Stwórz tabele używając słownika ABAP ( SE11 ) zgodnie z konwencją nazewniczą która będzie zawierała pola:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Name | Surname | Salary | Position |
| Zdefiniuj swój Data element niech będzie miał typ przykładowo NUMC5 | Zdefiniuj swój Data element | Zdefiniuj swój Data element | Zdefiniuj swój typ i domene. | Zdefiniuj swój Data element |

Po wygenerowaniu, stwórz dla tej tabeli maintenance view i wypełnij go losowymi danymi, na których będziesz w stanie operować NP.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Name | Surname | Salary | Position |
| 1000 | Adam | Eve | 5000 | Gardener |

Tworzenie raportu + selection screen:

utwórz raport w transakcji SE38

Stwórz selection screen który będzie zawierał

Pole ID jako SELECT-OPTION

Oraz pole pozycja jako PARAMETERS

Pole typu checkbox z podpisem „Pokaż dodatkowe parametry”

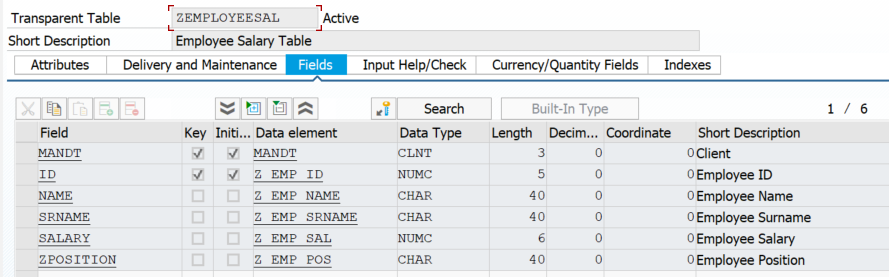
Tworzenie logiki raportu:

Napisz logikę która wykorzysta parametry podane na selection screenie ( miej na uwadze opcje checkboxa, który gdy zaznaczony wyświetli więcej parametrów ) pobierze dane ze stworzenej przez ciebie tabeli do tabeli wewnętrznej, a następnie wyświetli je w formie ALV na ekran użytkownika po wykonaniu programu.

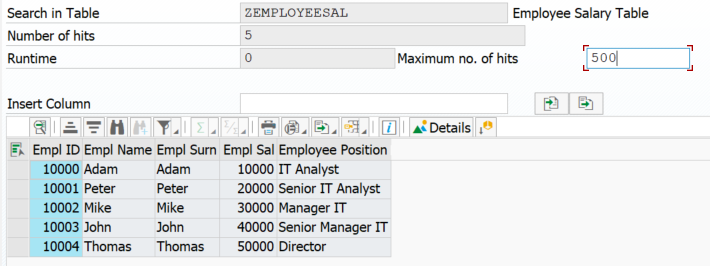
Dodatkowe:

Napisz dodatkową logike dodając kolejny checkbox ( Increase salary ), który gdy jest zaznaczony zwiększy pole ‘Salary’ każdego rekordu o 0.23%

Tabela:



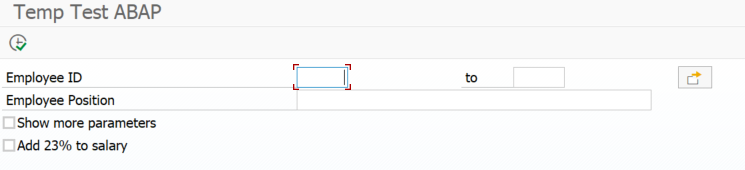
Wartości:



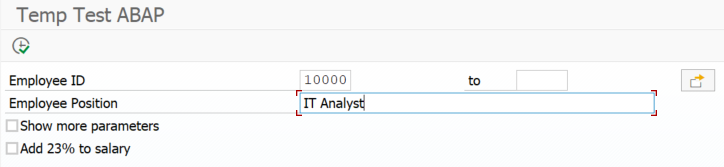
Kod:

REPORT ztst\_001.  
  
TABLES zemployeesal.  
  
SELECT-OPTIONS so\_id FOR zemployeesal-id.  
  
PARAMETERS: p\_zpos TYPE zemployeesal-zposition,  
            p\_name TYPE zemployeesal-name MODIF ID nam,  
            p\_srname TYPE zemployeesal-srname MODIF ID srn,  
            p\_salary TYPE zemployeesal-salary MODIF ID sal,  
            p\_hide AS CHECKBOX USER-COMMAND hid,  
            p\_vat AS CHECKBOX USER-COMMAND vat.  
  
DATA: go\_alv      TYPE REF TO cl\_salv\_table,  
      vat         TYPE P DECIMALS 2 value '1.23'.  
  
  
AT SELECTION-SCREEN OUTPUT.  
  
  LOOP AT SCREEN.  
    IF p\_hide = abap\_false.  
  
      IF screen-group1 = 'NAM' or screen-group1 = 'SRN' or screen-group1 = 'SAL'.  
        screen-active = 0.  
        MODIFY SCREEN.  
      ELSE.  
        screen-active = 1.  
        MODIFY SCREEN.  
      ENDIF.  
    ENDIF.  
  
  ENDLOOP.  
  
  
START-OF-SELECTION.  
  
IF p\_hide IS INITIAL.  
  
  SELECT zemployeesal~id, zemployeesal~zposition FROM zemployeesal INTO TABLE @DATA(lt\_employee) WHERE zposition = @p\_zpos AND id IN @so\_id.  
  
      TRY.  
      cl\_salv\_table=>factory(  
        IMPORTING  
          r\_salv\_table   = go\_alv  
        CHANGING  
          t\_table        = lt\_employee  
      ).  
    CATCH cx\_salv\_msg.  
  ENDTRY.  
  
  go\_alv->display( ).  
  
  ELSE.  
  
  SELECT \* FROM zemployeesal INTO TABLE @DATA(lt\_employee1) WHERE name = @p\_name AND srname = @p\_srname AND salary = @p\_salary AND zposition = @p\_zpos AND id IN @so\_id.  
  
    IF p\_vat IS NOT INITIAL.  
  
            LOOP AT lt\_employee1 ASSIGNING FIELD-SYMBOL(<fs\_data>).  
              <fs\_data>-salary = <fs\_data>-salary \* vat.  
              ENDLOOP.  
  
      ENDIF.  
  
     TRY.  
      cl\_salv\_table=>factory(  
        IMPORTING  
          r\_salv\_table   = go\_alv  
        CHANGING  
          t\_table        = lt\_employee1  
      ).  
    CATCH cx\_salv\_msg.  
  ENDTRY.  
  
  go\_alv->display( ).  
  
  ENDIF.

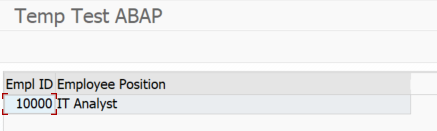
Po uruchomieniu:



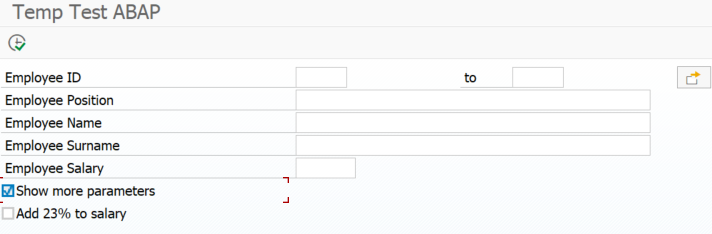
Po wypełnieniu dwóch pierwszych pól:



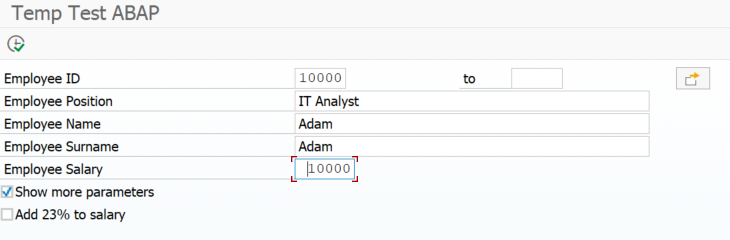
Wynik:



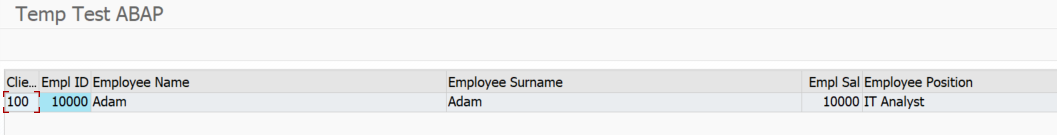
Po kliknieciu Show more parameters pojawiają się pola:



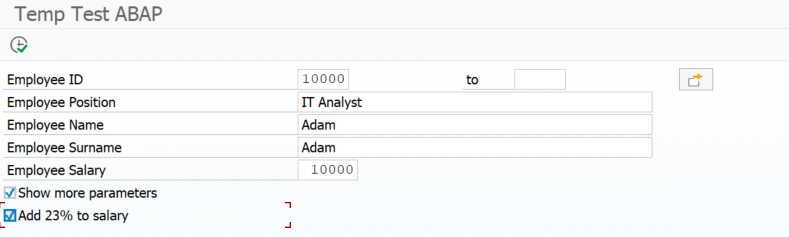
Po wypełnieniu:



Wynik:



Z zatickanym Salary:



Wynik:

