

Title: German Tax Deduction Calculation using BRF+

1. Objective

The objective of this requirement is to design and implement a tax calculation engine in BRF+ that automates the derivation of German tax deductions. The system should calculate VAT, Corporate Tax, Solidarity Surcharge, and Trade Tax based on financial input values.

2. Scope

Automate multi-step tax calculations using BRF+ formulas and rulesets.

Provide flexibility to adjust tax rates and formulas without modifying ABAP code.

Ensure transparency in tax calculation logic by maintaining all rules in BRF+.

Support simulation and testing of tax scenarios for validation.

3. Inputs

The function will accept the following input parameters (Data Object: `ZFI_FUNCTION_INPUT`):

- Total Revenue (`ZFI_TOTAL_REVENUE`)
- Total Expenses (`ZFI_TOTAL_EXPENSE`)

4. Processing Logic (Business Rules)

The system will apply the following rules sequentially through the ruleset (`ZFI_DE_DERIVE_TAXES_RULESET`):

- Calculate VAT which is 19% of Total Revenue
- Derive Taxable Profit which is Total Revenue – VAT – Total Expenses
- Calculate Corporate Tax which is 15% of Taxable Profit
- Calculate Solidarity Surcharge which is 5.5% of Corporate Tax
- Calculate Trade Tax which is 14% of Taxable Profit

5. Outputs: The function will return results in the output data object:

- VAT Amount (`ZFI_VAT_AMOUNT`)
- Taxable Profit (`ZFI_TAXABLE_PROFIT`)
- Corporate Tax (`ZFI_CORPORATE_TAX`)
- Solidarity Surcharge (`ZFI_SOLIDARITY_SURCHARGE`)
- Trade Tax (`ZFI_TRADE_TAX`)

RepositoryCatalog

Show: My Applications Search

Create Application

My ApplicationsSta...

ZFI_DE_TAX_DEDUCTIONS
Data Object
Element
Structure
ZFI_FUNCTION_INPUT
ZFI_FUNCTION_RESULT
Expression
Formula
ZFI_VAT_FORMULA
ZFI_TAXABLE_PROFIT_FC
ZFI_CORPORATE_TAX_FC
ZFI_SOLIDARITY_FORMUI
ZFI_TRADE_TAX_FORMUI
Function
ZFI_DERIVE_VAT
ZFI_DERIVE_TAXABLE_PROI
ZFI_DE_TAX_DEDUCTIONS..
ZFI_DERIVE_CORPORATE_T
ZFI_DERIVE_SOLIDARITY
ZFI_DERIVE_TRADE_TAX
Ruleset
ZFI_DE_DERIVE_TAXES_RUI

Function: ZFI_DE_TAX_DEDUCTIONS, DE Tax Deductions

BackEditCheckSaveActivateDeleteMore

General
Detail
Simulation

Mode: Event Mode

SignatureAssigned Rulesets

Context

| Component Name | Text |
|---|-------------------|
| <input type="checkbox"/> ZFI_FUNCTION_INPUT | Revenue & Expense |
| <input type="checkbox"/> ZFI_TOTAL_REVENUE | Total Revenue |
| <input type="checkbox"/> ZFI_TOTAL_EXPENSE | Total Expenses |

Result Data Object

Data Object: ZFI_FUNCTION_RESULT

RepositoryCatalog

Show: My Applications Search

Create Application

My ApplicationsSta...

ZFI_DE_TAX_DEDUCTIONS
Data Object
Element
Structure
ZFI_FUNCTION_INPUT
ZFI_FUNCTION_RESULT
Expression
Formula
ZFI_VAT_FORMULA
ZFI_TAXABLE_PROFIT_FC
ZFI_CORPORATE_TAX_FC
ZFI_SOLIDARITY_FORMUI
ZFI_TRADE_TAX_FORMUI
Function
ZFI_DERIVE_VAT
ZFI_DERIVE_TAXABLE_PROI
ZFI_DE_TAX_DEDUCTIONS..
ZFI_DERIVE_CORPORATE_T
ZFI_DERIVE_SOLIDARITY
ZFI_DERIVE_TRADE_TAX
Ruleset
ZFI_DE_DERIVE_TAXES_RUI

Function: ZFI_DE_TAX_DEDUCTIONS, DE Tax Deductions

BackEditCheckSaveActivateDeleteMore

General
Detail
Simulation

Mode: Event Mode

SignatureAssigned Rulesets

Ruleset

Create Ruleset

| Name | Text | Priority |
|--|---------------------------|-------------|
| <input type="checkbox"/> ZFI_DE_DERIVE_TAXES_RULESET | Derive DE Taxes (Ruleset) | [undefined] |
| | | |
| | | |
| | | |

Functions are executed in sequence to derive tax components

Repository Catalog

Show: My Applications Search

Create Application

My Applications Sta...

- ZFI_DE_TAX_DEDUCTIONS
 - Data Object
 - Element
 - Structure
 - ZFI_FUNCTION_INPUT
 - ZFI_FUNCTION_RESULT
 - Expression
 - Formula
 - ZFI_VAT_FORMULA
 - ZFI_TAXABLE_PROFIT_FC
 - ZFI_CORPORATE_TAX_FC
 - ZFI_SOLIDARITY_FORMUI
 - ZFI_TRADE_TAX_FORMUI
 - Function
 - ZFI_DERIVE_VAT
 - ZFI_DERIVE_TAXABLE_PROI
 - ZFI_DE_TAX_DEDUCTIONS..
 - ZFI_DERIVE_CORPORATE_T
 - ZFI_DERIVE_SOLIDARITY
 - ZFI_DERIVE_TRADE_TAX
 - Ruleset
 - ZFI_DE_DERIVE_TAXES_RUI

Ruleset: ZFI_DE_DERIVE_TAXES_RULESET, Derive DE Taxes (Ruleset)

Back Display Check Save Activate Delete More

General

Detail

Show Ruleset Header Context Overview

Rules

Insert Rule Insert Exit Condition

Disable Rule (1) Rule: Calculate VAT (19% of Total Revenue) - Unlimited Validity

(1) Change ZFI_FUNC...ZFI_VAT_AMOUNT after processing ZFI_VAT_FORMULA

Disable Rule (2) Rule: Derive Total Taxable Profit (Total Revenue - VAT) - Rule has not been defined

(1) Change ZFI_FUNC...ZFI_TAXABLE_PROFIT after processing ZFI_TAXABLE_PROFIT_FORMULA

Disable Rule (3) Rule: Calculate Corporate Tax (15% of Taxable Profit) - Rule has not been defined

(1) Change ZFI_FUNC...ZFI_CORPORATE_TAX after processing ZFI_CORPORATE_TAX_FORMULA

Disable Rule (4) Rule: Calculate Solidarity Surcharge (5.5% of Corporate Tax) - Rule has not been defined

(1) Change ZFI_FUNC...ZFI_SOLIDARITY_SURCHARGE after processing ZFI_SOLIDARITY_FORMULA

Disable Rule (5) Rule: Calculate Trade Tax (14% of Taxable Profit) - Rule has not been defined

(1) Change ZFI_FUNC...ZFI_TRADE_TAX after processing ZFI_TRADE_TAX_FORMULA

How BRF+ are consumed in ABAP programs:

Repository Browser

K Package

ZFI_BRF_APP

Object Name

- ZFI_BRF_APP
- Dictionary Objects
- Structures
 - ZFI_FUNCTION_INPUT
 - ZFI_FUNCTION_RESULT
- Data Elements
 - ZFI_DE_AMOUNT
- Class Library
- Classes
 - ZCL_PROCESS_BRF_APP
- Programs
 - ZFI_DE_TAX_CALCULATION

Report: ZFI_DE_TAX_CALCULATION Active

```

1  REPORT zfi_de_tax_calculation.
2
3
4  * SELECTION-SCREEN: BEGIN OF BLOCK b1 WITH FRAME TITLE TEXT-001.
5  * PARAMETERS : p_revenue TYPE zfi_de_amount,
6  *               p_expense TYPE zfi_de_amount.
7  * SELECTION-SCREEN: END OF BLOCK b1.
8
9  START-OF-SELECTION.
10 * Process the Function and Capture the Result
11 DATA(iv_result) = zcl_process_brf_app=>get_instance( EXPORTING iv_func_id = 'DD090909B95BA1FE0A2A90543E4A18A7B' )->process( EXPORTING iv_total_revenue = p_revenue
12                                                             iv_total_expense = p_expense ).
13
14 END-OF-SELECTION.
15
16 WRITE: /10 'German Tax Calculation'.
17 SKIP.
18 WRITE: /10 'Total Revenue:',    35 p_revenue,          55 'Total Revenue'.
19 WRITE: /10 'Total Expenses:',   35 p_expense,          55 'Total Expenses'.
20 SKIP.
21 WRITE: /10 'Total VAT:',        35 iv_result-zfi_vat_amount,  55 '19% of Total Revenue'.
22 WRITE: /10 'Taxable Profit:',   35 iv_result-zfi_taxable_profit,  55 'Total Revenue - VAT Amount'.
23 WRITE: /10 'Corporate Tax:',    35 iv_result-zfi_corporate_tax,  55 '15% of Taxable Profit'.
24 WRITE: /10 'Solidarity Surcharge:', 35 iv_result-zfi_solidarity_surcharge, 55 '5.5% of Corporate Tax'.
25 WRITE: /10 'Trade Tax:',        35 iv_result-zfi_trade_tax,      55 '14% of Taxable Profit'.

```

```

21  " Function Context Variables are assigned with input values
22  <fs_total_revenue> = iv_total_revenue.
23  <fs_total_expense> = iv_total_expense.
24  ls_name_value-value = lr_zfi_function_input.
25
26  cl_fdt_function_process=>move_data_to_data_object( EXPORTING ir_data          = lr_zfi_function_input
27                                                         iv_function_id = mv_func_id
28                                                         iv_timestamp  = lv_timestamp
29                                                         iv_data_object  = 'DD090909B95BA1FE0A1FF809F07A1CA7B'
30                                                         IMPORTING er_data    = ls_name_value-value ).
31
32  INSERT ls_name_value INTO TABLE lt_name_value.
33  CLEAR ls_name_value.
34

```

```

35 |-----*
36 |** Process the Function and Capture the Result
37 |-----*
38 | cl_fdt_function_process=>get_data_object_reference( EXPORTING iv_function_id   = mv_func_id
39 |                                                    iv_data_object     = '_V_RESULT'
40 |                                                    iv_timestamp      = lv_timestamp
41 |                                                    iv_trace_generation = abap_false
42 |                                                    IMPORTING er_data       = lr_zfi_function_result ).
43 | ASSIGN lr_zfi_function_result->* TO FIELD-SYMBOL(<fs_zfi_function_result>).
44 |
45 | TRY.
46 |   cl_fdt_function_process=>process( EXPORTING iv_function_id = mv_func_id      " Unique ID of the BRF+ function
47 |                                     iv_timestamp   = lv_timestamp
48 |                                     IMPORTING ea_result = <fs_zfi_function_result>
49 |                                     CHANGING ct_name_value = lt_name_value ). " Result Table
50 | CATCH cx_fdt.
51 | ENTRY.

```

DE Tax Calculations

German Tax Calculation

| | | |
|-----------------------|-----------|----------------------------|
| Total Revenue: | 10.000,00 | Total Revenue |
| Total Expense: | 5.000,00 | Total Expenses |
| Total VAT: | 1.900,00 | 19% of Total Revenue |
| Taxable Profit: | 3.100,00 | Total Revenue - VAT Amount |
| Corporate Tax: | 465,00 | 15% of Taxable Profit |
| Solidarity Surcharge: | 25,58 | 5.5% of Corporate Tax |
| Trade Tax: | 434,00 | 14% of Taxable Profit |