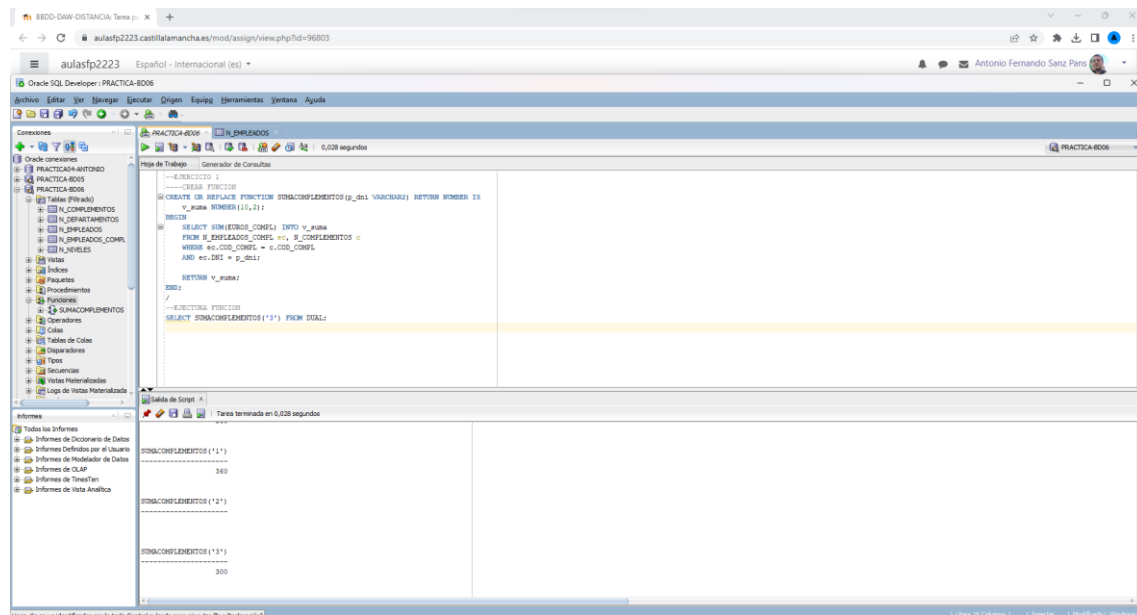


TAREA UT00 – BASES DE DATOS

ANTONIO SANZ PANS
IES MAESTRE DE CALATRAVA

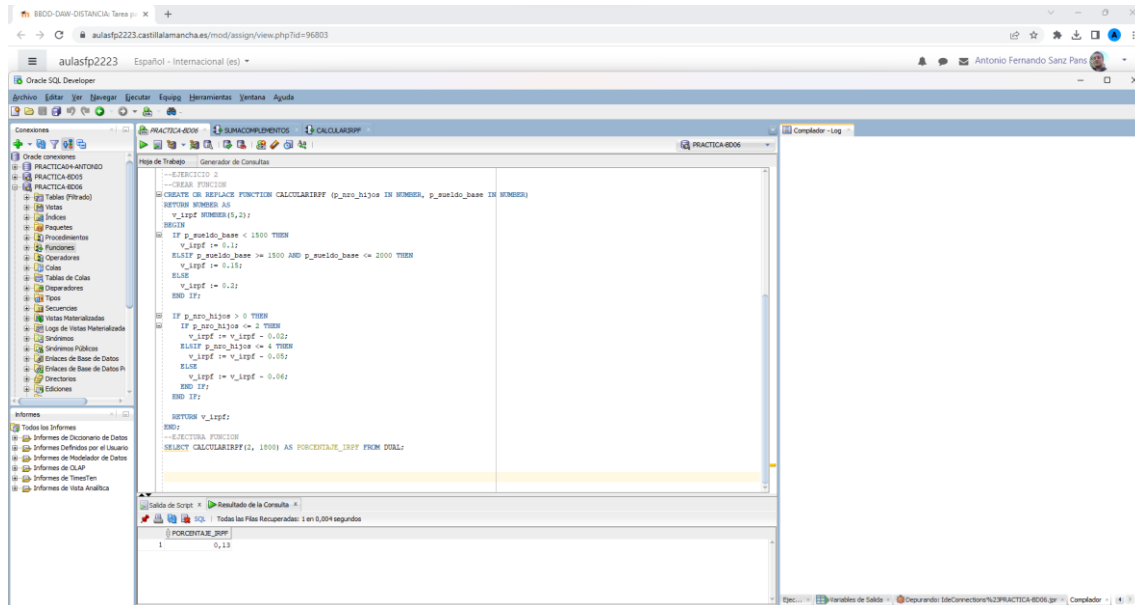
EJERCICIO 1



```
--EJERCICIO 1
--CREAR FUNCION
CREATE OR REPLACE FUNCTION SUMACOMPLEMENTOS(p_dni VARCHAR2) RETURN NUMBER IS
    v_suma NUMBER(10,2);
BEGIN
    SELECT SUM(EUROS_COMPL) INTO v_suma
    FROM N_EMPLEADOS_COMPL ec, N_COMPLEMENTOS c
    WHERE ec.COD_COMPL = c.COD_COMPL
    AND ec.DNI = p_dni;

    RETURN v_suma;
END;
/
--EJECTURA FUNCION
SELECT SUMACOMPLEMENTOS('3') FROM DUAL;
```

EJERCICIO 2



```
--CREAR FUNCION
CREATE OR REPLACE FUNCTION CALCULARIRPF (p_nro_hijos IN NUMBER, p_sueldo_base IN NUMBER)
RETURN NUMBER AS
  v_irpf NUMBER(5,2);
BEGIN
  IF p_sueldo_base < 1500 THEN
    v_irpf := 0.1;
  ELSIF p_sueldo_base >= 1500 AND p_sueldo_base <= 2000 THEN
    v_irpf := 0.15;
  ELSE
    v_irpf := 0.2;
  END IF;

  IF p_nro_hijos > 0 THEN
    IF p_nro_hijos <= 2 THEN
      v_irpf := v_irpf - 0.02;
    ELSIF p_nro_hijos <= 4 THEN
      v_irpf := v_irpf - 0.05;
    ELSE
      v_irpf := v_irpf - 0.06;
    END IF;
  END IF;

  RETURN v_irpf;
END;
/

--EJECTURA FUNCION
SELECT CALCULARIRPF(2, 1800) AS PORCENTAJE_IRPF FROM DUAL;
```

EJERCICIO 3

The screenshot shows the Oracle SQL Developer interface. The main window displays the SQL script for creating a procedure named `CALCULARSUeldoDEPT`. The script includes comments in Spanish and SQL code for declaring variables, creating cursors, and looping through departments and levels to calculate net salaries.

```
1  P_COD_NIVEL IN S_NIVELES.COD_NIVEL%TYPE,  
2  P_TOTAL_SUeldo OUT NUMBER  
3  ) AS  
4  
5  vsuma_completoes NUMBER(10,2);  
6  v_sueldo_neto NUMBER(10,2);  
7  v_sueldo_neto NUMBER(10,2);  
8  v_lvl NUMBER(1,2);  
9  v_sueldo_neto NUMBER(10,2);  
10  
11  --Declaramos las excepciones  
12  NO_EXISTE_DEPARTAMENTO EXCEPTION;  
13  NO_HAY_EMPLEADOS_DEPARTAMENTO EXCEPTION;  
14  NO_HAY_EMPLEADOS_CURSOR EXCEPTION;  
15  
16  -- Declaramos el cursor para recorrer los empleados del departamento  
17  CURSOR c_empleados IS  
18  SELECT DEPT, NIVEL, SAL, SUeldo_NETO, NRO_TRIENIOS, NRO_NIVEL  
19  FROM S_EMPLEADOS WHERE S_NIVELES = DEPT AND S_COD_NIVEL = P_COD_NIVEL;  
20  WHERE S_COD_DEPT = P_COD_DEPT AND S_COD_NIVEL = P_COD_NIVEL;  
21  --Declaramos las excepciones
```

The status bar at the bottom indicates: "Tarea terminada en 0,073 segundos".

The screenshot shows the Oracle SQL Developer interface with the procedure `CALCULARSUeldoDEPT` executed. The main window displays the SQL script for the procedure, and the status bar at the bottom indicates the execution result.

```
1  SET SERVEROUTPUT ON;  
2  DECLARE  
3  v_total_sueldo NUMBER;  
4  BEGIN  
5  CALCULARSUeldoDEPT(p_cod_dep => '01', p_cod_nivel => '4', p_total_sueldo => v_total_sueldo);  
6  END;
```

The status bar at the bottom indicates: "Tarea terminada en 0,064 segundos". Below the status bar, the output of the procedure is displayed: "El total de los sueldos netos de los empleados de departamento 01 y nivel 4 es: 3494,75".

```
--CREAR PROCEDIMIENTO
CREATE OR REPLACE PROCEDURE CALCULARSUELDODEPT (
  p_cod_dep IN N_DEPARTAMENTOS.COD_DEP%TYPE,
  p_cod_nivel IN N_NIVELES.COD_NIVEL%TYPE,
  p_total_sueldo OUT NUMBER
) AS
  v_suma_complementos NUMBER(10,2);
  v_sueldo_bruto NUMBER(10,2);
  v_sueldo_netto NUMBER(10,2);
  v_irpf NUMBER(5,2);
  v_sueldo_total NUMBER(10,2);
  --Declaramos las excepciones
  NO_EXISTE_DEPARTAMENTO EXCEPTION;
  NO_HAY_EMPLEADOS_DEPART EXCEPTION;
  NO_HAY_EMPLEADOS_CURSOR EXCEPTION;

  -- Declaramos el cursor para recorrer los empleados del departamento
  CURSOR c_empleados IS
    SELECT DNI, SUELDO_BASE, EUROS_UN_TRIENIO, NRO_TRIENIOS, NRO_HIJOS
    FROM N_EMPLEADOS e JOIN N_NIVELES n ON e.COD_NIVEL = n.COD_NIVEL
    WHERE e.COD_DEP = p_cod_dep AND e.COD_NIVEL = p_cod_nivel;

BEGIN
  -- Comprobamos que el departamento exista
  SELECT COUNT(*) INTO p_total_sueldo FROM N_DEPARTAMENTOS WHERE COD_DEP = p_cod_dep;
  IF p_total_sueldo = 0 THEN
    RAISE NO_EXISTE_DEPARTAMENTO;
    p_total_sueldo := -1;
    RETURN;
  END IF;

  -- Comprobamos que haya empleados en el departamento
  SELECT COUNT(*) INTO p_total_sueldo FROM N_EMPLEADOS WHERE COD_DEP = p_cod_dep;
  IF p_total_sueldo = 0 THEN
    RAISE NO_HAY_EMPLEADOS_DEPART;
    p_total_sueldo := -2;
    RETURN;
  END IF;

  v_sueldo_total := 0;
  FOR emp IN c_empleados LOOP
    v_suma_complementos := SUMACOMPLEMENTOS(emp.DNI);
    v_sueldo_bruto := emp.SUELDO_BASE + emp.NRO_TRIENIOS * emp.EUROS_UN_TRIENIO +
    v_suma_complementos;
    v_irpf := CALCULARIRPF(emp.NRO_HIJOS, v_sueldo_bruto);
    v_sueldo_netto := v_sueldo_bruto * (1 - v_irpf/100);
    v_sueldo_total := v_sueldo_total + v_sueldo_netto;
  END LOOP;

  IF v_sueldo_total = 0 THEN
    RAISE NO_HAY_EMPLEADOS_CURSOR;
    p_total_sueldo := -3;
    RETURN;
  ELSE
    p_total_sueldo := v_sueldo_total;
    DBMS_OUTPUT.PUT_LINE('El total de los sueldos netos de los empleados de
    departamento ' || p_cod_dep || ' y nivel ' || p_cod_nivel || ' es: ' || v_sueldo_total);
  END IF;
EXCEPTION
  WHEN NO_EXISTE_DEPARTAMENTO THEN
    DBMS_OUTPUT.PUT_LINE('No existe el departamento con código ' || p_cod_dep);
    P_TOTAL_SUELDO := -1;
  WHEN NO_HAY_EMPLEADOS_DEPART THEN
    DBMS_OUTPUT.PUT_LINE('El departamento ' || p_cod_dep || ' no tiene empleados.');
```

Página 4 | 4