# Vorlage

SmartMaster-Vorlage

Eingaben

Ergebnisse

## Arbeitsbereich

## Variables

%Version1_C	2		: version counter
%Width #N/	AME?		; Monitor width in pixels
%Custom_Sia	#NAME?		; Custom size based on lookı
%CurrentSize	87		; current size
%Scrn_Size_	Width 640 800 1024 1152 1280	CustomSize 85 95 125 140 155	: Screen size lookup table
#NAME?			
% <b>Pop_Looku</b> ?1	م	x:E6	
?2	م	x:G6	

%Lookup\_CeA:E6 %Current\_Po? 1 Arbeitsbereich

up into table

Bold items a	re referenced to duplica	<b>ate s</b> :Translators, p	lease ignore this	s table
	Translate this column	Xlat Name	Range Name	Range
	Amortization Sheet St	ring/sCURRENTSI2	CURRENTSIZ	B:B6B:B6
<u>RangeNames</u>	<u>Strings</u>	%CURRENT_P	CURRENT_PC	B:B22B:B22
A_1	SmartMaster-Vorlage	%CUSTOM_SIZ	CUSTOM_SIZ	B:B5B:B5
A_2		%LOOKUP_CE		B:B21B:B21
A_3	Ergebnisse	%POP_LOOKU	JPOP_LOOKU	B:B17B:C18
A_4	Eingaben	%SCRN_SIZE_T	CRN_SIZE_TA	B:B9B:C13
		%VERSION1_CO	ERSION1_COU	B:B3B:B3
		%WIDTH	%WIDTH	B:B4B:B4

### Button Strings

<u>RangeNames</u> <u>Strings</u> %B\_1 Neue Version

# Sheet names must be in order

# Sheet Strings

- RangeNames Strings
- %S\_1 Vorlage %S 2 Arbeitsbereich
- %S\_2 Arbeitsbereid
- %S\_3 Texte
- %S\_4 Scripts

## **Function Strings**

### RangeNames Strings

%F\_1 Bildschirm-Breite Do Not translate "Range Strings" starting with "%" *Range Strings* 

## RangeNames Strings

- %R\_1 %Current\_PorDo Not Translate
- %R\_2 Vorlage
- %R\_3 %Width Do Not Translate
- %R\_4 %Custom\_Siz Do Not Translate
- %R\_5 %CurrentSize Do Not Translate
- %R\_6 %Scrn\_Size\_1Do Not Translate
- %R\_7 %Version1\_C(Do Not Translate
- %R\_8 %Lookup\_CellDo Not Translate
- %R\_10 %Amort\_rang(Do Not Translate
- %R\_11 Leer
- %R\_12 Vorlage 1
- %R\_13 Original
- %R\_15 %Pop\_LookurDo Not Translate

## FileInfo Strings

## RangeNames Strings

- %D\_1 SmartMaster Do Not Translate
- %D\_2 Eigenen SmartMaster erstellen
- %D\_3 Verwenden Sie diese Vorlage, um eigene SmartMaster-Schablonen zu erstellen.

# Pop-up Strings

# RangeNames Strings

%P\_1Doppelklicken Sie in diesen Textblock, und geben Sie Text ein, um einen Hilfetext zu e%P\_2Wenn der Hilfetext fertig ist, "ffnen Sie den Script-Editor, und w,hlen Sie das Objekt "?

# Ranges Counter Current Name Xlat Name Current Range	8 9 #NAME? #NAME? #NAME?	
١R	{ Create ranges based on new range names} {FOR K3;1;K2;1;K11}	
Loop	{ Create range name for current cell} \U {CALC} /rnd@@(K4){calc}~ /rnc@@(K5){calc}~@@(K6){calc}~	١W

### RUN this macro before the one above

# Sheets Counter Current Sheet Cur SheetName Current Range	4 5 #NAME? Scripts %s_5
١S	{ Renames all Sheets} {FOR K20;1;K19;1;K28}
Loop	{ rename sheet} {CALC} #NAME? {IF K22=@@(K23)}{RETURN} {SHEET-NAME @@(K23);K22}
١X	{edit-copy d:b2d:b200}

<b>Pseudo-tranlator</b>	testing tool; pads all strings by %30 and adds INTL chars
١Y	{GOTO}Strings:b5~
١Q	{IF @cellpointer("row")>100}{QUIT}
	<pre>{IF @length(@cellpointer("contents"))&lt;3}{Down}{BRANCF</pre>
	{IF @cellpointer("type")="b"}{Down}{BRANCH \Q}
	#NAME?
	#NAME?
	{Down}{BRANCH \Q}
PAD	€□,ƒ"…†‡^‰Š‹Œ□Ž□□''""•—— <sup>~</sup> ™šœ¾P ¡¢£¤¥¦§¨_«¬€□,ƒ

## Sub Translate rstellen. Ziehen Sie den Text und das daz Dim X As Document 1" aus. L"schen Sie das Kommentarzeicher Set X = Bind( [].path + [].ShortName ) [Button 1].Text = [%B\_1].CellValue

[Text 1].Text = [%P\_1].CellValue [Text 2].Text = [%P\_2].CellValue

[%A\_R1.Vorlage 1].Name = [%R\_12].CellValue

[%A\_R1.Leer].Name = [%R\_11].CellValue X.Subject = [%D\_1].CellValue X.Title = [%D\_2].CellValue X.Description = [%D\_3].CellValue X.NoDocInfo = FALSE End Sub

	Temp	script an	d macro	for	conversion	from	.WK4	to .	123
--	------	-----------	---------	-----	------------	------	------	------	-----

١Z	{ Sets up For Gemini} {Create123}
Button script	Sub Create123 .NewButton 33200,30000,49800,34000 [Button 1].Text = [%B_1].CellValue
PopUp Script	[Picture 4].Name = "? 1" [Picture 5].Name = "? 2" .NewEditControl 11000,3500,85000,10500 [Text 1].Text = [%P_1].CellValue [Text 1].Background.BackColor.ColorIndex = 2 [Text 1].EdgeLineWidth = \$LineWidth2 [Text 1].TextHorizontalAlign = \$AlignCenter [Text 1].DesignerFrameStyle = \$DesignerFrame2 .NewEditControl 55000,23200,95000,29000 [Text 2].Text = [%P_2].CellValue [Text 2].Background.BackColor.ColorIndex = 2 [Text 2].EdgeLineWidth = \$LineWidth2 [Text 2].TextHorizontalAlign = \$AlignCenter [Text 2].TextHorizontalAlign = \$AlignCenter [Text 2].DesignerFrameStyle = \$DesignerFrame2

CreateVersion Script	[%A_R1].NewVersion "Leer" [%A_R1].Clear [%A_R1].VersionBorderVisible = True <b>[%A_R1.Original].Name = "Vorlage 1"</b> <b>[%A_R1.Vorlage 1].MakeCurrent</b>
FileInformation Script	[].Subject = [%D_1].CellValue [].Title = [%D_2].CellValue [].Description = [%D_3].CellValue [].NoDocInfo = FALSE
Outline Script	[Vorlage].RowFolding = \$ParentBefore [Vorlage].RowOutlineVisible = \$Yes [A:A6A:A10].DemoteRow 1 [A:A14A:A27].DemoteRow 1 [].PrintWhat = \$None End Sub

\*\*\*\*\*

Translate all Strings in Strings Sheet. Run macro \W to call Global Sub Translate, \S, \R and \X Delete all Global Subs Paste into IDE under Globals. Cleanup file as you like Rename from OS as \*.12M Test

Tool to unhide sheets /whdb:a1..d:a1~

{Translate} {\S} {\R} {\X}

+\Q}

'"…†‡^‰Š‹Œ□Ž□□''""•—<sup>~</sup>™šœ¾P j¢£¤¥¦§"\_«¬€□,*f* "…†‡^‰Š‹Œ□Ž□□''""•—<sup>~</sup>™šœ¾P j¢£¤¥¦§"\_«

\*\*\*\*\*\*\*

copy sub create123 to clip and paste to IDE copy sub Translate to clip and paste to IDE run macro \z check all test blocks and buttons for looks **Set a title horizontal title from A:A5** Delete create123 script from IDE **Run Macro \X to copy scripts to Clip. Paste into IDE under Globals.** Set up button, image and textblock scripts as per table below **hide and lock text blocks** Cleanup file Hide scripts, strings and WorkArea sheets test

Object	Event	Script
Picture 1	Selected	SM1
Picture 2	Selected	SM2
Picture 3	Selected	SM3
? 1,2, 3	Selected	Select_H
Button 1	Click	SM4
TextControl 1	, Selected	DeSelect_H

# Page 11

<¬€□,ƒ"…†‡^‰Š<Œ□Ž□□''""•—~<sup>~</sup>™šœ¾P j¢£¤¥¦§"\_«¬€□,ƒ"…†‡^‰Š<Œ□Ž□□''""•—~<sup>~</sup>™šœ¾P j¢£¤

¥¦§¨\_«¬€□,*f*,,...†‡^‰Š‹Œ□Ž□□''""•—<sup>~</sup>™šœ¾P j¢£¤¥¦§¨\_«¬€□,*f*,,...†‡^‰Š‹Œ□Ž□□''""•—<sup>~</sup>™šœ¾F

' j¢£¤¥¦§`\_«¬€□,*f* "…†‡^‰Š‹Œ□Ž□□''""•—~<sup>~</sup>™šœ¾P j¢£¤¥¦§`\_«¬€□,*f* "…†‡^‰Š‹Œ□Ž□□''""•—~<sup>~</sup>™

šœ¾P ;¢£¤¥¦§``\_«¬€□,ƒ"…†‡^‰Š‹Œ□Ž□□''""•—~~™šœ¾P ;¢£¤¥¦§``\_«¬

Script Name Assign to Op	CODE Sub Init	DESCRIPTION
or sub Init fo	[].Maximize	; need to add code here to cł
	End Sub	
Print	Sub SM1 [E5].Select .ShowAutomaticPageBreaks=False .CurrentPrintSettings.PrintWhat = \$CurrentSheet CurrentPrintSettings FooterCenter = ""	; select a cell to remove hanc ;turn off the display of the princ ; Prints spreadsheet page
	.CurrentPrintSettings.Orientation = \$Portrait .CurrentPrintSettings.FitToPage = \$FitRowsAndColumns .CurrentPrintSettings.LeftMargin = 1080 .CurrentPrintSettings.RightMargin = 1080 .CurrentPrintSettings.BottomMargin = 1080 [].ShowDrawLayer = False	: set print options
	[].ShowVersionBorders = False CurrentApplication.Print [].ShowDrawLayer = True [].ShowVersionBorders = True End Sub	; hide bitmaps ; turn off display of VM fames ; Print ; unhide bitmaps ; turn on display of VM fames
Help	Sub SM3 [E5].Select [].HelpContents End Sub	; select a cell to remove hanc ; raise help with right topic
Create Versio	Sub SM4	
	DIM VMR AS Version [%A_R1].NewVersion [%R_2].CellValue + " " + Cstr([% [%A_R1].Clear [%Version1_Count].Contents = CSTR([%Version1_Co End Sub	; dim VMS as Version 6; Create new Version 1; Increment counter
Send Results	Sub SM2 [E5].Select [].ShowDrawLayer = False	; select a cell to remove hanc
	[].SolowversionBorders – False [].SelectAll [].CopyToClipboard \$BitMapFormat	; copy resuls to clipboard
	[].SendMail , , "SmartMaster-Vorlage", , , , \$Clipboard [].ShowDrawLayer = True [].ShowVersionBorders = True	; send mail

End Sub

TextBlock	Sub Select_H	
Scripts	Dim X As String	; to hold popup number
Generic Sele	ecDim A As EditText	;edit object
	Dim CS as Range	; for range object for selecting
	X = Right\$( Cstr( .Name), 1)	; the number of the object
	[%Current_Pop].Contents = "? " + X	; enter popup name into cell
	[%Lookup_Cell].RecalcRange \$Columns	; recalc cell
	Set CS = Bind( [%Lookup_Cell].CellValue)	; bind cell ref to range object
	CS.Select	; finally select the object
	Set A = Bind ("Text " + X)	; set textcontrol object name
	A.IsLocked = False	; unlock it either way
	If A.Visible = True Then	; if visible
	A.Visible = False	; make invisible
	Else	; else
	A.Visible = True	; make it visible
	End IF	

A.IsLocked = True End Sub

If they click the TextBlock then deselect it and hide it.

Generic DeSe Sub DeSelect\_H Dim X As String Dim A As EditText Dim CS as Range X = Right\$( Cstr( .Name), 1) Set A = Bind (.Name)

> [%Current\_Pop].Contents = "? " + X [%Lookup\_Cell].RecalcRange \$Columns Set CS = Bind( [%Lookup\_Cell].CellValue) CS.Select

A.IsLocked = False A.Visible = False A.IsLocked = True End Sub

neck for If Canada

lles from image nt range

3

3

lles from image

lles from image

Help Icons should be named ? 1; ? 2 etc. Their corresponding lines, ellipses and text blocks will also g cell