



# Informatyzacja przedsiębiorstw

Izabela Szczęch

*Politechnika Poznańska*

# Agenda

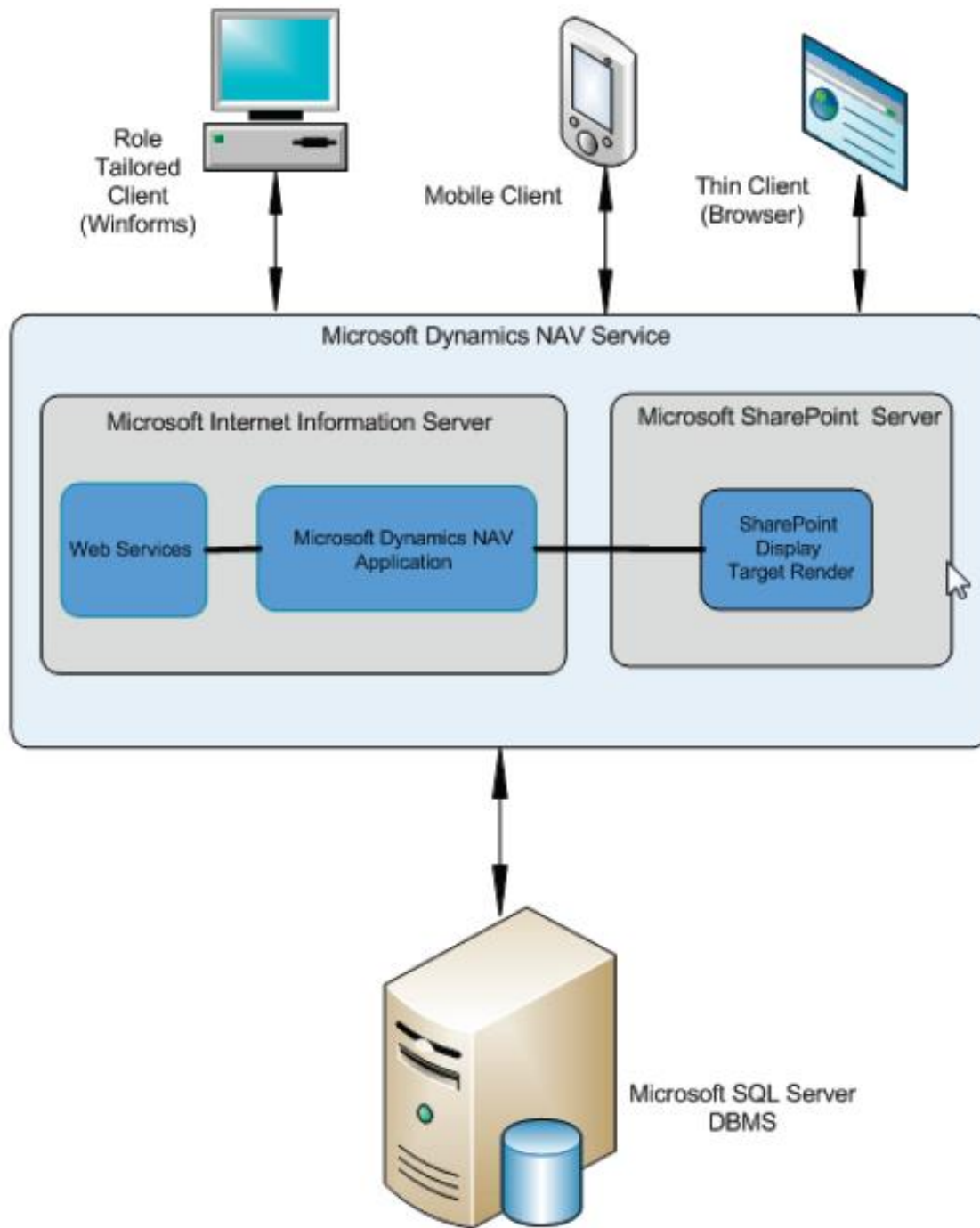
---

## Programming in MS Dynamics NAV

- NAV architecture
- Development environment
  - Tables
  - Pages
  - C/AL programming

# NAV architecture

- Microsoft Dynamics NAV 2016 is designed according to the **three-tier architecture** model:
  - Presentation logic/layers (User Interface) on the client computer,
  - Business logic on another layer available in the three-tier architecture called the service tier,
  - Data and data manipulation layers (DML) on the database server tier.



Development environment

## Development environment

---

**Client/Server Integrated Development Environment (C/SIDE)** - the development environment for Microsoft Dynamics NAV.

**Client Application Language (C/AL)** is the language used for writing functions in C/SIDE.

The C/SIDE user interface is composed of the following:

- **Object Designer**, which contains designers for each of the object types.
- **Navigation Pane Designer**, which is used to arrange MenuSuites on the Navigation Pane.
- Various Tools and Editors, including a C/AL Editor for editing code, a Properties window, a Debugger.

# Development environment

---

- Microsoft Dynamics NAV is not object-oriented but object-based.
- Developers cannot create new types of objects based on the ones that are already in the system.
- Objects created in a database are visible for all companies defined in this database.



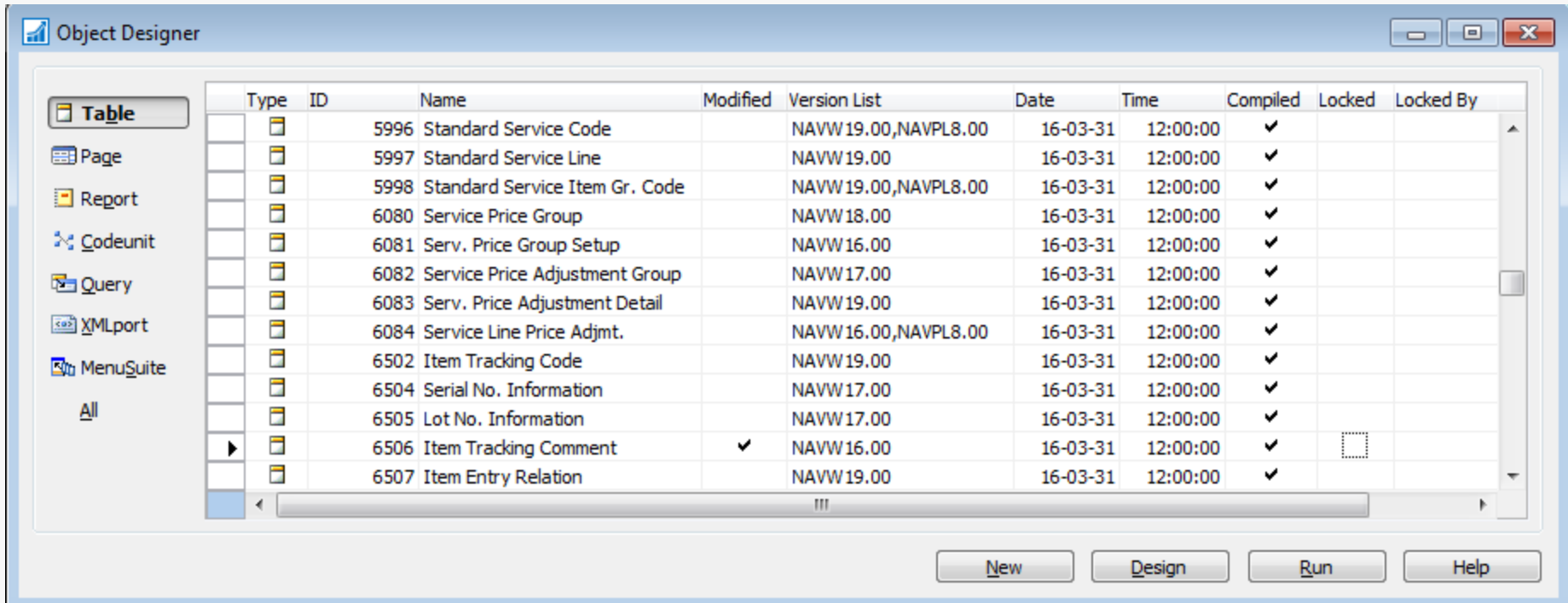
## **Objects available in Microsoft Dynamics NAV:**

- Tables (tabele)
- Forms (formatki)
- Pages (strony)
- Reports (Raporty)
- Codeunits (jednostki kodu)
- Query (zapytania)
- Dataports (Dataporty)-export or import table data in text format
- XMLports (XMLporty)-export or import table data in XML or text format
- MenuSuites (menu)

## **Information about objects:**

- ID
  - standard objects ranges
  - add-on range
  - for student license range from 50 000..50 099
- Name
- Modified
- Version List
- Date
- Time
- Compiled
- Locked/Locked By

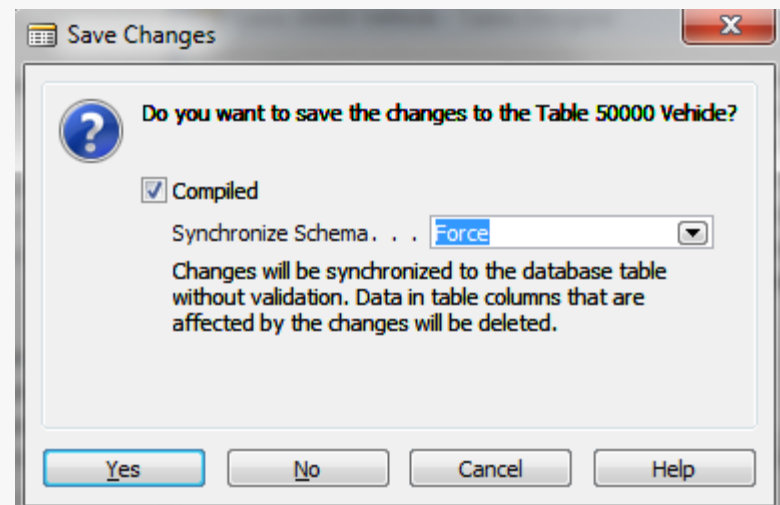
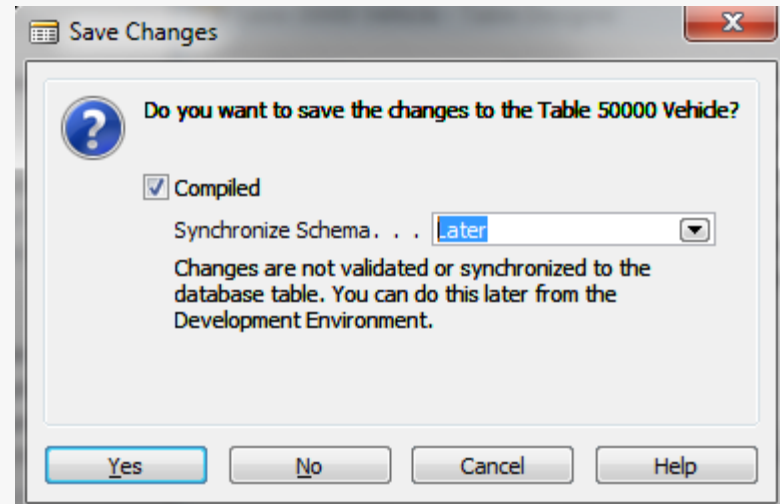
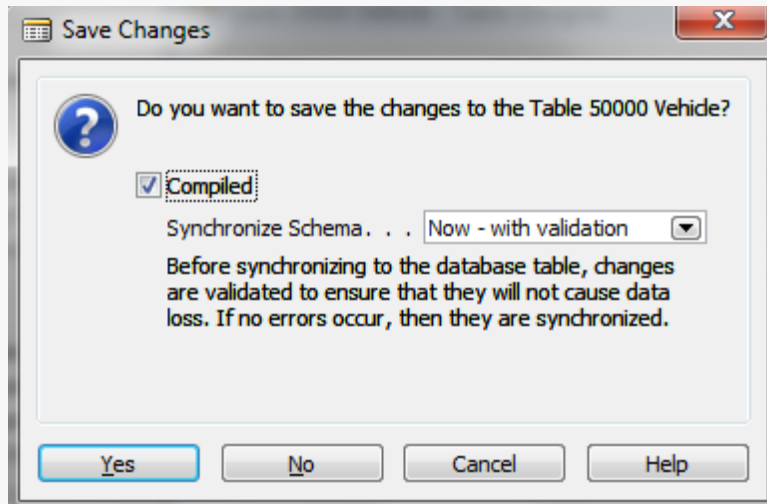
# Information about objects



The screenshot shows the 'Object Designer' window with a list of database objects. The window title is 'Object Designer'. On the left, there is a navigation pane with icons for 'Table', 'Page', 'Report', 'Codeunit', 'Query', 'XMLport', and 'MenuSuite', along with an 'All' option. The main area contains a table with the following columns: Type, ID, Name, Modified, Version List, Date, Time, Compiled, Locked, and Locked By. The table lists 12 objects, with the last one, 'Item Tracking Comment', selected. At the bottom of the window, there are four buttons: 'New', 'Design', 'Run', and 'Help'.

Type	ID	Name	Modified	Version List	Date	Time	Compiled	Locked	Locked By
		5996 Standard Service Code		NAVW19.00,NAVPL8.00	16-03-31	12:00:00	✓		
		5997 Standard Service Line		NAVW19.00	16-03-31	12:00:00	✓		
		5998 Standard Service Item Gr. Code		NAVW19.00,NAVPL8.00	16-03-31	12:00:00	✓		
		6080 Service Price Group		NAVW18.00	16-03-31	12:00:00	✓		
		6081 Serv. Price Group Setup		NAVW16.00	16-03-31	12:00:00	✓		
		6082 Service Price Adjustment Group		NAVW17.00	16-03-31	12:00:00	✓		
		6083 Serv. Price Adjustment Detail		NAVW19.00	16-03-31	12:00:00	✓		
		6084 Service Line Price Adjmt.		NAVW16.00,NAVPL8.00	16-03-31	12:00:00	✓		
		6502 Item Tracking Code		NAVW19.00	16-03-31	12:00:00	✓		
		6504 Serial No. Information		NAVW17.00	16-03-31	12:00:00	✓		
		6505 Lot No. Information		NAVW17.00	16-03-31	12:00:00	✓		
		6506 Item Tracking Comment	✓	NAVW16.00	16-03-31	12:00:00	✓		
		6507 Item Entry Relation		NAVW19.00	16-03-31	12:00:00	✓		

# Object compilation



# Tables

# Tables

---

## **Data types:**

- Integer
- Boolean
- Code – uppercase, 1..250 characters

# Tables

## DOST0001 · Hurtkomp

Ogólne

Nr: DOST0001

Nazwa: Hurtkomp

Adres: ul. Silikonowa 5

Adres 2:

Kod pocztowy:

Kod kraju/regionu:

Nr telefonu:

Nr kontaktu podstawowe

Kontakt:

Miasto: Poznań

Nazwa szukana: HURTKOMP

Saldo (PLN): 0,00

Komunikacja

Fakturowanie

Płatności

Przyjęcie

Handel zagraniczny

Table 18 Customer - Table Designer

E..	Field No.	Field Name	Data Type	Length	Description
✓	1	No.	Code	20	
✓	2	Name	Text	50	
✓	3	Search Name	Code	50	
✓	4	Name 2	Text	50	
✓	5	Address	Text	50	
✓	6	Address 2	Text	50	
✓	7	City	Text	30	
✓	8	Contact	Text	50	
✓	9	Phone No.	Text	30	
✓	10	Telex No.	Text	20	
▶	14	Our Account No.	Text	20	

Pomoc

Table 18 Customer - C/AL Editor

```
Name - OnValidate()  
IF ("Search Name" = UPPERCASE(xRec.Name)) OR ("Search Name" = '') THEN  
    "Search Name" := Name;
```

# Tables

---

## **Data types:**

- Integer
- Boolean
- Code – uppercase, 1..250 characters
- Date
- DateTime
- Decimal
- Option



Edycja - Plan kont

Narzędzia główne Akcje Nawiguj Raporty

Edycja Widok Edytuj listę

Rejestr K/G Wcięcie w planie kont Zapisy księgi Komentarze Wymiary

Plan kont

Sortowanie: Nr

Nr	Na
073-001	U
074-001	Ś
079-999	Um
100-000	Kas
100-001	K
100-002	K
100-999	Kas
131-000	Rac
131-001	W
131-002	W

Table 15 G/L Account - Table Designer

E.	Field No.	Field Name	Data Type	Length	Description
✓	1	No.	Code	20	
✓	2	Name	Text	30	
✓	3	Search Name	Code	30	
✓	4	Account Type	Option		
✓	6	Global Dimension 1 Code	Code	20	
✓	7	Global Dimension 2 Code	Code	20	
✓	9	Income/Balance	Option		
✓	10	Debit			
✓	11	No. 2			
✓	12	Com			
✓	13	Block			

Income/Balance - Properties

Property	Value
Field No.	9
Name	Income/Balance
Caption	Wynikowe/bilansowe
CaptionML	ENU=Income/Balance;PLK=Wynikowe/bilansowe
Description	<>
Data Type	Option
Enabled	<Yes>
InitValue	<Undefined>
FieldClass	<Normal>
AltSearchField	<Undefined>
OptionString	Income Statement,Balance Sheet
OptionCaption	Rachunek wyników,Bilans
OptionCaptionML	ENU=Income Statement,Balance Sheet;PLK=Rachunek wyników,Bilans

# Tables

---

## **Data types:**

- Integer
- Boolean
- Code – uppercase, 1..250 characters
- Date
- DateTime
- Decimal
- Option
- Text – 1..1024 characters
- Time

# Tables

---

## **Field properties** (Shift+F4):

- Field No.
- Name
- Caption, CaptionML
- Description
- Enabled
- FieldClass: „Normal“, „FlowField“, „FlowFilter“
- Editable
- NotBlank
- TableRelation (Conditional Table Relation, TableFilter in Relation)

# Tables

## Field properties example:

Ogólne

Nr:

Nazwa:

Adres:

Kontakt:

Nazwa szukana:

Saldo (PLN):

Saldo (PLN) wg dostawcy:

Limit kredytu (PLN):

Kod sprzedawcy:

Centrum kompetencyjne:

Kod strefy serwisu:

Zablokowane:

Data ostatniej modyfikacji:

Balance - Properties

Property	Value
Field No.	58
Name	Balance
Caption	Saldo
CaptionML	ENU=Balance;PLK=Saldo
Description	<>
Data Type	Decimal
Enabled	<Yes>
InitValue	<Undefined>
FieldClass	FlowField
CalcFormula	Sum("Detailed Cust. Ledg. Entry".Amount WHERE (Customer No.=FIELD(No....
DecimalPlaces	<Undefined>
BlankNumbers	<DontBlank>
BlankZero	<No>
SignDisplacement	<0>
AutoFormatType	
AutoFormatExpr	"Currency Code"
CaptionClass	<>
Editable	No
MinValue	<>
MaxValue	<>
NotBlank	<No>
ValuesAllowed	<>
TableRelation	<Undefined>
ValidateTableRelation	<Yes>
ExtendedDatatype	<None>

Table 18 Customer - Table Designer

E.	Field No.	Field Name	Data Type	Length
<input checked="" type="checkbox"/>	57	Global Dimension 2 Filter	Code	20
<input checked="" type="checkbox"/>	58	Balance	Decimal	
<input checked="" type="checkbox"/>	59	Balance (LCY)	Decimal	
<input checked="" type="checkbox"/>	60	Net Change	Decimal	
<input checked="" type="checkbox"/>	61	Net Change (LCY)	Decimal	
<input checked="" type="checkbox"/>	62	Sales (LCY)	Decimal	

Pomoc

# Tables

## Field properties example

Post Code - Properties

Property	Value
Field No.	91
Name	Post Code
Caption	Kod pocztowy
CaptionML	ENU=Post Code;PLK=
Description	<>
Data Type	Code
Enabled	<Yes>
MaxLength	
InitValue	<Undefined>
FieldClass	<Normal>
AltSearchField	<Undefined>
AutoFormatType	
AutoFormatExpr	<>
CaptionClass	<>
Editable	<Yes>
NotBlank	<No>
Numeric	<No>
CharAllowed	<Undefined>
DateFormula	<No>
ValuesAllowed	<>
SQL Data Type	<Undefined>
TableRelation	IF (Country/Region Code=CONST()) "Post Code" ELSE IF ...
ValidateTableRelation	No
TestTableRelation	No
ExtendedDatatype	<None>

Ogólne

Nr: NAB0001 Kontakt:

Nazwa: ElectronicBay Nazwa szukana: ELECTRONIC...

Adres: ul. Basenowa 4 Saldo (PLN): 12 300,00

Adres 2:  Saldo (PLN) wg dostawcy: 0,00

Kod pocztowy: 60-042 Limit kredytu (PLN):  0,00

▼ Kod	Miasto	Wyszukiw...	Kod kraju...	Województwo
02-515	Warszawa	WARSZAWA		
11-430	Korsze	KORSZE		
14-510	Orneta	ORNETA		
15-660	Białystok	BIAŁYSTOK		
45-418	Opole	OPOLE		
59-300	Lubin	LUBIN		

Komunikacja

Fakturowanie

Płatności

Wydanie

Table 18 Customer - Table Designer

E.	Field No.	Field Name	Data Type	Length
<input checked="" type="checkbox"/>	86	VAT Registration No.	Text	20
<input checked="" type="checkbox"/>	87	Combine Shipments	Boolean	
<input checked="" type="checkbox"/>	88	Gen. Bus. Posting Group	Code	10
<input checked="" type="checkbox"/>	89	Picture	BLOB	
<input checked="" type="checkbox"/>	91	Post Code	Code	20
<input checked="" type="checkbox"/>	92	County	Text	30

Pomoc

# Tables

---

## **Table properties** (under the last field Shift+F4):

- ID
- Name
- Caption, CaptionML
- Description
- DataPerCompany
- Permissions – additional permissions to other tables
- LookupPageID
- DrillDownPageID
- DataCaptionFields – these fields are displayed in the title bar of this table's forms/pages

# Tables

## Table properties example:

The screenshot shows a 'Table - Properties' dialog box with a table of properties and values. The 'LookupPageID' property is set to 'Customer List', and a red box highlights the value 'NAB0001 · ElectronicBay'. Below the table, a detailed view of the customer record is shown, including fields for 'Nr', 'Nazwa', 'Adres', 'Kontakt', 'Saldo (PLN)', and 'Limit kredytu (PLN)'.

Property	Value
ID	18
Name	Customer
Caption	Nabywca
CaptionML	ENU=Customer;PLK=Nabywca
Description	<>
DataPerCompany	<Yes>
Permissions	TableData Cust. Ledger Entry=r
LookupPageID	Customer List
DrillDownPageID	Customer List
DataCaptionFields	No.,Name
PasteIsValid	<Yes>
LinkedObject	<No>

**Ogólne**

Nr:	NAB0001	Kontakt:	
Nazwa:	ElectronicBay	Nazwa szukana:	ELECTRONICBAY
Adres:	ul. Basenowa 4	Saldo (PLN):	12 300,00
Adres 2:		Saldo (PLN) wg dostawcy:	0,00
Kod pocztowy:	60-042	Limit kredytu (PLN):	0,00
Miasto:	Poznań	Kod sprzedawcy:	
Kod kraju/regionu:	BN	Centrum kompetencyjne:	

# Tables

---

## **Field triggers (F9):**

- OnValidate – data is entered in a field or when <Record>.VALIDATE is executed in C/AL code
- OnLookup – Lookup (or F4) is activated

## **Table triggers (F9):**

- OnInsert
- OnModify
- OnDelete
- OnRename
  
- Documentation



# Tables

---

## Field triggers example:

Name - OnValidate()

```
IF ("Search Name" = UPPERCASE(xRec.Name)) OR ("Search Name" = '') THEN  
    "Search Name" := Name;
```

Name - OnLookup()

Search Name - OnValidate()

Search Name - OnLookup()

# Tables

## Table triggers example:

Documentation()

OnInsert()

```
IF "No." = '' THEN BEGIN
    GetInvtSetup;
    InvtSetup.TESTFIELD("Item Nos.");
    NoSeriesMgt.InitSeries(InvtSetup."Item Nos.",xRec."No. Ser
END;
```

```
DimMgt.UpdateDefaultDim(
    DATABASE::Item,"No.",
    "Global Dimension 1 Code","Global Dimension 2 Code");
```

OnModify()

```
"Last Date Modified" := TODAY;
```

```
PlanningAssignment.ItemChange(Rec,xRec);
```

OnDelete()

```
BOMComp.RESET;
BOMComp.SETCURRENTKEY(Type,"No.");
BOMComp.SETRANGE(Type,BOMComp.Type::Item);
BOMComp.SETRANGE("No.,"No.");
IF BOMComp.FIND('-') THEN
    ERROR(Text023, TABLECAPTION,"No.",BOMComp.TABLECAPTION);
```

## **Table Relationships**

There are three kinds of relationships between tables in relational database design:

- One-to-Many Relationships
- Many-to-Many Relationships
- One-to-One Relationships

The fields can use relationships to do the following:

- Validate data entries
- Perform Lookup in other tables
- Automatically propagate changes from one table to other tables

# Tables

---

## **Keys:**

- up to 40 keys can be associated to a table
- the primary key is composed of up to 20 fields in a record
- the number of fields in the primary key together with all the fields in each secondary key must not exceed 20
- first key on the list is the primary key
- Enable (only a secondary key can be changed into an inactive key)
- SumIndexField

# Tables

## Keys example:

E.. Key		SumIndexFields
<input checked="" type="checkbox"/>	Document Type, Document No., Line No.	Amount, Amount Including VAT...
<input checked="" type="checkbox"/>	Document No., Line No., Document Type	
<input checked="" type="checkbox"/>	Document Type, Type, No., Variant Code, Drop Shipment, Location Code, Shipment Date	Outstanding Qty. (Base)
<input checked="" type="checkbox"/>	Document Type, Bill-to Customer No., Currency Code	Outstanding Amount, Shipped ...
	Document Type, Type, No., Variant Code, Drop Shipment, Shortcut Dimension 1 Code, Shortcut Dimension 2 C...	Outstanding Qty. (Base)
	Document Type, Bill-to Customer No., Shortcut Dimension 1 Code, Shortcut Dimension 2 Code, Currency Code	Outstanding Amount, Shipped ...
<input checked="" type="checkbox"/>	Document Type, Blanket Order No., Blanket Order Line No.	
<input checked="" type="checkbox"/>	Document Type, Document No., Location Code	
<input checked="" type="checkbox"/>	Document Type, Shipment No., Shipment Line No.	
<input checked="" type="checkbox"/>	Type, No., Variant Code, Drop Shipment, Location Code, Document Type, Shipment Date	
<input checked="" type="checkbox"/>	Document Type, Sell-to Customer No.	
<input checked="" type="checkbox"/>	Job Contract Entry No.	

## FlowFields

- A FlowField is a virtual field that extends the table data. It is not a permanent part of the table data.
- It is used to calculate values from another table. The information in the FlowFields exists only at run time.
- If a FlowField is a direct source expression of a control on a page, the FlowField is automatically calculated when the page is displayed to update a FlowField.
- Developers use the CALCFIELDS function:  
`Customer.CALCFIELDS (Balance) ;`

# Tables

## FlowFields - example

Table 18 Customer - Table Designer

E..	Field No.	Field Name
✓	54	Last Date Modified
✓	55	Date Filter
✓	56	Global Dimension 1 Filter
✓	57	Global Dimension 2 Filter
▶ ✓	58	Balance
✓	59	Balance (LCY)
✓	60	Net Change
✓	61	Net Change (LCY)
✓	62	Sales (LCY)
✓	63	Profit (LCY)
✓	64	Inv. Discounts (LCY)

Balance - Properties

Property	Value
Field No.	58
Name	Balance
Caption	Saldo
CaptionML	ENU=Balance;PLK=Saldo
Description	<>
Data Type	Decimal
Enabled	<Yes>
InitValue	<Undefined>
FieldClass	FlowField
CalcFormula	Sum("Detailed Cust. Ledg. Entry".Amount WHERE (Custo...
DecimalPlaces	<Undefined>
BlankNumbers	<DontBlank>
BlankZero	<No>
SignDisplacement	<0>
AutoFormatType	
AutoFormatExpr	
CaptionClass	
Editable	
MinValue	
MaxValue	
NotBlank	
ValuesAllowed	
TableRelation	
ValidateTableRelat	
ExtendedDatatype	
Width	

Calculation Formula

Method . . . . . Sum

Reverse Sign . . . . .

Table . . . . . Detailed Cust. Ledg. Entry

Field . . . . . Amount

Table Filter . . . . . Customer No. =FIELD(No.),Initial E...

OK Anuluj Pomoc

## **There are seven kinds of FlowFields:**

- Sum - The sum of a specified set in a column in a table.
- Average - The average value of a specified set in a column in a table.
- Exists - Indicates whether any records exist in a specified set in a table.
- Count - The number of records in a specified set in a table.
- Min - The minimum value in a column in a specified set in a table.
- Max - The maximum value in a column in a specified set in a table.
- Lookup - Looks up a value in a column in another table.



## FlowFields

- A FlowField is always associated with a calculation formula that determines how the value in the FlowField is calculated.
- The following example shows a possible value for the CalcFormula property:

```
Sum("Detailed Cust. Ledger Entry".Amount
```

```
WHERE (Customer No.=FIELD(No.),
```

```
.../
```

```
(Currency Code=FIELD(Currency Filter)));
```

# Tables

## FlowFields - example

Table 18 Customer - Table Designer

E..	Field No.	Field Name
✓	54	Last Date Modified
✓	55	Date Filter
✓	56	Global Dimension 1 Filter
✓	57	Global Dimension 2 Filter
▶ ✓	58	Balance
✓	59	Balance (LCY)
✓	60	Net Change
✓	61	Net Change (LCY)
✓	62	Sales (LCY)
✓	63	Profit (LCY)
✓	64	Inv. Discounts (LCY)

Balance - Properties

Property	Value
Field No.	58
Name	Balance
Caption	Saldo
CaptionML	ENU=Balance;PLK=Saldo
Description	<>
Data Type	Decimal
Enabled	<Yes>
InitValue	<Undefined>
FieldClass	FlowField
CalcFormula	Sum("Detailed Cust. Ledg. Entry".Amount WHERE (Custo...
DecimalPlaces	<Undefined>
BlankNumbers	<DontBlank>
BlankZero	<No>
SignDisplacement	<0>
AutoFormatType	
AutoFormatExpr	
CaptionClass	
Editable	
MinValue	
MaxValue	
NotBlank	
ValuesAllowed	
TableRelation	
ValidateTableRelat	
ExtendedDatatype	
Width	

Calculation Formula

Method . . . . . Sum

Reverse Sign . . . . .

Table . . . . . Detailed Cust. Ledg. Entry

Field . . . . . Amount

Table Filter . . . . . Customer No. =FIELD(No.),Initial E... ..

OK Anuluj Pomoc

## **SumIndexFields**

- A SumIndexField is a decimal field that can be attached to a key definition.
- This is the fundamental feature of the Microsoft Dynamics NAV database that constructs the basis for FlowFields.
- SumIndexFields enable fast calculation of numeric columns in tables, even in tables with thousands of records.
- This is because SumIndexFields are maintained when the database record is updated.

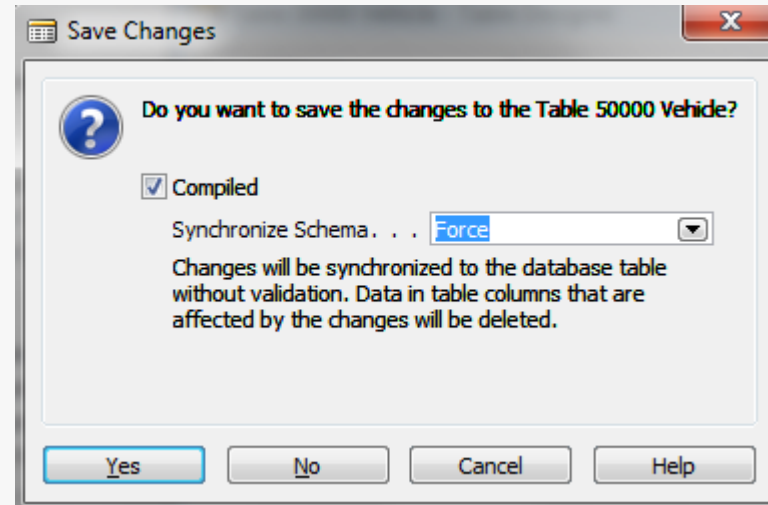
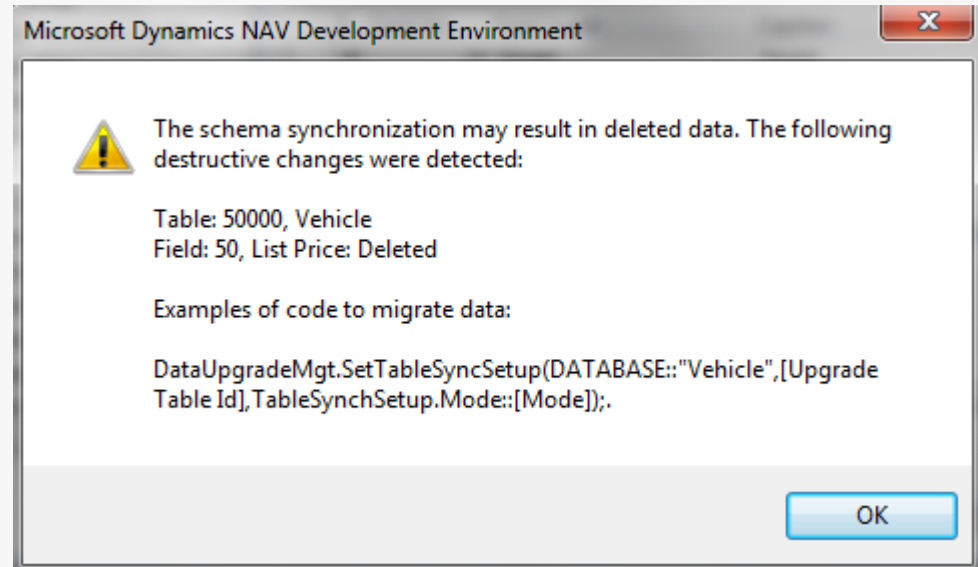
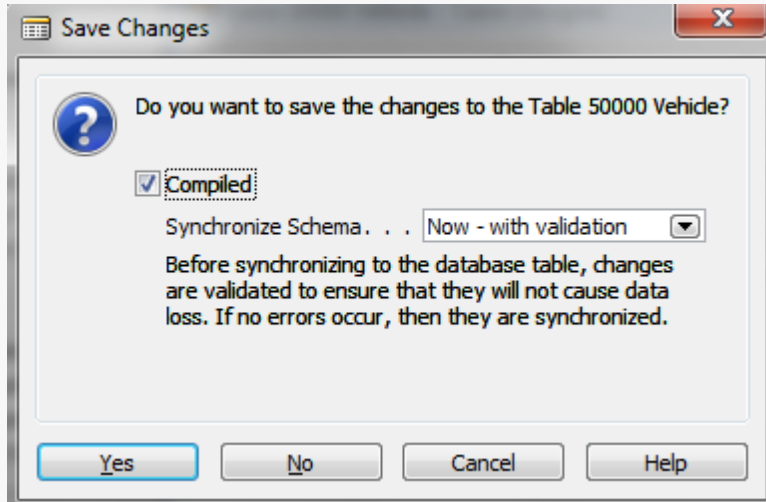
# Tables

## SumIndexFields – example

E.. Key	SumIndexFields
<input checked="" type="checkbox"/> Entry No.	
<input checked="" type="checkbox"/> Cust. Ledger Entry No.,Posting Date	
<input checked="" type="checkbox"/> Cust. Ledger Entry No.,Entry Type,Posting Date	Amount,Amount (LCY),Debit A...
<input checked="" type="checkbox"/> Customer No.,Initial Entry Due Date,Posting Date,Currency Code	Amount,Amount (LCY),Debit A...
<input checked="" type="checkbox"/> Customer No.,Initial Entry Due Date,Posting Date	Amount,Amount (LCY)
<input checked="" type="checkbox"/> Customer No.,Posting Date,Entry Type,Currency Code,Advance	Amount,Amount (LCY),Debit A...
<input checked="" type="checkbox"/> Document No.,Document Type,Posting Date	
<input checked="" type="checkbox"/> Customer No.,Initial Document Type,Document Type,Entry Type,Posting Date	Amount,Amount (LCY)
<input checked="" type="checkbox"/> Customer No.,Initial Entry Due Date,Posting Date,Initial Entry Global Dim. 1,Initial Entry Global Dim. 2,Currenc...	Amount,Amount (LCY),Debit A...
<input checked="" type="checkbox"/> Customer No.,Posting Date,Entry Type,Initial Entry Global Dim. 1,Initial Entry Global Dim. 2,Currency Code	Amount,Amount (LCY)
<input checked="" type="checkbox"/> Customer No.,Initial Document Type,Document Type,Entry Type,Initial Entry Global Dim. 1,Initial Entry Global ...	Amount,Amount (LCY)
<input checked="" type="checkbox"/> Applied Cust. Ledger Entry No.,Entry Type	
<input checked="" type="checkbox"/> Transaction No.,Customer No.,Entry Type	

# Tables

- FieldClass changed from Normal to FlowField



## **FlowFilter**

- Users may want to limit calculations so that they include only those values in a column that have some specific properties.
- For example, the user may want to sum up only the amounts of customer entries that are entered in April.
- This is possible if the application is designed by using FlowFilter fields for the FlowFields.
- A FlowFilter lets users dynamically change the Table Filter part of the FlowField.

# Tables

## FlowFilter – example

The image shows two overlapping windows from the SAP Table Designer. The background window, titled "Table 18 Customer - Table Designer", displays a table of fields. The foreground window, titled "Date Filter - Properties", shows the configuration for the selected field.

**Table 18 Customer - Table Designer**

Field No.	Field Name	Data Type	Length	Description
55	Date Filter	Date		
56	Global Dimension 1 Filter	Code	20	
57	Global Dimension 2 Filter	Code	20	
58	Balance	Decimal		
59	Balance (LCY)	Decimal		
60	Net Change	Decimal		
61	Net Change (LCY)	Decimal		
62	Sales (LCY)	Decimal		
63	Profit (LCY)			
64	Inv. Discount			
65	Pmt. Discount			

**Date Filter - Properties**

Property	Value
Field No.	55
Name	Date Filter
Caption	Filtr daty
CaptionML	ENU=Date Filter;PLK=Filtr daty
Description	<>
Data Type	Date
Enabled	<Yes>
InitValue	<Undefined>
FieldClass	FlowFilter
BlankNumbers	<DontBlank>
SignDisplacement	<0>

# Tables

## FlowFilter – example

The screenshot shows the 'Net Change - Properties' dialog box with two sub-dialogs open: 'Calculation Formula' and 'Table Filter'.

**Net Change - Properties**

Property	Value
Field No.	
Name	Net Change
Caption	Obroty netto
CaptionML	ENU=Net Change;PLK=C
Description	<>
Data Type	Decimal
Enabled	<Yes>
InitValue	<Undefined>
FieldClass	FlowField
CalcFormula	Sum("Detailed Cust. Ledg. Entry".Amount WHERE (Customer No.=FIELD(No....
DecimalPlaces	<Undefined>
BlankNumbers	<DontBlank>
BlankZero	<No>
SignDisplacement	<0>

**Calculation Formula**

Method . . . . . Sum

Reverse Sign . . . . .

Table . . . . . Detailed Cust. Ledg. Entry

Field . . . . . Amount

Table Filter . . . . . Customer No.=FIELD(No.),Initial E...

Buttons: OK, Anuluj, Pomoc

**Table Filter**

Field	Type	Value	OnlyMaxL...	ValueIsFilter
Customer No.	FIELD	No.		
Initial Entry Global Dim. 1	FIELD	Global Dimension 1 Filter		
Initial Entry Global Dim. 2	FIELD	Global Dimension 2 Filter		
▶ Posting Date	FIELD	Date Filter	<input checked="" type="checkbox"/>	
Currency Code	FIELD	Currency Filter		

Buttons: OK, Anuluj, Pomoc



# Tables

## FlowFilter – example

The screenshot displays a software interface with a table and several panels. The top panel shows a search bar with the text "Pisz, aby filtrować (...)" and a dropdown menu set to "Nr". Below this is a "Sortowanie" section with a "Zmień sortowanie" button and a keyboard shortcut "Ctrl+T". The "Filtry" section contains three options: "Filtr zaawansowany" (Shift+F3), "Ogranicz sumy" (Ctrl+Shift+F3), and "Wyczyść filtr" (Ctrl+Shift+A). The "Ogranicz sumy" option is highlighted with a red box. The "Widoki" section has a "Zapisz widok jako..." button. The main table has columns: "Centrum ...", "Kod lokali...", "Nr telefonu", and "Kontakt". The visible rows are:

Centrum ...	Kod lokali...	Nr telefonu	Kontakt
KRAKÓW	NIEBIESKI		Pan Daniel Durrer
			Pan Jacek Maśliński
			Pani Barbara Moreland

A secondary window titled "Ogranicz sumy do pozycji:" is overlaid on the table. It contains a search bar "Gdzie" with a dropdown menu set to "Filtr daty" and the text "jest" and "Wprowadź wartość". Below this is a "Dodaj filtr" section with a dropdown menu set to "Filtr daty". The table in the background of this window has columns: "Nr", "Kod lokali...", "Nr telefonu", and "Kontakt". The visible rows are:

Nr	Kod lokali...	Nr telefonu	Kontakt
10000	NIEBIESKI		Pan Daniel Durrer
20000			Pan Jacek Maśliński
30000			Pani Barbara Moreland
40000	NIEBIESKI		Pan Artur Rybka
50000	Filtry Polskie	WARSZAWA	Pan Grzegorz Grunwald
61000	Dobry Dźwięk Sp. z o.o.	WARSZAWA	NIEBIESKI
NAB0001	ElectronicBay		NIEBIESKI

Pages

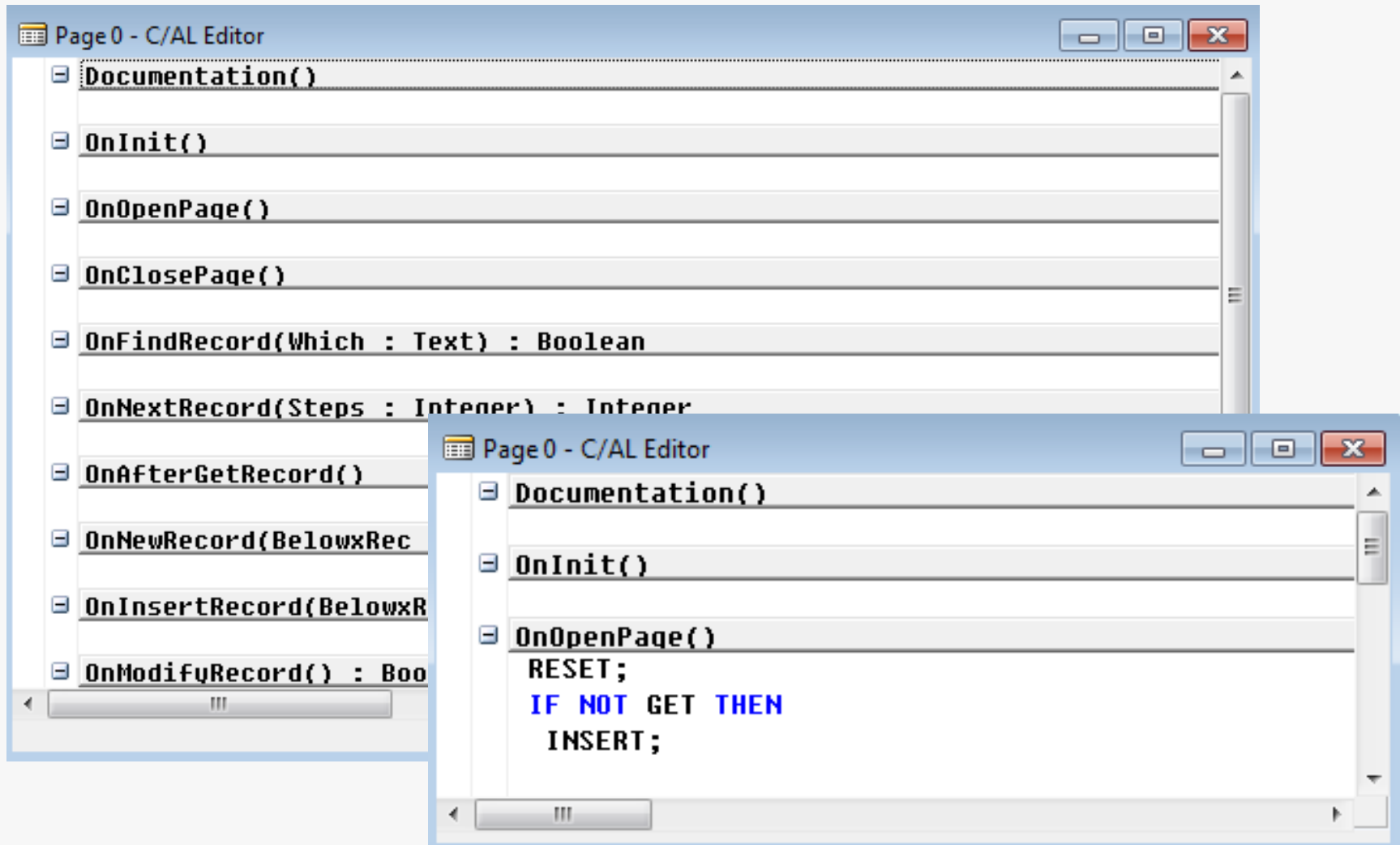
## **Page components:**

- Page properties
- Triggers
- Actions
  - Properties
  - Triggers

Property	Value
ID	17
Name	G/L Account Card
Caption	Kartoteka konta K/G
CaptionML	ENU=G/L Account Card;PLK=Kar...
Editable	<Yes>
Description	<>
Permissions	<Undefined>
PageType	Card
InstructionalTextML	<Undefined>
CardPageID	<Undefined>
DataCaptionExpr	<Undefined>
RefreshOnActivate	Yes
PromotedActionCategoriesML	<Undefined>
SourceTable	G/L Account
SourceTableView	<Undefined>
InsertAllowed	<Yes>
ModifyAllowed	<Yes>
DeleteAllowed	<Yes>
DelayedInsert	<No>
ShowFilter	<Yes>
MultipleNewLines	<No>
SaveValues	<No>
AutoSplitKey	<No>
DataCaptionFields	<Undefined>
SourceTableTemporary	<No>
LinksAllowed	<Yes>
PopulateAllFields	<Undefined>

# Pages

## Page triggers example:



# Pages

## Page actions example:

The screenshot displays the Microsoft Dynamics NAV interface for the 'Nabywcy' (Suppliers) page. The breadcrumb path is 'Politechnika Poznańska > Działy > Zarządzanie Finansami > Należności > Nabywcy'. The ribbon includes 'Narzędzia główne', 'Akcje', 'Nawiguj', and 'Raporty'. The 'Akcje' ribbon contains buttons for 'Faktura korygująca sprzedaży', 'Nota odsetkowa', 'Edycja', 'Widok', 'Usuń', 'Dziennik sprzedaży', 'Nabywca - saldo na dzień', 'Konta bankowe', 'Karty kredytowe', and 'Oferty'. The main area shows a table of suppliers with columns: Nr, Nazwa, Centrum..., Kod lokali..., Nr telefonu, Kontakt, and Nazwa. The table is sorted by 'Nr' and contains 7 rows of data.

Nr	Nazwa	Centrum ...	Kod lokali...	Nr telefonu	Kontakt	Nazwa
10000	Grupa Cronos S.A.	KRAKÓW	NIEBIESKI		Pan Daniel Durrer	GRUPA
20000	Selangor Sp. z o.o.				Pan Jacek Maśliński	SELANGO
30000	Jan Ubezpieczenia S.A.				Pani Barbara Moreland	JAN UB
40000	Drukarnia Akcydensowa Gryf		NIEBIESKI		Pan Artur Rybka	DRUKAR
50000	Filtry Polskie	WARSZAWA			Pan Grzegorz Grunwald	FILTRY
61000	Dobry Dźwięk Sp. z o.o.	WARSZAWA	NIEBIESKI			DOBRY
NAB0001	ElectronicBay		NIEBIESKI			ELECTR

# Pages

## Page actions example:

Expanded	Type	SubType	Name	Caption
<input type="checkbox"/>	<b>ActionContainer</b>	<b>NewDocumentItems</b>	<b>&lt;Action1900000005&gt;</b>	<b>&lt;Action190000000...</b>
	Action		<Page Blanket Sales Order >	Zamówienie zbiorcze...
	Action		<Page Sales Quote >	Oferta sprzedaży
	Action		<Page Sales Invoice >	Faktura sprzedaży
	Action		<Page Sales Order >	Zamówienie sprzedaży
	Action		<Page Sales Credit Memo >	Faktura korygująca ...
	Action		<Page Sales Return Order >	Zamówienie zwrotu ...
	Action		<Page Service Quote >	Oferta serwisu
	Action		<Page Service Invoice >	Faktura serwisu
	Action		<Page Service Order >	Zlecenie serwisowe
<input checked="" type="checkbox"/>	Action		<Page Service Credit Memo >	Faktura korygująca ...
	Action		<Page Reminder >	Monit
	Action		<Page Finance Charge Memo >	Nota odsetkowa
<input type="checkbox"/>	<b>ActionContainer</b>	<b>ActionItems</b>	<b>&lt;Action1900000004&gt;</b>	<b>&lt;Action190000000...</b>

- Actions
  - Properties
  - Triggers

## **Page types and characteristics**

Basic kinds of pages in C/SIDE:

- Dialog boxes
- Request pages
- Unbound pages – not associated with any table, mostly used for menu purposes or intermediate means to collect information from users for additional processing
- One-record pages: Customer Card, Vendor Card
- Multi-record pages





Widok



Edycja

Nominały  
pieniężne

OneNote



Komentarze



Łącza

Zarządzaj

Przetwarzanie

Pokaż załączone

## Ustawienia księgi głównej

## Ogólne

Dozw. księgowanie od:

Dozw. księgowanie do:

Rejestruj czas:

Format adresu lokalnego:

Kod poczto... ▼

Nazwa kontaktu w adresie:

Po nazwie f... ▼

Dokładność zaokr. faktur (PLN):

0,01

Typ zaokr. faktury (PLN):

Najbliższy ▼

Dozw. usuwanie kont K/G przed:

Sprawdzaj użycie kont K/G:

Sprawdź księgowanie debetu/kredy...:

Oznacz ilość ujemną jako korektę:

Oznacz faktury kor. jako korekty:

Waluta koszyka EURO:

Kod waluty lokalnej:

PLN

Rab. termin. bez VAT:

Korekta rabatu termin.:

VAT niezrealizowany:

Niezrealizowany VAT przedpłaty:

Maks. dozwolona różnica VAT:

0,00

Typ zaokrąglenia VAT:

Najbliższy ▼

Obl. VAT dla odb. fakt./nabywcy:

Nr odb. fak... ▼

Użyj poprzednich zasad blokowania tabeli Za...:

Numeracja ▼

Wymiary ▼

Raportowanie ▼

Rozliczenie ▼

## **Types of standard pages:**

- Card Page - One Record, example: Customer Card
- Statistics Page - One Record, example: Customer Statistics
- Tabular Page - Multi-Record, example: Currencies
- List Page - Multi-Record, example: Customer List
- Worksheet Page - Multi-Record, example: General Journal
- Header/Line Page- Both (actually two pages), example: Sales Invoice
- Setup Page - One Record, example: General Ledger Setup, Company Information

# C/AL programming

## Introduction to C/AL programming

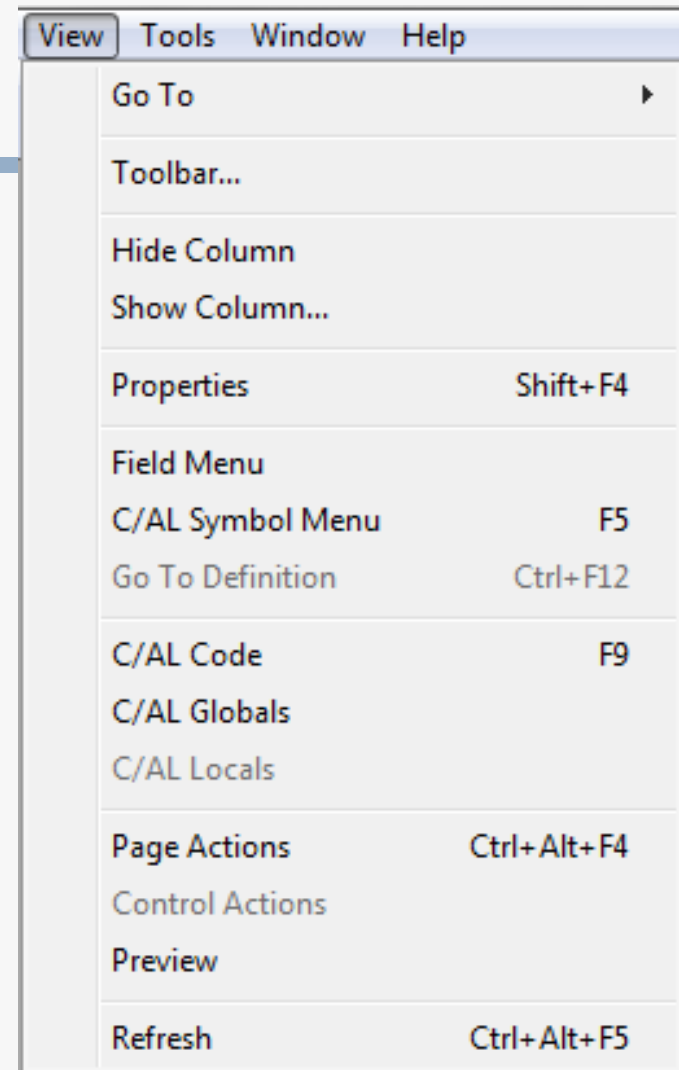
There are three kinds of triggers:

- Documentation triggers
- Event triggers
- Function Triggers

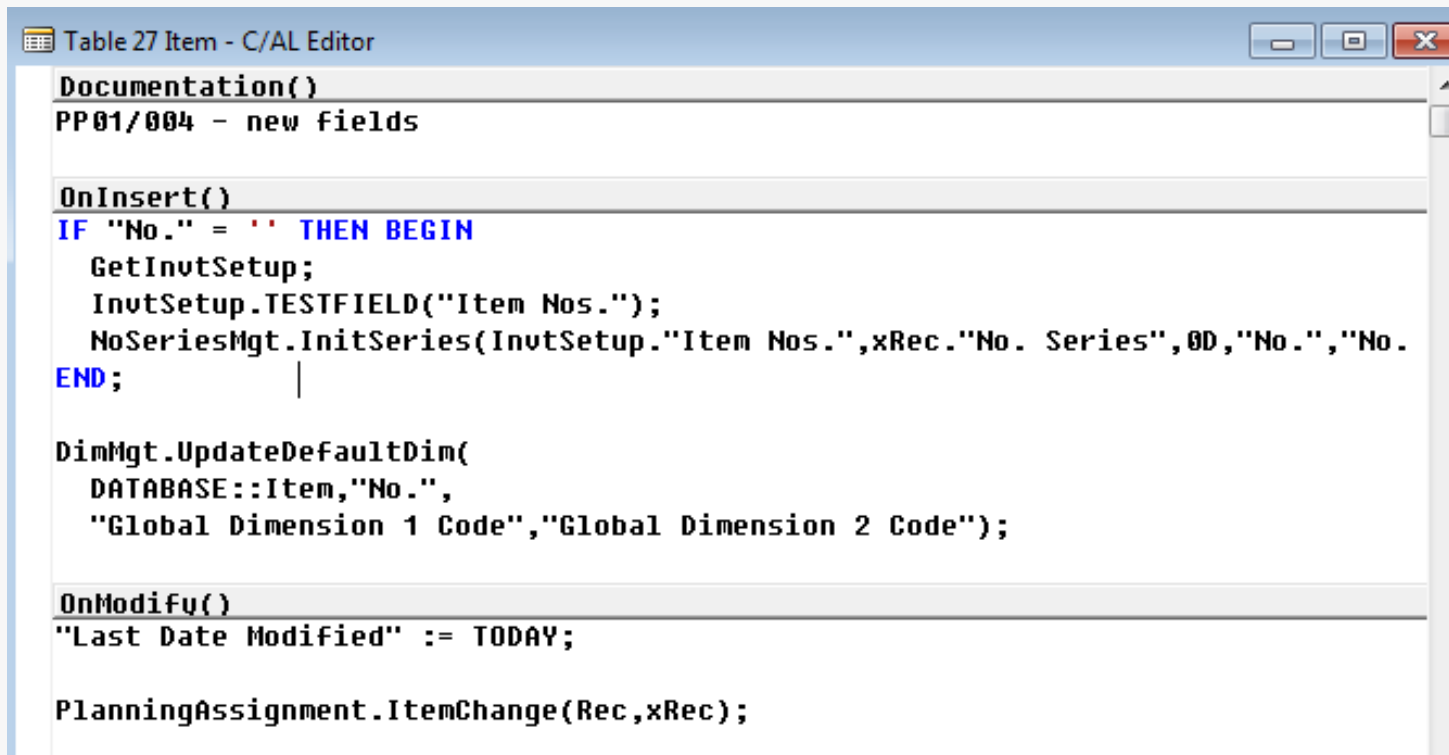
Variable scope:

- Global - it can be accessed anywhere in an object
- Local - it can only be accessed in a single trigger in an object

No variables can be accessed outside the object in which they are defined.



## Documentation/Event triggers example:



```
Table 27 Item - C/AL Editor
Documentation()
PP01/004 - new fields

OnInsert()
IF "No." = '' THEN BEGIN
    GetInvtSetup;
    InvtSetup.TESTFIELD("Item Nos.");
    NoSeriesMgt.InitSeries(InvtSetup."Item Nos.",xRec."No. Series",0D,"No.,"No.
END;

DimMgt.UpdateDefaultDim(
    DATABASE::Item,"No.",
    "Global Dimension 1 Code","Global Dimension 2 Code");

OnModify()
"Last Date Modified" := TODAY;

PlanningAssignment.ItemChange(Rec,xRec);
```

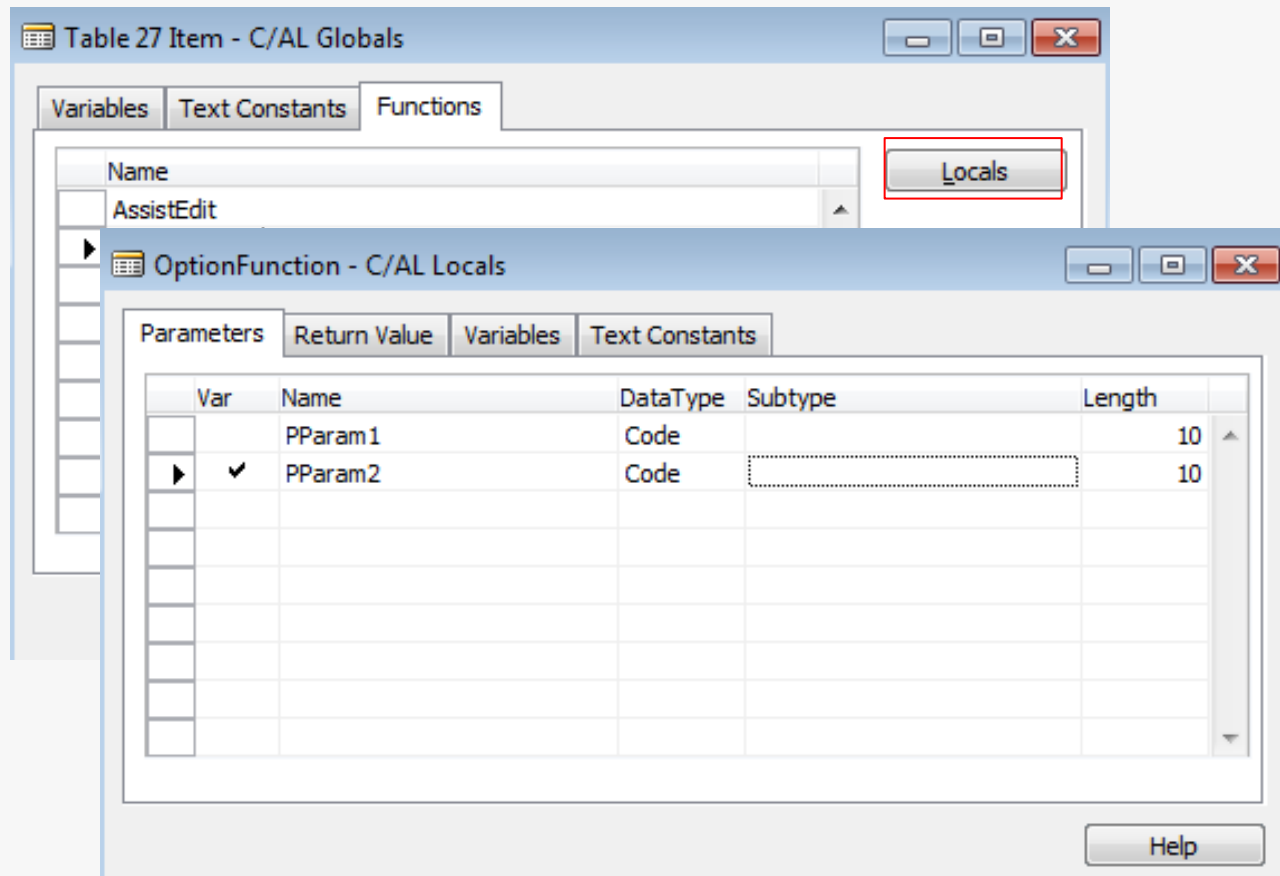
## Function triggers example:

The screenshot displays the SAP C/AL programming environment. On the left, the 'View' menu is open, with 'C/AL Globals' highlighted by a red rectangle. On the right, the 'Table 27 Item - C/AL Globals' window is shown, displaying a list of functions under the 'Functions' tab. The 'FindItemVend' function is selected in the list.

Name
AssistEdit
FindItemVend
ValidateShortcutDimCode
TestNoEntriesExist
TestNoOpenEntriesExist
ItemSKUGet
GetInvntSetup
IsMfgItem
GetGLSetup

## Parameters to functions can be passed by:

- Value
- Reference



```
OptionFunction(PParam1 : Code[10];VAR PParam2 : Code[10])
```

## Function triggers example:

The screenshot displays the Microsoft Dynamics NAV development environment. On the left is the Object Designer with 'Codeunit' selected. The main window shows 'Codeunit 50010 Seminar Management - C/AL Globals' with the 'Functions' tab active. A 'CreateSalesInvoice' function is listed. A 'CreateSalesInvoice - Properties' dialog box is open, showing the 'Local' property set to 'Yes', which is highlighted with a red box. Below the dialog, the C/AL code is visible, with the 'LOCAL' keyword in the function signature highlighted with a red box.

Property	Value
ID	1106000001
Local	Yes
TryFunction	<No>
Event	<No>

```
61 ELSE  
62 MES  
63 }  
64  
65 OnRun  
66  
67 LOCAL CreateSalesInvoice(PSeminarRegHeader :
```



## Variable example:

The screenshot shows the SAP C/AL Globals window for 'Table 27 Item - C/AL Globals'. The window has a menu bar with 'View', 'Tools', 'Window', and 'Help'. Below the menu bar is a toolbar with options like 'Go To', 'Toolbar...', 'Hide Column', 'Show Column...', 'Properties', 'Field Menu', 'C/AL Symbol Menu', 'Go To Definition', 'C/AL Code', 'C/AL Globals', 'C/AL Locals', 'Keys', 'Field Groups', and 'Refresh'. The 'C/AL Globals' menu item is highlighted with a red box. The main area of the window is divided into three tabs: 'Variables', 'Text Constants', and 'Functions'. The 'Variables' tab is selected and highlighted with a red box. It contains a table with the following data:

Name	DataType	Subtype	Length
ProdBOMHeader	Record	Production BOM Header	
ProdBOMLine	Record	Production BOM Line	
ItemIdent	Record	Item Identifier	
RequisitionLine	Record	Requisition Line	
ItemBudgetEntry	Record	Item Budget Entry	
ItemAnalysisViewEntry	Record	Item Analysis View Entry	
ItemAnalysisBudgViewEntry	Record	Item Analysis View Budg. E...	
NoSeriesMgt	Codeunit	NoSeriesManagement	
MoveEntries	Codeunit	MoveEntries	
DimMgt	Codeunit	DimensionManagement	
NonstockItemMgt	Codeunit	Nonstock Item Management	
ItemCostMgt	Codeunit	ItemCostManagement	
ResSkillMgt	Codeunit	Resource Skill Mgt.	

The 'MoveEntries' variable is highlighted with a red box. A 'Help' button is located at the bottom right of the window.

## Text Constants example:

The screenshot displays the SAP C/AL programming environment. The main window is titled "Table 27 Item - C/AL Globals" and has three tabs: "Variables", "Text Constants", and "Functions". The "Text Constants" tab is active and contains a table with the following data:

Name	ConstValue
Text000	You cannot delete %1 %2 because there is at lea...
Text001	You cannot delete %1 %2 because there is at lea...
Text002	You cannot delete %1 %2 because there are one...
Text003	Do you want to change %1?
Text004	You cannot delete %1 %2 because there are one...
Text006	
Text007	
Text008	
Text014	

Below the table, the "Table 27 Item - C/AL Editor" window shows the following code snippet:

```
END;  
END ELSE BEGIN  
IF CurrFieldNo <> 0 THEN  
IF NOT CONFIRM(Text003,FALSE,FIELDCAPTION("Phys Invt Counting Period Code")) THEN  
ERROR(Text7380);  
"Next Counting Period" := '';  
"Last Counting Period Update" := 00;  
END;  
MESSAGE('teksty komunikatów należy robić przez stałe tekstowe !')
```

The "C/AL Globals" menu item is highlighted with a red box. The "Multilanguage Editor" window is also visible, showing the translation of the text constant for different languages:

Language	Value
Angielski (Stany Zjednoczone)	Do you want to change %1?
Polski	Czy zmienić wartość %1?

## **There are six arithmetic operators in C/AL:**

- Plus Operator (+)
- Minus Operator (-)
- Times Operator (\*)
- Divide Operator (/)
- Integer Divide Operator (DIV)
- Modulus Operator (MOD)

### **Before any C/AL code is executed, all developer-defined variables are initialized with certain values:**

- For all variables with a numeric type, the value is zero (**0**).
- For string variables, the value is an empty string ("").
- For Boolean variables, the initial value is **FALSE**.
- For Date and Time type variables, the initial value is **0D** (the undefined date) and **0T** (the undefined time).
- The DateTime variable is initialized to **0DT**.

## **The available relational operators are as follows:**

- = (equal to)
- < (less than)
- > (greater than)
- <= (less than or equal to)
- >= (greater than or equal to)
- <> (not equal to)
- IN (included in set)

- IN - example

```
InOperatorFunction()  
IF 5 IN [2,4,6,8,10] THEN  
    MESSAGE( 'TRUE' )  
ELSE  
    MESSAGE( 'FALSE' );
```

```
InOperatorFunction()  
IF 5 IN [2,4..6,8,10] THEN  
    MESSAGE( 'TRUE' )  
ELSE  
    MESSAGE( 'FALSE' );
```

```
InOperatorFunction()  
IF 'M' IN ['A'..'Z'] THEN  
    MESSAGE( 'TRUE' )  
ELSE  
    MESSAGE( 'FALSE' );
```

## **The available logical operators are as follows:**

- AND
- OR
- XOR
- NOT

## **C/AL Statements**

- IF <Boolean expression> THEN <statement>
- EXIT - C/SIDE exits the current trigger, back to the object that calls the trigger, if any, or back to the user.
- CASE <Expression> OF
  - <Value set 1>: <Statement 1>;
  - <Value set 2>: <Statement 2>; ... ..
  - <Value set n>: <Statement n>;
  - [ELSE <Statement n+1>] END;



## **C/AL Statements**

- FOR
- FOR...DOWNTO
- WHILE...DO
- REPEAT...UNTIL
- WITH

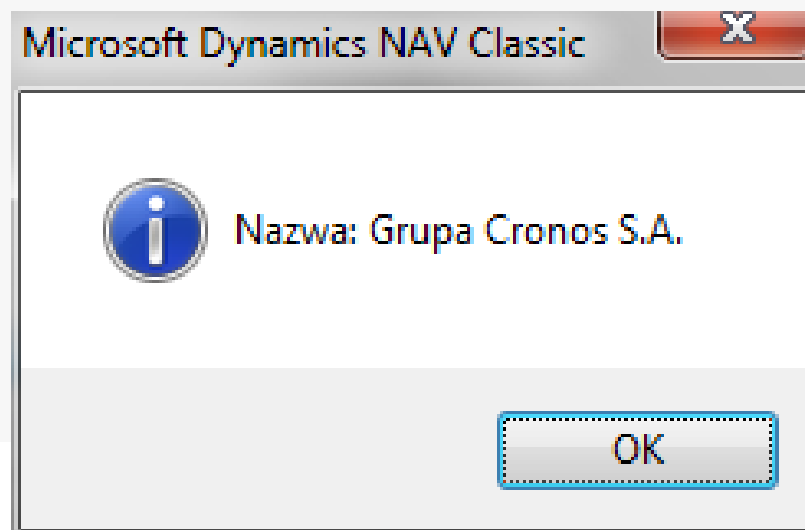
## C/AL Statements

- FOR – example

```
FORExample()  
FOR i := 0 TO 5 DO BEGIN  
    MESSAGE('%1', i);  
END;
```

- REPEAT...UNTIL – example

```
IF Customer.FINDSET THEN BEGIN  
    REPEAT  
        MESSAGE('%1: %2', Customer.FIELDCAPTION(Name), Customer.Name);  
    UNTIL Customer.NEXT = 0;  
END;
```



## C/AL Statements

- WITH – example

```
SalesHeader.GET(SalesHeader."Document Type"::Invoice, 'FUS0001');  
WITH SalesHeader DO BEGIN  
    "Sell-to Customer No." := '10000';  
    "Order Date" := WORKDATE;  
    MODIFY;  
END;
```

## Syntax of comments

- Single-Line Comment //
- Block of Comments {..}

```
//single-line comment  
  
{block  
of  
comments  
}
```

## Arrays

- an array can have one or more dimensions (up to 10 dims. in C/AL)
- Functions : CLEAR function, ARRAYLEN
- no dynamic arrays

# C/AL programming

## Arrays example

The screenshot shows two windows from the C/AL development environment. The 'ArrayFunction - C/AL Locals' window has the 'Variables' tab selected, showing a table with the following data:

Name	DataType
IntArray	Integer
IntArrayTwoDim	Integer
i	Integer

The 'IntArray - Properties' window is open, showing the following data:

Property	Value
ID	1000000000
Dimensions	4

The screenshot shows two windows from the C/AL development environment. The 'ArrayFunction - C/AL Locals' window has the 'Variables' tab selected, showing a table with the following data:

Name	DataType
IntArray	Integer
IntArrayTwoDim	Integer
i	Integer

The 'IntArrayTwoDim - Properties' window is open, showing the following data:

Property	Value
ID	1000000002
Dimensions	4;2

## Arrays example

```
ArrayFunction()  
IntArray[0] := 1; // Error  
CLEAR(IntArray);  
  
IntArray[1] := 1;  
IntArray[2] := 2;  
IntArray[3] := 3;  
  
FOR i := 1 TO 3 DO BEGIN  
    MESSAGE('Element %1: %2', i, IntArray[i]);  
END;  
  
IntArrayTwoDim[1][1] := 1;  
  
MESSAGE('Table size: %1', ARRAYLEN(IntArray));  
  
MESSAGE('Table size: %1', ARRAYLEN(IntArrayTwoDim));
```

### **Built-in functions:**

- Searching for records: GET, FIND, FINDFIRST, FINDLAST, FINDSET, NEXT
- Sorting and filtering records: SETCURRENTKEY, SETRANGE, SETFILTER
- Inserting, Modifying, and Deleting Records: INSERT, MODIFY, MODIFYALL, DELETE, DELETEALL

## Built-in functions example:

```
IF ("Seminar Price" <> xRec."Seminar Price") AND (Status <> Status::Canceled) THEN BEGIN
    SeminarRegLine.RESET;
    SeminarRegLine.SETRANGE("Seminar Registration No.,"No.");
    SeminarRegLine.SETRANGE(Registered,FALSE);
    IF SeminarRegLine.FINDSET THEN
        IF CONFIRM(Text003,FALSE,FIELDCAPTION("Seminar Price"), SeminarRegLine.TABLECAPTION) THEN BEGIN
            REPEAT
                SeminarRegLine.VALIDATE("Seminar Price","Seminar Price");
                SeminarRegLine.MODIFY;
            UNTIL SeminarRegLine.NEXT = 0;
            MODIFY;
        END;
    END;
END;
```

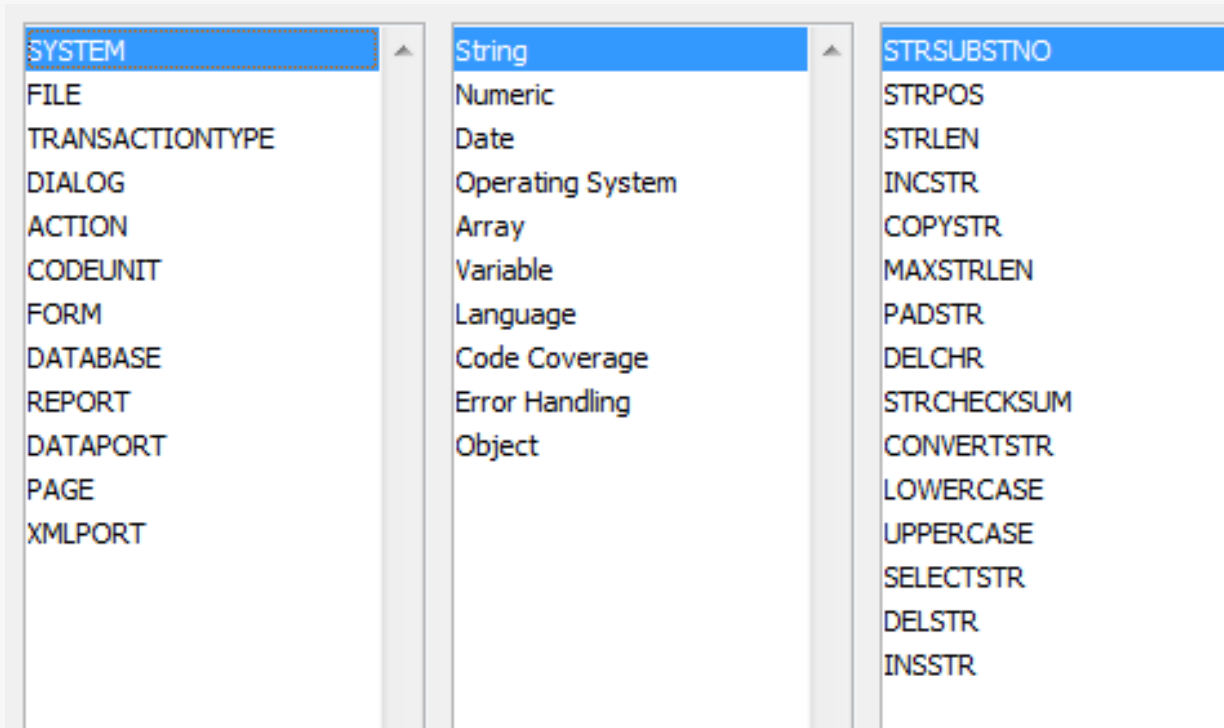


## Built-in functions example:

```
GetPreviousDocDefaultDim(TableID : Integer;DocType : Option;DocNo : Code)
GetGLSetup;
TempDimBuf1.RESET;
TempDimBuf1.DELETEALL;
DocDim.SETRANGE("Table ID",TableID);
DocDim.SETRANGE("Document Type",DocType);
DocDim.SETRANGE("Document No.",DocNo);
DocDim.SETRANGE("Line No.",LineNo);
IF DocDim.FINDSET THEN BEGIN
    REPEAT
        TempDimBuf1.INIT;
        TempDimBuf1."Table ID" := FromTableID;
        TempDimBuf1."Entry No." := 0;
        TempDimBuf1."Dimension Code" := DocDim."Dimension Code";
        TempDimBuf1."Dimension Value Code" := DocDim."Dimension Value Code";
        TempDimBuf1.INSERT;
        IF GLSetupShortcutDimCode[1] = TempDimBuf1."Dimension Code" THEN
            GlobalDim1Code := TempDimBuf1."Dimension Value Code";
        IF GLSetupShortcutDimCode[2] = TempDimBuf1."Dimension Code" THEN
            GlobalDim2Code := TempDimBuf1."Dimension Value Code";
    UNTIL DocDim.NEXT = 0;
END;
```

# C/AL programming - functions

- View → C/AL Symbol Menu (F5)



### **The following functions perform actions on fields:**

- **CALCFIELDS** - is used to update FlowFields
- **CALCSUMS** is used to calculate the sum of one or more fields that are SumIndexFields in the record
- **FIELDERROR** - triggers a run-time error after displaying a fieldrelated error message
- **INIT** - initializes a record, does not initialize the fields of the primary key
- **TESTFIELD** - is used to test a field against a value
- **VALIDATE** - is used to call the OnValidate trigger of a field
- **FIELDNAME/FIELDCAPTION** - returns the name/caption of a field

# C/AL programming - functions

## Examples: CALCFIELDS

10000 · Grupa Cronos S.A.

Ogólne			
Nr:	<input type="text" value="10000"/>	Kontakt:	<input type="text" value="Pan Daniel Durrer"/>
Nazwa:	<input type="text" value="Grupa Cronos S.A."/>	Nazwa szukana:	<input type="text" value="GRUPA CRONOS ..."/>
Adres:	<input type="text" value="ul. Parkowa 22/14"/>	Saldo (PLN):	<input type="text" value="3 813,00"/>
Adres 2:	<input type="text"/>	Saldo (PLN) wg dosta...	<input type="text" value="0,00"/>
Kod pocztowy:	<input type="text" value="11-430"/>	Limit kredytu (PLN):	<input type="text" value="0,00"/>
Miasto:	<input type="text" value="Korsze"/>	Kod sprzedawcy:	<input type="text" value="JR"/>
Kod kraju/regionu:	<input type="text" value="PL"/>	Centrum kompetency...	<input type="text" value="KRAKÓW"/>
Nr telefonu:	<input type="text"/>	Kod strefy serwisu:	<input type="text" value="CENTR"/>
Nr kontaktu podstawo...	<input type="text"/>	Zablokowane:	<input type="text"/>
		Data ostatniej modyfi...	<input type="text" value="10.12.2016"/>

```
Customer.GET('10000');  
MESSAGE('%1', Customer.Balance);
```

```
Customer.GET('10000');  
Customer.CALCFIELDS(Balance);  
MESSAGE('%1', Customer.Balance);
```

# C/AL programming - functions

## Examples:

### CALCSUMS

Table 21 Cust. Ledger Entry - Keys

E.. Key	
<input checked="" type="checkbox"/> Entry No.	
<input checked="" type="checkbox"/> Customer No.,Posting Date,Currency Code	
<input checked="" type="checkbox"/> Customer No.,Currency Code,Posting Date	
<input checked="" type="checkbox"/> Document No.,Document Type,Customer No.	
<input checked="" type="checkbox"/> External Document No.,Document Type,Customer No.	
<input checked="" type="checkbox"/> Customer No.,Open,Positive,Due Date,Currency Code	
<input checked="" type="checkbox"/> Open,Due Date	
<input checked="" type="checkbox"/> Document Type,Customer No.,Posting Date,Currency Code	
<input checked="" type="checkbox"/> Salesperson Code,Posting Date	
<input checked="" type="checkbox"/> Closed by Entry No.	
<input checked="" type="checkbox"/> Transaction No.	
Customer No.,Open,Positive,Calculate Interest,Due Date	
<input checked="" type="checkbox"/> Customer No.,Global Dimension 1 Code,Global Dimension 2 Code,Posting Date,Currency Code	Profit (LCY),Inv. Discount (LCY)
Customer No.,Open,Global Dimension 1 Code,Global Dimension 2 Code,Positive,Due Date,...	Sales (LCY)
Open,Global Dimension 1 Code,Global Dimension 2 Code,Due Date	
Document Type,Customer No.,Global Dimension 1 Code,Global Dimension 2 Code,Posting ...	Sales (LCY)
<input checked="" type="checkbox"/> Customer No.,Applies-to ID,Open,Positive,Due Date	
<input checked="" type="checkbox"/> Customer No.,Currency Code,Document Type	
<input checked="" type="checkbox"/> Document No.,Posting Date,Currency Code	Sales (LCY)
<input checked="" type="checkbox"/> Customer No.,Prepayment Type,Prepayment	
<input checked="" type="checkbox"/> Customer No.,Customer Posting Group,Prepayment,Posting Date	
<input checked="" type="checkbox"/> Document Type,Document No.	
<input checked="" type="checkbox"/> Posting Date	
<input checked="" type="checkbox"/> Customer No.	Sales (LCY)

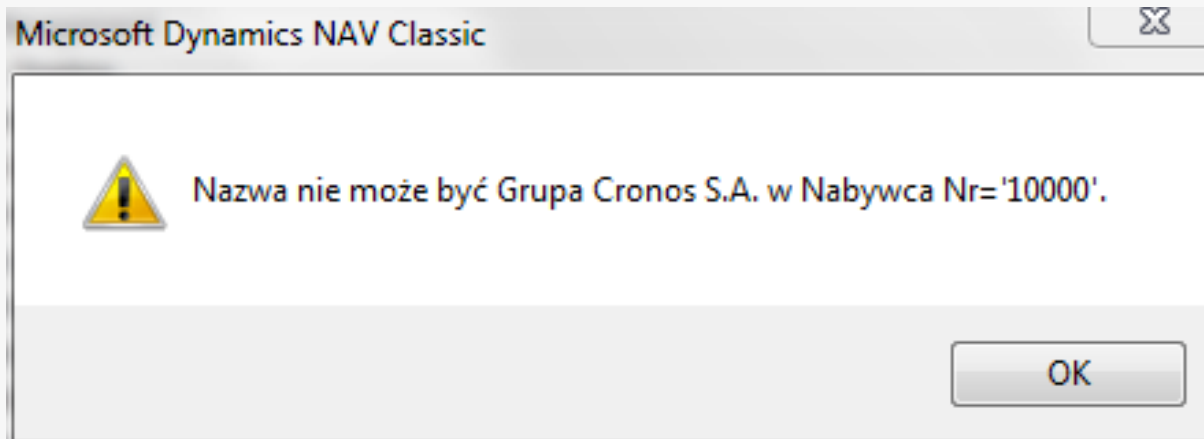
```
CustLedgerEntry.SETCURRENTKEY("Customer No.");  
CustLedgerEntry.SETRANGE("Customer No.", '10000');  
MESSAGE('%1', CustLedgerEntry."Sales (LCY)");
```

```
CustLedgerEntry.SETCURRENTKEY("Customer No.");  
CustLedgerEntry.SETRANGE("Customer No.", '10000');  
CustLedgerEntry.CALCSUMS("Sales (LCY)");  
MESSAGE('%1', CustLedgerEntry."Sales (LCY)");
```

## Examples:

### FIELDERROR

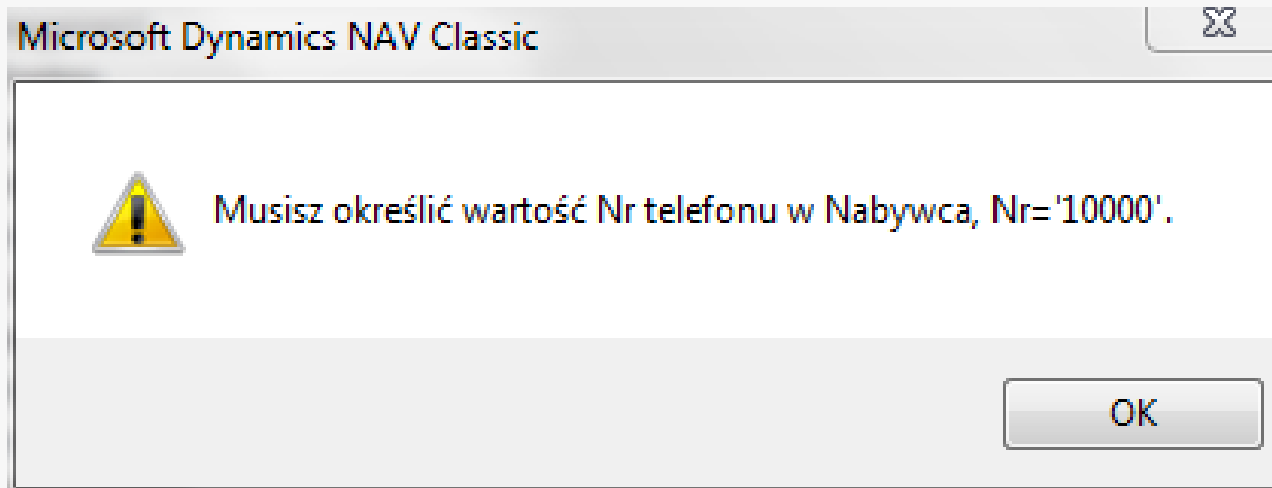
```
Customer.GET( '10000' );  
Customer.FIELDERROR(Name);
```



## Examples:

### FIELDERROR

```
Customer.GET('10000');  
Customer.FIELDERROR("Phone No.");
```



## Examples:

INIT

```
Customer.INIT;  
Customer."No." := '40000';  
Customer.Name := 'Cronus';  
Customer.INSERT;
```



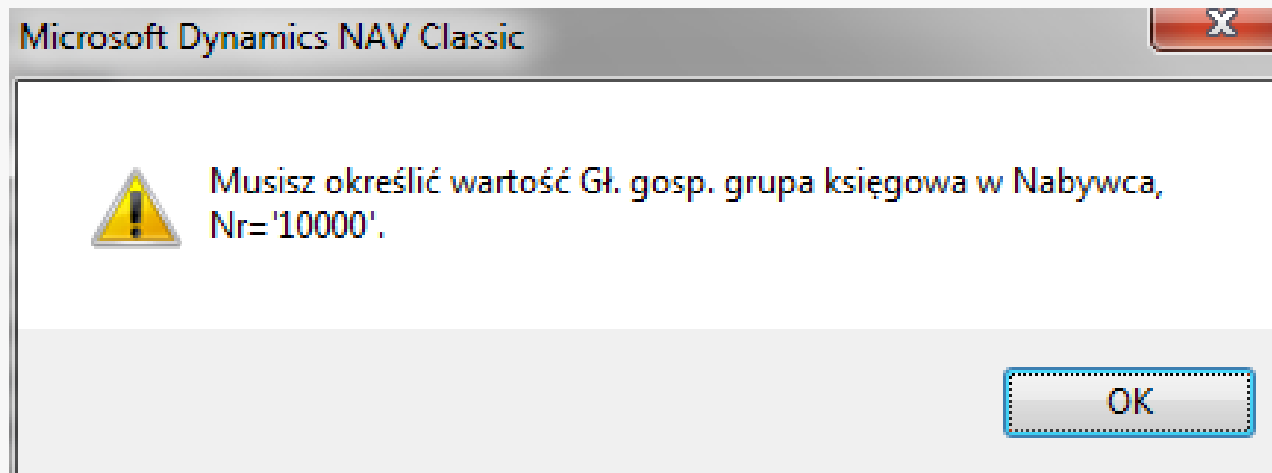
# C/AL programming - functions

---

## Examples:

TESTFIELD

```
Customer.GET('10000');  
Customer.TESTFIELD("Gen. Bus. Posting Group");
```



# C/AL programming - functions

## Examples:

### TESTFIELD

```
Seminar.GET ("Seminar Code") THEN BEGIN  
Seminar.TESTFIELD(Seminar.Blocked, FALSE);
```

### R\_01

✘ Pole Zablockowane musi być równe 'Nie' w Szkolenie: Kod=S002. Bieżąca wartość to 'Tak'. (Wybierz przycisk Odśwież, aby odrzucić błędy)

#### Ogólne

Nr:	<input type="text" value="R_01"/>	Stan:	<input type="text" value="Zakończzone"/>
Data początkowa:	<input type="text" value="2016-11-24"/>	Czas trwania szkolenia:	<input type="text" value="24"/>
Kod szkolenia:	<input type="text" value="✘ S002"/>	Min. liczba uczestników:	<input type="text" value="5"/>
Nazwa szkolenia:	<input type="text" value="SQL Server 2016"/>	Maks. liczba uczestników:	<input type="text" value="10"/>
Kod instruktora:	<input type="text" value="3"/>	Cena szkolenia:	<input type="text" value="200,00"/>
Nazwa instruktora:	<input type="text" value="Postmaster Warszawa"/> ...	Kwota:	<input type="text" value="630,00"/> ...
Data księgowania:	<input type="text" value="2016-11-24"/>		

# C/AL programming - functions

---

## Examples:

### VALIDATE

```
Customer .INIT;  
Customer ."No." := '40000';  
Customer .VALIDATE(Name, 'Cronus');  
Customer .INSERT;
```

Name - OnValidate()

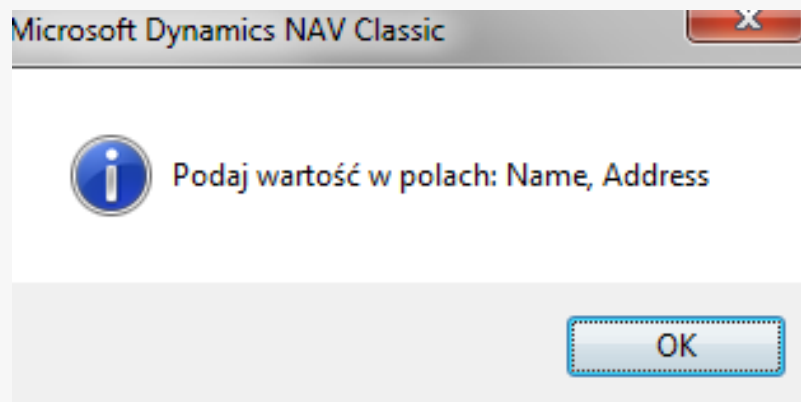
```
IF ("Search Name" = UPPERCASE(xRec.Name)) OR ("Search Name" = '') THEN  
    "Search Name" := Name;
```

# C/AL programming - functions

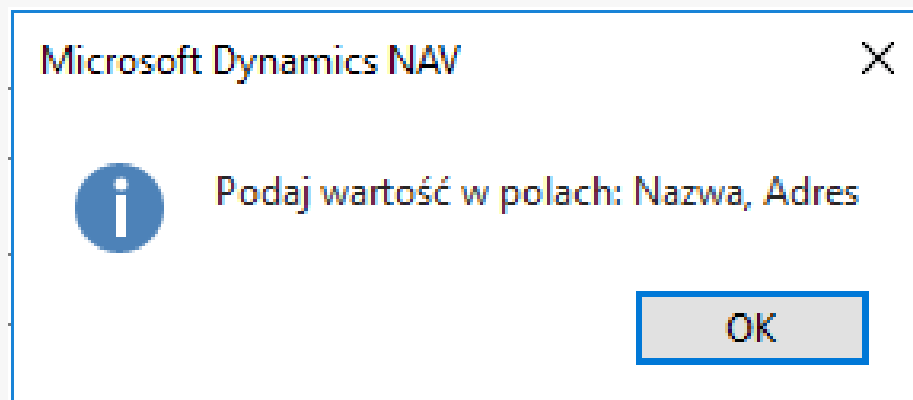
## Examples:

### FIELDNAME, FIELDCAPTION

```
MESSAGE('Podaj wartość w polach: %1, %2', Customer.FIELDNAME(Name), Customer.FIELDNAME(Customer.Address));
```



```
MESSAGE('Podaj wartość w polach: %1, %2', Customer.FIELDCAPTION(Name), Customer.FIELDCAPTION(Address));
```



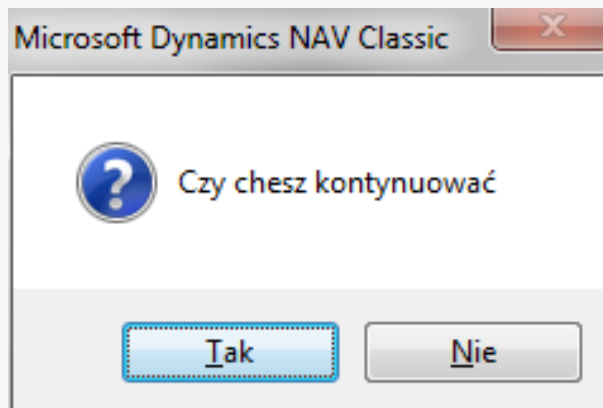
## **Communication functions:**

- MESSAGE: MESSAGE (String [, Value1, ...])
- CONFIRM: CONFIRM (String [, Default] [, Value1, ...])
- ERROR: ERROR(String [, Value1, ...])
- STRMENU: STRMENU (OptionString [, DefaultNumber])

## Communication functions examples:

MESSAGE, CONFIRM, ERROR

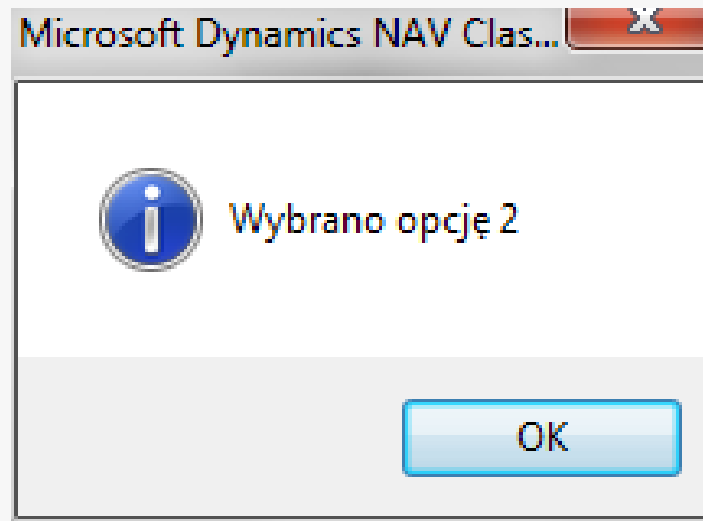
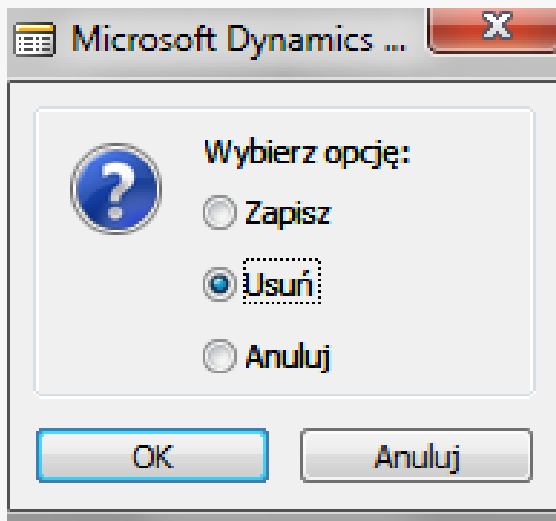
```
IF CONFIRM('Czy chcesz kontynuować', TRUE) THEN  
    MESSAGE('OK')  
ELSE  
    ERROR('Akcja przerwana');
```



## Communication functions examples:

### STRMENU

```
SelectedOption := STRMENU('Zapisz,Usuń,Anuluj', 2, 'Wybierz opcję:');  
MESSAGE('Wybrano opcję %1', SelectedOption);
```



## **String functions:**

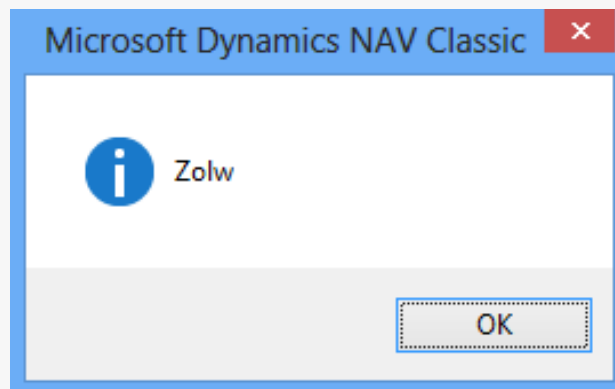
- STRPOS: Position := STRPOS(String, SubString)
- COPYSTR: NewString := COPYSTR(String, Position [, Length])
- STRLEN: Length := STRLEN(String)
- MAXSTRLEN: MaxLength := MAXSTLEN(String)
- LOWERCASE and UPPERCASE: NewString := LOWERCASE(String)
- CONVERTSTR: NewString := CONVERTSTR(String, FromCharacters, ToCharacters)
- DELSTR: NewString := DELSTR(String, Position [, Length])



## String functions examples:

### CONVERTSTR

```
TextToConv := 'Zółw';  
FromStr := 'ąćęłńóśźżĄĆĘŁŃÓŚŻŻ';  
ToStr := 'acelnoszzACELNOSZZ';  
ConvText := CONVERTSTR(TextToConv, FromStr, ToStr);  
MESSAGE(ConvText);
```



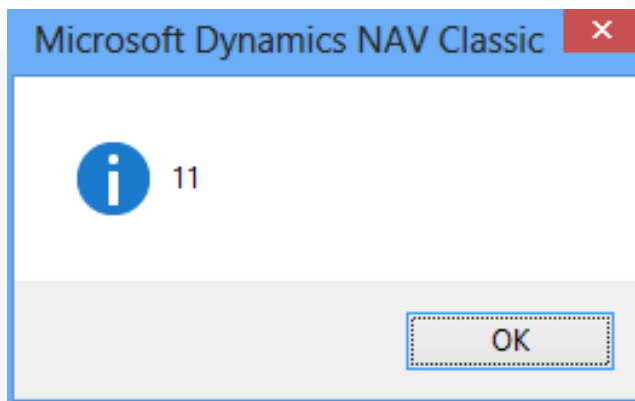
## **System functions:**

- `USERID: Name := USERID;`
- `COMPANYNAME: Name := COMPANYNAME;`
- `TODAY` and `TIME`
- `WORKDATE : [WorkDate] := WORKDATE([NewDate]);`

## Date functions:

- DATE2DMY: IntegerVar := DATE2DMY(Date, Integer);  
Second parameter valid options are 1, 2, and 3.
  - The value 1 corresponds to Day (1-31).
  - The value 2 corresponds to Month (1-12).
  - The value 3 corresponds to Year.

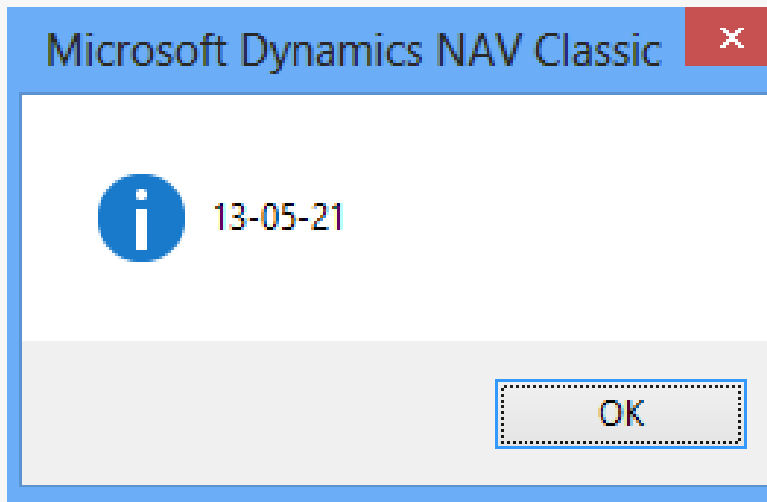
```
MESSAGE('%1',DATE2DMY(171112D, 2));
```



## Date functions:

- CALCDATE: CALCDATE(DateExpression [, Date]);

```
NewDate := CALCDATE('+1D', 130520D);  
MESSAGE('%1', NewDate);
```





## Number functions:

- ABS: NewNumber := ABS(Number)
- ROUND: NewNumber := ROUND(Number [,Precision] [, Direction])
  - **Precision Parameter:** the default value is .01
  - **Direction Parameter:**
    - = Rounds to the nearest value (default)
    - < Rounds down
    - > Rounds up

```
DiscountVATAmount := ROUND("Inv. Discount Amt. (ACY)" * "VAT %" / (1 + "VAT %" / 100) / 100,  
                          Currency."Amount Rounding Precision");
```

- RANDOMIZE: Randomize([Seed])
- RANDOM: Number := RANDOM(MaxNumber)

## Other functions:

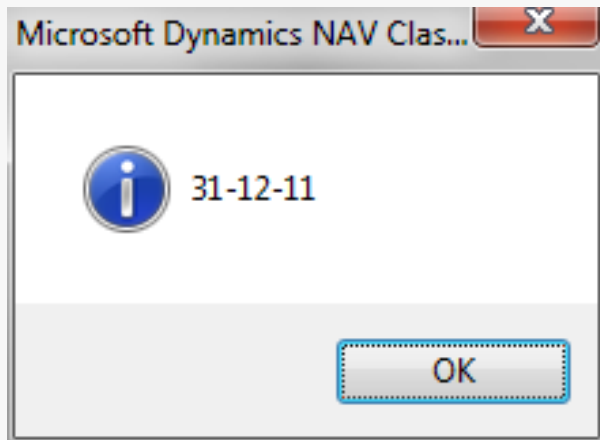
- EXIT: EXIT (Value)
- CLEAR: CLEAR(Variable)
- CLEARALL
- EVALUATE: [Ok :=] EVALUATE(Variable, String[, Number])  
Converts a string expression into another appropriate data type.  
The result is assigned to the Variable parameter.  

```
EVALUATE(IntVar, '100');
```
- FORMAT: String := FORMAT(Value [, Length] [, FormatNumber | FormatString])  
Converts any type of variable to a string variable.

## Other functions examples:

- FORMAT

```
DateVar := 311211D;  
MESSAGE(FORMAT(DateVar));
```





## Other functions examples:

- FORMAT

```
DateVar := 311213D;  
MESSAGE(FORMAT(DateVar, 0, '<Day> <Month Text> <Year4>'));
```

