COSMO CONSULT
Business-Software für Menschen

Microsoft AL

GUI & In-Client Designer

Prerequisite

Use Workshop-VM / Workshop-Computer
follow the instructions of the Boarding Pass

Visual Studio Code

https://code.visualstudio.com

Competence Team DevTech

Microsoft AL

VS Code
Useful Extensions

> **AL Formatter**
>> Indentation
>> Keyword case style
>> Sort variable definitions
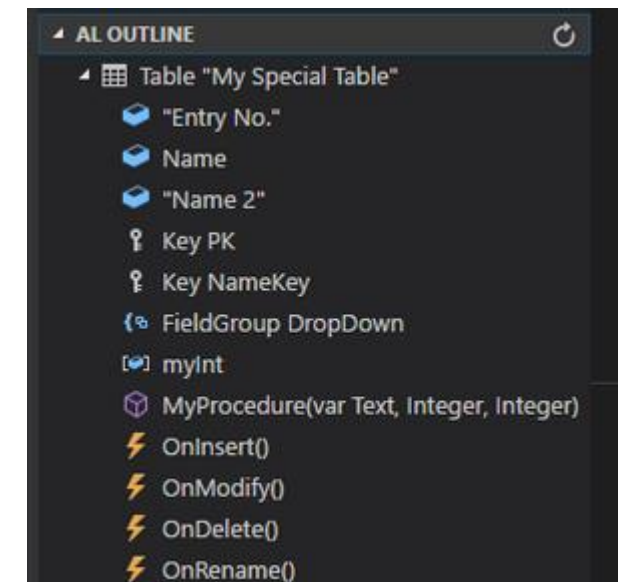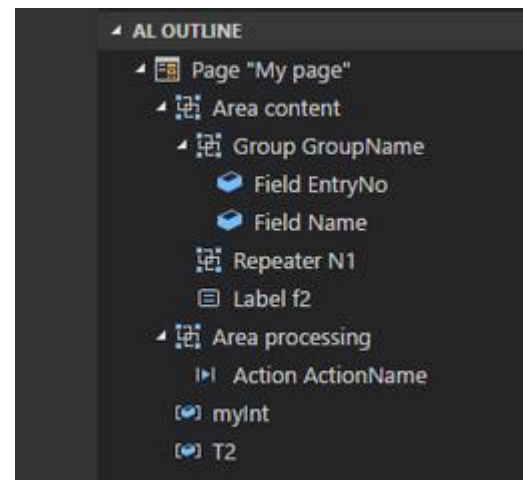>> Readability Guidelines - Spacing and newlines (experimental)

> **AL Code Outline**
>> Outlining – AL Object Structure
>> *"Create List Page"* from Table
>> *"Create Card Page"* from Table

> **al-util**
>> *"Rename & Move"* Objects
>> Templates e.g. "Readme.md"

GUI & In-Client Designer
Agenda

1 Language & Types

2 In-Client Designer

3 Hands On

Microsoft AL

Language & Types

> Page

> Page Extension

> Page Customization

Microsoft AL
Language & Types

> Recommendation for VS Code
>> 1 File per Object
>> Organize Object Files in Folders (Pages, PageExtension, …)

> Use Snippets in AL
>> ttable
>> tpageext
>> tcodeunit
>> …

> Documentation - Development in AL
>> https://docs.microsoft.com/.../developer/devenv-dev-overview

C/Side	Microsoft AL	
	Extending	*New*
Codeunit	-	codeunit
Table	tableextension	table
Page	pageextension pagecustomization	page
XMLport	-	xmlport
Query	-	query
Report	-	report
Menu Suite	-	-
		profile
		controladdin

Microsoft AL
Page

page <ID> „<Unique Name>"

> Content Structure
>> Page Properties
>> Layout Section
>> Actions Section
>> Global Variables
>> Procedures & Events & Page-Triggers

> Sections
>> Layout → Areas → Groups → *Group-Properties* & Fields → *Field-Properties* & Triggers
>> Action → Areas → *Area-Properties* & Actions → *Action-Properties* & Trigger

```
page 58215 "Test Page ccAI"
{
    PageType = Card;
    UsageCategory = Administration;
    ApplicationArea = All;
    Caption = 'cc TEST';

    layout
    {
        0 references
        area(content)
        {
            0 references
            group(GroupName)
            {
            }
        }
    }

    actions
    {
        0 references
        area(processing)
        {
            0 references
            action(ActionName)
            {
                Caption = 'Test';
                Image = TestDatabase;
                ApplicationArea = All;

                trigger OnAction();
                var CU: Codeunit "Intent Handler Test ccAI";
                begin
                    CU.TestStopWorkcenter;

                end;
            }
        }
    }
}
```

Microsoft AL
Page Extension

pageextension <ID> "<Unique Name>" **extends** "< Target Page Name>" / or <ID>

❯ Extensions / Modifications

    ❯ Add   *addfirst(<ID>), addbefore(<ID>), addafter(<ID>), addlast(<ID>)*

    ❯ Move   *movefirst(<ID>), movebefore(<ID>), moveafter(<ID>), movelast(<ID>)*

    ❯ Modify   *modify(<ID>)*

❯ Changes in Sections

    ❯ Layout ➔ <Extension/Modification> ➔ new / changed page content

    ❯ Action ➔ <Extension/Modification> ➔ new / changed page actions

❯ Additional Content

    ❯ Global Variables

    ❯ Procedures & Events & Page-Triggers

```al
pageextension 58205 "Work Center List ccAI" extends "Work Center List"
{
    layout
    {
        addafter(Name)
        {
            0 references
            field(Status; Status)
            { StyleExpr = StatusStyle; }
            0 references
            field("Fault Information"; "Status Message")
            { StyleExpr = StatusStyle; }
        }

        addfirst(FactBoxes)
        {
            2 references
            part("Bot"; "Bot ccAI")
            { }
        }
    }

    actions
    {
        addlast(Processing)
        {
            0 references
            action("Stop Work Center")
            {
                CaptionML = DEU = 'Stop', ENU = 'Stop';
                Image = Stop;
                Promoted = true;
                PromotedIsBig = true;
                ApplicationArea = All;

                trigger OnAction();
                begin
                    Rec.Validate(Status, Status::Down);
                    Rec.Modify;
                end;
            }
        }
    }

    5 references
    var StatusStyle: Text;

    trigger OnAfterGetRecord();
    begin
```

Microsoft AL
Page Customization

pagecustomization "<Unique Name>" **customizes** "<Target Page Name>" / or <ID>

> Modifications
>> Move *movefirst(<ID>), movebefore(<ID>), moveafter(<ID>), movelast(<ID>)*
>> Modify *modify(<ID>)*

> Changes in Sections
>> Layout → <Modification> → changed page content
>> Action → <Modification> → changed page actions

> **No Additional Content!!!**

```
pagecustomization "Item List Cust" customizes "Item List"
{
    layout
    {
        // Modify fields / groups
        modify("No.")
        {
            Visible = false;
        }

        // Move Fields / Groups relative to an Anchor
        movefirst(Content; Description, "Search Description")
    }

    actions
    {
        // Add changes to page actions here
        modify(Action29)
        {
            Visible = false;
        }
    }

    // Variables, procedures and triggers are
    // not allowed on Page Customizations
}
```

In-Client Designer

WSIWYG Designer (Web Client)

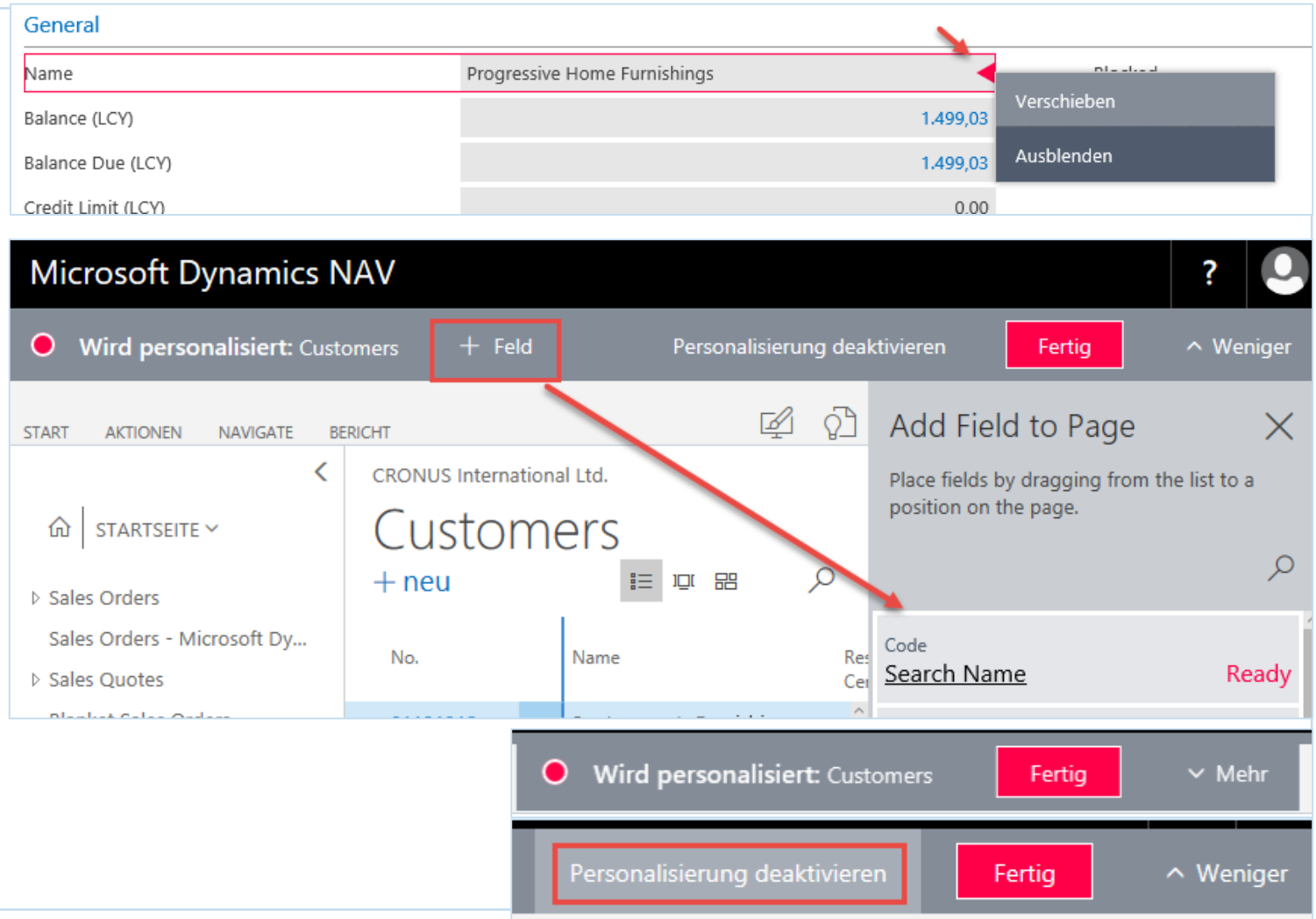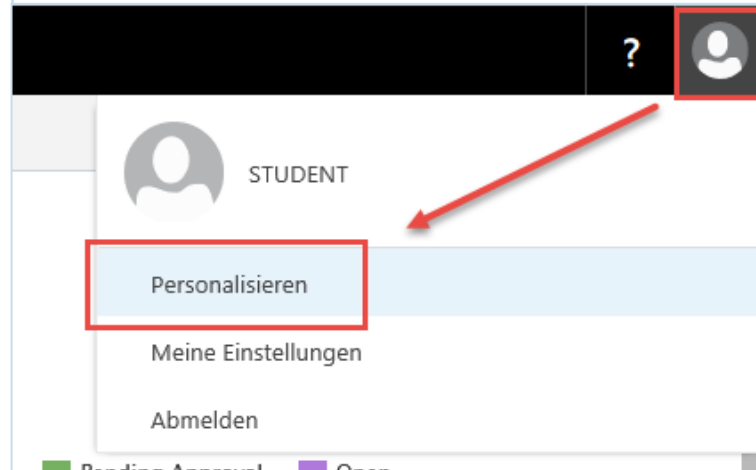> Page Customization

> Page Extension

In-Client Designer
Page Customization

> Customization
>> Individual per User
>> On-Top of all modifications
>> User-Menu → Personalization

> Actions
>> Show / Hide Fields
>> Move Fields

In-Client Designer
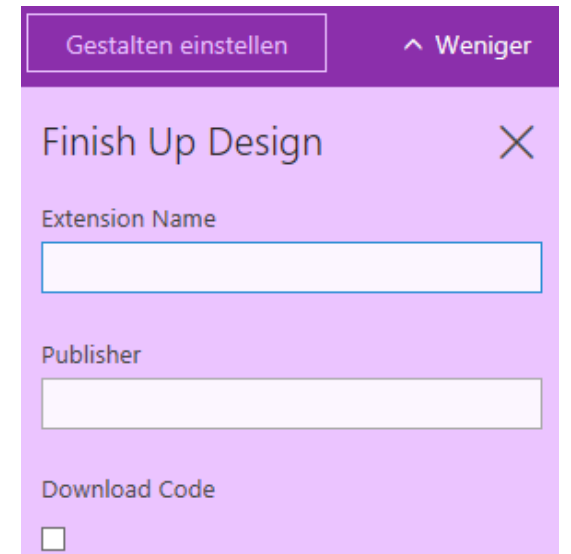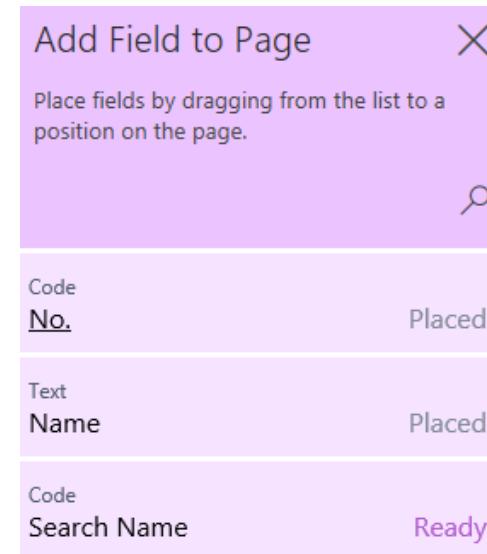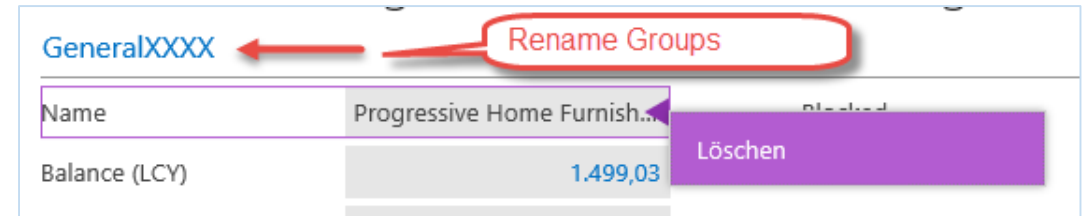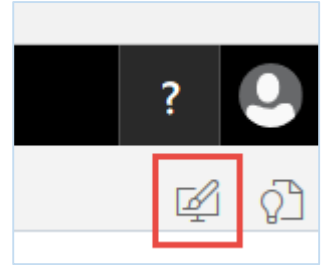## Page Extension

> Extension
>> For all Users
>> On-Top of all **existing** modifications
>> *except Customizations*
>> Designer-Menu

> Designer Actions
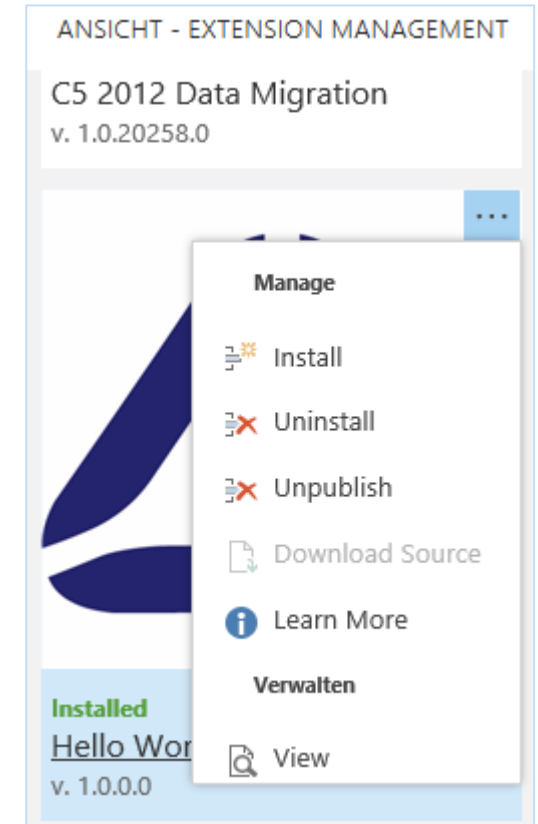>> Rename Groups (e.g. Card Pages)
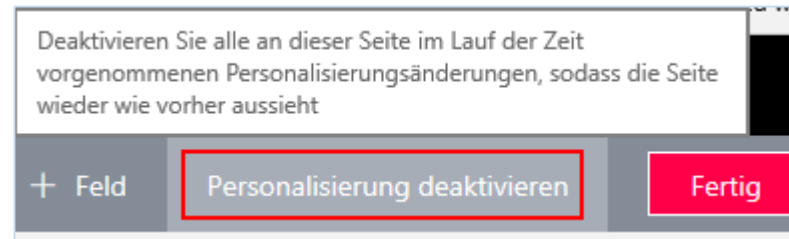>> Add, Delete, Move Fields

> Finish Up
>> Download Extension package (zip)

In-Client Designer
Extension / Customization Management

> Remove an Extension
>> Page "Extension Management"
>> Uninstall → "Deactivate"
>> Unpublish → Remove from System

> Remove a Customization
>> Open Personalization → Deactivate the Personalization

It's Lab Time

Hands On

> In-Client Designer

> In-Client Designer & VS Code

In-Client Designer
Hands On

› Open the Web Client

› Personalize the „Item Card"
  › Hide some Fields
  › Rearrange some Fields
  › Save the Customization

› Create an Extension for the Item Card
  › Unlock Designer when necessary
  › Rename the Item Group
  › Rearrange some Fields / Factboxes
  › Add Fields if you want
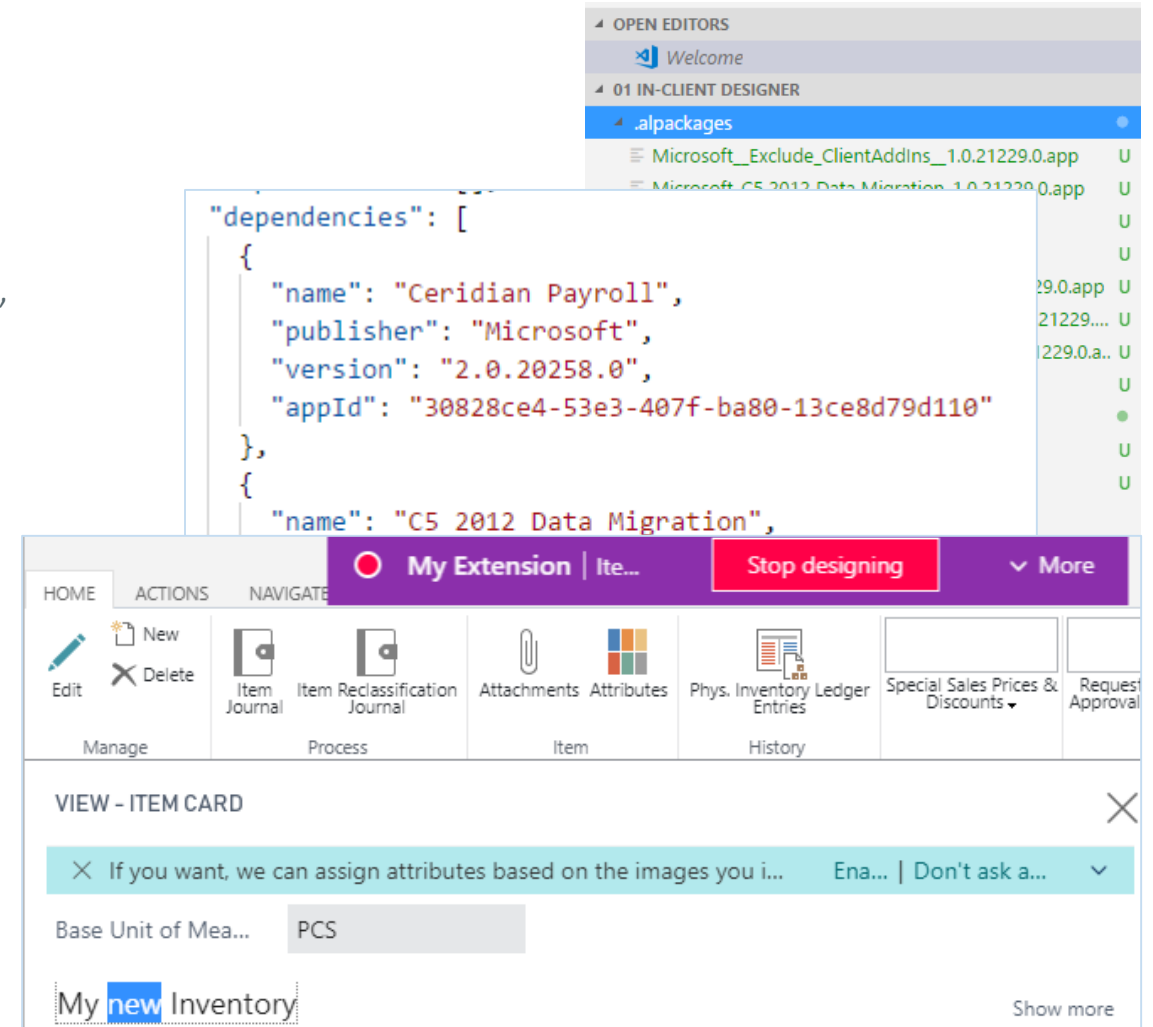 › Finish the Extension and Download the Package

In-Client Designer
Hands On

> **Unzip your Extension**
>> Clean up the dependencies in app.json

> **Open in VS Code**
>> Download Symbols   [Ctrl] + [Shift] + P … "Downlaod Symbols"
>> Compile   [Ctrl] + B
>> Publish   [Ctrl] + F5

> **Run the Designer from VS Code**
>> Check *app.json*   "showMyCode": true
>> Start the Designer   F6
>> Edit the Page   "Item Card"
>> Rename the Group   "InventoryGrp"
>> Save the changes
>> Download Source Code   F7

Q & A

Visual Studio Code

https://code.visualstudio.com

Competence Team DevTech

Microsoft AL