```
report  zzztest00.

parameters pa_name type string default sy-uname.
parameters p_ertek type string.

AT SELECTION-SCREEN.
if p_ertek is initial.
*  MESSAGE 'Kérem adja meg az értéket' type 'E'.
*  write: 'dflsjfks'.
    MESSAGE 'Nem adott meg értéket' type 'W'.
endif.

START-OF-SELECTION.

*MESSAGE 'Report kezdete' type 'I'.
*MESSAGE 'Nane' type 'X'.

write: 'Hello'(001), 'world'. "comment
skip.
write at 40 'Új sor'.

* sor elején * az is comment
" ez már nem comment

if not pa_name is initial.
  write: / 'Hello', pa_name.
endif.

data lv_karakter type c.
data gv_karakter type c.

data lv_tobb(10).
lv_tobb = 'Karakter'.

data lv_string type string.
lv_tobb = 'karakter'.
lv_string = 'karakter'.


data lv_szamlalo type i.
lv_szamlalo = 1.

data lv_datum type d.
lv_datum = '20161028'.

data lv_decimal type dec16_3.
data lv_ptipus type p.

data lv_dec163 type p length 16 decimals 3.

types t_dec163 type p length 16 decimals 3.
```

```
data lv_dec type t_dec163.

data lv_numc5 type numc5.
lv_numc5 = 1.
write: / lv_numc5.
data lv_num5 type n length 5.

lv_datum = '20160228'.
write: / lv_datum.
lv_datum = lv_datum + 1.
write: / lv_datum.
lv_datum = lv_datum + 1.
write: / lv_datum.

data lv_datum2 like lv_datum.

data lv_ddicdec163 type z00dec16_3.

constants lv_const_c value 'konstans'.
constants lv_const_i type i value 10.

data lv_ertek type i.
lv_ertek = 1.
lv_ertek = lv_ertek + 1.
add 1 to lv_ertek.
subtract 1 from lv_ertek.

data lv_ertek2 type i.
move lv_ertek to lv_ertek2.
lv_ertek2 = lv_ertek.

lv_ertek2 = lv_ertek * 2 + 1 + ( 10 * 5 ).

data lv_len type i.
lv_len = strlen( lv_tobb ).
write: / 'Karakter esetén', lv_len.
lv_len = strlen( lv_string ).
write: / 'String esetén', lv_len.

data lv_ures.
data lv_num type i.
lv_num = 0.

if ( lv_ertek < lv_ertek2 )
  or lv_len = 1.

elseif lv_len = 8 and sy-datum = '20161028'.

elseif lv_num = 0
  or lv_num is initial.
```

```
elseif not lv_num is initial.

endif.

case lv_len.
  when 1.
* ha lv_len = 1
  when 2.

  when 3 or 4.

  when others.

endcase.

do.
  write: / 'Külső', sy-index.
  do.
    write: / 'Belső', sy-index.
    if lv_len = 0.
      exit.
    endif.
    lv_len = lv_len - 1.
  enddo.
  write: / 'Külső', sy-index.
  exit.
enddo.

do 10 times.
  if sy-index = 5.
    continue.
  endif.
  write: / sy-index.
enddo.

lv_len = 10.

while lv_len > 0.
  lv_len = lv_len - 1.
  write: / 'Index', sy-index, 'lv_len=', lv_len.
endwhile.
*
*data ls_t001 type t001.
*select * from  t001
*    into ls_t001
**    where bukrs = '1000'
*    .
*  write: / ls_t001-BUTXT.
*endselect.
*
data lt_t001 type table of t001.
```

```
*select * from  t001
*    into table lt_t001.
*loop at lt_t001 into ls_t001.
*  write: / ls_t001-BUTXT.
*endloop.

clear lt_t001.

data lv_subrc type sysubrc.

FIELD-SYMBOLS <fs_t001> type t001.
select * from  t001
    into table lt_t001
*    where bukrs = 'zzzz'
  .
lv_subrc = sy-subrc.

*if not sy-subrc = 0.
if lv_subrc ne 0.
  write: / 'Nincs zzzz vállalat'.
endif.
* not =     NE

loop at lt_t001
      ASSIGNING <fs_t001>
      where bukrs = '1000'.

  write: / 'Tabix', sy-tabix, 'Index', sy-index.
  write: / <fs_t001>-BUTXT.
endloop.
if sy-subrc ne 0.
  write: / 'Loop nem dolgozott fel semmit'.
endif.

write: / sy-repid.

perform szubrutin using sy-datum 1
                CHANGING lv_subrc.


MESSAGE 'Report lefutott' type 'S'.
*&---------------------------------------------------------------------*
*&      Form  SZUBRUTIN
*&---------------------------------------------------------------------*
*       text
*----------------------------------------------------------------------*
*      -->P_SY_DATUM  text
*----------------------------------------------------------------------*
form SZUBRUTIN  using    p_sy_datum type d
                         p_sorszam type i
                CHANGING p_subrc type sysubrc.
```

```
  write: / 'Ez egy szubrutin'.
  write:/ p_sy_datum.
  p_subrc = 0.
endform.                    " SZUBRUTIN
```