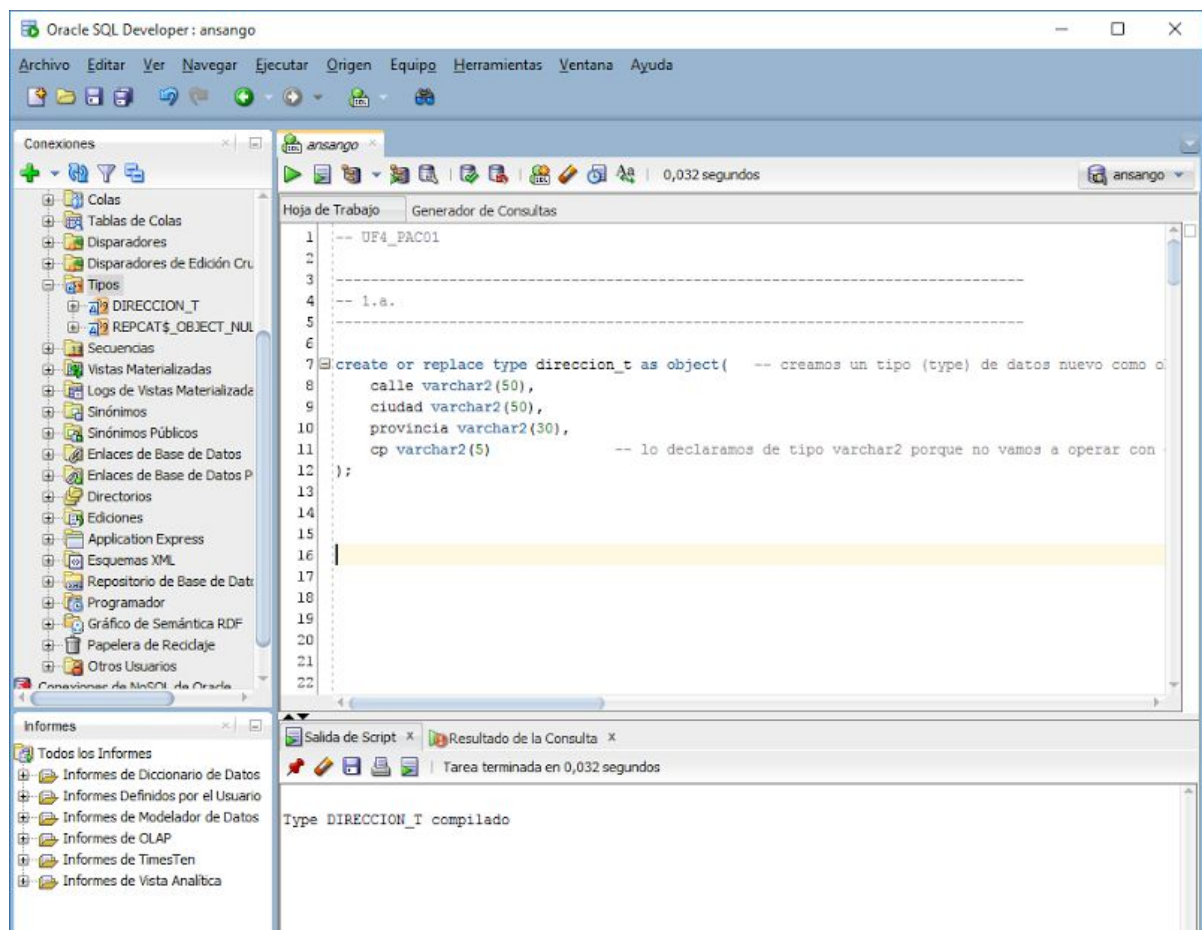


## 1.a

### DIRECCION\_T

-- creamos un tipo (type) de datos nuevo como objeto (object). Utilizamos el sufijo \_t por convenio para indicar que es un tipo de datos

```
create or replace type direccion_t as object(  
    calle varchar2(50),  
    ciudad varchar2(50),  
    provincia varchar2(30),  
    cp varchar2(5)          -- lo declaramos de tipo varchar2 porque no vamos  
a operar con él  
);
```



## 1.B

### EMPLEADO\_T

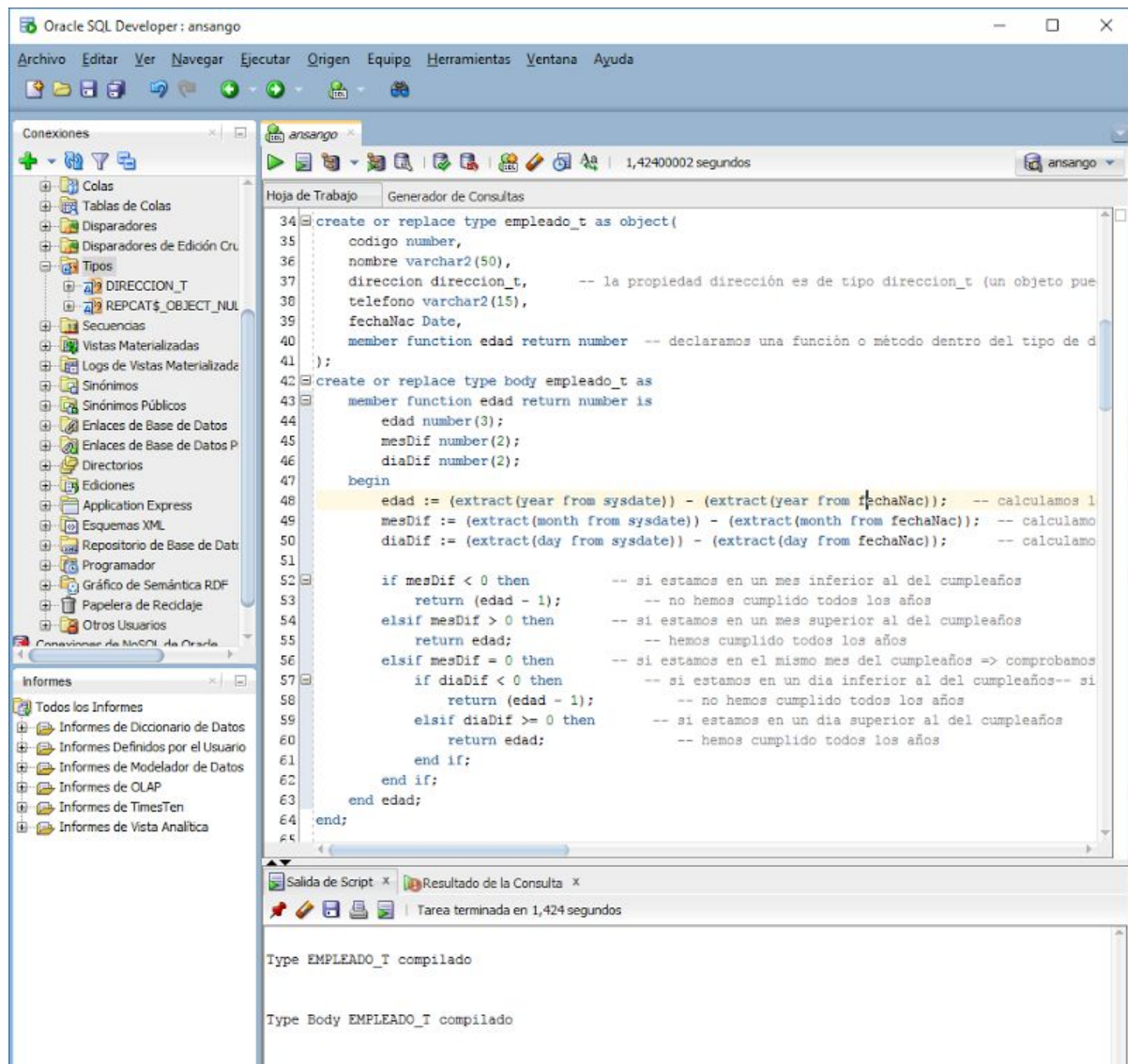
```
create or replace type empleado_t as object(
```

```

    codigo number,
    nombre varchar2(50),
    direccion direccion_t,      -- La propiedad dirección es de tipo direccion_t
    (un objeto puede contener propiedades de otros tipos de objetos)
    telefono varchar2(15),
    fechaNac Date,
    member function edad return number -- declaramos una función o método dentro
    del tipo de datos empleado_t que se llama edad y devuelve number
);
create or replace type body empleado_t as
    member function edad return number is
        edad number(3);
        mesDif number(2);
        diaDif number(2);
    begin
        edad := (extract(year from sysdate)) - (extract(year from fechaNac)); --
        calculamos los años de diferencia
        mesDif := (extract(month from sysdate)) - (extract(month from fechaNac));
        -- calculamos los meses
        diaDif := (extract(day from sysdate)) - (extract(day from fechaNac));
        -- calculamos los días

        if mesDif < 0 then                -- si estamos en un mes inferior al del
        cumpleaños
            return (edad - 1);            -- no hemos cumplido todos los años
        elsif mesDif > 0 then              -- si estamos en un mes superior al del
        cumpleaños
            return edad;                  -- hemos cumplido todos los años
        elsif mesDif = 0 then              -- si estamos en el mismo mes del cumpleaños
        => comprobamos el día
            if diaDif < 0 then              -- si estamos en un día inferior al del
        cumpleaños-- si
                return (edad - 1);        -- no hemos cumplido todos los años
            elsif diaDif >= 0 then          -- si estamos en un día superior al del
        cumpleaños
                return edad;              -- hemos cumplido todos los años
            end if;
        end if;
    end edad;
end;

```



2.

## DIRECCION\_T

-- hemos de eliminar el tipo empleado\_t antes de modificar direccion\_t debido a que forma parte de él

```
drop type empleado_t;
```

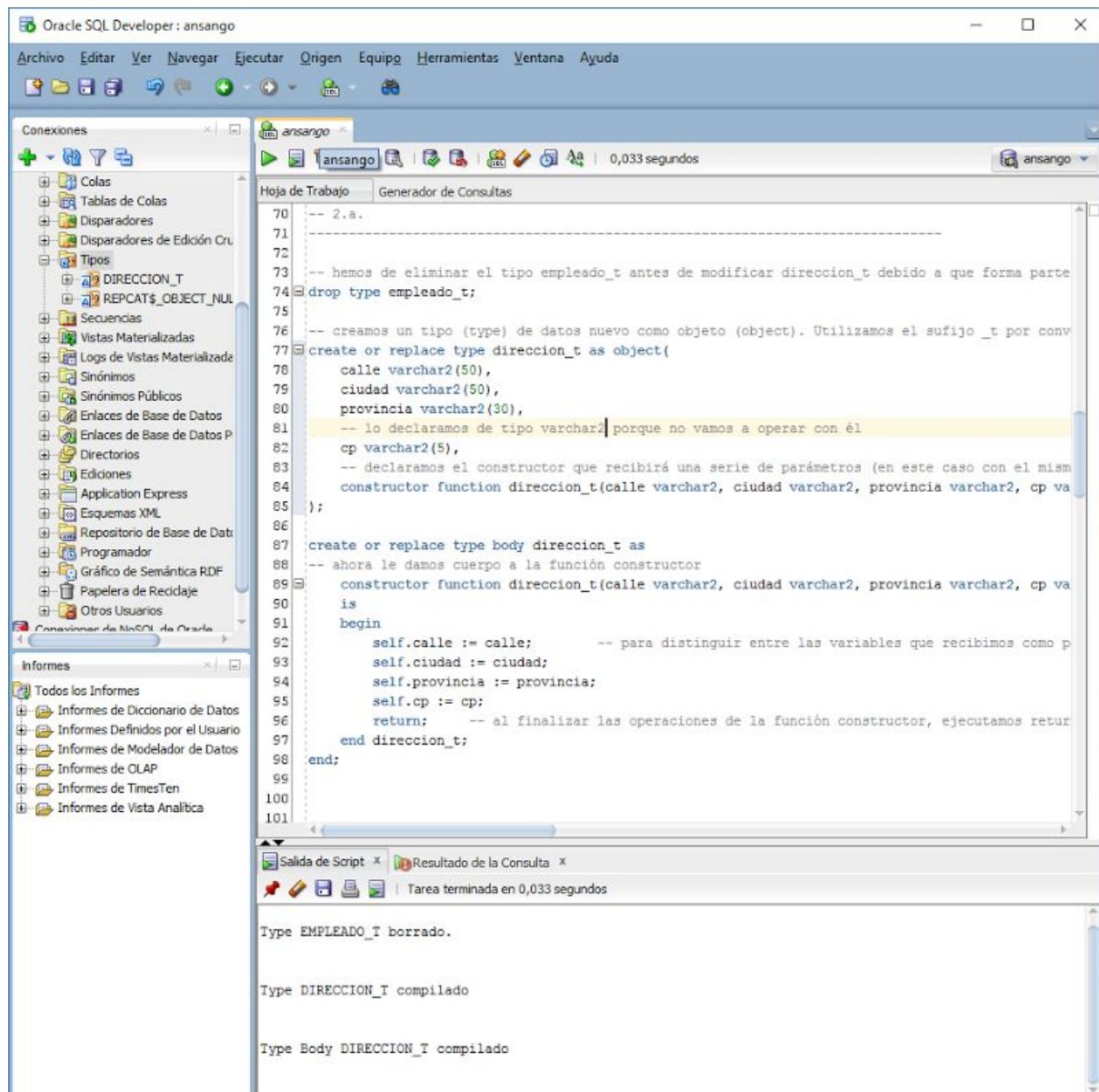
-- creamos un tipo (type) de datos nuevo como objeto (object). Utilizamos el sufijo \_t por convenio para indicar que es un tipo de datos

```
create or replace type direccion_t as object(
    calle varchar2(50),
```

```

ciudad varchar2(50),
provincia varchar2(30),
-- lo declaramos de tipo varchar2 porque no vamos a operar con él
cp varchar2(5),
-- declaramos el constructor que recibirá una serie de parámetros (en este
-- caso con el mismo nombre que las propiedades del objeto y, lógicamente, del mismo
-- tipo). Esta función especial devuelve el propio objeto como resultado de su
-- ejecución
constructor function direccion_t(calle varchar2, ciudad varchar2, provincia
varchar2, cp varchar2) return self as result
);

create or replace type body direccion_t as
-- ahora le damos cuerpo a la función constructor
constructor function direccion_t(calle varchar2, ciudad varchar2, provincia
varchar2, cp varchar2) return self as result
is
begin
    self.calle := calle;      -- para distinguir entre las variables que
-- recibimos como parámetros de las propiedades (variables que forman el objeto) del
-- objeto nos referimos a las propiedades del objeto anteponiendo la palabra
-- reservada self seguida de un punto y el nombre de la propiedad
    self.ciudad := ciudad;
    self.provincia := provincia;
    self.cp := cp;
    return;      -- al finalizar las operaciones de la función constructor,
-- ejecutamos return para que devuelva el nuevo objeto creado
end direccion_t;
end;
```



## EMPLEADO\_T

```

create or replace type empleado_t as object(
    codigo number,
    nombre varchar2(50),
    direccion direccion_t,      -- La propiedad dirección es de tipo direccion_t
    (un objeto puede contener propiedades de otros tipos de objetos)
    telefono varchar2(15),
    fechaNac Date,
    constructor function empleado_t(codigo number, nombre varchar2, direccion
    direccion_t, telefono varchar2, fechaNac Date) return self as result,
    member function edad return number -- declaramos una función o método dentro
    del tipo de datos empleado_t que se llama edad y devuelve number
);

```

```

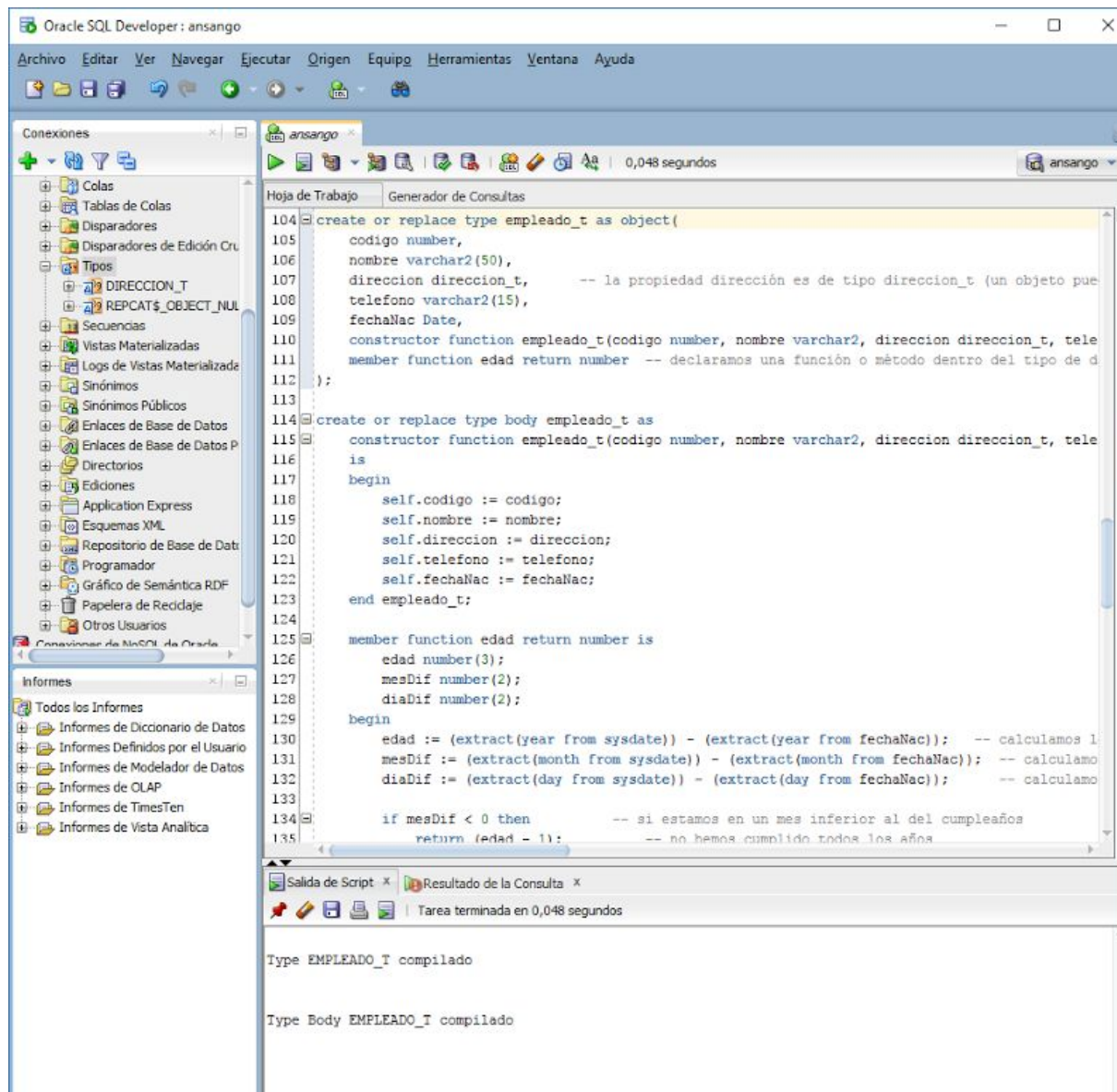
create or replace type body empleado_t as
    constructor function empleado_t(codigo number, nombre varchar2, direccion
direccion_t, telefono varchar2, fechaNac Date) return self as result
    is
    begin
        self.codigo := codigo;
        self.nombre := nombre;
        self.direccion := direccion;
        self.telefono := telefono;
        self.fechaNac := fechaNac;
        return;
    end empleado_t;

    member function edad return number is
        edad number(3);
        mesDif number(2);
        diaDif number(2);
    begin
        edad := (extract(year from sysdate)) - (extract(year from fechaNac));
        mesDif := (extract(month from sysdate)) - (extract(month from fechaNac));
        diaDif := (extract(day from sysdate)) - (extract(day from fechaNac));

        if mesDif < 0 then
            return (edad - 1);
        elsif mesDif > 0 then
            return edad;
        elsif mesDif = 0 then
            if diaDif < 0 then
                return (edad - 1);
            elsif diaDif >= 0 then
                return edad;
            end if;
        end if;
    end edad;
end;

```





## Ejecución

declare

```

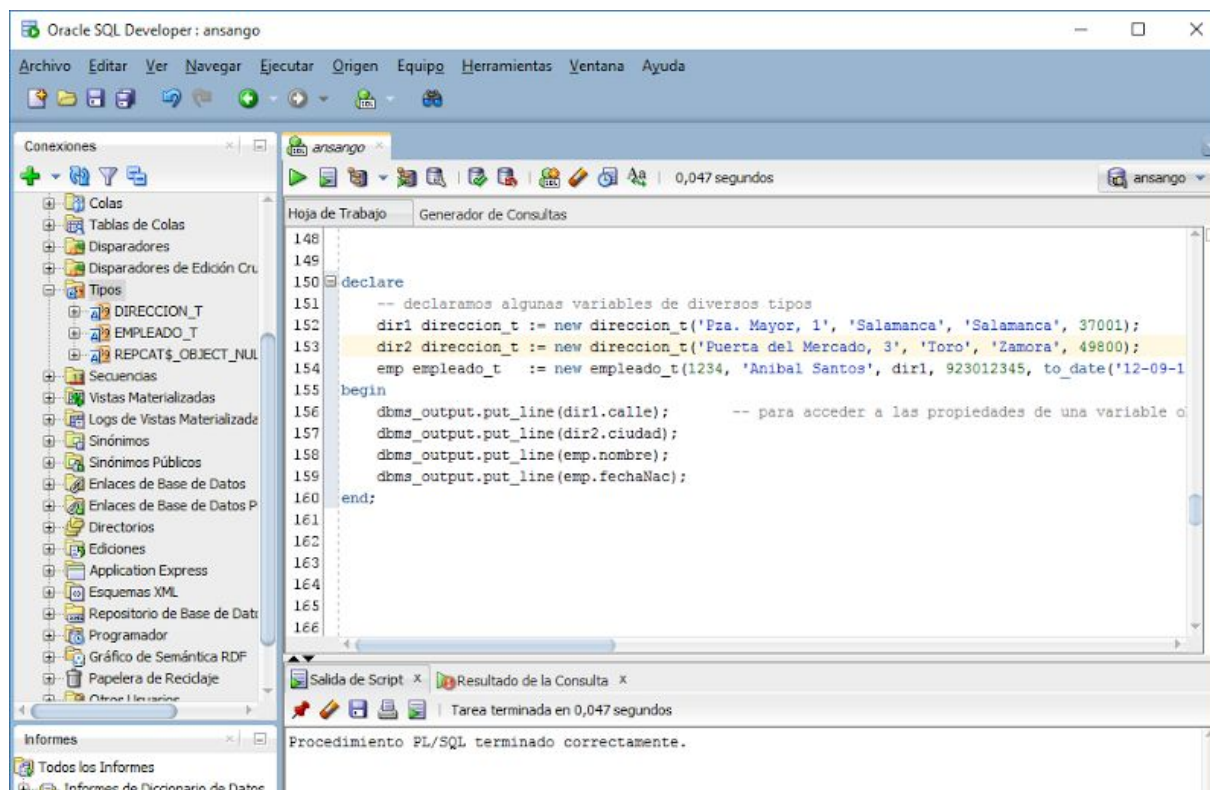
-- declaramos algunas variables de diversos tipos
dir1 direccion_t := new direccion_t('Pza. Mayor, 1', 'Salamanca', 'Salamanca',
37001); -- ejecutamos el constructor de la clase (objeto) direccion_t para
crear un nuevo objeto de ese tipo que habíamos declarado; Y guardamos este objeto
en la variable correspondiente
dir2 direccion_t := new direccion_t('Puerta del Mercado, 3', 'Toro', 'Zamora',
49800);
emp empleado_t := new empleado_t(1234, 'Anibal Santos', dir1, 923012345,
to_date('12-09-1991', 'dd/mm/yyyy'));
begin

```

```

        dbms_output.put_line(dir1.calle);          -- para acceder a las propiedades de
una variable objeto, ponemos el nombre de la misma seguida de . (punto) y el
nombre de la propiedad
        dbms_output.put_line(dir2.ciudad);
        dbms_output.put_line(emp.nombre);
        dbms_output.put_line(emp.fechaNac);
end;

```



### 3

-- creamos un nuevo tipo de datos llamado "coleccion\_empleados" como un array/vector de objetos de tipo empleado\_t

```

create or replace type coleccion_empleados as varray(10) of empleado_t;

```

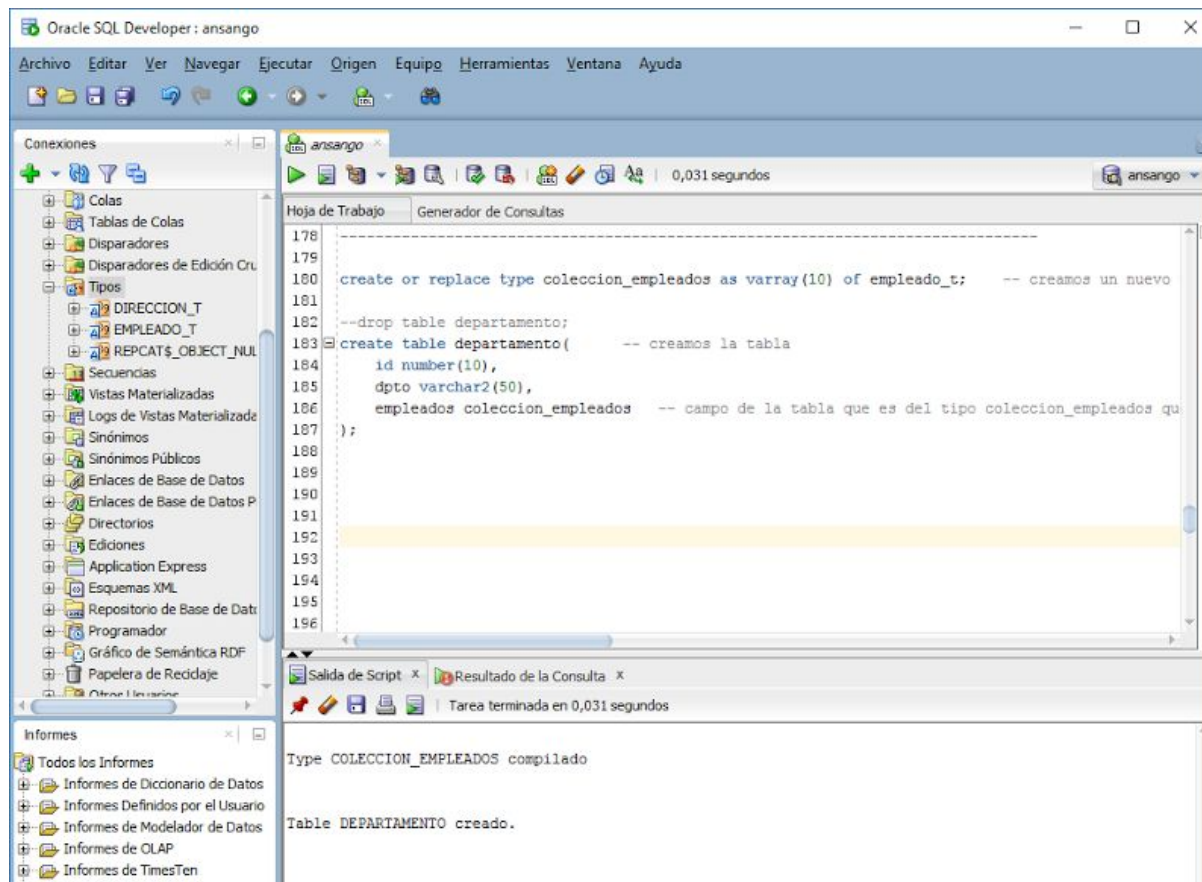
-- creamos la tabla

```

create table departamento(
    id number(10),
    dpto varchar2(50),
    -- campo de la tabla que es del tipo coleccion_empleados que acabamos de crear
    empleados coleccion_empleados
);

```





declare

```

    dir1 direccion_t;
    dir2 direccion_t;
    dir3 direccion_t;
    dir4 direccion_t;
    emp1 empleado_t;
    emp2 empleado_t;
    emp3 empleado_t;
    emp4 empleado_t;

```

begin

```

dir1 := new direccion_t('Pza. Mayor, 1', 'Salamanca', 'Salamanca', 37001);
dir2 := new direccion_t('Puerta del Mercado, 3', 'Toro', 'Zamora', 49800);
dir3 := new direccion_t('Rua Mayor, 13', 'Salamanca', 'Salamanca', 37002);
dir4 := new direccion_t('Corredera', 'Toro', 'Zamora', 49800);

```

```

emp1 := new empleado_t(1234, 'Anibal Santos', dir1, 923012345,
to_date('12-09-1991', 'dd/mm/yyyy'));
emp2 := new empleado_t(5678, 'Perico Palotes', dir2, 980963852,
to_date('05-02-1986', 'dd/mm/yyyy'));
emp3 := new empleado_t(9632, 'Fulanito de Tal', dir3, 923951470,
to_date('7-05-1995', 'dd/mm/yyyy'));
emp4 := new empleado_t(1475, 'Perantanito de Cual', dir4, 980654823,
to_date('25-11-2000', 'dd/mm/yyyy'));

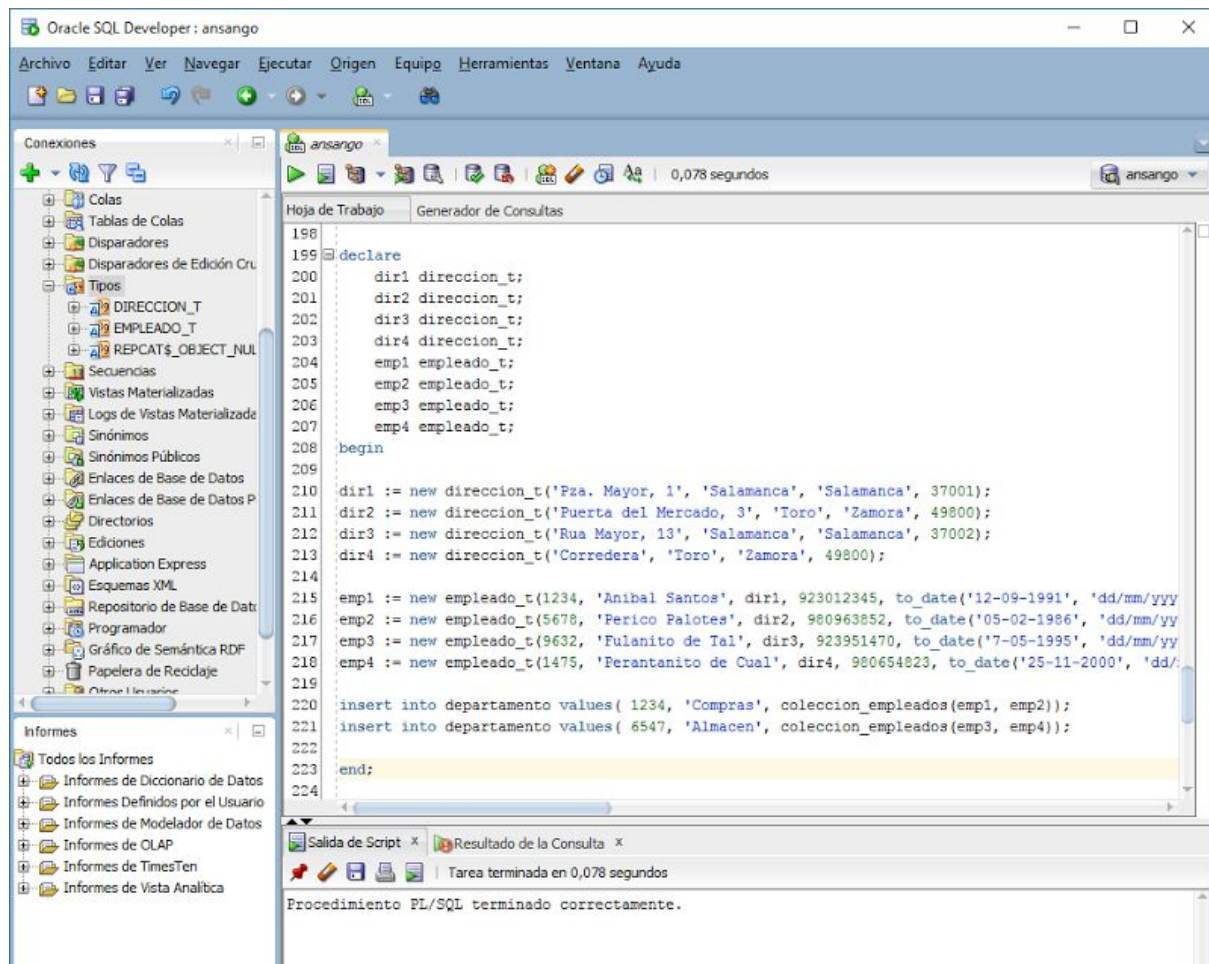
```

```

insert into departamento values( 1234, 'Compras', coleccion_empleados(emp1,
emp2));
insert into departamento values( 6547, 'Almacen', coleccion_empleados(emp3,
emp4));

end;

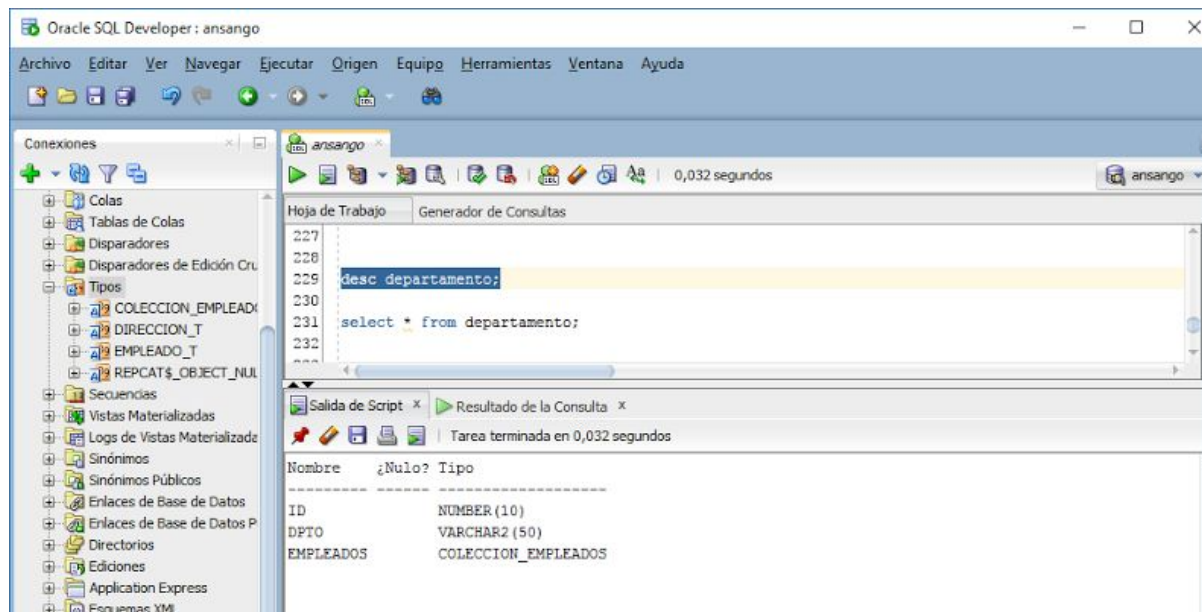
```



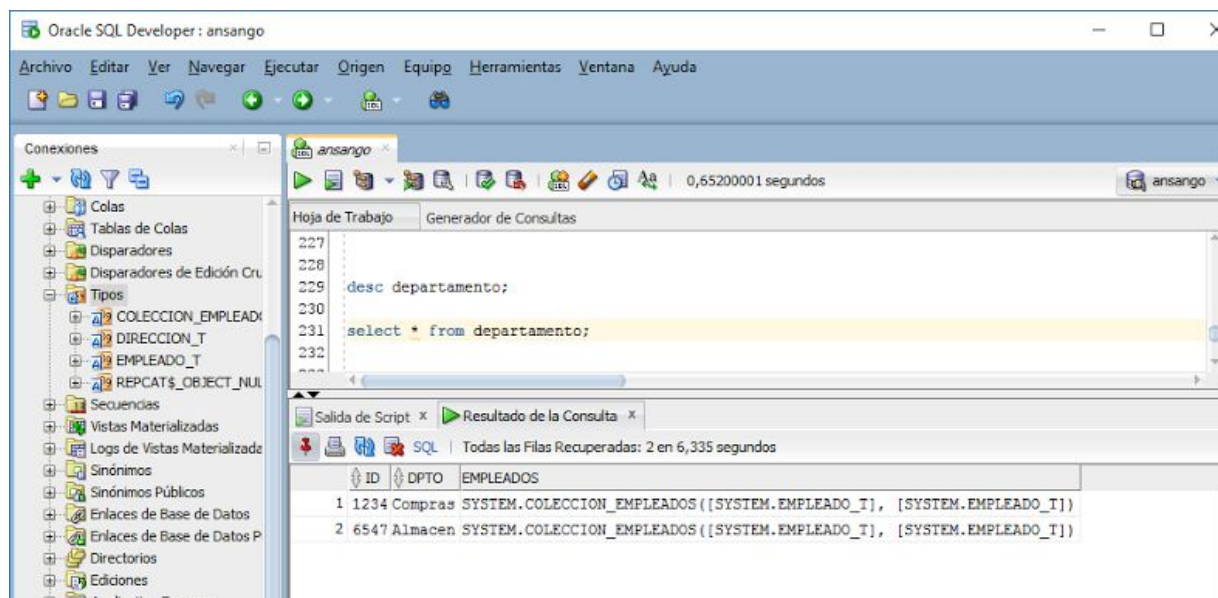
```

desc departamento;

```



select \* from departamento;



4

create or replace type empleados\_2017 as table of empleado\_t;

create table tabla\_empleados\_2017(  
id number,

```

        anio number default 2017,
        empleado empleados_2017
    )
    nested table empleado store as emp_2017;

```

```

create table empleados_current_year (
    empleado empleado_t
)

```

