

Spotlight API Documentation

A documentation for customer friendly API

Page 2/10

Introduction

This documentation describes two different ways to display data from Spotlight on companies' web pages. This document also shows the how to authenticate to have access to the Spotlight API to be able to fetch data.

Graph and IR Web URL or addresses have two different ways display data. The different addresses are ways to implement a customer friendly view generated by multiple API calls into a IFrame solution, it can be placed anywhere on the customer web site. The IFrame solution does not need authentication from the company.

Choose the Right Path

The API is divided into three webs that you can call with a customer API key. Depending on what the company is looking for, the company can use any of these 3 URLs:

- <u>https://api.spotlightstockmarket.com</u>
- <u>https://graph.spotlightstockmarket.com</u>
- <u>https://ir.spotlightstockmarket.com</u>

Get Started with api.spotlightstockmarket.com

The first step is to gain an "access token". This token is to gain permission to retrieve information from the endpoint <u>https://api.spotlightstockmarket.com</u>. The two other endpoints can be implemented immediately on to the customer solution or web page without a token. The API-token can be reused until it expires after **60 minutes**. The expiration time of the token will be displayed in the response of the API.

Authentication

Each client that is granted permission to use the Spotlight Web API will receive a personal "client id" and "client secret" to their email address. These are called keys and will be used to gain the access token. The first step is to make an http POST-request to Microsoft's authentication service. The keys that will be used to gain the access token are:

- ClientID
- Client Secret
- Resource
- Grant Type

A recommendation is to use the application "Postman" to call the API.

Download the app here: https://www.getpostman.com/apps.

Documentation of how to use postman in depth: https://docs.api.getpostman.com/

Page 3/10

Gain the Access Token

Build the request:

- Make sure you select POST
- Copy and paste the address below into the input field: https://login.microsoftonline.com/spotlightstockmarket.com/oauth2/token
- Set parameters
 - Select the upper tab Body (below the top address field)
 - o Select radio button "x-www-form-urlencoded"
 - Input the values into separate "Keys" by adding a new Key.
 - 1st Key. resource to the Key column. The URL you want to call in this case: <u>https://api.spotlightstockmarket.com</u> to the Value column.

2nd. Client id to the left. The Client Id sent to client's email to the right

- 3rd. Client secret to the left. The Client Secret is sent to client's email to the right
- 4th. Grant type to the left. Client credentials to the right.
- Hit the Send button.
- · The response will be shown in the bottom text area within the lower Body tab
- In the section of "access token" Copy the right-hand side long character combination, within the quotation marks and save it for later.

"access_token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsIng1dCI6I .eyJhdWQiOiJodHRwczovL2FwaS1ha3RpZXRvcmd1dC5henVyZXd1YnNpd(yOTUyNiwiYWLvIjoiNDJC211EaDdidDFDUjRFUTNISØNBNDNLUHRXSEFRP1 4NzQ4Zjd1MC8iLCJvaWQiOiIXNGJhNWFiMy1hMzFhLTQ5MTKtoWR1My02M: 1MCIsInV0aSI6IktpVnRtbjZXV1VpQm5hVEEyaE5vQUEiLCJ2ZXIiOiIXL: -s9CAKPKDba6KexYs69fAcL8HR8Iap1VxA8eSGpR8uNdXGNAGqK05rhU83 -ug6z3buBU_jkfMSRxudYPKF0s901jVXPVUDL1r621biLCLRg8Xh-s2KQM1 Page 4/10

This token should be in the **Authentication header** in the **Headers** section of all the future http **GET**-requests to the Spotlight Web API to be able to fetch the data. If the **Status Code** returns to **200** it is done correctly. The flow below demonstrates the program Postman used to retrieve the access token.

POST * https://login.microso	oftonline.com/spotlightstockmarket.com/oauth2/to	oken Params So	end •
uthorization Headers (1) Body	Pre-request Script Tests		
form-data	fed 🗣 raw 🗣 binary		
Key	Value	Description	
resource	https://api.spotlightstockmarket.co	om/	
client_secret			
client_id			
grant_type	client_credentials		
New key	Value	Description	
Ady Cookies (4) Headers (14) Pretty Raw Preview JSON	Test Results	Status: 200 OK T	ime: 281 ms
Cookies (4) Headers (14) Pretty Raw Preview JSON 1 * (*********************************	Test Results	Status: 200 OK T	ime: 281 ms
<pre>ody Cookies (4) Headers (14) Pretty Raw Preview JSON 1 * (2 "token_type": "Bearer", 3 "expires_in": "3599", 4 "ext expires_in": "0", </pre>	Test Results	Status: 200 OK T	ime: 281 ms
dy Cookies (4) Headers (14) Pretty Raw Preview JSON 1 • (2 "token_type": "Bearer", 3 "expires_in": "3599", 4 "ext_expires_in": "3591931; 5 "expires_in": "153191321;	Test Results	Status: 200 OK T	ime: 281 ms
dy Cookies (4) Headers (14) Pretty Raw Preview JSON 1 * (2 "token_type": "Bearer", 3 "expires_in": "3599", 4 "ext_expires_in": "0", 5 "expires_on": "1531915111 6 "not_before": "1531915111	Test Results	Status: 200 OK T	ime: 281 ms
dy Cooldes (4) Headers (14) Pretty Raw Preview JSON 1 * (2 "token_type": "Bearer", 3 "expires_in": "3599", 4 "ext_expires_in": "0", 5 "expires_on": "IS31913211 6 "not_before": "IS31913211 7 "resource": "IS3191311 8 "access token": "evg@exdit"	Test Results	Status: 200 OK T	ime: 281 ms
dy Cookies (4) Headers (14) Pretty Raw Preview JSON 1 • (2 "token_type": "Bearer", 3 "expires_in": "3599", 4 "ext_expires_in": "0", 5 "expires_in": "153191311 6 "not_before": "153191311 7 "resource": "https://api. 8 "zk	Test Results	Statum: 200 OK T	ime: 281 ms
dy Cookies (4) Headers (14) Pretty Raw Preview JSON 1 • (2 "token_type": "Bearer", 3 "extres_in": "SSON", 4 "ext_expires_in": "SSON 5 "ext_expires_in": "SSONJIJII 7 "resource": "https://api. 8 "acces_token: "eyJ0eXAi 2k .c	Test Results Test Results Solution Test Results Test R	Status: 200 OK T	sImtpZCI6
bdy Cooldes (4) Headers (14) Pretty Raw Preview JSON 1 * (2 "token_type": "Bearer", 3 "expires_in": "3599", 4 "ext_expires_in": "0", 5 "expires_on": "ISSIBIS211 6 "not_before": "ISSIBIS211 7 "resource": "https://api. 8 "ecces_token": "eyJ@eXAi 24 Ae 4 Ae	Test Results Test Results Solution Test Results Test R	Status: 200 OK T	ime: 281 ms
dy Cookies (4) Headers (14) Pretty Raw Preview JSON 1 • (2 "token_type": "Bearer", 3 "expires_in": "3599", 4 "ext_expires_in": "0", 5 "expires_in": "153191311 6 "not_before": "153191311 7 "resource": "https://api. 8 Ze 4	Test Results Test Results Solution Test Results Test R	Statum: 200 OK T	ime: 281 ms
bdy Cookies (4) Headers (14) Pretty Raw Preview JSON 1 * { 2 ************************************	Test Results	Status: 200 OK T	ime: 281 ms
dy Cookies (4) Headers (14) Pretty Raw Preview JSON 1 ~ (2 "token_type": "Bearer", 3 "expires_in": "3599", 4 "ext_expires_in": "3599", 5 "expires_on": "153191311 6 "not_before": "153191311 7 "resource": "https://daj. 8 Zk 4 Ap 3 Ck 4 Ckk 4 Ck 4 Ckk 4 Ckk 4 Ckk 4 Ck 4	Test Results Te	Status: 200 OK T	ime: 281 ms
bdy Cookies (4) Headers (14) Pretty Raw Preview JSON 1 ~ { "token_type": "Bearer", 3 "expires_in": "3599", 4 "ext_expires_in": "0", 5 "expires_in": "1531915211 6 "not_before": "1531915211 7 "resource": "https://doi. 8 "access_token": "eyJ0eX4j 2 .e Ae 3 .c Ae 3 .c 2 .e .e 3 .e .e 4 .e .e 5 .e .e 6 .e .e 7 .e .e 8 .e .e 9 .e .e 7 .e .e	Test Results Test Results Solution Test Results Test R	Statum: 200 OK T	ime: 281 ms
bdy Cookies (4) Headers (14) Pretty Raw Preview JSON 1 - { 2 "token_type": "Bearer", 3 "rextres_int: "SSON", 4 "rext_expires_int: "or,] 5 "rext_expires_int: "isSISISII 6 "not_before": "IsSISISII 7 "resource: "thttps://api 8 "access_token": "eyJBeXAI 24 26 36 70 70 70 70 70 70 70 70 70 70	Test Results Test Results Total State St	Status: 200 OK T	Ime: 281 ms
dy Cookies (4) Headers (14) Pretty Raw Preview JSON 1 - (2 "token_type": "Bearer", 3 "expires_in": "3599", 4 "ext_expires_in": "0", 5 "expires_in": "153191311 6 "not_before": "153191311 7 "resource": "https://aph. 8 ZA 4 AA 5 Carter State S	Test Results Te	Status: 200 OK T	Ime: 281 ms
ody Cookies (4) Headers (14) Pretty Raw Preview JSON 1 ~ { "token_type": "Bearer", "expires_in": "3599", "expires_in": "3599", "expires_in": "1531915211 6 "not_before": "1531915211 7 5 "expires_in": "1531915211 7 "not_before": "1531915211 7 7 "expires_int": "1531915211 7 "access_token": "eyJ@eXAi 2 8 22 4 7 "koncess_token": "eyJ@eXAi 5 7 "koncess_token": "eyJ@eXAi 5 7 7 7 8 9	Test Results Test Results Total State St	Status: 200 OK T d5d3dsaHZkRmJYWjgxM1dwUGF5OUFsV5I turusukasecukrzebwinningguuniewido	Ime: 281 ms

Warning

Make sure that the application Fiddler (<u>https://www.telerik.com/fiddler</u>) is not running when doing the steps below. That will interfere with Postman. If you don't know what Fiddler is, do not worry, you are probably not running it.

Second step - retrieve data using the authentication token

The general principle for an approved API call to the endpoint api.spotlightstockmarket.com is that you have to send the access token, the base URL <u>http://api.spotlightstockmarket.com/</u> with the current version v2/ along with the relative path of the URL including Quotes, Circuit Breaker, Trade, Summarized Trades. Example:

http://api.spotlightstockmarket.com/v2/quotes + Access token http://api.spotlightstockmarket.com/v2/trade + Access token

Using API with parameters

Page 5/10

Demonstration using Postman

- Change the **POST** to **GET**
- Change the address to <u>http://api.spotlightstockmarket.com/v2/summarizedtrades.</u>
 This will return information about the **all trades for an instrument.**
- Add a new Key by clicking the upper Param tab and add a new Key. Add a key called
 OrderBookId and put the order book id for the instrument in the upper Key section.
- Click Headers-tab (below the address field).
- Type Authorization in the Key section. Add "Bearer" blank space {your access token} in the same row.
- Hit Send

The result will now show all summarized trades for an instrument, from when the request was sent.

GET	https://api.spotlightstockmarket.com/v1.0/summarizedtrades?orderbookide	~20 ₉₅	Params	Send V Save V
Key		Value	Description	Bulk Edi
 orderbookid 		40ys		
Authorization	Headers (1) Body Pre-request Script Tests			Con
Key		Value	Description	••• Bulk Edit Presets *
 Authorization 	n	Bearer eyj0eXAiOijKV1QiLCJhbGciOijSUzl1NilsIng1dCl6Imk2bEdrM0ZaenhSY1ViMikMzbk		
Body Cookies	Headers (8) Test Results			Status: 200 OK Time: 1170 ms
Pretty Raw	Preview JSON V			
· 7 € 9 № 11 12 13 № 18 № 22 22 23 № 26 № 21 23 23 № 25 № 12 13 № 18 № 22 22 23 № 26 № 28 № 23 23 № 25 № 19 № 40-	-Closepartip": 19995, -Closepartip": 19995, -Closeftmoder: 199056, -Terference: null: -Terference: null: -Terference: 10, -Closeftmoder: 116, -Closeftmoder: 116, -Closeftmoder: 116, -Closeftmoder: 116, -Closeftmoder: 116, -Closeftmoder: 128-14 -Closeftmoder: 128-			

Page 6/10

The Full API Documentation

The full API documentation for api.spotlightstockmarket.com can be found on: <u>https://api.spotlightstockmarket.com/swagger</u> This is the part of swagger documentation that describes most traded

(<u>https://api.spotlightstockmarket.com/v2/summarizedtrades</u>). You can see that it is a Http GET-method and that has the relative path "/summarizedtrades" (case-insensitive). A Parameter called OrderBookId is where the instrument order book id is set. Note that you cannot use the "try it out"-button, because this is a protected api. (you can of course use Postman as in the authentication flow above)

SummarizedTrades		Show/Hide L	ist Operations Expand Operations
GET /SummarizedTrades			
Response Class (Status 200) OK Model Model Schema			
<pre>[{ "NumTrades": 0, "CloseParice": 0, "CloseDaylumPrice": 0, "CloseDaylumPrice": 0, "CloseQuantity": 0, "CloseTurnOver": 0, "TradeDate": "2018-09-14T11:12:00.177Z", "Reference": 0, "TradeDate": 0, "TradeDate:": 0, "TradeDate:":</pre>			•
Response Content Type application/json •			
Parameters Parameter Value	Description	Parameter Type	Data Type
OrderBookID (required)		query	string
Try it out!			

Another example below, a method that takes a parameter. You can see the name (isin), required datatype (string) and how to pass the parameter (query string).
Quotes
Show/Hide List Operations Expand

(Showinge	isc operations	expand operations
GET /Quotes	s/GetQuoteByIsin				
Response Class OK	(Status 200)				
Model Model So	thema				
<pre>{ "MarketMaker "TradeState" "Differential "BidPrice": "AskPrice": "DayHighPrice" "DayLowPrice "Quantity": "TradeState" </pre>	": "string", : 0, IPreviousDay": 0, 0, 0, e": 0, *: 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,				Ŷ
Parameters					
raiameters					
Parameter	Value	Description	Parameter Type	Data Type	
isin	(required)		query	string	
Try it out!					

A correct call to get stock quote for the instrument with isin "SE0006091476" would look like: <u>https://api.spotlightstockmarket.com/v2/quotes/getquotebyisin?isin=SE0006091476.</u> Please note that you cannot test this in the Swagger UI (by clicking "Try it out", nor directly in a web browser without an access token (see Authentication flow above). Page 7/10

Another example below, a method that takes a parameter (companyld) and returns a list of all the company news.

GET	/v1/nev	vs/getteedbycompanyld				
Respo OK	onse Clas	ss (Status 200)				
Model	Example	e Value				
[{	"NewsId": "Companylo "CompanyNa "CategoryN "PublishDa "Title": ' "Teaser": "NewsText"	0, "": 0, mme": "string", lame": "string", tte": "2019-01-24T12:28:10.303Z", string", "string", : "string", untHimp", "string"				•
Respo	nse Conte	ent Type application/json 🔻				
Paran	neters					
Param	neter	Value	Description	Parameter Type	Data Type	
Compa	nyID	(required)		query	integer	
Try it	out!					

The following list contains all the values you can get from this method:

```
"NewsId": Our unique Id of the news,
"CompanyId": The Id of the company,
"CompanyName": The name of the company
"CategoryName": The name of the category
"PublishDate": When the news text was published
"Title": The header for the news text
"Teaser": The news text teaser
"NewsText": The bodytext
"CompanyShortName": Shortname for the company
"CompanyOrgNr": Company registrationnumber
"LEI": LEI-code
"SVUrl": The url that takes you to the Swedish version of the news text on
Spotlightstockmarket.com
"ENUrl": The url that takes you to the English version of the news text on
Spotlightstockmarket.com
"DKUrl": The url that takes you to the Danish version of the news text on
Spotlightstockmarket.com
"CategoryId": The Id for that specific category
"ExternalDocument1": The url to the first document
"ExternalDocument2": The url to the second document
"ExternalDocument3": The url to the third document
"ExternalDocument4": The url to the forth document
"ExternalDocument5": The url to the fifth document
"ExternalDocument1Title": The title for the first document
"ExternalDocument2Title": The title for the second document
"ExternalDocument3Title": The title for the third document
```

Graph and IR API

The Graph and IR API is essentially a package of API combined into one web page view. This can be used in a section in a company solution, web page etc. it is called IFrame. The IFrame solution will run from Spotlight's server and the customer cannot manipulate the data inside the Iframe (unless cross-origin policy is added <u>https://en.wikipedia.org/wiki/Cross-origin resource sharing</u>).

Page 8/10

Access the endpoints

Graph API

For Graph API you only need the base URL <u>https://graph.spotlightstockmarket.com/</u>. Choose the node language English, Swedish or Danish (not yet finished). Swedish: "sv", English: "en", Danish: "dk".

Add the relative URL for that node Swedish: "bolag", English: "companies" Danish "" (in progress).

Add the rest of the relative URL "/irabout".

Lastly add the Parameter to the instrument. "?InstrumentId={your instrument id here}" Final URL for the graph API should look similar to: https://graph.spotlightstockmarket.com/sv/bolag/irabout?InstrumentId=XSAT01000315 or

https://graph.spotlightstockmarket.com/en/companies/irabout?InstrumentId=XSAT01000573

Page 9/10

IR API

•

For the IR API you need the base URL https://ir.spotlightstockmarket.com/.

- Choose the node language English, Swedish or Danish (not yet finished).
 Swedish: "sv", English: "en", Danish: "dk".
- Add the relative URL for that node
 - Swedish: "bolag", English: "companies" Danish "" (tbd).
 - Add the rest of the relative URL "/irabout", "/irtrade" or "/ircontact"

Lastly add the Parameter to the instrument. "?InstrumentId={your instrument id here}" Final URL should look similar to:

<u>https://ir.spotlightstockmarket.com/sv/bolag/irabout?InstrumentId=XSAT01000315</u> or:

https://ir.spotlightstockmarket.com/en/companies/irtrade?InstrumentId=XSAT01000573

For graphical illustration see pictures below.



Page 10/10

Libraries and sample projects

If you are a .NET Developer, please see the NuGet-package "Microsoft.IdentityModel.Clients.ActiveDirectory"

For .NET-developers, please see the code samples on: https://docs.microsoft.com/en-us/azure/active-directory/develop/active-directory-codesamples#web-applications-signing-in-users-calling-microsoft-graph-or-a-web-api-with-the-users-<u>identity</u>

You can find it under "Daemon applications", "ASP.NET Web App"