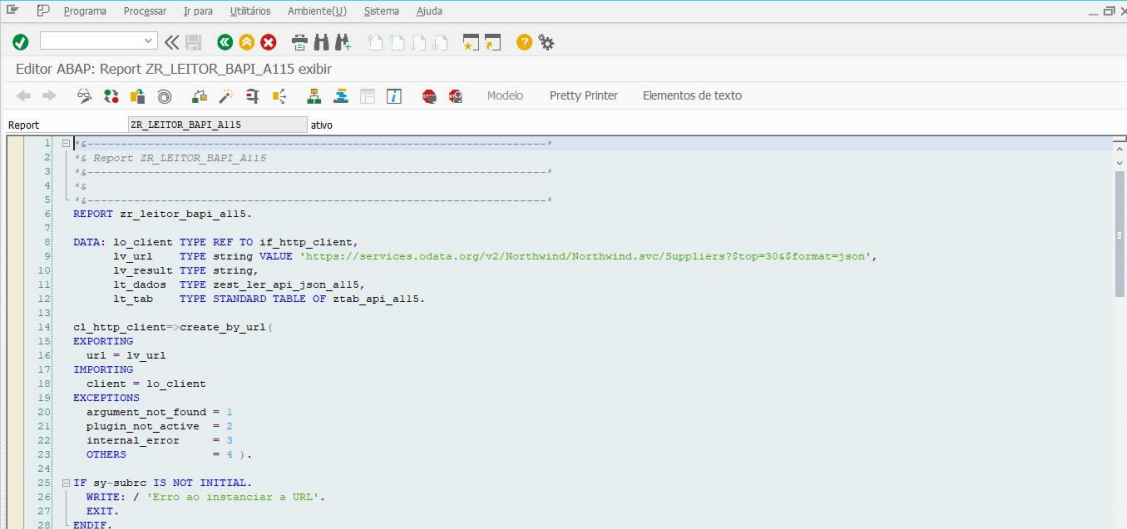


## Programa que consome API

1. Crie dois programas, sendo um que lê os dados do fornecedor disponíveis na API Northwind e salve em uma tabela Z nova.



```
1  *-----*
2  *% Report ZR_LEITOR_BAPI_A115
3  *%
4  *%
5  *%-----*
6  REPORT zr_leitor_bapi_a115.
7
8  DATA: lo_client TYPE REF TO if_http_client,
9         lv_url    TYPE string VALUE 'https://services.odata.org/v2/Northwind/Northwind.svc/Suppliers?$top=30&$format=json',
10        lv_result  TYPE string,
11        lv_dados  TYPE zest_lei_api_json_a115,
12        lv_tab    TYPE STANDARD TABLE OF ztab_api_a115.
13
14  cl_http_client=>create_by_url(
15  EXPORTING
16    url = lv_url
17  IMPORTING
18    client = lo_client
19  EXCEPTIONS
20    argument_not_found = 1
21    plugin_not_active  = 2
22    internal_error     = 3
23    OTHERS              = 4 ).
24
25  IF sy-subrc IS NOT INITIAL.
26    WRITE: / 'Erro ao instanciar a URL'.
27    EXIT.
28  ENDIF.
```

Editor ABAP: Report ZR\_LEITOR\_BAPI\_A115 exibir

Editor ABAP: Report ZR\_LEITOR\_BAPI\_A115 exibir

Report ZR\_LEITOR\_BAPI\_A115 ativo

```
28: ENDIF.
29:
30:
31: lo_client->propertytype_logon_popup = lo_client->co_disabled. "comando para nao precisar de usuario e senha.
32:
33: lo_client->send(
34: EXCEPTIONS
35:   http_communication_failure = 1
36:   http_invalid_state         = 2
37:   http_processing_failed     = 3
38:   http_invalid_timeout       = 4
39:   OTHERS                     = 5 ).
40:
41: IF sy-subrc IS NOT INITIAL.
42:   WRITE: / 'Erro no comando send'.
43:   EXIT.
44: ENDIF.
45:
46: lo_client->receive(
47: EXCEPTIONS
48:   http_communication_failure = 1
49:   http_invalid_state         = 2
50:   http_processing_failed     = 3
51:   OTHERS                     = 4 ).
52:
53: IF sy-subrc IS NOT INITIAL.
54:   WRITE: / 'Erro ao receber resposta'.
55:   EXIT.
56: ENDIF.
57:
```

ABAP | Li 1 Co 1

Editor ABAP: Report ZR\_LEITOR\_BAPI\_A115 exibir

Editor ABAP: Report ZR\_LEITOR\_BAPI\_A115 exibir

Report ZR\_LEITOR\_BAPI\_A115 ativo

```
52:
53: IF sy-subrc IS NOT INITIAL.
54:   WRITE: / 'Erro ao receber resposta'.
55:   EXIT.
56: ENDIF.
57:
58: CLEAR lv_result.
59: lv_result = lo_client->response->get_cdata( ).
60:
61:
62: CHECK lv_result IS NOT INITIAL.
63:
64: CLEAR lt_dados.
65:
66: /ui2/cl_json=>deserialize(
67: EXPORTING
68:   json = lv_result
69:   pretty_name = /ui2/cl_json=>pretty_mode-camel_case "Verifica maiusculas e minusculas.
70: CHANGING
71:   data = lt_dados ).
72:
73: *INSERT stab_api_a115 FROM lt_dados.
74:
75:
76: IF lt_dados IS NOT INITIAL.
77:   cl_demo_output=>display( lt_dados ).
78:
79:
80: ENDIF.
```

ABAP | Li 1 Co 1

SAP

Output

LT\_DADOS

_METADATA		SUPPLIERID	COMPANYNAME	CONTACTNAME	CONTACT
URI	TYPE				
https://services.odata.org/V2/Northwind/Northwind.svc/Suppliers (1)	NorthwindModel.Supplier	1	Exotic Liquids	Charlotte Cooper	Purchasing M
URI	TYPE				
https://services.odata.org/V2/Northwind/Northwind.svc/Suppliers (2)	NorthwindModel.Supplier	2	New Orleans Cajun Delights	Shelley Burke	Order Admin
URI	TYPE				
https://services.odata.org/V2/Northwind/Northwind.svc/Suppliers (3)	NorthwindModel.Supplier	3	Grandma Kelly's Homestead	Regina Murphy	Sales Represent
URI	TYPE				
https://services.odata.org/V2/Northwind/Northwind.svc/Suppliers (4)	NorthwindModel.Supplier	4	Tokyo Traders	Yoshi Nagase	Marketing M
URI	TYPE				
https://services.odata.org/V2/Northwind/Northwind.svc/Suppliers (5)	NorthwindModel.Supplier	5	Cooperativa de Quesos 'Las Cabras'	Antonio del Valle Saavedra	Export Admin
URI	TYPE				
https://services.odata.org/V2/Northwind/Northwind.svc/Suppliers (6)	NorthwindModel.Supplier	6	Mayumi's	Mayumi Ohno	Marketi Represent
URI	TYPE				
https://services.odata.org/V2/Northwind/Northwind.svc/Suppliers (7)	NorthwindModel.Supplier	7	Pavlova, Ltd.	Ian Devling	Marketing M

- O segundo programa deve conter os dados do fornecedor no SAP e os dados que foram salvos da API.

Programa Processar Ir para Sistema Ajuda

Desafio de APIs e Interfaces

Fornecedor

Código Fornecedor	<input type="text"/>	até	<input type="text"/>	<input type="button" value="F5"/>
SupplierID	<input type="text"/>	até	<input type="text"/>	<input type="button" value="F5"/>



Editor ABAP: Include ZR\_DES\_API\_A115\_S01 exibir

Editor ABAP: Include ZR\_DES\_API\_A115\_S01 exibir

Include ZR\_DES\_API\_A115\_S01 ativo

```
1  *-----*
2  * Include      ZR_DES_API_A115_S01
3  *-----*
4
5  *-----*
6  * Tabelas
7  *-----*
8  TABLES: lfal, ztab_api_all5, ztab_dpara_all5.
9
10
11 *-----*
12 * Types
13 *-----*
14 TYPES: BEGIN OF ty_lfal,
15         lifnr      TYPE lfal-lifnr,
16         landl      TYPE lfal-landl,
17         namel      TYPE lfal-namel,
18         ort01      TYPE lfal-ort01,
19         regio      TYPE lfal-regio,
20         supplierid TYPE ZTAB_dpara_All5-supplierid,
21     END OF ty_lfal.
22
23
24 TYPES: BEGIN OF ty_api,
25         supplierid TYPE ztab_api_all5-supplierid,
26         companyname TYPE ztab_api_all5-companyname,
27         contactname TYPE ztab_api_all5-contactname,
28         contacttitle TYPE ztab_api_all5-contacttitle,
29         address      TYPE ztab_api_all5-address,
30         city         TYPE ztab_api_all5-city.
```

| ABAP | Li 1 Co 1

Editor ABAP: Include ZR\_DES\_API\_A115\_S01 exibir

Editor ABAP: Include ZR\_DES\_API\_A115\_S01 exibir

Include ZR\_DES\_API\_A115\_S01 ativo

```
22
23
24 TYPES: BEGIN OF ty_api,
25         supplierid TYPE ztab_api_all5-supplierid,
26         companyname TYPE ztab_api_all5-companyname,
27         contactname TYPE ztab_api_all5-contactname,
28         contacttitle TYPE ztab_api_all5-contacttitle,
29         address      TYPE ztab_api_all5-address,
30         city         TYPE ztab_api_all5-city,
31         region       TYPE ztab_api_all5-region,
32         postalcode   TYPE ztab_api_all5-postalcode,
33         country       TYPE ztab_api_all5-country,
34         phone        TYPE ztab_api_all5-phone,
35         fax          TYPE ztab_api_all5-fax,
36         homepage     TYPE ztab_api_all5-homepage,
37         codigo_p     TYPE ztab_dpara_all5-codigo_p,
38     END OF ty_api.
39
40
41 *-----*
42 * Data
43 *-----*
44 DATA: gt_lfal      TYPE STANDARD TABLE OF ty_lfal,
45        gt_api       TYPE STANDARD TABLE OF ty_api,
46        gt_fornece   TYPE STANDARD TABLE OF ztab_api2_all5,
47        gs_fornece   TYPE ztab_api2_all5,
48        go_alv       TYPE REF TO cl_salv_table,
49        r_funcoes    TYPE REF TO cl_salv_functions_list.
50
51
```

| ABAP | Li 1 Co 1

Editor ABAP: Include ZR\_DES\_API\_A115\_S01 exibir

Editor ABAP: Include ZR\_DES\_API\_A115\_S01 exibir

Include ZR\_DES\_API\_A115\_S01 ativo

```
40
41 *-----*
42 *& Data
43 *-----*
44 DATA: gt_lfal    TYPE STANDARD TABLE OF ty_lfal,
45        gt_api     TYPE STANDARD TABLE OF ty_api,
46        gt_fornec  TYPE STANDARD TABLE OF ztab_api2_all5,
47        gs_fornec  TYPE ztab_api2_all5,
48        go_alv     TYPE REF TO cl_salv_table,
49        r_funcoes  TYPE REF TO cl_salv_functions_list.
50
51
52 *-----*
53 *& Tela de seleção
54 *-----*
55 SELECTION-SCREEN BEGIN OF BLOCK b1 WITH FRAME TITLE TEXT-001.
56   SELECT-OPTIONS: s_for FOR lfal-lifnr,
57                  s_sup FOR ztab_api_all5-supplierid.
58 SELECTION-SCREEN END OF BLOCK b1.
```

ABAP | Li 1 Co 1

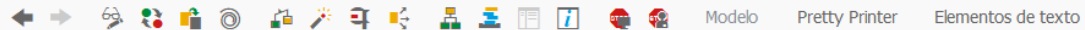
Editor ABAP: Include ZR\_DES\_API\_A115\_F01 exibir

Editor ABAP: Include ZR\_DES\_API\_A115\_F01 exibir

Include ZR\_DES\_API\_A115\_F01 ativo

```
25 FROM lfal
26 LEFT OUTER JOIN ztab_dpara_all5
27 ON lfal-lifnr = ztab_dpara_all5-codigo_p.
28
29 SELECT ztab_api_all5~supplierid
30        ztab_api_all5~companyname
31        ztab_api_all5~contactname
32        ztab_api_all5~contacttitle
33        ztab_api_all5~address
34        ztab_api_all5~city
35        ztab_api_all5~region
36        ztab_api_all5~postalcode
37        ztab_api_all5~country
38        ztab_api_all5~phone
39        ztab_api_all5~fax
40        ztab_api_all5~homepage
41        ztab_dpara_all5~codigo_p
42 INTO TABLE gt_api
43 FROM ztab_api_all5
44 INNER JOIN ztab_dpara_all5
45 ON ztab_api_all5~supplierid = ztab_dpara_all5~supplierid
46 WHERE ztab_api_all5~supplierid IN s_sup.
47
48
49 CLEAR gt_fornec.
50
51
52 LOOP AT gt_lfal INTO DATA(lt_lfal).
53   CLEAR gs_fornec.
54
```

ABAP | Li 1 Co 1



Include	ZR_DES_API_A115_F01	ativo
---------	---------------------	-------

```

52 LOOP AT gt_lfal INTO DATA(lt_lfal).
53     CLEAR gs_fornec.
54
55
56     gs_fornec-fornecedor      = lt_lfal-lifnr.
57     gs_fornec-pais           = lt_lfal-landl.
58     gs_fornec-nome           = lt_lfal-namel.
59     gs_fornec-cidade         = lt_lfal-ort01.
60     gs_fornec-regiao         = lt_lfal-regio.
61
62
63 LOOP AT gt_api INTO DATA(lt_api)
64 WHERE supplierid = lt_lfal-supplierid.
65
66     gs_fornec-supplierid      = lt_api-supplierid.
67     gs_fornec-companyname     = lt_api-companyname.
68     gs_fornec-contactname     = lt_api-contactname.
69     gs_fornec-contacttitle    = lt_api-contacttitle.
70     gs_fornec-address         = lt_api-address.
71     gs_fornec-city            = lt_api-city.
72     gs_fornec-region          = lt_api-region.
73     gs_fornec-postalcode      = lt_api-postalcode.
74     gs_fornec-country         = lt_api-country.
75     gs_fornec-phone           = lt_api-phone.
76     gs_fornec-fax             = lt_api-fax.
77     gs_fornec-homepage        = lt_api-homepage.
78
79
80
81 IF gs_fornec-fornecedor IN s_for[

```



Include	ZR_DES_API_A115_F01	ativo
---------	---------------------	-------

```

79      ENDLOOP.
80
81      IF  gs_fornec-fornecedor IN s_for[]
82      AND gs_fornec-supplierid IN s_sup[].
83
84          APPEND gs_fornec TO gt_fornec.
85
86      ENDIF.
87
88  ENDLOOP.
89
90  ENDFORM.
91
92
93
94
95  *%-----
96  *% Form f_imprime_alv
97  *%-----
98  *% text
99  *%-----
100 *% --> p1      text
101 *% <-- p2      text
102 *%-----
103 FORM f_imprime_alv .
104
105
106 IF gt_fornec IS NOT INITIAL.
107
108

```



Include ZR\_DES\_API\_A115\_F01 ativo

```
109 cl_salv_table=>factory(
110     IMPORTING
111     r_salv_table = go_alv
112     CHANGING
113     t_table = gt_fornec
114 ).
115
116 DATA: lo_columns TYPE REF TO cl_salv_columns_table,
117       lr_column  TYPE REF TO cl_salv_column.
118
119 " Obter configuração de todas as colunas
120 lo_columns = go_alv->get_columns( ).
121
122
123 TRY.
124     lr_column = lo_columns->get_column( columnname = 'MANDT' ).
125     lr_column->set_visible( value = abap_false ).
126 CATCH cx_salv_not_found.
127
128 ENDTRY.
129
130 r_funcoes = go_alv->get_functions( ). "Retorna o objeto de funções
131 r_funcoes->set_all( abap_true ). "Opções padrão do ALV
132
133 go_alv->display( ).
134
135 ENDIF.
136
137 ENDFORM.
```