

APARTADO 1

ENUNCIADO:

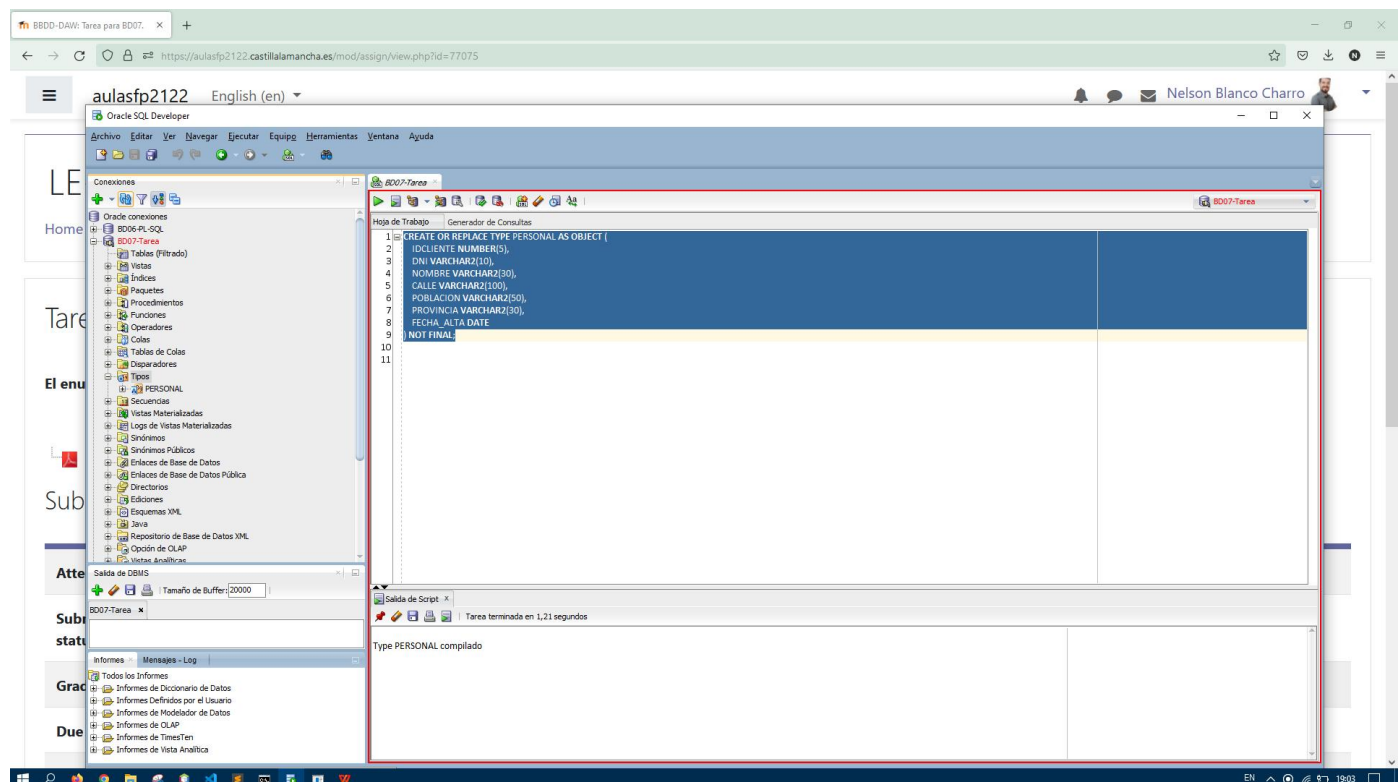
Crea un tipo de objetos llamado PERSONAL con los siguientes atributos:

IDCLIENTE NUMBER(5)
DNI VARCHAR2(10)
NOMBRE VARCHAR2(30)
CALLE VARCHAR2(100)
POBLACION VARCHAR2(50)
PROVINCIA VARCHAR2(30)
FECHA_ALTA DATE

SENTENCIA SQL:

```
CREATE OR REPLACE TYPE PERSONAL AS OBJECT (  
    IDCLIENTE NUMBER(5),  
    DNI VARCHAR2(10),  
    NOMBRE VARCHAR2(30),  
    CALLE VARCHAR2(100),  
    POBLACION VARCHAR2(50),  
    PROVINCIA VARCHAR2(30),  
    FECHA_ALTA DATE  
) NOT FINAL;
```

CAPTURA EJECUCIÓN EN SQL DEVELOPER:



APARTADO 2

ENUNCIADO:

Crea el tipo de objeto llamado CLIENTE como tipo heredado de PERSONAL, con los siguientes atributos propios:

NRO_CUENTA VARCHAR2(20) TIPO_CLI NUMBER(1)

Crea un primer método constructor para el tipo de objetos CLIENTE, en el que se indiquen como parámetros idcliente, dni, nombre, apellidos, calle, población y provincia.

A la fecha de alta se le asignará la fecha del sistema y al atributo NOMBRE se le asignará apellidos concatenado al nombre que se han pasado como parámetros, uniéndolos con un espacio entre ellos.

Crea un segundo método constructor para el tipo de objetos CLIENTE, en el que se indiquen como parámetros todos los atributos del cliente.

Crea un método getDireccion para el tipo de objeto CLIENTE que permita obtener su dirección completa concatenando la calle, población y provincia.

Crea también en cuerpo (BODY) del tipo de objetos CLIENTE.

SENTENCIA SQL:

Creación del objeto CLIENTE

```
CREATE OR REPLACE TYPE CLIENTE UNDER PERSONAL(
    NRO_CUENTA VARCHAR2(20),
    TIPO_CLI NUMBER(1),
    -- Primer constructor
    CONSTRUCTOR FUNCTION CLIENTE(P_IDCLIENTE NUMBER,P_DNI VARCHAR2, P_NOMBRE VARCHAR2, P_APELLIDOS
    VARCHAR2, P_CALLE VARCHAR2, P_POBLACION VARCHAR2, P_PROVINCIA VARCHAR2)
        RETURN SELF AS RESULT,
    -- Segundo constructor
    CONSTRUCTOR FUNCTION CLIENTE(P_IDCLIENTE NUMBER, P_DNI VARCHAR2, P_NOMBRE VARCHAR2, P_CALLE VARCHAR2,
    P_POBLACION VARCHAR2, P_PROVINCIA VARCHAR2, P_FECHA_ALTA DATE)
        RETURN SELF AS RESULT,
    -- Metodo getDireccion
    MEMBER FUNCTION GETDIRECCION RETURN VARCHAR2,
    -- Metodos setter
    MEMBER PROCEDURE SETIDCLIENTE(IDCLIENTE NUMBER),
    MEMBER PROCEDURE SETDNI(DNI VARCHAR2),
    MEMBER PROCEDURE SETNOMBRE(NOMBRE VARCHAR2, APELLIDO VARCHAR2),
    MEMBER PROCEDURE SETNOMBRE(NOMBRE VARCHAR2),
    MEMBER PROCEDURE SETCALLE(CALLE VARCHAR2),
    MEMBER PROCEDURE SETPOBLACION(POBLACION VARCHAR2),
    MEMBER PROCEDURE SETPROVINCIA(PROVINCIA VARCHAR2),
    MEMBER PROCEDURE SETFECHALTA(FECHA NUMBER),
    MEMBER PROCEDURE SETFECHALTA(FECHA_ALTA DATE)
);
```

Body de CLIENTE

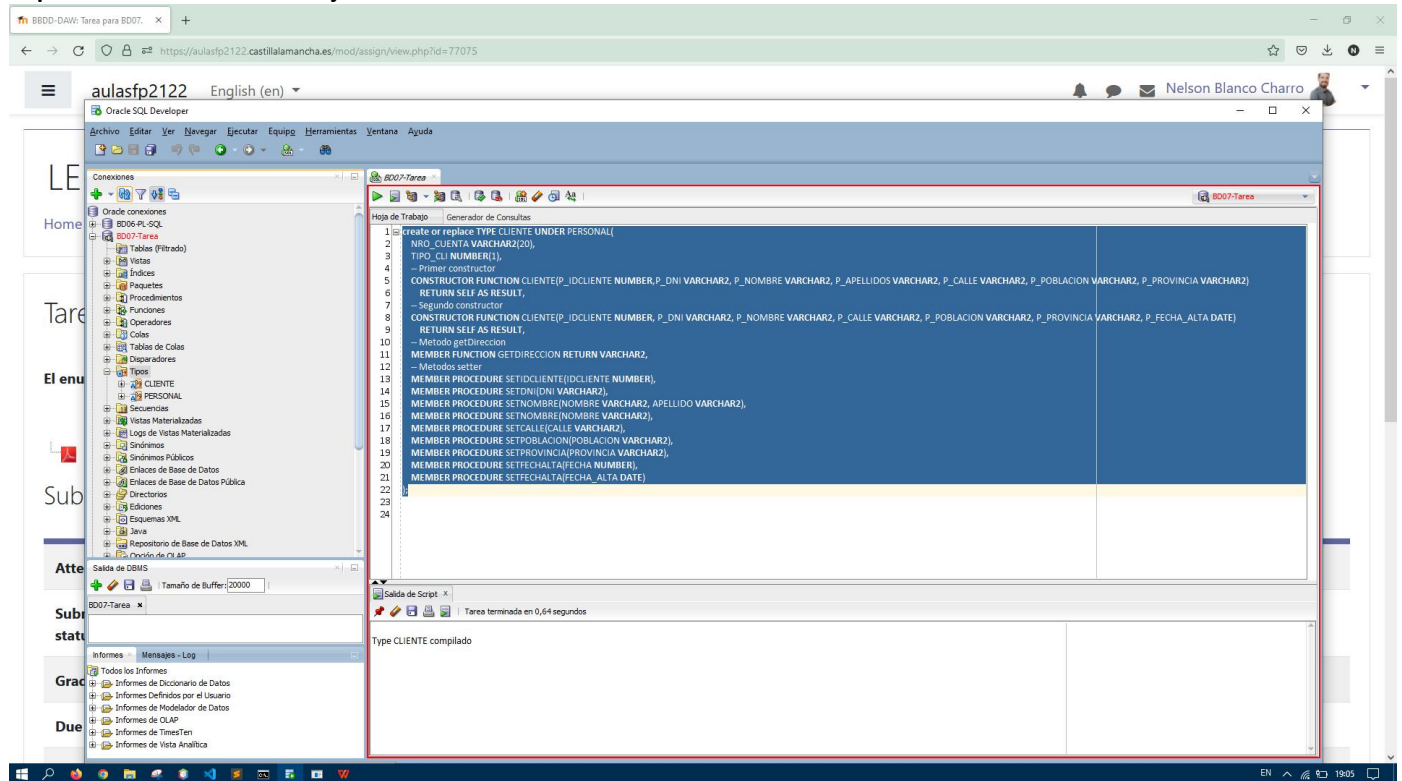
```
CREATE OR REPLACE TYPE BODY CLIENTE AS
    -- Primer constructor
    CONSTRUCTOR FUNCTION CLIENTE(P_IDCLIENTE NUMBER,P_DNI VARCHAR2, P_NOMBRE VARCHAR2, P_APELLIDOS
    VARCHAR2, P_CALLE VARCHAR2, P_POBLACION VARCHAR2, P_PROVINCIA VARCHAR2)
        RETURN SELF AS RESULT AS
    BEGIN
        SELF.SETIDCLIENTE(P_IDCLIENTE);
        SELF.SETDNI(P_DNI);
        SELF.SETNOMBRE(P_NOMBRE, P_APELLIDOS);
        SELF.SETCALLE(P_CALLE);
        SELF.SETPOBLACION(P_POBLACION);
        SELF.SETPROVINCIA(P_PROVINCIA);
        SELF.SETFECHALTA(0);
    RETURN;
    END CLIENTE;
    -- Segundo constructor
    CONSTRUCTOR FUNCTION CLIENTE(P_IDCLIENTE NUMBER, P_DNI VARCHAR2, P_NOMBRE VARCHAR2, P_CALLE VARCHAR2,
    P_POBLACION VARCHAR2, P_PROVINCIA VARCHAR2, P_FECHA_ALTA DATE)
        RETURN SELF AS RESULT AS
    BEGIN
```

```

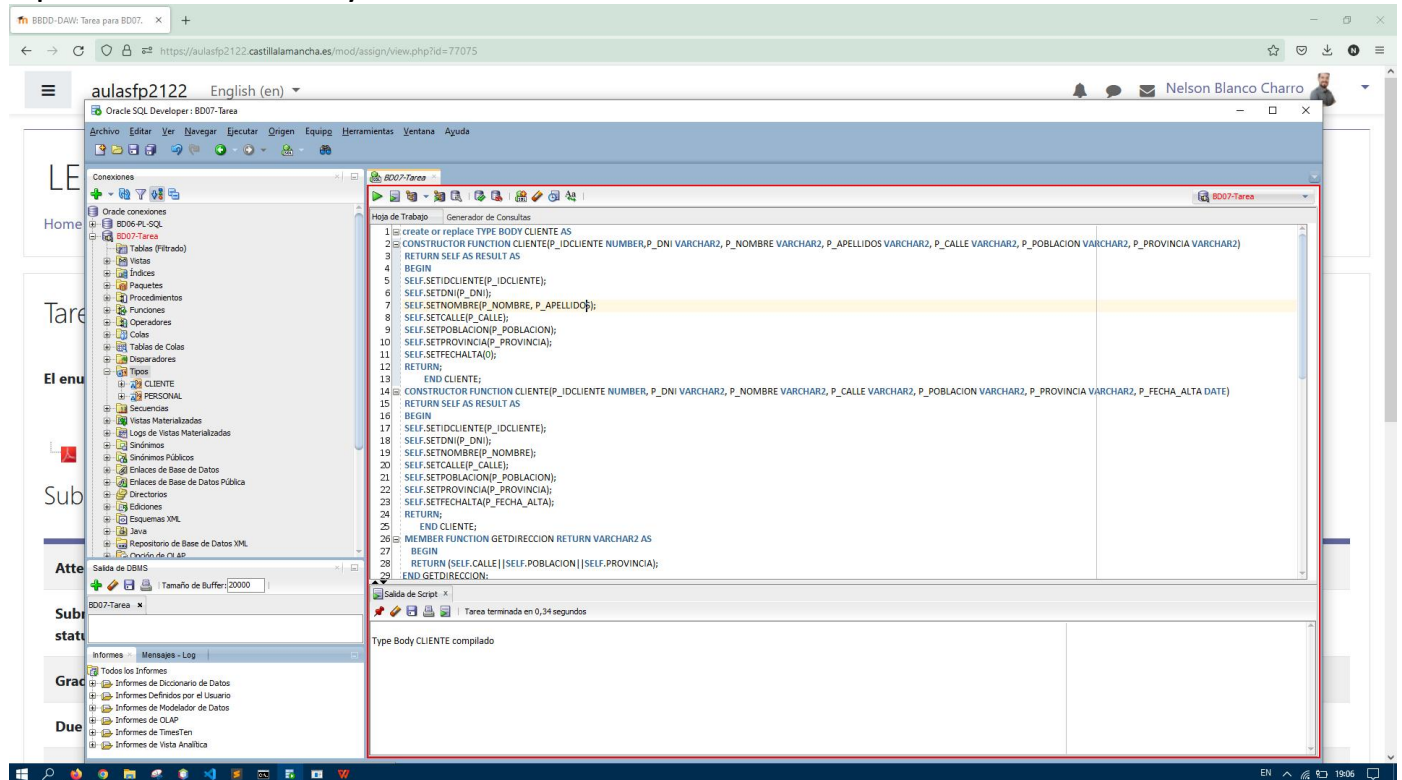
        SELF.SETIDCLIENTE(P_IDCLIENTE);
        SELF.SETDNI(P_DNI);
        SELF.SETNOMBRE(P_NOMBRE);
        SELF.SETCALLE(P_CALLE);
        SELF.SETPOBLACION(P_POBLACION);
        SELF.SETPROVINCIA(P_PROVINCIA);
        SELF.SETFECHALTA(P_FECHA_ALTA);
    RETURN;
END CLIENTE;
-- Metodo getDireccion
MEMBER FUNCTION GETDIRECCION RETURN VARCHAR2 AS
    BEGIN
        RETURN (SELF.CALLE || SELF.POBLACION || SELF.PROVINCIA);
    END GETDIRECCION;
-- Metodos setter
MEMBER PROCEDURE SETIDCLIENTE(IDCLIENTE NUMBER) AS
    BEGIN
        SELF.IDCLIENTE:=IDCLIENTE;
    END SETIDCLIENTE;
MEMBER PROCEDURE SETDNI(DNI VARCHAR2) AS
    BEGIN
        SELF.DNI:=DNI;
    END SETDNI;
MEMBER PROCEDURE SETNOMBRE(NOMBRE VARCHAR2, APELLIDO VARCHAR2) AS
    BEGIN
        SELF.NOMBRE:=APELLIDO || ', ' || NOMBRE;
    END SETNOMBRE;
MEMBER PROCEDURE SETNOMBRE(NOMBRE VARCHAR2) AS
    BEGIN
        SELF.NOMBRE:=NOMBRE;
    END SETNOMBRE;
MEMBER PROCEDURE SETCALLE(CALLE VARCHAR2) AS
    BEGIN
        SELF.CALLE:=CALLE;
    END SETCALLE;
MEMBER PROCEDURE SETPOBLACION(POBLACION VARCHAR2) AS
    BEGIN
        SELF.POBLACION:=POBLACION;
    END SETPOBLACION;
MEMBER PROCEDURE SETPROVINCIA(PROVINCIA VARCHAR2) AS
    BEGIN
        SELF.PROVINCIA:=PROVINCIA;
    END SETPROVINCIA;
MEMBER PROCEDURE SETFECHALTA(FECHA NUMBER) AS
    BEGIN
        SELF.FECHA_ALTA:=SYSDATE;
    END SETFECHALTA;
MEMBER PROCEDURE SETFECHALTA(FECHA_ALTA DATE) AS
    BEGIN
        SELF.FECHA_ALTA:=FECHA_ALTA;
    END SETFECHALTA;
END;
```

CAPTURA EJECUCIÓN EN SQL DEVELOPER:

Captura de la creación del objeto CLIENTE



Captura de la creación del body de CLIENTE



ENUNCIADO:

Crea el tipo de objeto SERVICIO con los siguientes atributos:

IDSERVICIO NUMBER(5)
REFCLIENTE REF CLIENTE
NRO_HORAS NUMBER(3)
PRECIO_HORA NUMBER(5,2)
FECHA_SERVICIO DATE

SENTENCIA SQL:

Creación del objeto SERVICIO

```
CREATE OR REPLACE TYPE SERVICIO AS OBJECT (  
    IDSERVICIO NUMBER(5),  
    REFCLIENTE REF CLIENTE,  
    NRO_HORAS NUMBER(3),  
    PRECIO_HORA NUMBER(5,2),  
    FECHA_SERVICIO DATE,  
    -- CONSTRUCTOR  
    CONSTRUCTOR FUNCTION SERVICIO(IDSERVICIO NUMBER, REFCLIENTE REF CLIENTE, NRO_HORAS NUMBER, PRECIO_HORA  
NUMBER, FECHA_SERVICIO DATE)  
        RETURN SELF AS RESULT,  
    -- GETTER Y SETTER  
    MEMBER FUNCTION GETIDSERVICIO RETURN NUMBER,  
    MEMBER PROCEDURE SETIDSERVICIO(IDSERVICIO NUMBER),  
  
    MEMBER FUNCTION GETREFCLIENTE RETURN REF CLIENTE,  
    MEMBER PROCEDURE SETREFCLIENTE(REFCLIENTE REF CLIENTE),  
  
    MEMBER FUNCTION GETNRO_HORAS RETURN NUMBER,  
    MEMBER PROCEDURE SETNRO_HORAS(NRO_HORAS NUMBER),  
  
    MEMBER FUNCTION GETPRECIO_HORA RETURN NUMBER,  
    MEMBER PROCEDURE SETPRECIO_HORA(PRECIO_HORA NUMBER),  
  
    MEMBER FUNCTION GETFECHA_SERVICIO RETURN DATE,  
    MEMBER PROCEDURE SETFECHA_SERVICIO(FECHA_SERVICIO DATE)  
) NOT FINAL;
```

Body de SERVICIO

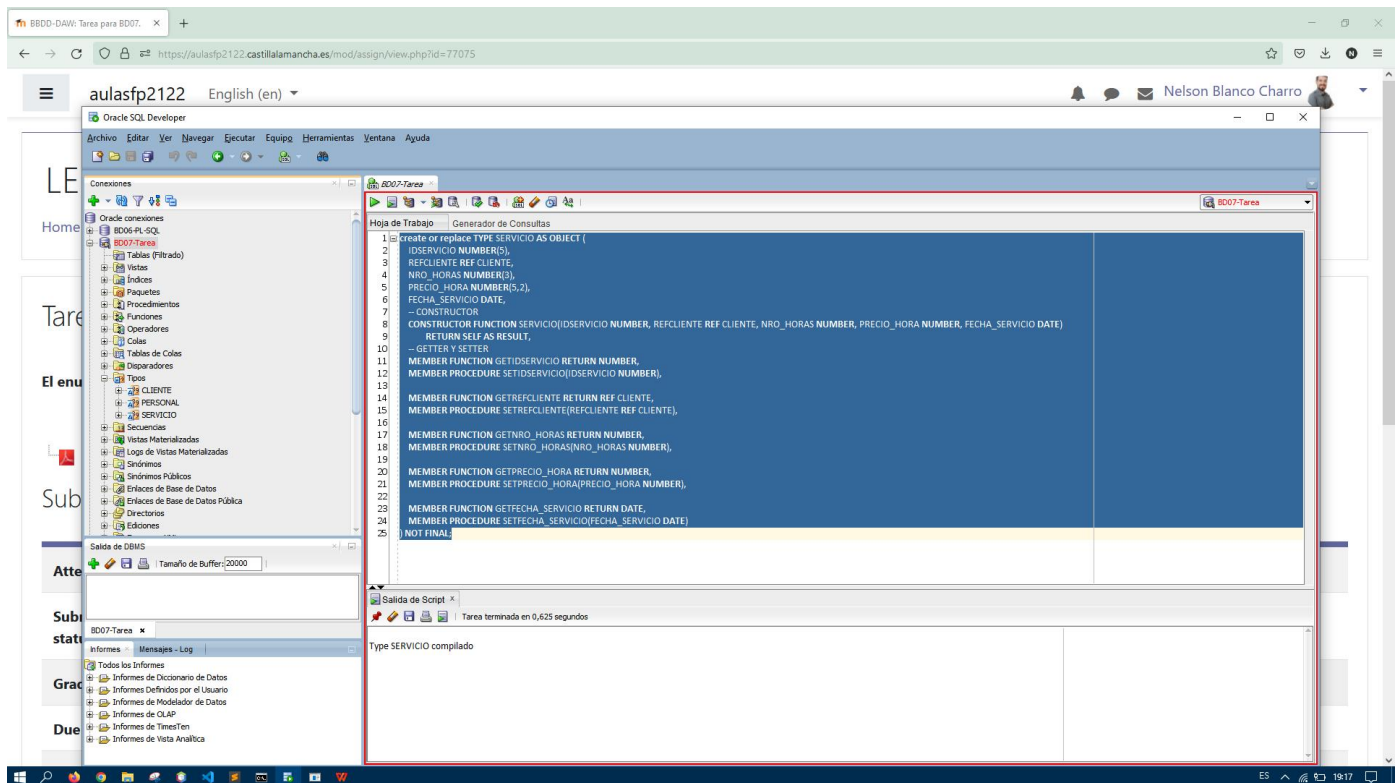
```
CREATE OR REPLACE TYPE BODY SERVICIO AS  
    -- CONSTRUCTOR  
    CONSTRUCTOR FUNCTION SERVICIO(IDSERVICIO NUMBER, REFCLIENTE REF CLIENTE, NRO_HORAS NUMBER, PRECIO_HORA  
NUMBER, FECHA_SERVICIO DATE)  
        RETURN SELF AS RESULT AS  
    BEGIN  
        SELF.SETIDSERVICIO(IDSERVICIO);  
        SELF.SETREFCLIENTE(REFCLIENTE);  
        SELF.SETNRO_HORAS(NRO_HORAS);  
        SELF.SETPRECIO_HORA(PRECIO_HORA);  
        SELF.SETFECHA_SERVICIO(FECHA_SERVICIO);  
        RETURN;  
    END SERVICIO;  
    -- GETTER Y SETTER  
    MEMBER FUNCTION GETIDSERVICIO RETURN NUMBER AS  
    BEGIN  
        RETURN SELF.IDSERVICIO;  
    END GETIDSERVICIO;  
    MEMBER PROCEDURE SETIDSERVICIO(IDSERVICIO NUMBER) AS  
    BEGIN  
        SELF.IDSERVICIO:=IDSERVICIO;  
    END SETIDSERVICIO;  
    MEMBER FUNCTION GETREFCLIENTE RETURN REF CLIENTE AS
```

```

BEGIN
RETURN SELF.REFCLIENTE;
END GETREFCLIENTE;
MEMBER PROCEDURE SETREFCLIENTE(REFCLIENTE REF CLIENTE) AS
BEGIN
SELF.REFCLIENTE:=REFCLIENTE;
END SETREFCLIENTE;
MEMBER FUNCTION GETNRO_HORAS RETURN NUMBER AS
BEGIN
RETURN SELF.NRO_HORAS;
END GETNRO_HORAS;
MEMBER PROCEDURE SETNRO_HORAS(NRO_HORAS NUMBER) AS
BEGIN
SELF.NRO_HORAS:=NRO_HORAS;
END SETNRO_HORAS;
MEMBER FUNCTION GETPRECIO_HORA RETURN NUMBER AS
BEGIN
RETURN SELF.PRECIO_HORA;
END GETPRECIO_HORA;
MEMBER PROCEDURE SETPRECIO_HORA(PRECIO_HORA NUMBER) AS
BEGIN
SELF.PRECIO_HORA:=PRECIO_HORA;
END SETPRECIO_HORA;
MEMBER FUNCTION GETFECHA_SERVICIO RETURN DATE AS
BEGIN
RETURN SELF.FECHA_SERVICIO;
END GETFECHA_SERVICIO;
MEMBER PROCEDURE SETFECHA_SERVICIO(FECHA_SERVICIO DATE) AS
BEGIN
SELF.FECHA_SERVICIO:=FECHA_SERVICIO;
END SETFECHA_SERVICIO;
END;

```

CAPTURA EJECUCIÓN EN SQL DEVELOPER:



APARTADO 4

ENUNCIADO:

Crea un tabla CLIENTES_OBJ de objetos de tipo CLIENTE cuya clave primaria sea el IDCLIENTE.

Inserta en dicha tabla tres objetos CLIENTE.

El primero de ellos usando el primer constructor, con los siguientes datos:

IDCLIENTE : 1
DNI : 111111111A
NOMBRE : MARÍA
APELLIDOS : LASO MAR
CALLE : TOLEDO, 21-1º C
POBLACION : CIUDAD REAL
PROVINCIA : CIUDAD REAL

El segundo de ellos usando el segundo constructor, con los siguientes datos:

IDCLIENTE : 2
DNI : 2222222B
NOMBRE : MIRTA GIL, JUAN
CALLE : COSO,10
POBLACION : DAIMIEL
PROVINCIA : CIUDAD REAL
FECHA_ALTA : 04/02/2021

El tercero de ellos usando el segundo constructor, con los siguientes datos:

IDCLIENTE : 3
DNI : 33333333C
NOMBRE : TOSAR MESA, PEDRO
CALLE : BARRIAL,10
POBLACION : ARGÉS
PROVINCIA : TOLEDO
FECHA_ALTA : 03/08/2020

SENTENCIA SQL:

```
CREATE TABLE CLIENTES_OBJ OF CLIENTE (PRIMARY KEY (IDCLIENTE));  
INSERT INTO CLIENTES_OBJ VALUES(CLIENTE(1, '111111111A', 'MARÍA', 'LASO MAR', 'TOLEDO, 21-1º C', 'CIUDAD REAL',  
'CIUDAD REAL' ));  
INSERT INTO CLIENTES_OBJ VALUES(CLIENTE(2, '2222222B', 'MIRTA GIL, JUAN', 'COSO,10', 'DAIMIEL', 'CIUDAD REAL',  
TO_DATE('04/02/2021','DD/MM/YYYY') ));  
INSERT INTO CLIENTES_OBJ VALUES(CLIENTE(3, '33333333C', 'TOSAR MESA, PEDRO', 'BARRIAL,10', 'ARGÉS', 'TOLEDO',  
TO_DATE('03/08/2020','DD/MM/YYYY') ));  
COMMIT;
```

CAPTURA EJECUCIÓN EN SQL DEVELOPER:

BBDD-DAW: Tarea para BD07. x

https://aulasfp2122.castillalamancha.es/mod/assign/view.php?id=77075

aulasfp2122 English (en)

Nelson Blanco Charro

Oracle SQL Developer: BD07-Tarea

Archivo Editar Ver Navegar Ejecutar Origen Equipos Herramientas Ventana Ayuda

Conexiones

Oracle conexiones
BD06-FI-SQL
BD07-Tarea
 Tablas (Filtrado)
 CLIENTES_OBI
 Vistas
 Indices
 Paquetes
 Procedimientos
 Funciones
 Operadores
 Colas
 Tablas de Colas
 Disparadores
 Tipos
 CLIENTE
 PERSONAL
 SERVICIO
 Secuencias
 Vistas Materializadas
 Logs de Vistas Materializadas
 Sinónimos
 Sinónimos Públicos
 Enlaces de Base de Datos
 Enlaces de Base de Datos Pública
 Directorios

BD07-Tarea

Hoja de Trabajo Generador de Consultas

```
1 CREATE TABLE CLIENTES_OBI OF CLIENTE (PRIMARY KEY (IDCLIENTE));  
2 INSERT INTO CLIENTES_OBI VALUES(CLIENTE(1, '1111111111A', 'MARIA', 'LASO MAR', 'TOLEDO, 21-18 C', 'CIUDAD REAL', 'CIUDAD REAL'));  
3 INSERT INTO CLIENTES_OBI VALUES(CLIENTE(2, '22222222B', 'MIRTA GIL, JUAN', 'COSO,10', 'DAIMIEL', 'CIUDAD REAL', 'TO_DATE('04/02/2021','DD/MM/YYYY')));  
4 INSERT INTO CLIENTES_OBI VALUES(CLIENTE(3, '33333333C', 'TOSAR MESA, PEDRO', 'BARRIAL,10', 'ARGÉS', 'TOLEDO', 'TO_DATE('03/08/2020','DD/MM/YYYY')));  
5 COMMIT;
```

Salida de DBMS

Salida de Script

Tarea terminada en 3,049 segundos

Table CLIENTES_OBI creado.
1 fila insertadas.
1 fila insertadas.
1 fila insertadas.
Confirmación terminada.

BD07-Tarea

Informes Mensajes - Log

Todos los Informes
 Informes de Diccionario de Datos
 Informes Definidos por el Usuario
 Informes de Modelador de Datos
 Informes de OLAP
 Informes de TimesTen
 Informes de Vista Analítica

ES

19:19

APARTADO 5

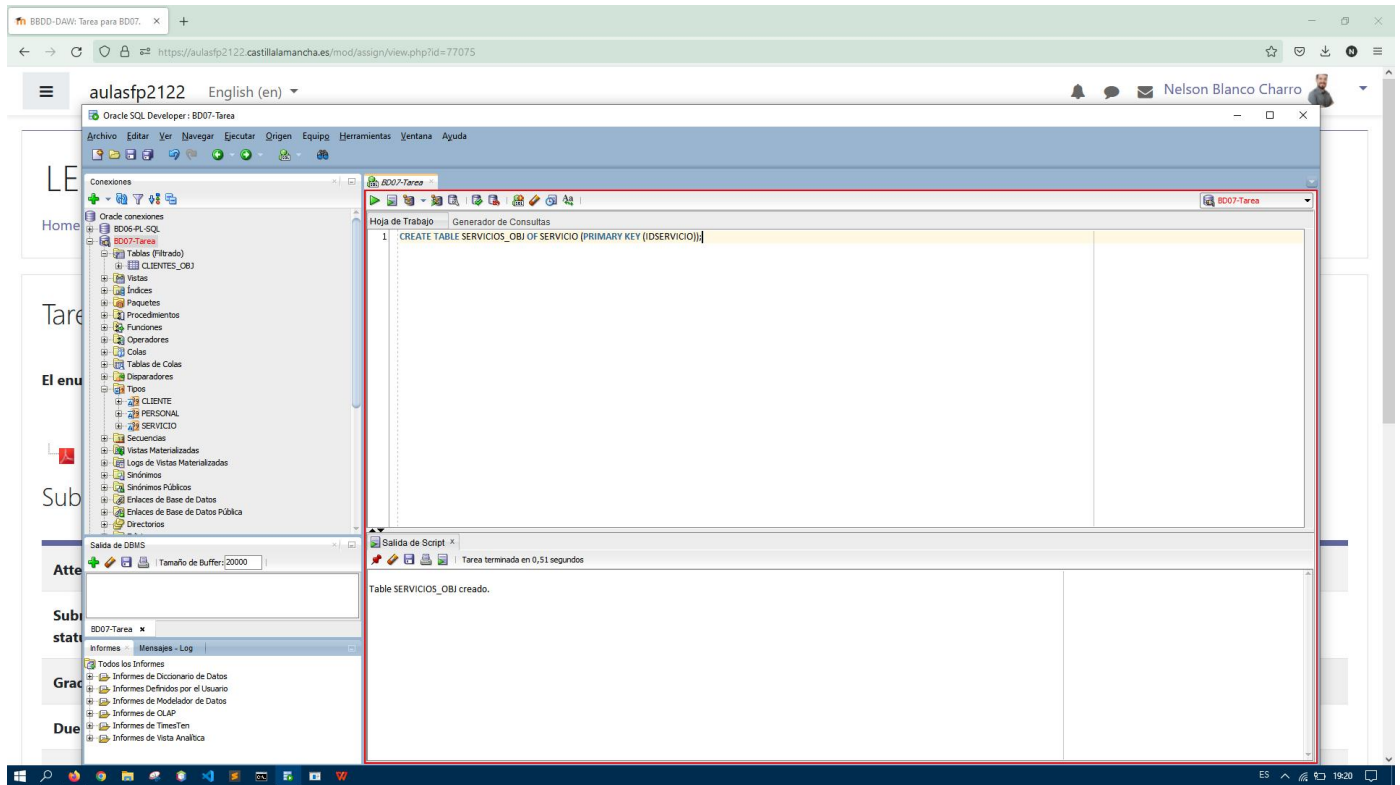
ENUNCIADO:

Crea una tabla SERVICIOS_OBJ de objetos SERVICIO la clave primaria es el IDSERVICIO.

SENTENCIA SQL:

CREATE TABLE SERVICIOS_OBJ OF SERVICIO (PRIMARY KEY (IDSERVICIO));

CAPTURA EJECUCIÓN EN SQL DEVELOPER:



APARTADO 6

ENUNCIADO:

Crea un bloque anónimo en PL/SQL que realice lo siguiente:

- Declara una variable llamada REFCLIENTE1. Obtener de la tabla CLIENTES_OBJ una referencia al cliente cuyo IDCLIENTE es 1
- Declara una variable llamada REFCLIENTE2. Obtener de la tabla CLIENTES_OBJ una referencia al cliente cuyo IDCLIENTE es 1
- Inserta en la tabla SERVICIOS_OBJ dos objetos

El primer servicio que debes almacenar en la tabla debe tener los siguientes datos:

```
IDSERVICIO : 11
REFCLIENTE : Referencia al cliente 1(REFCLIENTE1)
NRO_HORAS : 10
PRECIO_HORA : 30
FECHA_SERVICIO: 01/02/2022
```

El segundo servicio que debes almacenar en la tabla debe tener los siguientes datos:

```
IDSERVICIO : 22
REFCLIENTE : Referencia al cliente 1(REFCLIENTE2)
NRO_HORAS : 20
PRECIO_HORA : 40
FECHA_SERVICIO: 12/03/2022
```

- Seleccionar de la tabla CLIENTES_OBJ, el cliente que tiene el idcliente 2, asignándoselo a una variable llamada CLIENTE_2. Modifica el idcliente por un nuevo valor: 22, el dni del cliente guardado en esa variable CLIENTE_2, le asignaremos el valor 21111111B. Inserta CLIENTE_2 en la tabla CLIENTES_OBJ.
- Seleccionar de la tabla SERVICIOS_OBJ, el servicio cuyo id es 11 en una variable llamada SERVICIO_V. Modificaremos el id servicio y le asignaremos 12 y al cliente le asignaremos cliente cuyo idcliente sea 3. Insertaremos en la tabla SERVICIOS_OBJ un nuevo registro con los valores de SERVICIO_V

SENTENCIA SQL:

DECLARE

```
REFCLIENTE1 REF CLIENTE;
REFCLIENTE2 REF CLIENTE;
REFCLIENTE3 REF CLIENTE;
p1 SERVICIO;
CLIENTE_2 CLIENTE;
SERVICIO_V SERVICIO;
```

BEGIN

```
SELECT REF(u) INTO REFCLIENTE1 FROM CLIENTES_OBJ u WHERE u.IDCLIENTE = 1;
p1 := NEW SERVICIO(11, REFCLIENTE1, 10, 30, TO_DATE('01/02/2022','DD/MM/YYYY'));
INSERT INTO SERVICIOS_OBJ VALUES(p1);

SELECT REF(u) INTO REFCLIENTE2 FROM CLIENTES_OBJ u WHERE u.IDCLIENTE = 2;
INSERT INTO SERVICIOS_OBJ VALUES(22, REFCLIENTE2, 20, 40, TO_DATE('12/03/2022','DD/MM/YYYY'));

SELECT VALUE(u) INTO CLIENTE_2 FROM CLIENTES_OBJ U WHERE U.IDCLIENTE = 2;
CLIENTE_2.IDCLIENTE:=22;
CLIENTE_2.DNI:= '21111111B';
INSERT INTO CLIENTES_OBJ VALUES(CLIENTE_2);

SELECT VALUE(u) INTO SERVICIO_V FROM SERVICIOS_OBJ U WHERE U.IDSERVICIO = 11;
SERVICIO_V.IDSERVICIO:=12;
SELECT REF(u) INTO REFCLIENTE3 FROM CLIENTES_OBJ u WHERE u.IDCLIENTE = 3;
SERVICIO_V.REFCLIENTE:=REFCLIENTE3;
INSERT INTO SERVICIOS_OBJ VALUES(SERVICIO_V);

END;
```

CAPTURA EJECUCIÓN EN SQL DEVELOPER:

BBDD-DAW: Tarea para BD07. x

← → ↻ 🔍 <https://aulasfp2122.castillalamancha.es/mod/assign/view.php?id=77075> ☆ 📄 📌 📧

aulasfp2122 English (en) Nelson Blanco Charro

Oracle SQL Developer

Archivo Editar Ver Navegar Ejecutar Equipo Herramientas Ventana Ayuda

Conexiones

Oracle conexiones

BD06-PL-SQL

BD07-Tarea

Tablas (Filtrado)

CLIENTES_OBI

SERVICIOS_OBI

Vistas

Indices

Paquetes

Procedimientos

Funciones

Operadores

Tablas de Colas

Disparadores

Tipos

CLIENTE

PERSONAL

SERVICIO

Secuencias

Vistas Materializadas

Log de Vistas Materializadas

Síndromes

Síndromes Públicos

Enlaces de Base de Datos

Enlaces de Base de Datos Pública

Salida de DBMS

Tamaño de Buffer: 20000

BD07-Tarea

Informes Mensajes Log

Todos los Informes

Informes de Diccionario de Datos

Informes Definidos por el Usuario

Informes de Modelador de Datos

Informes de OLAP

Informes de TimesTen

Informes de Vista Analítica

Hoja de Trabajo

Generador de Consultas

0,81 segundos

BD07-Tarea

```
1 DECLARE
2   REFLIEN1 REF CLIENTE;
3   REFLIEN2 REF CLIENTE;
4   REFLIEN3 REF CLIENTE;
5   p1 SERVICIO;
6   CLIENTE_2 CLIENTE;
7   SERVICIO_V SERVICIO;
8 BEGIN
9   SELECT REF(u) INTO REFLIEN1 FROM CLIENTES_OBI u WHERE u.IDCLIENTE = 1;
10  p1 := NEW SERVICIO(11, REFLIEN1, 10, 30, TO_DATE('01/02/2022','DD/MM/YYYY'));
11  INSERT INTO SERVICIOS_OBI VALUES(p1);
12
13  SELECT REF(u) INTO REFLIEN2 FROM CLIENTES_OBI u WHERE u.IDCLIENTE = 2;
14  INSERT INTO SERVICIOS_OBI VALUES(22, REFLIEN2, 20, 40, TO_DATE('12/03/2022','DD/MM/YYYY'));
15
16  SELECT VALUE(u) INTO CLIENTE_2 FROM CLIENTES_OBI u WHERE u.IDCLIENTE = 2;
17  CLIENTE_2.IDCLIENTE:=22;
18  CLIENTE_2.DNI:= '21111111B';
19  INSERT INTO CLIENTES_OBI VALUES(CLIENTE_2);
20
21  SELECT VALUE(u) INTO SERVICIO_V FROM SERVICIOS_OBI u WHERE u.IDSERVICIO = 11;
22  SERVICIO_V.IDSERVICIO:=12;
23  SELECT REF(u) INTO REFLIEN3 FROM CLIENTES_OBI u WHERE u.IDCLIENTE = 3;
24  SERVICIO_V.REFLIEN:=REFLIEN3;
25  INSERT INTO SERVICIOS_OBI VALUES(SERVICIO_V);
26
27 END;
```

Salida de Script

Tarea terminada en 0,81 segundos

Procedimiento PL/SQL terminado correctamente.

ES

19:29

APARTADO 7

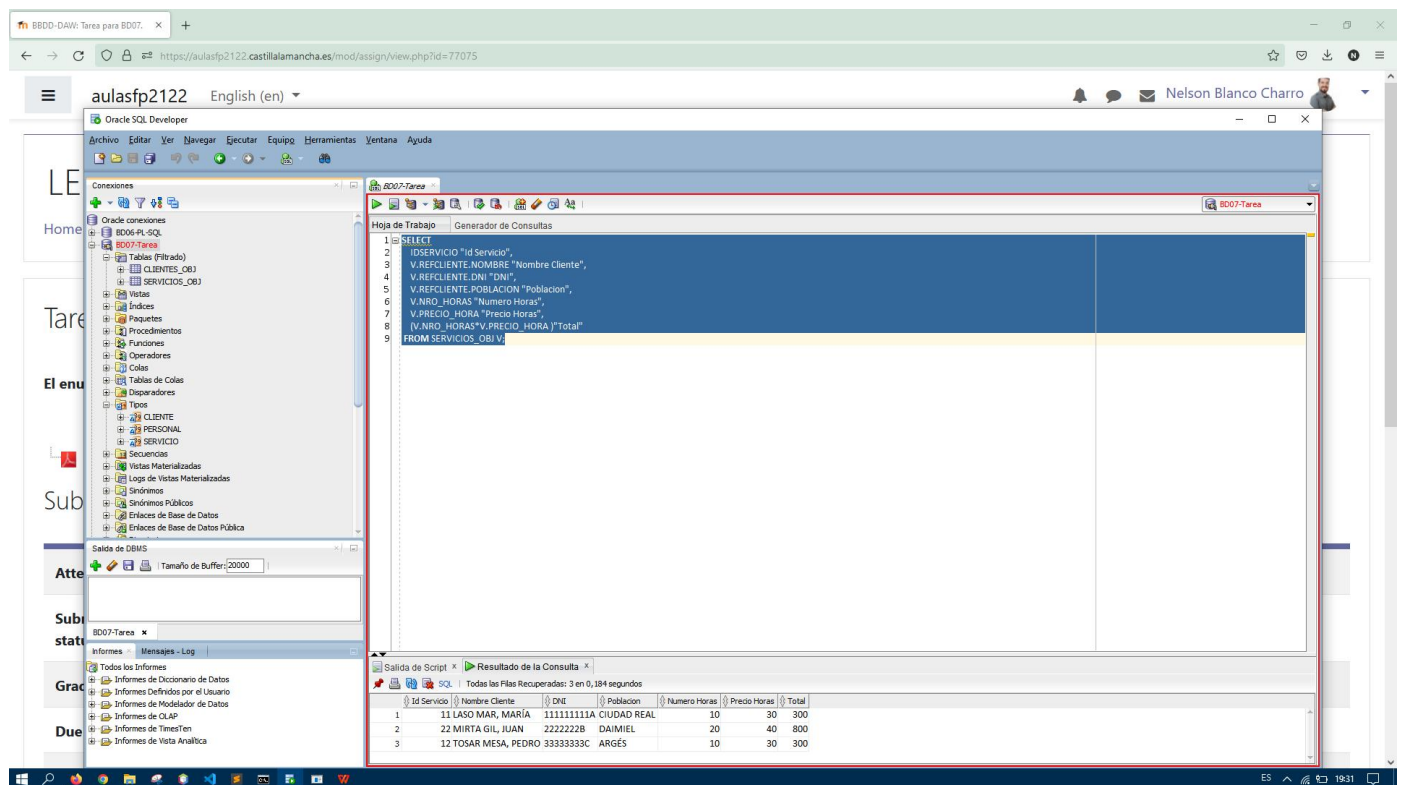
ENUNCIADO:

Realiza una consulta de la tabla SERVICIOS_OBJ en la que se visualicen de todos los registros los siguientes datos: IDSERVICIO, NOMBRE CLIENTE, DNI, POBLACION, NRO_HORAS, PRECIO HORA Y TOTAL (NRO_HORAS*PRECIO HORA)

SENTENCIA SQL:

```
SELECT
    IDSERVICIO "Id Servicio",
    V.REFCLIENTE.NOMBRE "Nombre Cliente",
    V.REFCLIENTE.DNI "DNI",
    V.REFCLIENTE.POBLACION "Poblacion",
    V.NRO_HORAS "Numero Horas",
    V.PRECIO_HORA "Precio Horas",
    (V.NRO_HORAS*V.PRECIO_HORA )"Total"
FROM SERVICIOS_OBJ V;
```

CAPTURA EJECUCIÓN EN SQL DEVELOPER:



The screenshot shows the Oracle SQL Developer interface. The 'Hoja de Trabajo' (Worksheet) tab is active, displaying the following SQL query:

```
1 SELECT
2     IDSERVICIO "Id Servicio",
3     V.REFCLIENTE.NOMBRE "Nombre Cliente",
4     V.REFCLIENTE.DNI "DNI",
5     V.REFCLIENTE.POBLACION "Poblacion",
6     V.NRO_HORAS "Numero Horas",
7     V.PRECIO_HORA "Precio Horas",
8     (V.NRO_HORAS*V.PRECIO_HORA )"Total"
9 FROM SERVICIOS_OBJ V;
```

The 'Resultado de la Consulta' (Query Result) tab shows the following data:

ID Servicio	Nombre Cliente	DNI	Poblacion	Numero Horas	Precio Horas	Total
1	11 LASO MAR, MARÍA	1111111111A	CIUDAD REAL	10	30	300
2	22 MIRTA GIL, JUAN	22222222B	DAIMIÉL	20	40	800
3	12 TOSAR MESA, PEDRO	33333333C	ARGÉS	10	30	300