Posted by pastor on Thu, 28 Dec 2023 13:47:10 GMT
View Forum Message <> Reply to Message

```
set term ^ ;

create table TEST(
ID bigint,
ID_SP bigint,
primary key (ID)
)
^

create procedure TEST_SP(
ID_IN bigint
) returns (
ID_SP bigint,
ID_OUT bigint
) as
begin
 insert into TEST(ID, ID_SP) values (:ID_IN, :ID_IN);
 ID_SP = :ID_IN;
 ID_OUT = :ID_OUT;
 suspend;
end
^

select sp.ID_SP, t.ID
from TEST_SP( 1) sp
left join TEST t on t.ID_SP = sp.ID_SP
^
```

Posted by pastor on Thu, 28 Dec 2023 14:18:27 GMT
View Forum Message <> Reply to Message

```
with S ( ID_SP, ID_OUT)
as (
select sp.ID_SP, sp.ID_OUT
from TEST_SP( :ID_IN ) sp
group by 1, 2
order by 2, 1
)
select s.*, t.*
from  S
left join  TEST t on t.ID_SP = s.ID_SP and t.id = s.ID_OUT
```

Posted by hvlad on Thu, 28 Dec 2023 14:52:09 GMT
View Forum Message <> Reply to Message

```
for select ... from sp_test()
  into ...
  do begin
    select ... from test where ... into...
    suspend;
  end
```

Posted by sim_84 on Thu, 28 Dec 2023 14:55:59 GMT
View Forum Message <> Reply to Message

```
execute block
as
  procedure inserts(y int)
  returns (a int, b int)
  as
  begin
    for
      select x
      from t
      into a
    do
    begin
      if (a > y) then
      begin
        insert into t2(a)
        values (:a)
        returning b
        into b;

        suspend;
      end
    end
  end
begin
  for
    select
      t2.c
    from
      inserts(5) i
      left join t2 on t2.b = i.b
  do
  begin


  end
end
```

Posted by SD on Thu, 28 Dec 2023 22:41:56 GMT
View Forum Message <> Reply to Message

Posted by shavluk on Thu, 28 Dec 2023 23:59:53 GMT
View Forum Message <> Reply to Message

Posted by pastor on Fri, 29 Dec 2023 05:38:13 GMT
View Forum Message <> Reply to Message

Posted by pastor on Fri, 29 Dec 2023 06:18:49 GMT
View Forum Message <> Reply to Message

```
execute block
(ID_IN bigint = :ID_IN)
returns (
ID_SP bigint,
ID_OUT bigint
) as
begin
  id_out = null;
  for select sp.ID_SP
     from TEST_SP( :ID_IN ) sp
     into :ID_SP
     do for select t.ID
              from  TEST t
                   where t.ID_SP = :ID_SP
              into :ID_OUT
              do suspend;
end
```

```
set term ^ ;

create table TEST(
ID bigint,
ID_SP bigint,
primary key (ID)
)
^
create or alter procedure TEST_SP(
ID_IN bigint
) returns (
ID_SP bigint,
ID_OUT bigint
) as
declare variable i integer;
begin
 i = 0;
 while (i < 3)
 do begin
     insert into TEST(ID, ID_SP) values (:ID_IN, :ID_IN);
     ID_SP = :ID_IN;  ID_OUT = :ID_IN;
     suspend;
     ID_IN = ID_IN + 1;
     i = i + 1;
    end
end
^



with S ( ID_SP, ID_OUT)
as (
select  first 100 sp.ID_SP, sp.ID_OUT
from TEST_SP( :ID_IN ) sp
order by  1 desc, 2 desc
)
select first 100 s.*, t.*
from  S
left join  TEST t on t.ID_SP = s.ID_SP and t.id = s.ID_SP
order by s.ID_SP


with S ( ID_SP, ID_OUT)
```

```
as (
select  sp.ID_SP, sp.ID_OUT
from TEST_SP( :ID_IN ) sp
order by 'const')
select s.*, t.*
from  S
left join  TEST t on t.ID_SP = s.ID_SP and t.id = s.ID_SP
```
