2.7 Mugarria

Taldea 5 2024/3/19 Aritz Lekube, Iñigo Arrizabalo eta Julen Garcia Mata

[Prozedura 4](#_Toc164855420)

[Funtzioa 5](#_Toc164855421)

[Paketea sortu 6](#_Toc164855422)

[Paketearen body-a 7](#_Toc164855423)

[Bloke anonimoa 8](#_Toc164855424)

[Irudia 1- Prozeduraren funtzionamendua ziurtatu. 4](#_Toc164855385)

[Irudia 2- Funtzioaren funtzionamendua ziurtatu. 6](#_Toc164855386)

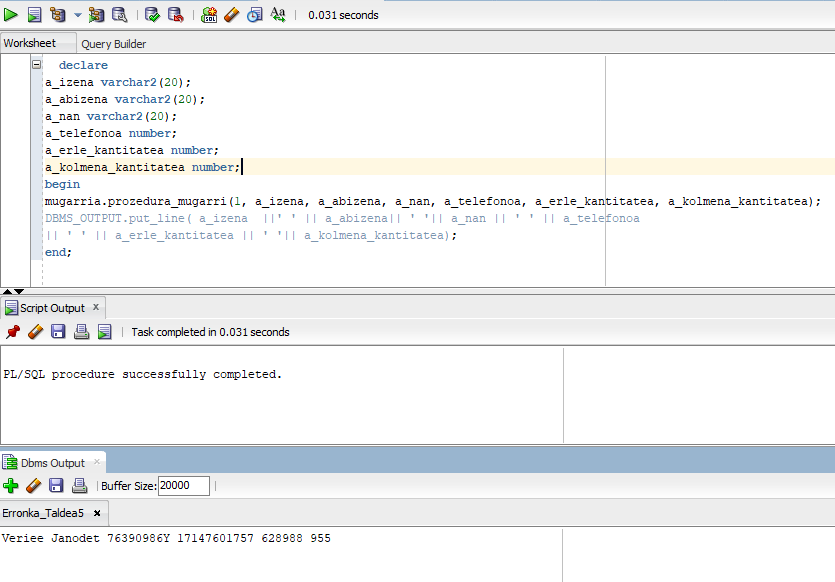
[Irudia 3- Paketea sortuta. 8](#_Toc164855387)

[Irudia 4- Bloke anonimoa exekutatuta. 10](#_Toc164855388)

# Prozedura

|  |
| --- |
| create or replace PROCEDURE Prozedura\_mugarri  (  a\_id\_sozioa IN NUMBER  , a\_izena OUT varchar2  , a\_abizena OUT varchar2  , a\_NAN OUT varchar2  , a\_telefonoa OUT number  , a\_erle\_kantitatea OUT number  , a\_kolmena\_kantitatea OUT number  )AS  begin  select sozio\_izena, sozio\_abizena, nan, telefonoa, erle\_kantitatea, kolmena\_kantitatea  into a\_izena, a\_abizena, a\_NAN, a\_telefonoa, a\_erle\_kantitatea, a\_kolmena\_kantitatea from sozioak where a\_id\_sozioa= id\_sozioa;  end Prozedura\_mugarri; |

|  |
| --- |
| declare  a\_izena varchar2(20);  a\_abizena varchar2(20);  a\_nan varchar2(20);  a\_telefonoa number;  a\_erle\_kantitatea number;  a\_kolmena\_kantitatea number;  begin  mugarria.prozedura\_mugarri(1, a\_izena, a\_abizena, a\_nan, a\_telefonoa, a\_erle\_kantitatea, a\_kolmena\_kantitatea);  DBMS\_OUTPUT.put\_line( a\_izena ||' ' || a\_abizena|| ' '|| a\_nan || ' ' || a\_telefonoa  || ' ' || a\_erle\_kantitatea || ' '|| a\_kolmena\_kantitatea);  end; |

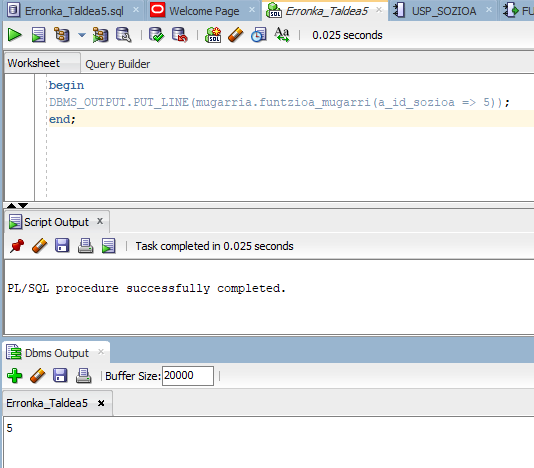


Irudia 1- Prozeduraren funtzionamendua ziurtatu.

# Funtzioa

|  |
| --- |
| create or replace NONEDITIONABLE FUNCTION Funtzioa\_mugarri  (  A\_ID\_SOZIOA IN NUMBER  ) RETURN VARCHAR2 AS  a\_parte number;  a\_kop number;  a\_mota number;  BEGIN  select kolmena\_kantitatea into a\_kop from sozioak where id\_sozioa= a\_id\_sozioa;  if a\_kop>500 then  select count(id\_asoziazioa) into a\_parte from asoziazio\_parte where a\_id\_sozioa=id\_sozioa;  RETURN a\_parte;    else  select count(id\_produktua) into a\_mota from ze\_ezti where a\_id\_sozioa=id\_sozioa;  RETURN a\_mota;  end if;  END Funtzioa\_mugarri; |

|  |
| --- |
| begin  DBMS\_OUTPUT.PUT\_LINE(mugarria.funtzioa\_mugarri(a\_id\_sozioa => 5));  end; |



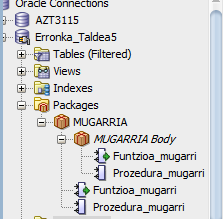
Irudia 2- Funtzioaren funtzionamendua ziurtatu.

# Paketea sortu

|  |
| --- |
| CREATE OR REPLACE  PACKAGE MUGARRIA AS  PROCEDURE Prozedura\_mugarri  (  a\_id\_sozioa IN NUMBER  , a\_izena OUT varchar2  , a\_abizena OUT varchar2  , a\_NAN OUT varchar2  , a\_telefonoa OUT number  , a\_erle\_kantitatea OUT number  , a\_kolmena\_kantitatea OUT number  );  FUNCTION Funtzioa\_mugarri  (  A\_ID\_SOZIOA IN NUMBER  ) RETURN VARCHAR2;  /\* TODO enter package declarations (types, exceptions, methods etc) here \*/  END MUGARRIA; |

# Paketearen body-a

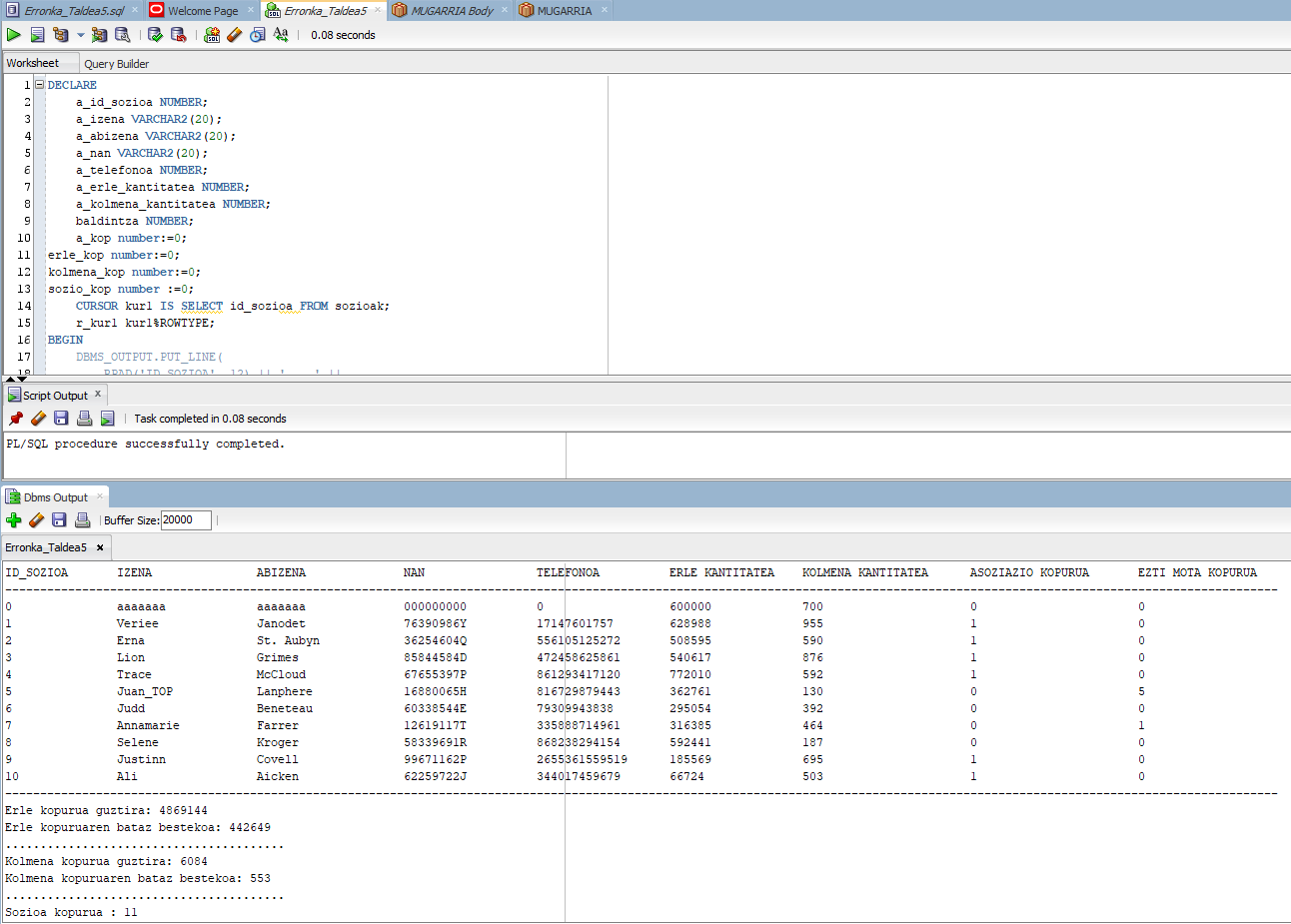
|  |
| --- |
| create or replace NONEDITIONABLE PACKAGE BODY Mugarria AS  -- Implementación del procedimiento  PROCEDURE Prozedura\_mugarri  (  a\_id\_sozioa IN NUMBER  , a\_izena OUT varchar2  , a\_abizena OUT varchar2  , a\_NAN OUT varchar2  , a\_telefonoa OUT number  , a\_erle\_kantitatea OUT number  , a\_kolmena\_kantitatea OUT number  )AS  begin  select sozio\_izena, sozio\_abizena, nan, telefonoa, erle\_kantitatea, kolmena\_kantitatea  into a\_izena, a\_abizena, a\_NAN, a\_telefonoa, a\_erle\_kantitatea, a\_kolmena\_kantitatea from sozioak where a\_id\_sozioa= id\_sozioa;  end Prozedura\_mugarri;    -- Implementación de la función  FUNCTION Funtzioa\_mugarri  (  A\_ID\_SOZIOA IN NUMBER  ) RETURN VARCHAR2 AS  a\_parte number;  a\_kop number;  a\_mota number;  BEGIN  select kolmena\_kantitatea into a\_kop from sozioak where id\_sozioa= a\_id\_sozioa;  if a\_kop>500 then  select count(id\_asoziazioa) into a\_parte from asoziazio\_parte where a\_id\_sozioa=id\_sozioa;  RETURN a\_parte;    else  select count(id\_produktua) into a\_mota from ze\_ezti where a\_id\_sozioa=id\_sozioa;  RETURN a\_mota;  end if;  END Funtzioa\_mugarri;  END Mugarria; |



Irudia 3- Paketea sortuta.

# Bloke anonimoa

|  |
| --- |
| DECLARE  a\_id\_sozioa NUMBER;  a\_izena VARCHAR2(20);  a\_abizena VARCHAR2(20);  a\_nan VARCHAR2(20);  a\_telefonoa NUMBER;  a\_erle\_kantitatea NUMBER;  a\_kolmena\_kantitatea NUMBER;  baldintza NUMBER;  a\_kop number:=0;  erle\_kop number:=0;  kolmena\_kop number:=0;  sozio\_kop number :=0;  CURSOR kur1 IS SELECT id\_sozioa FROM sozioak;  r\_kur1 kur1%ROWTYPE;  BEGIN  DBMS\_OUTPUT.PUT\_LINE(  RPAD('ID\_SOZIOA', 12) || ' ' ||  RPAD('IZENA', 16) || ' ' ||  RPAD('ABIZENA', 17) || ' ' ||  RPAD('NAN', 15) || ' ' ||  RPAD('TELEFONOA', 15) || ' ' ||  RPAD('ERLE KANTITATEA', 15) || ' ' ||  RPAD('KOLMENA KANTITATEA', 20)|| ' ' ||  RPAD('ASOZIAZIO KOPURUA', 20) || ' ' ||  RPAD('EZTI MOTA KOPURUA', 20)  );  DBMS\_OUTPUT.PUT\_LINE('--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------');  OPEN kur1;  FETCH kur1 INTO r\_kur1;  WHILE kur1%FOUND LOOP  mugarria.prozedura\_mugarri(r\_kur1.id\_sozioa, a\_izena, a\_abizena, a\_nan, a\_telefonoa, a\_erle\_kantitatea, a\_kolmena\_kantitatea);  baldintza:= mugarria.funtzioa\_mugarri(a\_id\_sozioa => r\_kur1.id\_sozioa);  select kolmena\_kantitatea into a\_kop from sozioak where id\_sozioa=r\_kur1.id\_sozioa;    if a\_kop >500 then  DBMS\_OUTPUT.PUT\_LINE(  RPAD(r\_kur1.id\_sozioa, 12) || ' ' ||  RPAD(a\_izena, 13) || ' ' ||  RPAD(a\_abizena, 18) || ' ' ||  RPAD(a\_nan, 12) || ' ' ||  RPAD(a\_telefonoa, 14) || ' ' ||  RPAD(a\_erle\_kantitatea, 12) || ' ' ||  RPAD(a\_kolmena\_kantitatea, 17) || ' '||  RPAD(baldintza, 17) || ' ' ||  RPAD('0', 12)  );  else  DBMS\_OUTPUT.PUT\_LINE(  RPAD(r\_kur1.id\_sozioa, 12) || ' ' ||  RPAD(a\_izena, 13) || ' ' ||  RPAD(a\_abizena, 18) || ' ' ||  RPAD(a\_nan, 12) || ' ' ||  RPAD(a\_telefonoa, 14) || ' ' ||  RPAD(a\_erle\_kantitatea, 12) || ' ' ||  RPAD(a\_kolmena\_kantitatea, 17) || ' '||  RPAD('0', 17) || ' ' ||  RPAD(baldintza, 12)  );  end if;  erle\_kop:=erle\_kop+a\_erle\_kantitatea;  kolmena\_kop:=kolmena\_kop+a\_kolmena\_kantitatea;  sozio\_kop:= sozio\_kop+1;  FETCH kur1 INTO r\_kur1;  END LOOP;  CLOSE kur1;  DBMS\_OUTPUT.PUT\_LINE('--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------');  DBMS\_OUTPUT.PUT\_LINE('Erle kopurua guztira: '||erle\_kop);  DBMS\_OUTPUT.PUT\_LINE('Erle kopuruaren bataz bestekoa: ' ||round((erle\_kop/sozio\_kop)));  DBMS\_OUTPUT.PUT\_LINE('........................................');  DBMS\_OUTPUT.PUT\_LINE('Kolmena kopurua guztira: '||kolmena\_kop);  DBMS\_OUTPUT.PUT\_LINE('Kolmena kopuruaren bataz bestekoa: ' ||round((kolmena\_kop/sozio\_kop)));  DBMS\_OUTPUT.PUT\_LINE('........................................');    DBMS\_OUTPUT.PUT\_LINE('Sozioa kopurua : '||sozio\_kop);  END;  / |



Irudia 4- Bloke anonimoa exekutatuta.