount=10.90 \
    -d currency="gbp"
\
    -d
description="Partial
refund due to"
```

EXAMPLE RESPONSE

```
{
  "id":
"bp_H1CMItIpHa63YiOMkPnJtj
  "object": "payment",
  "created_at": "2024-
01-16T10:40:06Z",
  "updated_at": null,
  "amount": 10.90,
  "currency": "gbp",
  "statement": "pencil
box",
  "description":
"Partial refund due
to",

"target_bank_account":
"ba_VHqmNr4cHDtW4ntYKUYK9r
  "receipt_email":
null,
  "receipt_number":
null,
  "receipt_url": null,
```

		funds will be transfered to. By default we'll use the bank account the source payment was initiated from if one exists, otherwise you should supply bank account details via child attributes.
description optional		An arbitrary string attached to a refund for your records.
receipt_email optional		The email address where an automatically-generated receipt will be sent to when the refund is completed.
receipt_number optional		The user-friendly number that will be attached to a refund receipt when the refund is completed. Maximum of 24 characters. Leave it empty, and we'll generate an easy-readable number.
return_url optional		The URL you want the user to be redirected to

```json
  "vendor": null,
  "customer": null,
  "initiator":
"vendor",
  "provider": null,
  "payment_link": null,
  "payment_source":
"bp_LiDYx1SaOBw57TOI9KPDlT
  "completed_at": null,
  "failure_code": null,
  "failure_message":
null,
  "type": "refund",
  "status": "pending",
  "redirect": {
    "return_url":
"https://bigtechapp.com/pa
callback",
    "consent_url":
"https://finexer.com/conne
consent=bp_LiDYx1SaOBw57TO
  },
  "metadata": {}
}
```

after they authorize the refund. Leave this empty, and we'll redirect the user to one of your redirect URLs saved in your app, or to our confirmation page.

metadata *optional*	Set of arbitrary key/value pairs you can attach to the refund.	

RESPONSE

Returns the newly created **payment** object if the request succeeded. Returns an **error** otherwise.

List refunds

Retrieves a **paginated** list of refunds you have previously created. You need only supply the unique payment `id` in the endpoint route, associated with the payment source object you want to get refunds for.

QUERY PARAMETERS

provider *optional*	Only return payments for the provider specified by	

ENDPOINT

```
GET: /payments
```

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/pay
\
    -u "API_KEY"
```

EXAMPLE RESPONSE

```
{
  "count": 2,
```

		specified by this provider ID.
status	optional	Limit to records whose status match the specified status value. List of available statuses can be found at **payment** object.
Expand ▸		
amount	optional	A positive integer representing the payment amount, e.g., 820.16
Expand ▸		
created_at	optional	Represents date time universal **ISO 8601** format. (e.g. 2019-05-25 or 2019-05-25T14:00:00)

To retrieve a complete set of paginated results, you can use **pagination** arguments too.

RESPONSE

Returns a list of existing **payment** objects that match filter criteria if the request succeeded. Returns an **error** otherwise.

```
"location":
"/payments/bp_VJd2SDmR65fX
  "paging": {
    "total": 2,
    "next": null,
    "prev": null
  },
  "data": [
    {
      "id":
"bp_H1CMItIpHa63YiOMkPnJtj
      "object":
"payment",
      "created_at":
"2024-01-16T10:40:06Z",
      "updated_at":
null,
      "amount": 10.90,
      "currency":
"gbp",
      "statement":
"pencil box",
      "description":
"Partial refund due
to",

"target_bank_account":
"ba_VHqmNr4cHDtW4ntYKUYK9r
      "receipt_email":
null,
      "receipt_number":
null,
      "receipt_url":
null,
      "vendor": null,
      "customer": null,
      "initiator":
"vendor",
      "provider": null,
      "payment_link":
null,
      "payment_source":
"bp_LiDYx1SaOBw57TOI9KPDlT
      "completed_at":
null,
      "failure_code":
null,

"failure_message":
null,
      "type": "refund",
      "status":
"pending",
      "redirect": {
        "return_url":
"https://bigtechapp.com/pa
callback",
        "consent_url":
"http://fx.local/connect/o
```

consent=bp_LiDYx1SaOBw57TO:
    },
    "metadata": {}
  },
  {...}
  ]
}

Payment Links

Payment links are URLs that you can send through the API, or from your dashboard to your customers through email or any other means (like copying and pasting a link) to collect payments. Accessing the secure link will take your customers straight into the payment flow.

You can also create a new payment for an existing payment link by specifying the payment link ID in **create a payment** method. Payment links support multiple payments for a single payment link object.

Payment link object

ATTRIBUTES

id string	Unique identifier for payment link object.
object string	The name of the object: `payment_link`
created_at	Time

EXAMPLE OBJECT

```
{
  "id":
"pl_y48R75HYhHMFKsBm8ltbNF
  "object":
"payment_link",
  "created_at": "2020-
10-20T11:58:41Z",
  "amount": 100.20,
  "amount_paid":
100.20,
  "currency": "gbp",
  "statement": "CL-INV-
065002",
  "description":
"Invoice paid by
```

timestamp	indicating when the payment link was created.
amount decimal	A positive integer representing the payment amount, e.g., `820.16`.
amount_paid decimal	A positive integer representing an allocated amount to the payment link.
currency string	3-letter **ISO** currency code e.g., gbp, eur, usd, etc., representing the payment link.
statement string	The reference that will appear at the user's bank statement. Maximum of 18 characters.
description string, nullable	An arbitrary string attached to a payment for your records.
vendor string expandable	The unique identifier of a **vendor** the payment link is issued by.
customer string expandable	The unique identifier of a **customer** object the payment link

```
David",
    "vendor": null,
    "customer":
"cus_qS5hHtII3XjLV5dB92EA"
    "receipt_email":
null,
    "expires_at": "2020-
11-20T13:19:36Z",

"target_bank_account":
null,
    "provider": null,
    "redirect": null,
    "sent_at": "2020-10-
20T12:59:28Z",
    "paid_at": "2020-10-
20T15:00:44Z",
    "type": "single",
    "status": "paid",
    "active": true,
    "metadata": {}
}
```

	payment link is issued for.
receipt_email string, nullable	The email address where an automatically-generated receipt is sent to for each completed payment.
expires_at timestamp	Time at which the payment link gets expired.
target_bank_account string, nullable	The unique identifier of the **bank account** object where money are sent to.
provider string expandable	The unique identifier of a **provider** object the payment link is issued for.
Expand ▸ redirect object, nullable	Available only if the link is not expired or fully paid.
sent_at timestamp, nullable	Time indicating when the link was sent to a user.
paid_at timestamp, nullable	Time indicating when the link has been paid the last time.
type string	Indicates how payments can be attributed to a

attributed to a link.
`single` - only a single payment with fixed amount can be made against the link;
`multiple` - multiple payments with fixed amount can be made against the link until the link expires;
`variable` - multiple payments with variable amount a customer wish to pay (e.g. donations) can be made against the link until the link expires. Variable amount must not exceed the value in the amount field.

status string	The status of the payment link. Available statuses: `new` - the link is created; `sent` - the link is sent to a customer; `paid` - the link is fully paid; `paid_part` - the link is partially paid.
active	Boolean

boolean		value indicates if the consent URL of the link is active. If false , no payment can be initiated.
metadata object, { }		Set of arbitrary key/value pairs you can attach to the payment link.

Create a payment link

Creates a new payment link object. Once created send it to a customer by calling **send a payment link** method, or send the URL supplied in redirect.consent_url attribute through any other means. Note the redirect attribute is populated only when the payment link object is not expired or fully paid.

ARGUMENTS

amount REQUIRED	A positive integer representing the payment amount for the payment link to be created.
currency REQUIRED	3-letter **ISO** currency code e.g., gbp, eur, usd, etc., representing the payment

ENDPOINT

```
POST: /payment_links
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/pa
\
    -u "API_KEY" \
    -d amount=100.20 \
    -d currency="gbp"
\
    -d statement="CL-
INV-065002" \
    -d
customer="cus_qS5hHtII3XjL
\
    -d
description="Invoice
paid by David"
```

EXAMPLE RESPONSE

```
{
  "id":
"pl_y48R75HYhHMFKsBm8ltbNF
  "object":
"payment_link",
  "created_at": "2020-
10 20T11:58:41Z"
```

		link to be created.
statement **REQUIRED**		The reference that will appear at the user's bank statement. Maximum of 18 characters.
customer optional		The ID of a **customer** object the new payment link will be issued for.
vendor optional		The ID of a **vendor** the new payment link will be issued by.
description optional		An arbitrary string attached to a payment for your records.
provider optional		The ID of a **provider** object that a new payment link will be issued for. Supplying a provider will lock the consent screen to only a single bank.
expires_at optional		Time at which a new payment link will

```
10 2011170170112 ,
  "amount": 100.20,
  "amount_paid": 0.00,
  "currency": "gbp",
  "statement": "CL-INV-
065002",
  "description":
"Invoice paid by
David",
  "vendor": null,
  "customer":
"cus_qS5hHtII3XjLV5dB92EA"
  "receipt_email":
null,
  "expires_at": "2020-
11-20T13:19:36Z",

"target_bank_account":
null,
  "provider": null,
  "redirect": {
    "return_url":
"https://bigtechapp.com/pay
callback",
    "consent_url":
"https://finexer.com/p/eej]
  },
  "type": "single",
  "sent_at": "2020-10-
20T12:59:28Z",
  "paid_at": null,
  "status": "new",
  "active": true,
  "metadata": {}
}
```

expire. Expired links won't work however you can extend the expiry date by calling the **update a payment link** method.

target_bank_account *optional*	The unique identifier of an existing **bank account** object where money will be sent to for each payment made with the link to be created. If you leave this empty, we'll use a default bank account stored in a Finexer account.
return_url *optional*	The URL you want the user to be redirected to when they authorize the payment. Leave this empty, and we'll redirect the user to one of your redirect URLs saved in your app, or to our

confirmation page.

type optional	The type of the link to be created that indicates how payments are attributed to the new link. Default value is `single`.	
active optional	Boolean value indicates if the consent URL of the new link will be active. Defaults to `true`.	
metadata optional	Set of arbitrary key/value pairs you can attach to the payment link object.	

RESPONSE

Returns the newly created **payment_link** object if the request succeeded. Returns an **error** otherwise.

Get a payment link

Retrieves a single existing payment link object that was previously

ENDPOINT

`GET: /payment_links/{id}`

created. You need only supply the unique payment link `id` in the endpoint route, associated with the payment link object you want to retrieve.

RESPONSE

Returns an existing **payment_link** object that matches provided payment link ID if the request succeeded. Returns an **error** otherwise.

```
curl -X GET
https://api.finexer.com/pay
\
    -u "API_KEY"
```

```
{
  "id":
"pl_y48R75HYhHMFKsBm8ltbNF
  "object":
"payment_link",
  "created_at": "2020-
10-20T11:58:41Z",
  "amount": 100.20,
  "amount_paid": 0.00,
  "currency": "gbp",
  "statement": "CL-INV-
065002",
  "description":
"Invoice paid by
David",
  "vendor": null,
  "customer":
"cus_qS5hHtII3XjLV5dB92EA"
  "receipt_email":
null,
  "expires_at": "2020-
11-20T13:19:36Z",

"target_bank_account":
null,
  "provider": null,
  "redirect": {
    "return_url":
"https://bigtechapp.com/pa
callback",
    "consent_url":
"https://finexer.com/p/eej
  },
  "sent_at": null,
  "paid_at": null,
  "type": "single",
  "status": "new",
  "active": true,
  "metadata": {}
}
```

Update a payment link

Updates an existing payment link object by its `id` found on the **payment_link** object. It can be used to modify attributes on the payment link, arguments of which were passed for this request. For example, you can change the amount, or extend an expiration date on an existing payment link object.

Note you can update payment links that have no attributed payment objects, or its metadata at all times.

ARGUMENTS

amount optional	A positive integer representing the payment amount for the payment link to be updated.	
currency optional	3-letter **ISO** currency code e.g., gbp, eur, usd, etc., representing the payment link to be updated.	
statement optional	The reference that will appear at the user's bank statement. Maximum of 18 characters.	

ENDPOINT

```
POST: /payment_links/{id}
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/pay
\
    -u "API_KEY" \
    -d statement="CL-
INV-065002-UPD" \
    -d
expires_at="2020-12-
31T23:59:59" \
    -d
metadata[track]="5001"
```

EXAMPLE RESPONSE

```
{
  "id":
"pl_y48R75HYhHMFKsBm8ltbNF
  "object":
"payment_link",
  "created_at": "2020-
12-31T23:59:59Z",
  "amount": 100.20,
  "amount_paid": 0,
  "currency": "gbp",
  "statement": "CL-INV-
065002-UPD",
  "description":
"Invoice paid by
David",
  "vendor": null,
  "customer":
"cus_qS5hHtII3XjLV5dB92EA"
  "expires_at": "2020-
12-31T23:59:59",

"target_bank_account":
null,
  "provider": null,
  "redirect": {
    "return_url":
"https://bigtechapp.com/pay
callback",
    "consent_url":
"https://finexer.com/p/eej
  },
  "sent_at": null,
```

description optional	An arbitrary string attached to a payment for your records.	
provider optional	The ID of a **provider** object that will lock the consent screen to only a single bank. Can be updated for non-used links.	
expires_at optional	Time at which an existing payment link will expire. Supply `now` if you want the payment link expire right away.	
target_bank_account optional	The unique identifier of an existing **bank account** object where money will be sent to for each payment made with the existing link.	
return_url optional	The URL you want the user to be redirected to when they authorize the	

```
    "paid_at": null,
    "type": "single",
    "status": "new",
    "active": true,
    "metadata": {
      "track": "5001"
    }
}
```

		payment.
type optional		The type indicates how payments are attributed to the link.
active optional		Indicates if the consent URL of the link is active. If `false` , no payment can be initiated.
metadata optional		Set of arbitrary key/value pairs you can attach to the payment link object.

RESPONSE

Returns the **payment_link** object that has been updated if the request succeeded. Returns an **error** otherwise.

List payment links

Retrieves a **paginated** list of payment links you have previously created. The returned payment links are sorted by creation date, with the most recently created links appearing first.

ENDPOINT

```
GET: /payment_links
```

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/pay
created_at.gte=2020-12-
01&created_at.lte=2021-
```

```
01-31 \
    -u "API_KEY"
```

QUERY PARAMETERS

vendor optional	Only return payment links that were issued by the vendor specified by this vendor ID.
customer optional	Only return payment links that were issued for the customer specified by this customer ID.
currency optional	Limit to records whose currency match the specified 3-letter **ISO** currency code e.g., gbp, eur, usd, etc.
Expand ▸ amount optional	A positive integer representing the payment amount, e.g., 820.16
Expand ▸ created_at optional	Represents date time universal **ISO 8601** format. (e.g. 2019-05-25 or 2019-05-25T14:00:00)

To retrieve a complete set of paginated results, you can use **pagination** arguments too.

EXAMPLE RESPONSE

```json
{
  "object": "list",
  "count": 10,
  "location":
"/payment_links",
  "paging": {
    "total": 108,
    "next":
"/payment_links?
offset=20",
    "prev": null
  },
  "data": [
    {
      "id":
"pl_y48R75HYhHMFKsBm8ltbNF
      "object":
"payment_link",
      "created_at":
"2020-12-31T23:59:59Z",
      "amount": 100.20,
      "amount_paid":
0.00,
      "currency":
"gbp",
      "statement": "CL-
INV-065002",
      "description":
"Invoice paid by
David",
      "vendor": null,
      "customer":
"cus_qS5hHtII3XjLV5dB92EA"
      "receipt_email":
null,
      "expires_at":
"2020-12-31T23:59:59",

"target_bank_account":
null,
      "provider": null,
      "redirect": {
        "return_url":
"https://bigtechapp.com/pa
callback",
        "consent_url":
"https://finexer.com/p/eej
      },
      "sent_at": null,
      "paid_at": null,
```

```
      "type": "single",
      "status": "new",
      "active": true,
      "metadata": {}
    },
    {...}
  ]
}
```

RESPONSE

Returns a list of existing **payment_link** objects that match filter criteria if the request succeeded. Returns an **error** otherwise.

Send a payment link

Sends a payment link to a customer to collect a payment. You need only supply the unique payment link `id` in the endpoint route associated with the payment link object you want to send.

ENDPOINT

```
POST:
/payment_links/{id}/send
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/pa
\
    -u "API_KEY"
```

ARGUMENTS

email optional	The email address to send a link to. If you leave this empty, we'll send a link to the email address saved on the **customer** object the payment link was issued for.
expires_at optional	Time at which a payment link will expire.

EXAMPLE RESPONSE

```
{
  "id":
"pl_y48R75HYhHMFKsBm8ltbNF
  "object":
"payment_link",
  "created_at": "2020-
10-15T14:06:24Z",
  "amount": 100.20,
  "amount_paid": 0.00,
  "currency": "gbp",
  "statement": "CL-INV-
065002-UPD",
  "description":
"Invoice paid by
David",
  "vendor": null,
  "customer":
"cus_qS5hHtII3XjLV5dB92EA"
  "receipt_email":
null,
  "expires_at": "2020-
12-31T23:59:59",
```

You can extend expiration date giving your customer more time to make a payment for example.

notes optional	Additional notes to be appended in the email sent to your customer.

```
  "target_bank_account":
null,
  "provider": null,
  "redirect": {
    "return_url":
"https://bigtechapp.com/pay
callback",
    "consent_url":
"https://finexer.com/p/eej
  },
  "sent_at": "2020-10-
15T14:10:16",
  "paid_at": null,
  "type": "single",
  "status": "sent",
  "active": true,
  "metadata": {
    "track": "5001"
  }
}
```

RESPONSE

Returns the **payment_link** object that has been sent if the request succeeded. Returns an **error** otherwise.

Get payment link QR code

```
GET:
/payment_links/{id}/qr_co
```

Retrieves QR code for a payment link. Note codes can be returned only for links that are active, i.e. have `redirect` attribute set on a payment link object. You need only supply the unique payment link `id` in the endpoint route, associated with the payment link object you want to get QR code for.

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/pa
\
    -u "API_KEY"
```

RESPONSE

Returns a simple object with relevant metadata if the request succeeded. Returns an **error** otherwise.

EXAMPLE RESPONSE

```
{
  "qr_code_url":
"https://blobs.finexer.com
}
```

Verification Checks

Verification is the process of verifying customer identities by checking information on supplied documents, and/or information provided to us by financial institutions (i.e. banks).

Depending on the type of the check, you can extract data from the ID document (e.g. passport) with the help of AI, or ask the customer to share their bank account information.

Once a verification is finished, the status of a processed check is switched to `completed`, and a new report is generated with a score indicating the data matching accuracy. To view the report, use **Get verification report** method supplying a unique identifier of the report found in the `last_report` attribute in **verification check** object.

Verification check object

ATTRIBUTES

id string	Unique identifier for the verification check object.
object string	The name of the object: `verification_check`

EXAMPLE OBJECT

```
{
  "id":
"vc_2dc67Kz2TBKQhH3uhYdkne(
  "object":
"verification_check",
  "created_at": "2021-
12-26T16:04:58Z",
  "name": "John
Williams",
  "provider": "rbs",
  "customer": null,
  "last_report":
"vr_h8b9w5Ra9Z1mR1xTEnxrgz]
  "mode": "auto",
  "type":
```

created_at timestamp	Time indicating when a verification check object is created.	

"name_account",
  "scopes": [
    "accounts"
  ],
  "redirect": null,
  "status":
"completed",
  "files": [],
  "metadata": {}
}

updated_at timestamp, nullable	Time indicating when the verification check object is last time updated.
name string	Contains the name of an individual to compare it to other sources, or simply the name of the check.
provider string, nullable	The unique identifier of supported **provider** objects the check is created or processed with.
customer string, nullable	The unique identifier of **customer** object the check is processed for.
last_report string, nullable	The unique identifier of last **verification report** object that is generated for the check.
mode	auto -

string		default mode, switches the check to the processing state automatically once data is collected. `manual` - once data is collected, the check remains in the pending state until **Process a check** method is called. It's useful when you need to review collected data before it's submitted for processing.
	type string	The type of the processing. `name_account` - individual's name is compared to the name in a financial institution. `name_id` - individual's name is compared to the name on the document i.e. passport. `name_id_account` - individual's name extracted from a document and

compared to the name in a financial institution. `face_id_selfi e` - individual's photo on a document is compared to the selife.

scopes *list*	The scope of data generated in the **verification report**. Available scopes: `accounts` - information about account name, number, currency; `balance` - includes available bank account balance; `transactions` - includes transactional account summary.

Expand ▸

redirect *object, nullable*	*See child attributes* This object is returned only when the check status is `requires_inp ut`, that means data is required for the check before the verification process gets started.

status *string*	The current state of the verification check. `requires_input` - requires customer data to be supplied (attach files, and/or authorize bank access depending on the check type). `pending` - data is collected, hit **Process a check** method to switch the check to the processing state. This state is skipped in auto mode. `canceled` - the verification check has been cancelled by the user or thru the API. `processing` - the verification process is started. `completed` - the verification is completed, report is generated.	
files *list*	The list of identifiers of **file** objects attached to the check	

metadata object, { }	Set of arbitrary key/value pairs you can attach to the verification check object.	

Create a verification check

Creates a new verification check object to verify a customer identity.

ARGUMENTS

name REQUIRED	The name of an individual to compare it to other sources, or simply the name of the check. Supply the full name of the individual for name_account or name_id check type.	
type REQUIRED	The type of the verification to perform.	
customer optional	The unique identifier of the **customer** object a new check will be processed for.	
provider	Supply the

ENDPOINT

```
POST:
/verification_checks
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/ve
\
    -u "API_KEY" \
    -d name="John
Williams" \
    -d
type="name_account" \
    -d
scopes[]="accounts" \
    -d
scopes[]="balance"
```

EXAMPLE RESPONSE

```
{
  "id":
"vc_2dc67Kz2TBKQhH3uhYdkne
  "object":
"verification_check",
  "created_at": "2021-
12-26T16:04:58Z",
  "name": "John
Williams",
  "provider": null,
  "customer": null,
  "last_report": null,
  "mode": "auto",
  "type":
"name_account",
```

provider optional	Supply the unique identifier of a supported **provider** if you already know your customer intends to share bank account data from a specific bank that will be used to verify identity.
mode optional	It can be either `auto` that will process the new check automatically once data is collected, or `manual` that will remain the check in the pending state until you call **Process a check** method.
files optional	The list of identifiers of **file** objects to attach to the check. For document-related verifications, the check will remain in status `requires_input` until files are attached to the check.
scopes optional	Use `scopes`

```json
  "scopes": [
    "accounts"
  ],
  "redirect": {
    "return_url":
"https://bigtechapp.com/data
callback",
    "consent_url":
"https://finexer.com/conne
verify=vc_2dc67Kz2TBKQhH3u
  },
  "status":
"requires_input",
  "files": [],
  "metadata": {}
}
```

		parameter to pass the data scopes you need to include in the **verification report**. By default, we apply `accounts`, `balance` and `transactions`.
return_url optional		The URL you want the user to be redirected to after they give access to their data. Leave this empty and we'll redirect the user to one of your redirect URLs saved in your app, or to our confirmation page.
metadata optional		Set of arbitrary key/value pairs you can attach to the verification check.

RESPONSE

Returns the newly created **verification check** object if the request succeeded. Returns an **error** otherwise.

Get a verification check

Retrieves the details of a single verification check object. You need only supply the unique verification check `id` in the endpoint route, associated with the check object you want to retrieve.

RESPONSE

Returns an existing **verification check** object that matches provided verification check ID if the request succeeded. Returns an **error** otherwise.

```
GET:
/verification_checks/{id}
```

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/ve
\
    -u "API_KEY"
```

EXAMPLE RESPONSE

```
{
  "id":
"vc_2dc67Kz2TBKQhH3uhYdkne
  "object":
"verification_check",
  "created_at": "2021-
12-26T16:04:58Z",
  "name": "John
Williams",
  "provider": null,
  "customer": null,
  "last_report":
"vr_h8b9w5Ra9Z1mR1xTEnxrgz
  "mode": "auto",
  "type":
"name_account",
  "scopes": [
    "accounts"
  ],
  "redirect": null,
  "status":
"completed",
  "files": [],
  "metadata": {}
}
```

Update a verification check

Updates the verification check object by its `id` found on the **verification check** object.

ENDPOINT

```
POST:
/verification_checks/{id}
```

> Note you can update checks that have `requires_input` status. Metadata attribute can be updated any time.

ARGUMENTS

name optional	The name of an individual to compare it to other sources, or simply the name of the check. Supply the full name of the individual for `name_account` or `name_id` check type.
type optional	The type of the verification to perform.
provider optional	Supply the unique identifier of a supported **provider** if you already know your customer intends to share bank account data from a specific bank that will be used to verify identity.
mode optional	It can be either `auto` that will process the new check

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/ver
\
    -u "API_KEY" \
    -d name="John
Williams Junior"
```

EXAMPLE RESPONSE

```
{
  "id":
"vc_2dc67Kz2TBKQhH3uhYdkne
  "object":
"verification_check",
  "created_at": "2021-
12-26T16:04:58Z",
  "name": "John
Williams Junior",
  "provider": null,
  "customer": null,
  "last_report": null,
  "mode": "auto",
  "type":
"name_account",
  "scopes": [
    "accounts"
  ],
  "redirect": {
    "return_url":
"https://bigtechapp.com/da
callback",
    "consent_url":
"https://finexer.com/conne
verify=vc_2dc67Kz2TBKQhH3u
  },
  "status":
"requires_input",
  "files": [],
  "metadata": {}
}
```

new check automatically once data is collected, or `manual` that will remain the check in the pending state until you call **Process a check** method.

files	optional	The list of identifiers of **file** objects to attach to the check. For document-related verifications, the check will remain in status `requires_input` until files are attached to the check.
metadata	optional	Set of arbitrary key/value pairs you can attach to the verification check.

Returns the **verification check** object that has been updated if the request succeeded. Returns an **error** otherwise.

Cancel a verification check

Cancels a previously created verification check. This action can only be performed while the verification process has not begun i.e., before the status transitions to processing .

RESPONSE

Returns an existing **verification check** object with status canceled if the request succeeded. Returns an **error** otherwise.

```
POST:
/verification_checks/{id}
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/ve:
\
    -u "API_KEY"
```

EXAMPLE RESPONSE

```json
{
  "id":
"bc_mtoK6Z9ZBtBLztQorkEm8c:
  "object":
"verification_check",
  "created_at": "2021-
12-26T16:04:58Z",
  "name": "John
Williams Junior",
  "provider": null,
  "customer": null,
  "last_report": null,
  "mode": "auto",
  "type":
"name_account",
  "scopes": [
    "accounts"
  ],
  "status": "canceled",
  "files": [],
  "metadata": {}
}
```

List verification checks

Retrieves a **paginated** list of verification checks. The returned checks are sorted by creation date, with the most recently created checks appearing first.

ENDPOINT

```
GET: /verification_checks
```

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/ve:
```

QUERY PARAMETERS

type optional	Limit to verification checks whose type contains the specified value.
status optional	Limit to verification checks whose status contains the specified value.
customer optional	Only return checks for the customer specified by this customer ID.
Expand ▸ created_at optional	Represents date time universal **ISO 8601** format. (e.g. 2019-05-25 or 2019-05-25T14:00:00)

To retrieve a complete set of paginated results, you can use **pagination** arguments too.

RESPONSE

Returns a list of existing **verification check** objects that match filter criteria if the request succeeded. Returns an **error** otherwise.

```
        \
        -u "API_KEY"
```

EXAMPLE RESPONSE

```
{
  "object": "list",
  "count": 2,
  "location":
"/verification_checks",
  "paging": {
    "total": 2,
    "next": null,
    "prev": null
  },
  "data": [
    {
      "id":
"vc_2dc67Kz2TBKQhH3uhYdkne
      "object":
"verification_check",
      "created_at":
"2021-12-26T16:04:58Z",
      "updated_at":
null,
      "name": "John
Williams",
      "provider": null,
      "customer": null,
      "verification":
null,
      "last_report":
"vr_h8b9w5Ra9Z1mR1xTEnxrgz
      "files": [

"fl_3kqeVuJDTmmog77ag4oIg"
      ],
      "mode": "auto",
      "type":
"name_id",
      "scopes": [
        "accounts"
      ],
      "redirect": null,
      "status":
"completed",
      "metadata": {}
    },
    {...}
  ]
}
```

Process a verification check

Checks with mode `manual` remain in pending state for you to review collected data before submitting them for the verification. Use this method to get a verification process started for a check, ID of which is supplied in the endpoint route.

ENDPOINT

```
POST:
/verification_checks/{id}
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/ver
\
     -u "API_KEY"
```

RESPONSE

Returns the **verification check** object with status `processing` if the request succeeded. Returns an **error** otherwise.

EXAMPLE RESPONSE

```
{
  "id":
"vc_2dc67Kz2TBKQhH3uhYdkne(
  "object":
"verification_check",
  "created_at": "2021-
12-26T16:04:58Z",
  "name": "John
Williams",
  "provider": null,
  "customer": null,
  "last_report": null,
  "mode": "manual",
  "type": "name_id",
  "scopes": [
    "accounts"
  ],
  "redirect": null,
  "status":
"processing",
  "files": [

"fl_3kqeVuJDTmmog77ag4oIg"
  ],
  "metadata": {}
}
```

Verification Reports

Verification reports are generated from collected data from customers and attached to **verification check** objects.

Each report contains the `score` attribute indicating the data matching accuracy when a report is verified. The score is ranged from `0` to `1` where 0.00 has no match and 1 is exact match.

You can view more examples of reports in the Appendix section **here**.

Verification report object

EXAMPLE OBJECT

```
{
  "id":
"vr_h8b9w5Ra9Z1mR1xTEnxrgzl
  "object":
"verification_report",
  "created_at": "2021-
12-26T16:05:03Z",
  "period_from": "2021-
12-26",
  "period_to": "2021-
12-26",
  "score": 0.92,
  "customer": null,
  "verification_check":
"vc_2dc67Kz2TBKQhH3uhYdkne(
  "type": "document",
  "document": {
    "type": "passport",
    "first_name":
"John",
    "last_name":
"Williams",
    "sex": "male",
    "address": null,
    "dob": {
      "month": 6,
      "day": 23,
      "year": 1970
    },
```

ATTRIBUTES

id string	Unique identifier for the verification report object.
object string	The name of the object: `verification_report`
created_at timestamp	Time indicating when the verification report object was created.
period_from timestamp	The period of time the report is generated for.
period_to report	The period of

period_to report timestamp	This period of time the report is generated for.
score integer	The data matching accuracy in the range from `0` to `1`.
customer string, nullable	The unique identifier of **customer** object this report is generated for.
verification_check string	The unique identifier of **verification check** object this report is linked to.
type string	The type of the generated report and the data it represents. The attribute name is determined by the object this report contains. Can be `document`, `bank_account`.
status string	The current state of the verification report. `verified` - a linked verification check is processed. `failed` - a linked verification check failed

```
    "expiry_date": {
      "month": 5,
      "day": 29,
      "year": 2023
    },
    "issued_date":
null,
    "issued_country":
"GB",
    "number":
"503240353",
    "files": [

"fl_3kqeVuJDTmmog77ag4oIg"
    ]
  },
  "status": "verified"
}
```

check failed
to extract and
analyze
collected
data.

Get a verification reports

Retrieves the details of a single verification report object. You need only supply the unique verification report `id` in the endpoint route, associated with the verification report object you want to retrieve.

RESPONSE

Returns an existing **verification report** object that matches provided verification report ID if the request succeeded. Returns an **error** otherwise.

ENDPOINT

```
GET:
/verification_reports/{id
```

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/ve
\
    -u "API_KEY"
```

EXAMPLE RESPONSE

```
{
  "id":
"vr_h8b9w5Ra9Z1mR1xTEnxrgz
  "object":
"verification_report",
  "created_at": "2021-
12-26T16:05:03Z",
  "period_from": "2021-
12-26",
  "period_to": "2021-
12-26",
  "score": 0.92,
  "customer": null,
  "verification_check":
"vc_2dc67Kz2TBKQhH3uhYdkne
  "type": "document",
  "document": {
    // report data
  },
  "status": "verified"
}
```

List verification reports

ENDPOINT

Retrieves a **paginated** list of verification reports. The returned reports are sorted by creation date, with the most recently created reports appearing first.

QUERY PARAMETERS

type optional	Limit to reports whose type contains the specified value.	
status optional	Limit to reports whose status contains the specified value.	
customer optional	Only return reports for the customer specified by this customer ID.	
verification_check optional	Only return reports for the verification check specified by this check ID.	
Expand ▸ created_at optional	Represents date time universal **ISO 8601** format. (e.g. 2019-05-25 or 2019-05-25T14:00:00)	

To retrieve a complete set of paginated results, you can use

```
GET:
/verification_reports
```

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/ve
\
    -u "API_KEY"
```

EXAMPLE RESPONSE

```
{
  "object": "list",
  "count": 3,
  "location":
"/verification_reports",
  "paging": {
    "total": 3,
    "next": null,
    "prev": null
  },
  "data": [
    {
      "id":
"vr_h8b9w5Ra9Z1mR1xTEnxrgzl
      "object":
"verification_report",
      "created_at":
"2021-12-26T16:05:03Z",
      "period_from":
"2021-12-26",
      "period_to":
"2021-12-26",
      "score": 0.92,
      "customer": null,

"verification_check":
"vc_2dc67Kz2TBKQhH3uhYdkne(
      "type":
"document",
      "document": {
        // report data
      },
      "status":
"verified"
    },
    {...}
  ]
}
```

pagination arguments too.

RESPONSE

Returns a list of existing **verification report** objects that match filter criteria if the request succeeded. Returns an **error** otherwise.

Providers

Returns providers that are used to indicate which bank to connect or use to initiate payments from. Provider object holds the bank information such as a unique identifier, a name, etc.

Currently we support banks only in the UK. We aim to support banks in Europe very soon to deliver a better value for you. Stay with us.

ENDPOINTS

GET: **/providers/{id}**

GET: **/providers**

Provider object

EXAMPLE OBJECT

```
{
  "id":
"lloyds_business",
  "object": "provider",
  "name": "Lloyds
Business",
  "roles": [
    "ais",
    "pis",
    "bpis"
  ],
  "ais": {
    "name": "Account
Data"
  },
  "pis": {
```

ATTRIBUTES

id string	Unique identifier for provider object.	
object string	The name of the object: `provider`	
name string	The name of the bank	

		institution.
roles list		Functionality supported by the provider. `ais` - bank account information; `pis` - payment initiation; `bpis` - batch payment initiation.
Expand ▸	ais object, nullable	See child attributes Account data config, returned only if appears in the `roles` list.
Expand ▸	pis object, nullable	See child attributes Single payment config, returned only if appears in the `roles` list.
Expand ▸	bpis object, nullable	See child attributes Batch payment config, returned only if appears in the `roles` list.
	logo_url string	Absolute URL path of a provider logo.

```
      "name": "Single
Payments",
      "min_amount": 0.10,
      "max_amount":
100000.00
    },
    "bpis": {
      "name": "Batch
Payments",
      "min_qty": 2,
      "max_qty": 50
    },
    "logo_url":
"https://finexer.blob.core
  }
```

Get a provider

Retrieves the details of a single provider. You need only supply the unique provider `id` in the endpoint route, associated with the provider object you want to retrieve.

RESPONSE

Returns an existing **provider** object that matches provided provider ID if the request succeeded. Returns an **error** otherwise.

```
GET: /providers/{id}
```

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/pro
\
    -u "API_KEY"
```

EXAMPLE RESPONSE

```json
{
  "id": "revolut",
  "object": "provider",
  "name": "Revolut",
  "roles": [
    "ais",
    "pis"
  ],
  "ais": {
    "name": "Account
Data"
  },
  "pis": {
    "name": "Single
Payments",
    "min_amount": 0.10,
    "max_amount":
250000.0
  },
  "logo_url":
"https://finexer.blob.core
}
```

List providers

Retrieves a **paginated** list of providers. The returned providers are sorted by name in alphabetical order.

ENDPOINT

```
GET: /providers
```

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/pro
\
    -d roles[]="ais" \
    -d roles[]="bpis"
\
    -u "API_KEY"
```

> Note the list of supported providers is returned for the environment that the API key belongs to. The returned list from one environment cannot be used in the other environment, i.e. pay with a sandbox bank supplying a production API key.

QUERY PARAMETERS

term optional	Limit to records whose name contains the specified value.
roles optional	Return the records which support the specified functionality.
limit optional, default is 20	A limit on the number of objects to be returned, between 1 and 100.
offset optional, default is 0	The number of objects to skip for the first returned result.

RESPONSE

Returns a list of existing **provider** objects that match filter criteria if the request succeeded. Returns an **error** otherwise.

EXAMPLE RESPONSE

```
{
  "object": "list",
  "count": 2,
  "location":
"/providers",
  "paging": {
    "total": 158,
    "next": "{to be
populated}",
    "prev": null
  },
  "data": [
    {
      "id": "aib",
      "object":
"provider",
      "name": "Allied
Irish Bank",
      "roles": [
        "ais",
        "pis"
      ],
      "ais": {
        "name":
"Account Data"
      },
      "pis": {
        "name": "Single
Payments",
        "min_amount":
1.00,
        "max_amount":
250000.00
      },
      "logo_url":
"https://finexer.blob.core
    },
    {
      "id": "boi_365",
      "object":
"provider",
      "name": "Bank of
Ireland 365 Online",
      "roles": [
        "ais",
```

```
        "pis",
        "bpis"
      ],
    "ais": {
      "name":
"Account Data"
    },
    "pis": {
      "name": "Single
Payments",
      "min_amount":
1.00,
      "max_amount":
250000.00
    },
    "bpis": {
      "name": "Batch
Payments",
      "min_qty": 2,
      "max_qty": 25
    },
    "logo_url":
"https://finexer.blob.core
    },
    {...}
  ]
}
```

Files

File objects contain information about files that have been uploaded to Finexer servers. Files can be shared with a permanent link or used within the Finexer API.

ENDPOINTS

POST: /files

GET: /files/{id}

POST: /files/{id}

DELETE: /files/{id}

GET: /files

File object

EXAMPLE OBJECT

ATTRIBUTES

id string	Unique identifier for the file object.
object	The name

```
{
  "id":
"fl_3kqeVuJDTmmog77ag4oIg"
  "object": "file",
  "created_at": "2021-
12-26T15:42:04Z",
  "updated_at": null,
  "title": "Homeland
passport",
  "name": "page-1",
```

```
    "size": 690270,
    "extension": "jpg",
    "type": "passport",
    "url":
"https://finexer.com/files
    "metadata": {}
}
```

string	of the object: file
created_at timestamp	Time indicating when the file object was created.
updated_at timestamp, nullable	Time indicating when the file object was last time updated.
title string, nullable	The title of the file.
name string	The original file name of the uploaded file without an extension (e.g. .jpg).
size integer	The absolute size of the uploaded file in bytes.
extension string	Extension of the uploaded file, for example png .
type string	The type of the document. Can be passport , selfie , driving_lic ense or logo

	url string	The permanent and unique URL to the uploaded file.
	metadata object, { }	Set of arbitrary key/value pairs you can attach to the file object.

Create a file

Creates a new file object with information supplied in the request including the file content that is sent via `multipart/form-data` .

> This method supports only `multipart/form-data` content type that follows the specifications of RFC 2388.

ARGUMENTS

type REQUIRED	The type of the document to upload.
file REQUIRED	The file itself.
title optional	The title of the file to upload.
metadata optional	Set of arbitrary key/value pairs you can attach to the file.

ENDPOINT

```
POST: /files
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/fil
\
    -u "API_KEY" \
    -f type="passport"
\
    -f
file="C://path/to/a/page-
1.jpg"
```

EXAMPLE RESPONSE

```json
{
  "id":
"fl_3kqeVuJDTmmog77ag4oIg"
  "object": "file",
  "created_at": "2021-
12-26T15:42:04Z",
  "updated_at": null,
  "title": null,
  "name": "page-1",
  "size": 690270,
  "extension": "jpg",
  "type": "passport",
  "url":
"https://finexer.com/files
  "metadata": {}
}
```

RESPONSE

Returns the newly created **file** object if the request succeeded. Returns an **error** otherwise.

Get a file

Retrieves the details of a single file object. You need only supply the unique file `id` in the endpoint route, associated with the file object you want to retrieve.

Returns an existing **file** object that matches provided file ID if the request succeeded. Returns an **error** otherwise.

ENDPOINT

```
GET: /files/{id}
```

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/fi
\
    -u "API_KEY"
```

EXAMPLE RESPONSE

```
{
  "id":
"fl_3kqeVuJDTmmog77ag4oIg"
  "object": "file",
  "created_at": "2021-
12-26T15:42:04Z",
  "updated_at": null,
  "title": null,
  "name": "page-1",
  "size": 690270,
  "extension": "jpg",
  "type": "passport",
  "url":
"https://finexer.com/files
  "metadata": {}
}
```

Update a file

Updates the file object by its `id` found on the **file** object. It can be used to modify attributes on the file, arguments of which were passed for this request.

ARGUMENTS

title optional	The title of the file to update.
metadata optional	Set of arbitrary key/value pairs you can attach to the file.

RESPONSE

Returns the **file** object that has been updated if the request succeeded. Returns an **error** otherwise.

```
POST: /files/{id}
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/fi
\
    -u "API_KEY" \
    -d title="My
Passport" \
    -d
metadata[authority]="80642
```

EXAMPLE RESPONSE

```json
{
  "id":
"fl_3kqeVuJDTmmog77ag4oIg"
  "object": "file",
  "created_at": "2021-
12-26T15:42:04Z",
  "updated_at": "2021-
12-26T16:10:28Z",
  "title": "My
Passport",
  "name": "page-1",
  "size": 690270,
  "extension": "jpg",
  "type": "passport",
  "url":
"https://finexer.com/files.
  "metadata": {
    "authority":
"806427"
  }
}
```

Delete a file

DELETE: /files/{id}

```
curl -X DELETE
https://api.finexer.com/fi
\
    -u "API_KEY"
```

Permanently deletes a file by its id found on the **file** object. You need to supply only the unique file id in the endpoint route.

RESPONSE

Returns an empty response with http status code of 204 if the request succeeded. Returns an **error** otherwise.

List files

Retrieves a **paginated** list of files. The returned files are sorted by creation date, with the most recently created files appearing first.

QUERY PARAMETERS

type optional		Limit to files whose type contains the specified value.
Expand ▸ created_at optional		Represents date time universal **ISO 8601** format. (e.g. 2019-05-25 or 2019-05-25T14:00:00)

To retrieve a complete set of paginated results, you can use **pagination** arguments too.

RESPONSE

Returns a list of existing **file** objects that match filter criteria if the request succeeded. Returns an **error** otherwise.

```
GET: /files
```

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/fi
\
    -u "API_KEY"
```

EXAMPLE RESPONSE

```json
{
  "object": "list",
  "count": 1,
  "location": "/files",
  "paging": {
    "total": 1,
    "next": null,
    "prev": null
  },
  "data": [
    {
      "id":
"fl_3kqeVuJDTmmog77ag4oIg",
      "object": "file",
      "created_at":
"2021-12-26T15:42:04Z",
      "updated_at":
null,
      "title": null,
      "name": "page-1",
      "size": 690270,
      "extension":
"jpg",
      "type":
"passport",
      "url":
"https://finexer.com/files
      "metadata": {}
    }
  ]
}
```

Events

Every time if something happens in your account or the state of an object changes, we record that event for security reasons and for you to track changes. For example, when a new consent gets created, we record `consent.created` event, when an existing consent gets revoked, we record `consent.canceled` event with a reference to the original object.

Event object

```json
{
  "id":
"evt_w2RJS3pWU2V3o7mzA6eP2V
  "object": "event",
  "created_at": "2020-
01-31T10:55:55Z",
  "type":
"customer.created",
  "data": {
    "id":
"cus_hjw5i1xO9zQK5kpS",
    "object":
"customer"
  }
}
```

ATTRIBUTES

id string	Unique identifier for event object.
object string	The name of the object: `event`
created_at timestamp	Time indicating when the event is created.
type string	Unique type of the event (e.g. `customer.created`). See a full list of available event types **here**.
Expand ▸ data object	Contains data associated with the event.

Get an event

Retrieves the details of a single existing event that was previously recorded on the API. You need only supply the unique event `id` in the endpoint route, associated with the event object you want to retrieve.

```
GET: /events/{id}
```

```
curl -X GET
https://api.finexer.com/ev
\
    -u "API_KEY"
```

RESPONSE

Returns an existing **event** object that matches provided event ID if the request succeeded. Returns an **error** otherwise.

```
{
  "id":
"evt_w2RJS3pWU2V3o7mzA6eP2
  "object": "event",
  "created_at": "2020-
01-31T10:55:55Z",
  "type":
"customer.created",
  "data": {
    "id":
"cus_hjw5i1xO9zQK5kpS",
    "object":
"customer"
  }
}
```

List events

Retrieves a **paginated** list of events we have previously recorded. The returned events are sorted by creation date, with the most recently created events appearing first.

QUERY PARAMETERS

object
optional

Only return events for the object specified by

```
GET: /customers
```

```
curl -X GET
https://api.finexer.com/ev
\
    -u "API_KEY"
```

		specified by this object ID.
linked_objects boolean, optional		If `true`, related events that depend on records whose object identifiers match the specified value of `object` are returned. For example, if a customer connected a bank, events such as `consent.accepted`, `bank_account.created` are returned in the response too, including events about changed state to the customer object.
type optional		Limit to events whose type match the specified value. Pass here one of supported event types (e.g. `customer.created`, `consent.canceled`, etc.).

```json
{
  "count": 2,
  "location":
"/events",
  "paging": {
    "total": 2,
    "next": null,
    "prev": null
  },
  "data": [
    {
      "id":
"evt_s6X1j78Id59s4IWiPu8hV:
      "object":
"event",
      "created_at":
"2020-02-02T15:32:30Z",
      "type":
"customer.created",
      "data": {
        "id":
"cus_hjw5i1xO9zQK5kpS",
        "object":
"customer"
      }
    },
    {...}
  ]
}
```

Expand ▸

created_at optional	Represents date time universal **ISO 8601** format. (e.g. 2019-05-25 or 2019-05-25T14:00:00)

To retrieve a complete set of paginated results, you can use **pagination** arguments too.

Returns a list of existing **event** objects that match filter criteria if the request succeeded. Returns an **error** otherwise.

Types of events

The list of available event types we currently send. Note as we constantly develop new features we aim to provide you with best coverage of all events happen in your account, subsequently extending this list.

TYPES

BANK ACCOUNTS

bank_account.created **bank_account** object	Triggered when a new bank account object is created via dashboard, API or gets imported from a bank.
bank_account.updated **bank_account** object	Triggered when an existing bank account object is

account object is updated.

bank_account.deleted **bank_account** object	Triggered when an existing bank account object is deleted.

BANK CONSENTS

consent.created **consent** object	Triggered when a new consent object is created.
consent.updated **consent** object	Triggered when an existing consent object is updated.
consent.opened **consent** object	Triggered when a user accesses the consent screen.
consent.initiated **consent** object	Triggered right before a user is directed to a bank side or banking app to approve access to their data.
consent.authorized **consent** object	Triggered when a user authenticates with their bank and authorizes access request to their bank account data.
consent.canceled **consent** object	Triggered when a user fails to authenticate at a bank, or denies access request to their bank account data at the consent flow.
consent.failed **consent** object	Triggered during the consent flow at any stage involving bank communication. This may occur if a bank encounters technical issues, or other disruptions

other disruptions.

consent.expired **consent** object	Triggered when a consent object gets expired.

CONSENT LINKS

consent_link.created **consent_link** object	Triggered when a new consent link object is created.
consent_link.updated **consent_link** object	Triggered when an existing consent link object is updated.
consent_link.sent **consent_link** object	Triggered when a consent link is sent to a customer, requesting them to connect their bank account(s).
consent_link.authorized **consent_link** object	Triggered when a customer authenticated at their bank and connected their bank account(s).
consent_link.canceled **consent_link** object	Triggered when an existing consent link is cancelled.

CUSTOMERS

customer.created **customer** object	Triggered when a new customer object is created.
customer.updated **customer** object	Triggered when an existing customer object is updated.
customer.deleted **customer** object	Triggered when an existing customer object is deleted.

FILES

file.created **file** object	Triggered when a new file object is created.

file.updated **file** object	Triggered when an existing file object is updated.	
file.deleted **file** object	Triggered when an existing file object is deleted.	

OAUTH

oauth.authorization.revoked **oauth** object	Triggered when access to managed account is revoked. Revocation can be initiated in the app account, managed account and through the API.

PAYMENT LINKS

payment_link.created **payment_link** object	Triggered when a new payment link object is created.
payment_link.updated **payment_link** object	Triggered when an existing payment link object is updated.
payment_link.sent **payment_link** object	Triggered when a payment link is sent to a customer, requesting payment.
payment_link.paid **payment_link** object	Triggered when a customer made a payment and amount was allocated to an existing payment link.
payment_link.canceled **payment_link** object	Triggered when an existing payment link is cancelled.

PAYMENTS

payment.created **payment** object	Triggered when a new payment

payment object		new payment object is created.
payment.updated **payment** object		Triggered when an existing payment object is updated.
payment.consent_opened **payment** object		Triggered when a user accesses the consent screen.
payment.consent_initiated **payment** object		Triggered right before a user is directed to a bank side or banking app to authorize the payment.
payment.authorized **payment** object		Triggered when a user authenticates with their bank and authorizes the initiation of a queued payment.
payment.completed **payment** object		Triggered when the bank processes the payment and the funds are successfully cleared.
payment.canceled **payment** object		Triggered when a user fails to authenticate at a bank, or denies a payment at the consent flow.
payment.failed **payment** object		Triggered during the consent flow at any stage involving bank communication. This may occur if a bank encounters a payment failure, technical issues, or other disruptions.

REMINDERS

reminder.consent_expiry.sent **reminder** object		Triggered when a reminder gets

| | **reminder** object | reminder gets triggered to remind about an expiring consent based on pre-set Consent Control settings. |

TEMPLATES

	template.created **template** object	Triggered when a new app template object is created.
	template.updated **template** object	Triggered when an existing app template object is updated.
	template.deleted **template** object	Triggered when an existing app template object is deleted.

VENDORS

	vendor.created **vendor** object	Triggered when a new vendor object is created.
	vendor.updated **vendor** object	Triggered when an existing vendor object is updated.
	vendor.deleted **vendor** object	Triggered when an existing vendor object is deleted.

VERIFICATION CHECKS

	verification_check.created **verification_check** object	Triggered when a new verification check is created.
	verification_check.updated **verification_check** object	Triggered when an existing verification check is updated.
	verification_check.pending **verification_check** object	Triggered for verification checks with flow set to "manual" whenever data was submitted/imported

	for review.
verification_check.canceled **verification_check** object	Triggered when an existing verification check is cancelled by the user or thru API.
verification_check.consent_opened **verification_check** object	Triggered when a user accesses the consent screen.
verification_check.consent_initiated **verification_check** object	Triggered right before a user is directed to a bank side or banking app to approve access to their data for identity verification.
verification_check.processing **verification_check** object	Triggered when an existing verification check is started for analyzing supplied data.
verification_check.completed **verification_check** object	Triggered when supplied data was analyzed for an existing verification check and it ran to the completion state, i.e. generated a report.

WEBHOOKS

webhook.created **webhook** object	Triggered when a new webhook object is created.
webhook.updated **webhook** object	Triggered when an existing webhook object is updated.
webhook.deleted **webhook** object	Triggered when an existing webhook object is deleted.

Templates

App templates represent trading styles of a business with a particular business name and branding theme that should be applied to the consent pages. For more information, please go to our **documentation** page.

> Note most of the attributes on a template object are optional. Leaving an attribute empty will bind value from your app.

Template object

ATTRIBUTES

id string	Unique identifier for template object.	
object string	The name of the object: app_template	
created_at timestamp	Time indicating when a template object was created.	
updated_at timestamp	Time indicating when a template object was last time updated.	

EXAMPLE RESPONSE

```
{
  "id": "yTRo67w1h1co",
  "object":
"app_template",
  "created_at": "2022-
08-08T11:51:32Z",
  "updated_at": "2022-
08-08T11:52:31Z",
  "name": "Template for
Acme Limited",
  "app_name": "Acme
Stars Limited",
  "file":
"fl_3qchjQoGMTrbZTDsB3BTF"
  "css": ".consent-
block { font-size:
16px; height: 24px; }",
  "header_html": null,
  "footer_html": "
<a>Terms &
Conditions</a>",
  "default": false,
  "metadata": {}
}
```

name string	The name of a template for your to recognize it.	
app_name string, nullable	The name of a business that appears on consent page.	
file string, nullable `expandable`	The unique identifier of the **file** object with the type `logo` that appears on consent pages as a business logo.	
css string, nullable	Custom CSS content that is embedded into the consent pages. This can be used to override existing styles of the page, or applied to your own HTML in the header and footer.	

header_html	Custom
string, nullable	HTML content that is embedded in the header of the consent pages.
footer_html	Custom HTML content that is embedded in the footer of the consent pages.
string, nullable	
default	When `true`, the template is applied by default to consent pages without a need to pass the template `id` in the URL.
boolean	
metadata	Set of arbitrary key/value pairs attached to the template.
object, { }	

Create a template

Creates a new template object. You need to supply the unique app `id` in the endpoint route found in your app settings.

ENDPOINT

```
POST:
/apps{app_id}/templates
```

EXAMPLE REQUEST

app settings.

ARGUMENTS

name REQUIRED	The name of the new template.	
app_name optional	The name of a business that will appear on consent page.	
file optional	The unique identifier of the **file** object with the type logo .	
css optional	Custom CSS content that is embedded into the consent pages.	
header_html optional	Custom HTML content that is embedded in the header of the consent pages.	
footer_html optional	Custom HTML content that is embedded in the footer of the consent pages.	
default optional	Provide true to	

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/app
\
    -u "API_KEY" \
    -d name="Staging
template for Acme
Limited" \
    -d app_name="Acme
Stars Limited" \
    -d
file="fl_3qchjQoGMTrbZTDsB
```

EXAMPLE RESPONSE

```json
{
  "id": "yTRo67w1h1co",
  "object":
"app_template",
  "created_at": "2022-
08-08T11:51:32Z",
  "updated_at": null,
  "name": "Staging
template for Acme
Limited",
  "app_name": "Acme
Stars Limited",
  "file":
"fl_3qchjQoGMTrbZTDsB3BTF"
  "css": null,
  "header_html": null,
  "footer_html": null,
  "default": false,
  "metadata": {}
}
```

	indicate the new template as a default template of the app, otherwise `false` .
metadata optional	Set of arbitrary key/value pairs you can attach to the template.

RESPONSE

Returns the newly created **template** object if the request succeeded. Returns an **error** otherwise.

Get a template

Retrieves the details of a single existing template that was previously created by **Create a template** method or in your Dashboard. You need to supply the unique app `id`, and the unique template `id` in the endpoint route, associated with the template object you want to retrieve.

RESPONSE

Returns an existing **template** object that matches provided template ID if the request succeeded. Returns an **error** otherwise.

ENDPOINT

```
GET:
/apps/{app_id}/templates/
```

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/app
\
     -u "API_KEY"
```

EXAMPLE RESPONSE

```
{
  "id": "yTRo67w1h1co",
  "object":
"app_template",
  "created_at": "2022-
08-08T11:51:32Z",
  "updated_at": null,
  "name": "Staging
template for Acme
Limited",
  "app_name": "Acme
Stars Limited",
  "file":
"fl_3qchjQoGMTrbZTDsB3BTF"
  "css": null,
  "header_html": null,
  "footer_html": null,
  "default": false,
  "metadata": {}
}
```

Update a template

Updates the template object by its `id` found on the **template** object. You need to supply the unique app `id`, and the unique template `id` in the endpoint route.

Note if you need to clear an attribute's value, you can pass an

ENDPOINT

```
POST:
/apps/{app_id}/templates/
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/app
\
```

empty value in an argument.

```
        -u "API_KEY" \
        -d name="Fixed
layout for Acme
Limited"
```

ARGUMENTS

name optional	The name of the existing template.
app_name optional	The business name on the existing template.
file optional	The unique identifier of the **file** object with the type `logo`.
css optional	Custom CSS content that is embedded into the consent pages.
header_html optional	Custom HTML content that is embedded in the header of the consent pages.
footer_html optional	Custom HTML content that is embedded in the footer of the consent pages.
default optional	Provide `true` to

EXAMPLE RESPONSE

```
{
  "id": "yTRo67w1h1co",
  "object":
"app_template",
  "created_at": "2022-
08-08T11:51:32Z",
  "updated_at": "2022-
08-08T11:52:31Z",
  "name": "Fixed layout
for Acme Limited",
  "app_name": "Acme
Stars Limited",
  "file":
"fl_3qchjQoGMTrbZTDsB3BTF"
  "css": null,
  "header_html": null,
  "footer_html": null,
  "default": false,
  "metadata": {}
}
```

true to
indicate the
existing
template as
a default
template of
the app,
otherwise
false .

metadata optional	Set of arbitrary key/value pairs you can attach to the existing template.

RESPONSE

Returns the **template** object that has been updated if the request succeeded. Returns an **error** otherwise.

Delete a template

Permanently deletes a template by its `id` found on the **template** object. You need to supply the unique app `id` , and the unique template `id` in the endpoint route.

RESPONSE

Returns an empty response with http status code of `204` if the request succeeded. Returns an **error** otherwise.

ENDPOINT

```
DELETE:
/apps/{app_id}/templates/
```

EXAMPLE REQUEST

```
curl -X DELETE
https://api.finexer.com/ap
\
    -u "API_KEY"
```

List templates

Retrieves a **paginated** list of templates you have previously created. The returned templates are sorted by creation date, with the most recently created templates appearing first. You need only supply the unique app `id` in the endpoint route.

> Note attributes `css`, `header_html` and `footer_html` are not returned in this method.

QUERY PARAMETERS

term optional	Only return templates whose name contain this value.

To retrieve a complete set of paginated results, you can use **pagination** arguments too.

RESPONSE

Returns a list of existing **template** objects that match filter criteria if the request succeeded. Returns an **error** otherwise.

```
GET:
/apps/{app_id}/templates
```

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/ap
\
    -u "API_KEY"
```

EXAMPLE RESPONSE

```
{
  "object": "list",
  "count": 7,
  "location":
"/templates",
  "paging": {
    "total": 7,
    "next": null,
    "prev": null
  },
  "data": [
    {
      "id":
"yTRo67w1h1co",
      "object":
"app_template",
      "created_at":
"2022-08-08T11:51:32Z",
      "updated_at":
null,
      "name": "Staging
template for Acme
Limited",
      "app_name": "Acme
Stars Limited",
      "file":
"fl_3qchjQoGMTrbZTDsB3BTF"
      "default": false,
      "metadata": {}
    },
    {...}
  ]
}
```

Webhooks

Webhooks allow you to be notified about events that happen on your Finexer account. You can configure what type of webhooks you would like to receive specifying certain events e.g., `webhook.created` , `payment.completed` when creating or updating a webhook.

You can also manage webhooks directly in your **account settings**.

Webhook object

ATTRIBUTES

id string	Unique identifier for webhook object.	
object string	The name of the object: `webhook`	
created_at timestamp	Time indicating when the webhook object was created.	
updated_at timestamp, nullable	Time indicating when the webhook object was last time updated.	
name	The name	

ENDPOINTS

POST: /webhooks

GET: /webhooks/{id}

POST: /webhooks/{id}

DELETE: /webhooks/{id}

GET: /webhooks

EXAMPLE RESPONSE

```
{
  "id":
"wh_4ZFeSoGHycUB78Z5OEdf1A
  "object": "webhook",
  "created_at": "2020-
02-10T10:42:22Z",
  "updated_at": null,
  "name": "Main App
Server",
  "url":
"https://bigtechapp.com/ho
  "status": "enabled",
  "events": [

"bank_account.created",

"bank_account.import",
    "consent.accepted",
    "consent.canceled",
    "consent.failed",
    "customer.created",
    "customer.deleted",
    "payment.accepted",

"payment.completed",
    "webhook.created",
    "..."
  ],
  "secret":
"r05L79dX5SDmFUOwDHrvzkX6"
  "metadata": {
    "server": "Apache
Tomcat"
```

```
    }
  }
```

string, nullable		of the webhook.
url	string	The URL of the webhook.
events	list	The list of events to trigger the webhook url for. Note this attribute is not declared on the object when calling **List webhook** method. See **Types of events** we support.
secret	string	The signature key of the webhook that is used to sign webhook payloads.
status	string	The status of the webhook. It can be enabled or disabled .
metadata	object, { }	Set of arbitrary key/value pairs you can attach to the webhook. It can be useful for storing

...ring
additional
information
about the
webhook.

Create a webhook

Creates a new webhook object.

ARGUMENTS

url **REQUIRED**	The URL of the webhook to be created.
name optional	The name of the webhook to be created.
events optional	The list of events to trigger the webhook url for to be created. To indicate all events, you can specify `events[]="*"` in the parameter. See **Types of events** we support.
metadata optional	Set of arbitrary key/value pairs you can attach to the

ENDPOINT

```
POST: /webhooks
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/web
\
    -u "API_KEY" \
    -d name="Main App
Server" \
    -d
url="https://bigtechapp.co
\
    -d
events[]="consent.accepted
\
    -d
events[]="payment.accepted
\
    -d
metadata[server]="Apache
Tomcat"
```

EXAMPLE RESPONSE

```
{
  "id":
"wh_4ZFeSoGHycUB78Z5OEdf1A
  "object": "webhook",
  "created_at": "2020-
02-10T10:42:22Z",
  "updated_at": null,
  "name": "Main App
Server",
  "url":
"https://bigtechapp.com/ho
  "status": "enabled",
  "events": [
    "consent.accepted",
    "payment.accepted"
```

webhook.

```
  ],
  "secret":
"r05L79dX5SDmFUOwDHrvzkX6"
  "metadata": {
    "server": "Apache
Tomcat"
  }
}
```

RESPONSE

Returns the newly created
webhook object if the request
succeeded. Returns an **error**
otherwise.

Get a webhook

```
GET: /webhooks/{id}
```

Retrieves the details of a single existing webhook that was previously created by **Create a webhook** method or in your **account settings**. You need only supply the unique webhook `id` in the endpoint route, associated with the webhook object you want to retrieve.

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/wel
\
    -u "API_KEY"
```

RESPONSE

Returns an existing **webhook** object that matches provided webhook ID if the request succeeded. Returns an **error** otherwise.

EXAMPLE RESPONSE

```
{
  "id":
"wh_4ZFeSoGHycUB78Z5OEdf1A
  "object": "webhook",
  "created_at": "2020-
02-10T10:42:22Z",
  "updated_at": null,
  "name": "Main App
Server",
  "url":
"https://bigtechapp.com/hoc
  "status": "enabled",
  "events": [
    "consent.accepted",
    "payment.accepted"
  ],
  "secret":
"r05L79dX5SDmFUOwDHrvzkX6"
  "metadata": {
    "server": "Apache
Tomcat"
  }
}
```

Update a webhook

Updates the webhook object by its `id` found on the **webhook** object. It can be used to modify attributes on the webhook, arguments of which were passed for this request.

ENDPOINT

```
POST: /webhooks/{id}
```

EXAMPLE REQUEST

```
curl -X POST
https://api.finexer.com/wel
```

Supports almost the same
arguments supported when
creating a webhook.

You can use this method to disable
or enable webhooks. Note if a
webhook is disabled, it will not
receive notifications until it is
enabled again, and Finexer will not
retry any notifications that are
generated while the webhook is
disabled.

ARGUMENTS

url optional	The URL of the webhook to be updated.
name optional	The name of the webhook to be updated.
events optional	The list of events to trigger the webhook url for to be updated. To indicate all events, you can specify events[]="*" in the parameter. See **Types of events** we support.
status optional	Set a status to the webhook.

```
      \
      -u "API_KEY" \
      -d name="nginx
server" \
      -d
url="https://bigtechapp.co
hooks" \
      -d events[]="*" \
      -d
metadata[server]="nginx"
```

EXAMPLE RESPONSE

```
{
  "id":
"wh_4ZFeSoGHycUB78Z5OEdf1A
  "object": "webhook",
  "created_at": "2020-
02-10T10:42:22Z",
  "updated_at": "2020-
02-12T15:20:11Z",
  "name": "nginx
server",
  "url":
"https://bigtechapp.com/la
hooks",
  "status": "enabled",
  "events": [
    "*"
  ],
  "secret":
"r05L79dX5SDmFUOwDHrvzkX6"
  "metadata": {
    "server": "nginx"
  }
}
```

	Possible values: enabled, disabled.
metadata *optional*	Set of arbitrary key/value pairs you can attach to the webhook.

RESPONSE

Returns the **webhook** object that has been updated if the request succeeded. Returns an **error** otherwise.

Delete a webhook

Permanently deletes a webhook by its `id` found on the **webhook** object.

RESPONSE

Returns an empty response with http status code of `204` if the request succeeded. Returns an **error** otherwise.

ENDPOINT

```
DELETE: /webhooks/{id}
```

EXAMPLE REQUEST

```
curl -X DELETE
https://api.finexer.com/wel
\
    -u "API_KEY"
```

List webhooks

Retrieves a list of webhooks you have previously created. The returned webhooks are sorted by name, then url in alphabetical order.

ENDPOINT

```
GET: /webhooks
```

EXAMPLE REQUEST

```
curl -X GET
https://api.finexer.com/wel
\
    -u "API_KEY"
```

RESPONSE

Returns a list of existing **webhook** objects if the request succeeded. Returns an **error** otherwise.

EXAMPLE RESPONSE

```
{
  "location":
"/webhooks",
  "count": 2,
  "data": [
    {
      "id":
"wh_rpC4XFqlZd3JqG5tCampbL!
      "object":
"webhook",
      "created_at":
"2020-08-15T10:56:12Z",
      "updated_at":
"2020-08-15T12:29:24Z",
      "url":
"https://bigtechapp.com?
auth=ABC90509030",
      "name": null,
      "status":
"enabled",
      "secret":
"Ufg3BOE2Dm0wrqbIPvujL5ic"
      "metadata": {}
    },
    {
      "id":
"wh_4ZFeSoGHycUB78Z5OEdf1A
      "object":
"webhook",
      "created_at":
"2020-02-10T10:42:22Z",
      "updated_at":
"2020-02-12T15:20:11Z",
      "name": "nginx
server",
      "url":
"https://bigtechapp.com/la
hooks",
      "status":
"enabled",
      "secret":
"r05L79dX5SDmFUOwDHrvzkX6"
      "metadata": {
        "server":
"nginx"
      }
    },
    {...}
```

```
    ]
}
```

Additional object definitions

There are objects that are not declared nor have corresponding API methods but are utilized to send you webhook payloads about certain events occur in your account. We listed such objects in this section with descriptions and JSON examples.

Account verification report

An example on the right illustrates a generated verification report object based on a verification check with the type `name_account`.

The verification report object may contain different types of objects and the names of which are always specified in the attribute `type`.

EXAMPLE RESPONSE

```
{
  "id":
"vr_ESIBf4F4yryiAPG3DEUrZk
  "object":
"verification_report",
  "created_at": "2021-
12-26T15:06:23Z",
  "period_from": "2020-
12-01",
  "period_to": "2021-
12-27",
  "score": 0.38,
  "customer": null,
  "verification_check":
"vc_EdTKgdaDWITsBooALzw6fX
  "type":
"bank_account",
  "bank_account": {
    "created_at":
"2021-12-26T15:06:22Z",
    "holder_name":
"Champagne Business
Account",
    "currency": "gbp",
    "identification": {
      "sort_code":
"102030",
      "account_number":
"20402040"
    },
    "nickname": null,
```

      "type": "business",
      "class": "savings",
      "balance": 99999,
      "provider": {
        "id": "orion",
        "object":
"provider",
        "name": "Orion",
        "logo_url":
"https://finexer.blob.core
      },
      "consent":
"bc_ByceAGOAx7Z7nV8eG3BDoC
      "customer": null,
      "fingerprint":
"HpPpBv8JlApQYuzq16O3",
      "metadata": {},
      "account_summary":
{
        "daily_balance":
null,

"transaction_count":
51,

"transaction_volume_in":
20293.8,

"transaction_volume_out":
-5257.2,

"transaction_count_per_mth
6.38,

"transaction_volume_in_per
5073.45,

"transaction_volume_out_pe
-1828.59
      }
    },
    "status": "verified"
}

Reminder event

Reminders are automatically sent at configured intervals to remind you about an event that is about to happen. They get sent via an **event** object that contains the reminder itself in the `data.object` attribute.

{
  "id":
"evt_neDTrEMfxUufGDseukKny
  "object": "event",
  "created_at": "2020-
09-04T17:48:31Z",
  "type":
"reminder.consent.expiry.s

For example, the Consent Control feature utilizes the reminders. An example on the right illustrates a payload with the event object containing a reminder object. The reminder object itself contains a consent object since it reminds about the expiration date of the given consent. For more information, please go to our **documentation** page.

ATTRIBUTES

id string	Unique identifier for reminder object.	
object string	The name of the object: `reminder`	
created_at timestamp	Time indicating when the reminder object was created.	
type string	The type of the reminder. It can be `consent_expiry`.	
target string	The recipient of the reminder. `account` – reminds about something in relation to your	

```
  "data": {
    "object": {
      "id":
"rmd_EIZnm5Idjq8QI5VWhKBA"
      "object":
"reminder",
      "created_at":
"2020-09-04T17:47:22Z",
      "type":
"consent_expiry",
      "consent": {
        "id":
"bc_KzSacn3DlAe69xt7tYHJFR'
        "object":
"consent",
        "created_at":
"2020-07-25T20:03:45Z",
        "updated_at":
"2020-07-25T20:04:43Z",
        "provider":
"barclays_personal",
        "customer":
"cus_YFilrr5jX378FCgpDFC0"
        "failure_code":
null,

"failure_message":
null,
        "expiry_date":
"2020-09-05",
        "retro_date":
"2020-07-25",
        "status":
"authorized",
        "metadata": {}
      },
      "target":
"customer",
      "method":
"webhook"
    }
  }
}
```

to your account. `customer` – reminds about something in relation to your customers.

| method | The |
| string | method how the reminders get sent. It can be `email` or `webhook`. For emails a short user-friendly message is sent. For webhooks full **reminder** object is sent, including an object that a reminder represents, e.g. **consent** object. |

OAuth event

EXAMPLE EVENT

```
{
  "id":
"evt_hiyAIbEzkCkJHa3QOr4Mv
  "object": "event",
  "created_at": "2021-
03-25T07:55:11Z",
  "type":
"oauth.authorization.revok
  "data": {
    "object": {
      "token_type":
"bearer",
      "account_id":
"acc_q3vEk3PuKrwrBMwETs86Q
      "account_name":
"The Great Company",
      "scope":
"customer_read,customer_wr
    }
  }
}
```

When an access token previously obtained via OAuth flow to manage your client's account is revoked by you, or your client, you receive a webhook with an event of the type `oauth.authorization.revoked`. This is a good way to be notified and take a corresponding action on your side, for example, remove a connectivity indicator. In the oauth object, `account_id` attribute contains a unique account identifier that you can correlate with your client.

ATTRIBUTES

token_type string	The type of access token. The value is always `bearer`.
account_id string	Unique identifier of an account for which the access token was issued for.
scope string	Permissions represented in CSV format indicating the scope of access the access token had.