

# PRÁCTICA 3 MÉTODOS

Se desea modelar un objeto triángulo para almacenar sus características (la base y la altura) y almacenar en la BBDD cientos

de triángulos pudiendo calcular el área de cada triángulo, se podría

crear el siguiente esquema:

- Un tipo llamado tipo\_triángulo con los atributos base number, altura number y una función llamada area return number.

(el área de un triángulo se calcula como  $\text{base} \times \text{altura} / 2$ )

- Crear el cuerpo body para dicho tipo.

- Crear una tabla relacional llamada triangulos para almacenar los triángulos, con las columnas Id number y triangulo de tipo\_triángulo.

- Insertar dos triángulos con los siguientes valores

Id=1, base=5, altura=5

Id=2, base=10, altura=10

- Listar todos los triángulos.

- Crear un bloque PL/SQL para recorrer la tabla triángulos e invocar al método área, de forma que en el resultado obtengamos esto:

```
el triangulo con id:1
```

```
con base: 5
```

```
y altura: 6
```

```
tiene un area de: 15
```

```
el triangulo con id:2
```

```
con base: 10
```

```
y altura: 10
```

```
tiene un area de: 50
```

```

1 drop table triangulos cascade constraints;
2 create or replace type tipo_triangulo as object(
3     base number,
4     altura number,
5     member function area return number);
6 /
7 create or replace type body tipo_triangulo as
8     member function area return number is
9         a number;
10        begin
11            a:=(base*altura)/2;
12            return a;
13        END;
14    END;
15 /
16 create table triangulos (id number, triangulo tipo_triangulo);
17 insert into triangulos values(1, tipo_triangulo(5,6));
18 insert into triangulos values(2, tipo_triangulo(10,10));
19 select * from triangulos;
20 declare
21 t tipo_triangulo;
22 begin
23 for i in (select * from triangulos) loop
24     t:=i.triangulo;
25     DBMS_OUTPUT.PUT_LINE('el triangulo con id: '||i.id);
26     DBMS_OUTPUT.PUT_LINE('con base: '||t.base);
27     DBMS_OUTPUT.PUT_LINE('y altura: '||t.altura);
28     DBMS_OUTPUT.PUT_LINE('tiene un area de: '||t.area);
29 end loop;
30 end;
31 /

```