

Programación PL/SQL: Bloques anónimos

```

Error que empieza en la línea : 1 Archivo @ C:\Creación_taller.sql
del comando -
DROP TABLE TRABAJOS CASCADE CONSTRAINT
Informe de error -
ORA-00942: la tabla o vista no existe
00942. 00000 - "table or view does not exist"
*Cause:
*Action:

Error que empieza en la línea : 3 Archivo @ C:\Creación_taller.sql
del comando -
DROP TABLE MECANICOS CASCADE CONSTRAINT
Informe de error -
ORA-00942: la tabla o vista no existe
00942. 00000 - "table or view does not exist"
*Cause:
*Action:

Error que empieza en la línea : 5 Archivo @ C:\Creación_taller.sql
del comando -
DROP TABLE COCHES CASCADE CONSTRAINT
Informe de error -
ORA-00942: la tabla o vista no existe
00942. 00000 - "table or view does not exist"
*Cause:
*Action:

Table COCHES creado.

Table MECANICOS creado.

Table TRABAJOS creado.

Table TRABAJOS alterado.

Table TRABAJOS alterado.

Table TRABAJOS alterado.

Confirmación terminada.
  
```

```

1 fila insertadas.

1 fila insertadas.

1 fila insertadas.

1 fila insertadas.

1 fila insertadas.

1 fila insertadas.

1 fila insertadas.

1 fila insertadas.

1 fila insertadas.

1 fila insertadas.

Confirmación terminada.
  
```

```

ALTER SESSION SET "_ORACLE_SCRIPT" = TRUE;
CREATE USER TALLER12 IDENTIFIED BY oracle;
ALTER USER TALLER12 QUOTA 5M ON USERS;

GRANT CONNECT, RESOURCE TO TALLER12;

Session alterado.

User TALLER12 creado.

User TALLER12 alterado.

Grant correcto.
  
```

Nueva / Seleccionar Conexión a Base de Datos

Nombre de Cone... Detalles de Cone...
 ALUMNO1 ALUMNO1@//loc...
 Conexion1 sys@//localhost:1...
 Conexion2 Sergio@//localho...
 Conexion3 E12V2@//localho...
 Ej1_Distribucion12 DISTRIBUCION1...
 Ej2_COCHES12 COCHES12@//loc...
 Ej3_CICLISMO12 CICLISMO12@//l...
 Ej4_JARDINERIA12 JARDINERIA12@//...
 Ej3 EMPRESA12@//l...
 Ej4 EMPRESA12-2@//...
 Ej5 INSTITUTO12@//l...
 Ej6 SUCURSAL12@//l...
 Ej7 PEDIDOS12@//l...
 EMPLEADO1 EMPLEADO1@//l...
 EMPLEADO2 EMPLEADO2@//l...
 EMPLEADO3 EMPLEADO3@//l...
 Examen_BDD_5... CURSOS12@//loc...
 nba12 nba12@//localho...
 PROFESOR1 PROFESOR1@//l...

Nombre: TALLER12

Tipo de Base de Datos: Oracle

Información de usuario: Usuario de Proxy

Tipo de autenticación: Por defecto

Usuario: TALLER12 Rol: valor por defecto

Contraseña: ***** Guardar Contraseña

Tipo de Conexión: Básico

Detalles: Avanzado

Nombre del Host: localhost

Puerto: 1521

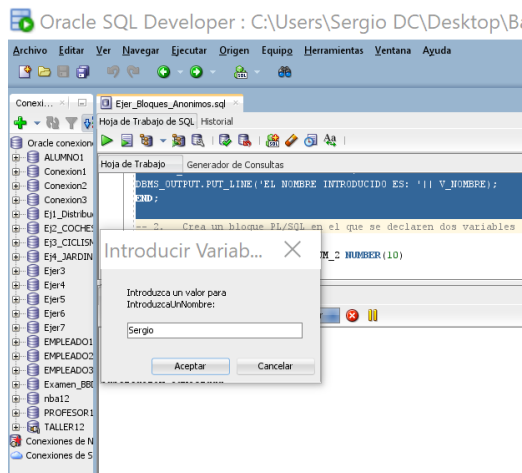
☒ SID ☐ Nombre del Servicio

Estado: Correcto

Ayuda Guardar Borrar Probar Conectar Cancelar

1. Crea un bloque PL/SQL que pida al usuario su nombre por teclado y que posteriormente lo visualice de la siguiente forma “El nombre introducido es: NOMBRE”.

```
DECLARE V_NOMBRE VARCHAR2(50);
BEGIN V_NOMBRE:='&IntroduzcaUnNombre';
DBMS_OUTPUT.PUT_LINE('EL NOMBRE INTRODUCIDO ES: '|| V_NOMBRE);
END;
```



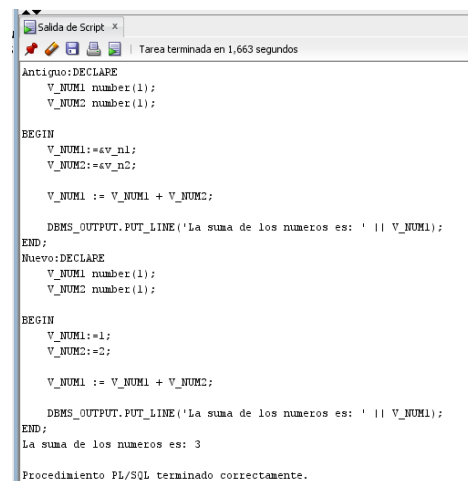
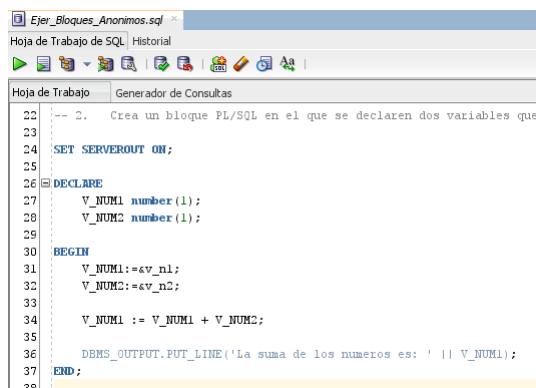
2. Crea un bloque PL/SQL en el que se declaren dos variables que almacenen dos números que se pidan por teclado y se muestre su suma por pantalla.

```
DECLARE
  V_NUM1 number(1);
  V_NUM2 number(1);

BEGIN
  V_NUM1:=&v_n1;
  V_NUM2:=&v_n2;

  V_NUM1 := V_NUM1 + V_NUM2;

  DBMS_OUTPUT.PUT_LINE('La suma de los numeros es: ' || V_NUM1);
END;
```



3. Crea un bloque PL/SQL que muestre la suma del salario de todos los mecánicos (tienes que hacer uso de %TYPE para declarar una variable).

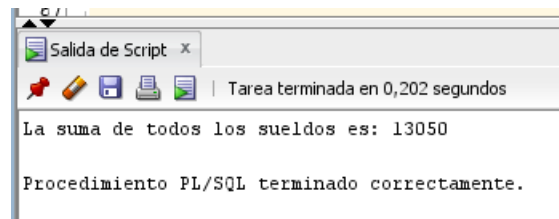
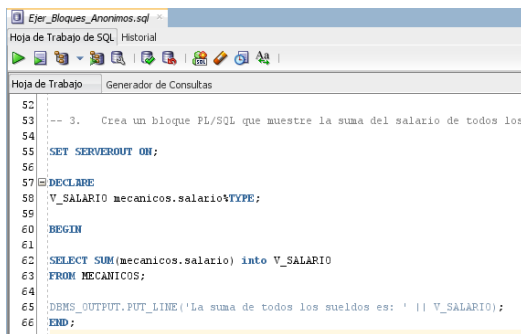
```

DECLARE
V_SALARIO mecanicos.salario%TYPE;

BEGIN

SELECT SUM(mecanicos.salario) into V_SALARIO
FROM MECANICOS;

DBMS_OUTPUT.PUT_LINE('La suma de todos los sueldos es: ' || V_SALARIO);
END;
```



4. Crea un bloque PL/SQL que muestre el número mecánicos y si no hay mecánicos que muestre por pantalla “No hay mecánicos”.

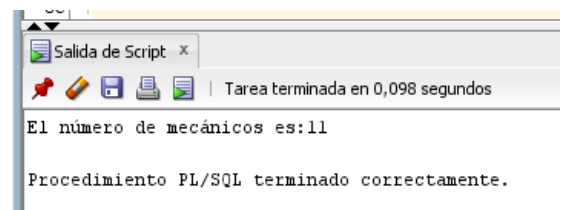
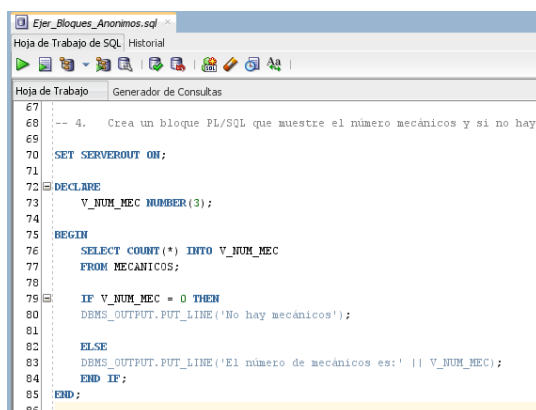
```

DECLARE
V_NUM_MEC NUMBER(3);

BEGIN
SELECT COUNT(*) INTO V_NUM_MEC
FROM MECANICOS;

IF V_NUM_MEC = 0 THEN
DBMS_OUTPUT.PUT_LINE('No hay mecánicos');

ELSE
DBMS_OUTPUT.PUT_LINE('El número de mecánicos es:' || V_NUM_MEC);
END IF;
END;
```



5. Crea un bloque PL/SQL que muestre los datos del mecánico con DNI '1001' (tienes que hacer uso de %ROWTYPE para declarar una variable).

```

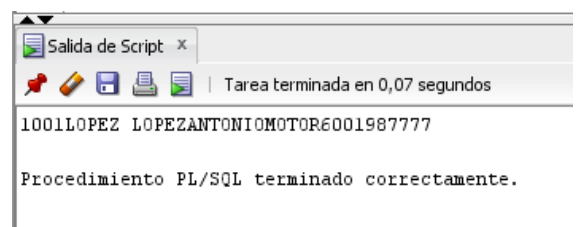
DECLARE
V_MEC MECANICOS%ROWTYPE;

BEGIN

    SELECT * INTO V_MEC
    FROM MECANICOS
    WHERE DNI='1001';

    DBMS_OUTPUT.PUT_LINE(V_MEC.DNI || V_MEC.APELLIDOS || V_MEC.NOMBRE ||
V_MEC.PUESTO || V_MEC.SALARIO || V_MEC.PARCIAL || V_MEC.TELEFONO);

END;
```

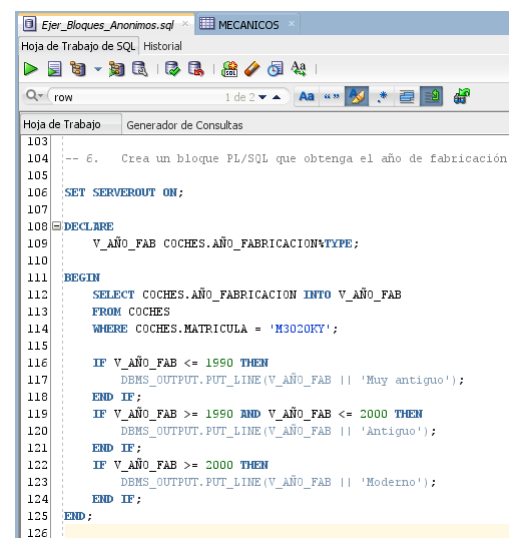
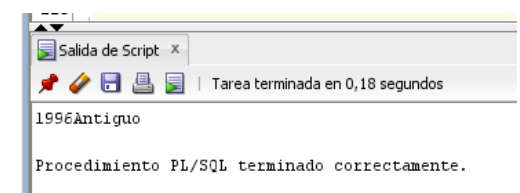
6. Crea un bloque PL/SQL que obtenga el año de fabricación del coche con matrícula 'M3020KY' y muestre por pantalla 'Muy antiguo' si el año es menor que 1990, 'Antiguo' si está entre 1990 y 2000, y 'Moderno' si es superior a 2000.

```

DECLARE
V_AÑO_FAB COCHES.AÑO_FABRICACION%TYPE;

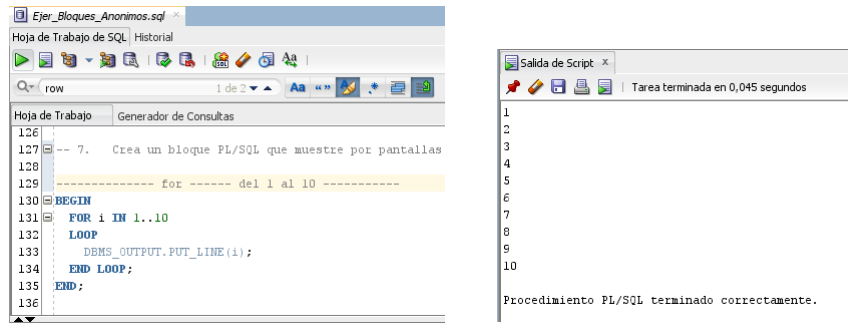
BEGIN
    SELECT COCHES.AÑO_FABRICACION INTO V_AÑO_FAB
    FROM COCHES
    WHERE COCHES.MATRICULA = 'M3020KY';

    IF V_AÑO_FAB <= 1990 THEN
        DBMS_OUTPUT.PUT_LINE(V_AÑO_FAB || 'Muy antiguo');
    END IF;
    IF V_AÑO_FAB >= 1990 AND V_AÑO_FAB <= 2000 THEN
        DBMS_OUTPUT.PUT_LINE(V_AÑO_FAB || 'Antiguo');
    END IF;
    IF V_AÑO_FAB >= 2000 THEN
        DBMS_OUTPUT.PUT_LINE(V_AÑO_FAB || 'Moderno');
    END IF;
END;
```

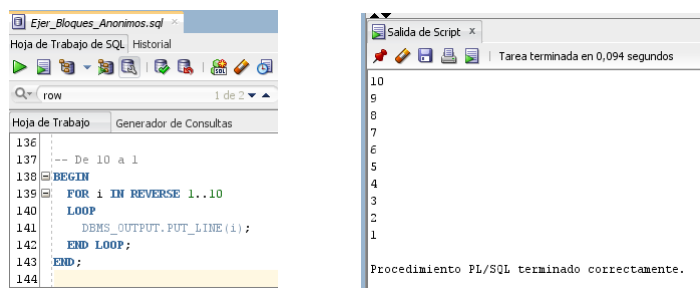



7. Crea un bloque PL/SQL que muestre por pantallas los números de 1 al 10 usando un bucle FOR, los números de 20 al 30 usando un bucle WHILE y los números del 40 al 50 usando un bucle LOOP.

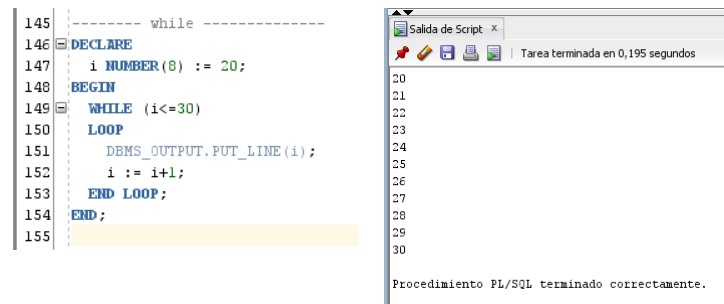
```
BEGIN
  FOR i IN 1..10
  LOOP
    DBMS_OUTPUT.PUT_LINE(i);
  END LOOP;
END;
```



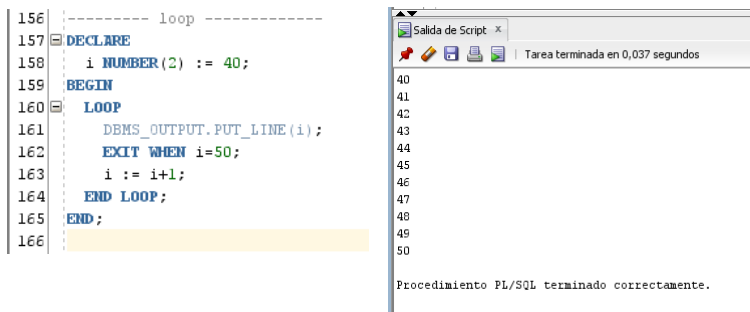
```
BEGIN
  FOR i IN REVERSE 1..10
  LOOP
    DBMS_OUTPUT.PUT_LINE(i);
  END LOOP;
END;
```



```
DECLARE
  i NUMBER(8) := 20;
BEGIN
  WHILE (i<=30)
  LOOP
    DBMS_OUTPUT.PUT_LINE(i);
    i := i+1;
  END LOOP;
END;
```



```
DECLARE
  i NUMBER(2) := 40;
BEGIN
  LOOP
    DBMS_OUTPUT.PUT_LINE(i);
    EXIT WHEN i=50;
    i := i+1;
  END LOOP;
END;
```



Todo junto:

```

DECLARE
  I NUMBER;
BEGIN

FOR I IN 1..10 LOOP
  DBMS_OUTPUT.PUT_LINE(I);
END LOOP;

FOR I IN REVERSE 1..10 LOOP
  DBMS_OUTPUT.PUT_LINE(I);
END LOOP;

I:= 20;
WHILE I>19 AND I<31 LOOP
  DBMS_OUTPUT.PUT_LINE(I);
  I:=I+1;
END LOOP;

I:=40;
LOOP
  DBMS_OUTPUT.PUT_LINE(I);
  I:=I+1;
  IF I>50 THEN
    EXIT;
  END IF;
END LOOP;

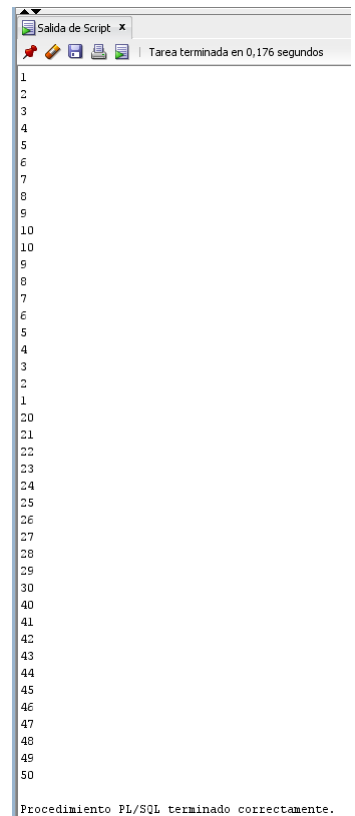
```

END;

```

168 DECLARE
169 I NUMBER;
170 BEGIN
171
172 FOR I IN 1..10 LOOP
173   DBMS_OUTPUT.PUT_LINE(I);
174 END LOOP;
175
176 FOR I IN REVERSE 1..10 LOOP
177   DBMS_OUTPUT.PUT_LINE(I);
178 END LOOP;
179
180 I:= 20;
181 WHILE I>19 AND I<31 LOOP
182   DBMS_OUTPUT.PUT_LINE(I);
183   I:=I+1;
184 END LOOP;
185
186 I:=40;
187 LOOP
188   DBMS_OUTPUT.PUT_LINE(I);
189   I:=I+1;
190   IF I>50 THEN
191     EXIT;
192   END IF;
193   END LOOP;
194 END;

```



8. Crea un bloque aumente el 100 el salario de todos los mecánicos cuyo puesto es CHAPA.

```
SELECT * FROM MECANICOS WHERE MECANICOS.PUESTO = 'CHAPA';
```

```
SET SERVEROUT ON;
```

```

DECLARE
  V_SAL_INCREASE MECANICOS.SALARIO%TYPE := 100;

```

```

BEGIN
  UPDATE MECANICOS
  SET MECANICOS.SALARIO = MECANICOS.SALARIO + V_SAL_INCREASE
  WHERE MECANICOS.PUESTO = 'CHAPA';

```

END;

```
SELECT * FROM MECANICOS WHERE MECANICOS.PUESTO = 'CHAPA';
```

1ºDAM – Bases de datos

```

190 -- 8. Crea un bloque aumente el 100 el salario de todos los mecánicos cuyo puesto es CHAPA.
191
192 SELECT * FROM MECANICOS WHERE MECANICOS.PUESTO = 'CHAPA';
193
194 SET SERVEROUT ON;
195
196 DECLARE
197     V_SAL_INCREASE MECANICOS.SALARIO%TYPE := 100;
198
199 BEGIN
200     UPDATE MECANICOS
201     SET MECANICOS.SALARIO = MECANICOS.SALARIO + V_SAL_INCREASE
202     WHERE MECANICOS.PUESTO = 'CHAPA';
203
204 END;
205
206 SELECT * FROM MECANICOS WHERE MECANICOS.PUESTO = 'CHAPA';
207
208

```

DNI	APELLIDOS	NOMBRE	PUESTO	SALARIO	PARCIAL	TELEFONO
1 1004	VEGA CAMPOS	LOLA	CHAPA	1800 1		987777
2 1006	LOPEZ SANCHEZ	EMILIO	CHAPA	2300 0		875666
3 1008	FERNANDEZ NAVARRO	LUIS	CHAPA	1400 0		(null)
4 1009	HERDANDEZ SANCHEZ	LOLA	CHAPA	1400 0		(null)
5 1010	LOPEZ GARCIA	ANGEL	CHAPA	1400 0		454545

```

193
196 DECLARE
197     V_SAL_INCREASE MECANICOS.SALARIO%TYPE := 100;
198
199 BEGIN
200     UPDATE MECANICOS
201     SET MECANICOS.SALARIO = MECANICOS.SALARIO + V_SAL_INCREASE
202     WHERE MECANICOS.PUESTO = 'CHAPA';
203
204 END;
205
206 SELECT * FROM MECANICOS WHERE MECANICOS.PUESTO = 'CHAPA';
207
208

```

Procedimiento PL/SQL terminado correctamente.

```

206
207 SELECT * FROM MECANICOS WHERE MECANICOS.PUESTO = 'CHAPA';
208

```

DNI	APELLIDOS	NOMBRE	PUESTO	SALARIO	PARCIAL	TELEFONO
1 1004	VEGA CAMPOS	LOLA	CHAPA	1900 1		987777
2 1006	LOPEZ SANCHEZ	EMILIO	CHAPA	2400 0		875666
3 1008	FERNANDEZ NAVARRO	LUIS	CHAPA	1500 0		(null)
4 1009	HERDANDEZ SANCHEZ	LOLA	CHAPA	1500 0		(null)
5 1010	LOPEZ GARCIA	ANGEL	CHAPA	1500 0		454545

9. Crea un bloque PL/SQL que que reciba una cadena de texto por teclado y la muestre por pantalla al revés.

SET SERVEROUT ON;

DECLARE

V_CADENA VARCHAR(10);

V_CADENA_REV VARCHAR(10);

BEGIN

V_CADENA:= '&v_cad';

FOR i IN REVERSE 1..LENGTH(V_CADENA) LOOP

V_CADENA_REV := V_CADENA_REV || SUBSTR(V_CADENA,i,1);

END LOOP;

DBMS_OUTPUT.PUT_LINE(V_CADENA_REV);

END;

```

209 -- 9. Crea un bloque PL/SQL que que reciba una cadena de texto
210
211 SET SERVEROUT ON;
212
213 DECLARE
214     V_CADENA VARCHAR(10);
215     V_CADENA_REV VARCHAR(10);
216
217 BEGIN
218     V_CADENA:='&v_cad';
219     FOR i IN REVERSE 1..LENGTH(V_CADENA) LOOP
220         V_CADENA_REV := V_CADENA_REV || SUBSTR(V_CADENA,i,1);
221     END LOOP;
222     DBMS_OUTPUT.PUT_LINE(V_CADENA_REV);
223 END;

```

Salida de Script x Resultado de la Consulta x

Tarea terminada en 2,011 segundos

```

Antiguo:DECLARE
  V_CADENA VARCHAR(10);
  V_CADENA_REV VARCHAR(10);
BEGIN
  V_CADENA:='&v_cad';
  FOR i IN REVERSE 1..LENGTH(V_CADENA) LOOP
    V_CADENA_REV := V_CADENA_REV || SUBSTR(V_CADENA,i,1);
  END LOOP;
  DBMS_OUTPUT.PUT_LINE(V_CADENA_REV);
END;
Nuevo:DECLARE
  V_CADENA VARCHAR(10);
  V_CADENA_REV VARCHAR(10);
BEGIN
  V_CADENA:='hola';
  FOR i IN REVERSE 1..LENGTH(V_CADENA) LOOP
    V_CADENA_REV := V_CADENA_REV || SUBSTR(V_CADENA,i,1);
  END LOOP;
  DBMS_OUTPUT.PUT_LINE(V_CADENA_REV);
END;
aloh

```

Procedimiento PL/SQL terminado correctamente.

10. Crea un bloque PL/SQL que muestre el número de años completos que hay entre dos fechas (crea dos variables de tipo fecha con las fechas que quieras para probar).

```

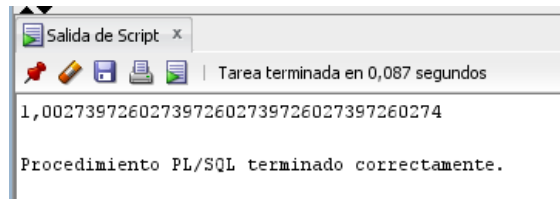
DECLARE
  V_FECHA1 DATE := '01-01-2021';
  V_FECHA2 DATE := '01-01-2020';
  V_RESULTADO NUMBER;

BEGIN
  V_RESULTADO := (V_FECHA1 - V_FECHA2)/365;
  DBMS_OUTPUT.PUT_LINE(V_RESULTADO);

END;
```

```

224 -- 10. Crea un bloque PL/SQL que muestre el número de años
225
226 SET SERVEROUT ON;
227
228 DECLARE
229     V_FECHA1 DATE := '01-01-2021';
230     V_FECHA2 DATE := '01-01-2020';
231     V_RESULTADO NUMBER;
232
233 BEGIN
234     V_RESULTADO := (V_FECHA1 - V_FECHA2)/365;
235     DBMS_OUTPUT.PUT_LINE(V_RESULTADO);
236
237 END;
```



Salida de Script x

Tarea terminada en 0,087 segundos

1,0027397260273972602739726027397260274

Procedimiento PL/SQL terminado correctamente.

11. Crea un bloque PL/SQL que inserte dos mecánicos en la tabla coches y que aumente el salario de un mecánico en 100.

```

DECLARE
  V_MEC1 MECANICOS%ROWTYPE;
  V_MEC2 MECANICOS%ROWTYPE;

  V_DNIMEC MECANICOS.DNI%TYPE;
  V_AUMENTO MECANICOS.SALARIO%TYPE := 100;

  V_MECACT MECANICOS%ROWTYPE; --mecanico actualizado
  V_MECACT2 MECANICOS%ROWTYPE; --mecánico actualizado

BEGIN

  V_MEC1.DNI:='1020';
  V_MEC1.APELLIDOS:='Diaz';
  V_MEC1.NOMBRE:='Sergio';
  V_MEC1.PUESTO:='MOTOR';
  V_MEC1.SALARIO:=1000;
  V_MEC1.PARCIAL:='0';
  V_MEC1.TELEFONO:='6666666';

  V_MEC2.DNI:='3040';
  V_MEC2.APELLIDOS:='Diaz';
  V_MEC2.NOMBRE:='Fernando';
  V_MEC2.PUESTO:='CHAPA';
  V_MEC2.SALARIO:=1100;
  V_MEC2.PARCIAL:='0';
  V_MEC2.TELEFONO:='7777777';
```


V_DNIMEC := V_MEC2.DNI;

INSERT INTO MECANICOS VALUES (V_MEC1.DNI, V_MEC1.APELLIDOS, V_MEC1.NOMBRE, V_MEC1.PUESTO, V_MEC1.SALARIO, V_MEC1.PARCIAL, V_MEC1.TELEFONO);

INSERT INTO MECANICOS VALUES (V_MEC2.DNI, V_MEC2.APELLIDOS, V_MEC2.NOMBRE, V_MEC2.PUESTO, V_MEC2.SALARIO, V_MEC2.PARCIAL, V_MEC2.TELEFONO);

UPDATE MECANICOS

SET SALARIO = SALARIO + V_AUMENTO
WHERE DNI = V_DNIMEC;

SELECT * INTO V_MECACT
FROM MECANICOS
WHERE DNI=V_DNIMEC;

V_DNIMEC := V_MEC1.DNI;

SELECT * INTO V_MECACT2
FROM MECANICOS
WHERE DNI=V_DNIMEC;

END;

The screenshot shows a SQL Developer window titled 'Ejer_Bloques_Anonimos.sql'. The main editor contains a PL/SQL script with the following code:

```

SET SERVEROUTPUT ON;

DECLARE
    V_MEC1 MECANICOS%ROWTYPE;
    V_MEC2 MECANICOS%ROWTYPE;

    V_DNIMEC MECANICOS.DNI%TYPE;
    V_AUMENTO MECANICOS.SALARIO%TYPE := 100;

    V_MECACT MECANICOS%ROWTYPE; --mecanico actualizado
    V_MECACT2 MECANICOS%ROWTYPE; --mecánico actualizado

BEGIN

    V_MEC1.DNI:='1020';
    V_MEC1.APELLIDOS:='Perez';
    V_MEC1.NOMBRE:='FACO';
    V_MEC1.PUESTO:='CHAPA';
    V_MEC1.SALARIO:=1000;
    V_MEC1.PARCIAL:='0';
    V_MEC1.TELEFONO:='6666432';

    V_MEC2.DNI:='1030';
    V_MEC2.APELLIDOS:='Sucino';
    V_MEC2.NOMBRE:='ANER';
    V_MEC2.PUESTO:='MOTOR';
    V_MEC2.SALARIO:=1010;
    V_MEC2.PARCIAL:='0';
    V_MEC2.TELEFONO:='4343434';

    V_DNIMEC := V_MEC2.DNI;

    INSERT INTO MECANICOS VALUES (V_MEC1.DNI, V_MEC1.APELLIDOS, V_MEC1.NOMBRE, V_MEC1.PUESTO, V_MEC1.SALARIO, V_MEC1.PARCIAL, V_MEC1.TELEFONO);
    INSERT INTO MECANICOS VALUES (V_MEC2.DNI, V_MEC2.APELLIDOS, V_MEC2.NOMBRE, V_MEC2.PUESTO, V_MEC2.SALARIO, V_MEC2.PARCIAL, V_MEC2.TELEFONO);

    UPDATE MECANICOS

    SET SALARIO = SALARIO + V_AUMENTO
    WHERE DNI = V_DNIMEC;

    SELECT * INTO V_MECACT
    FROM MECANICOS
    WHERE DNI=V_DNIMEC;

    V_DNIMEC := V_MEC1.DNI;

    SELECT * INTO V_MECACT2
    FROM MECANICOS
    WHERE DNI=V_DNIMEC;

END;
    
```

At the bottom, a 'Salida de Script' (Script Output) window shows the message: 'Procedimiento PL/SQL terminado correctamente.' (PL/SQL procedure terminated successfully).