**Data exchange with AL language:**

For comfortable API tests, I advise you to install POSTMAN ([www.getpostman.com](http://www.getpostman.com))

**Task 1.**

Parsing simple json request. Importing Share rates to Customer’s card.

**Sources:**

1. Rates source: <https://www.alphavantage.co>
2. Getting API key: <https://www.alphavantage.co/support/#api-key>

For tests you can use MSFT symbols and apikey = demo

1. API URL: https://www.alphavantage.co/query
2. GitHub Repository with source code: github.com/abaludin/SharesRate
3. Request sample:

https://www.alphavantage.co/query?function=GLOBAL\_QUOTE&symbol=MSFT&apikey=demo

**Step 1**

Create Integration Setup table and Page (card type) with next fields:

* Entry no. (As you prefer) \*key
* API URL (Text) = https://www.alphavantage.co/query
* API Token (Text)
* Last Rate Function (Text) = GLOBAL\_QUOTE
* Daily Rate Function (Text) = TIME\_SERIES\_DAILY

**Step 2**

Create Customer Table extension with next fields:

* Ticker Symbols (code[10])
* Rate (Decimal)

Create Page Extension for Customer Card (or list) to add these fields

**Step 3**

* Create management codeunit
* Get response from Last Rate API:

URL := IntegrationSetup."API URL" +

'?function=' + IntegrationSetup."Last Rate function" +

'&symbol=' + Symbols +

'&apikey=' + IntegrationSetup."API Token";

* Parse response and get decimal value from “Global Quote”.”05. Price” token:

QuoteToken.AsObject().Get('05. price', PriceToken);

**Step 4**

Create Get Last Rate action on Customer Card Page extension. Pass Symbols from customer and update Rate field with response

**Step 5**

Create Daily Rates Table and list page with next fields:

* Symbols (code[10]) \*key
* Rate Date (Date) \*key
* Open (Decimal)
* High (Decimal)
* Low (Decimal)
* Close (Decimal)

**Step 6**

* Create Get Daily Rate function with Symbols parameter in management codeunit
* Get response from Daily Rate API:

URL := IntegrationSetup."API URL" +

'?function=' + IntegrationSetup."Daily Rate function" +

'&symbol=' + Symbols +

'&apikey=' + IntegrationSetup."API Token";

* Parse response. A little bit tricky – because token name equals rate date. By default response contains last 100 dates. Remember that weekend and holiday dates does not exists. You could create date loop and select any quantity of days.
* Create a record for each rate of each Symbols

**Step 7**

Create action for Customer card calling Daily rates page filtered by customer’s symbols

**Final Result:**

