

Examen 111Mil – Bicerleteria

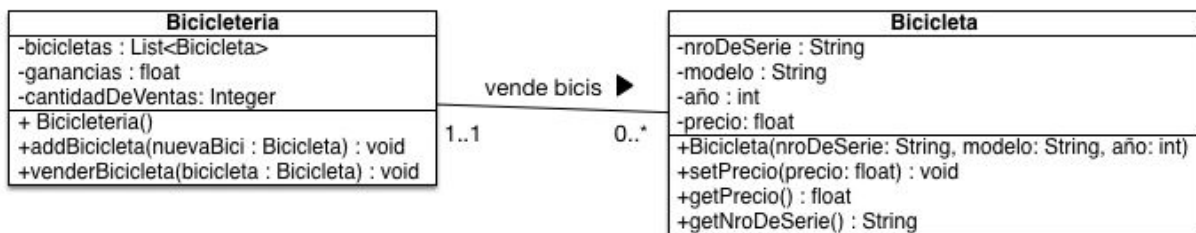
Carlos tiene un negocio especializado en la venta de bicicletas. Hasta el momento registra todas sus operaciones en un cuaderno. Sin embargo, dado el éxito de su empresa, cada día le es más difícil poder tener control sobre las ventas de sus bicicletas y las ganancias que registra. Por esta razón, Carlos necesita desarrollar un sistema que lo ayude en su negocio.

Como sabe que cursamos el programa 111Mil se puso en contacto con nosotros para que lo ayudemos a construir el sistema, que en esta primera etapa del proyecto deberá registrar las bicicletas en venta y las ventas registradas.

Ejercicio 1. Implementar desde el diagrama de clases

Dado que Carlos es experto en modelado de software, ha creado un diagrama de clases UML para indicarnos lo que se desea. Otros programadores comenzaron con la codificación, pero nos piden que completemos el código para:

- Declarar las variables bicicletas, ganancias y cantidadDeVentas de la clase Bicerleteria.
- Implementar el método addBicicleta de la clase Bicerleteria. El mismo deberá agregar una nueva bicicleta en la lista bicicletas.
- Implementar los métodos getPrecio y setPrecio de la clase Bicicleta.



```
public class Bicicleteria {
    private List<Bicicleta> bicicletas;
    private float ganancias;
    private Integer cantidadDeVentas;

    public Bicicleteria() {
        bicicletas = new ArrayList<Bicicleta>();
        cantidadDeVentas = 0;
        ganancias = 0;
    }
    public void venderBicicleta(Bicicleta bicicleta){
        bicicletas.remove(bicicleta);
        cantidadDeVentas++;
        ganancias = ganancias + bicicleta.getPrecio();
    }
    public void addBicicleta(Bicicleta nuevaBici){
        bicicletas.add(nuevaBici);
    }
}
```

```
public class Bicicleta {
    private String nroDeSerie;
    private String modelo;
    private int año;
    private float precio;
```

```

public Bicicleta(String nroDeSerie, String modelo, int año) {
    this.nroDeSerie = nroDeSerie;
    this.modelo = modelo;
    this.año = año;
}
public void setPrecio(float precio) {
    this.precio = precio;
}
public float getPrecio() {
    return precio;
}
public String getNroDeSerie() {
    return nroDeSerie;
}
}

```

Ejercicio 2. Implementar un método a partir de un enunciado

Programar en Java el código del método *buscarBicicleta* en la clase *Bicicleteria*. El mismo recibirá por parámetros una variable llamada *nroDeSerie* de tipo String y retornará la bicicleta con dicho *nroDeSerie*. En caso de no encontrarlo deberá retornar null.

```

public Bicicleta buscarBicicleta(String nroDeSerie){
    for (Bicicleta bici : bicicletas) {
        if(bici.getNroDeSerie().equals(nroDeSerie))
            return bici;
    }
    return null;
}

```

Ejercicio 3. Interpretación de código

Un desarrollador implementó el siguiente método en la clase *Bicicleteria* pero no usó nombres representativos. Indique cuál de las siguientes opciones explica lo que hace el código:

- A. Calcula el total de bicicletas de la bicicletería
- B. Calcula el precio total de las bicicletas de la bicicletería
- C. Calcula el promedio del precio de las bicicletas de la bicicletería
- D. Ninguna de las anteriores.

La respuesta correcta es: B

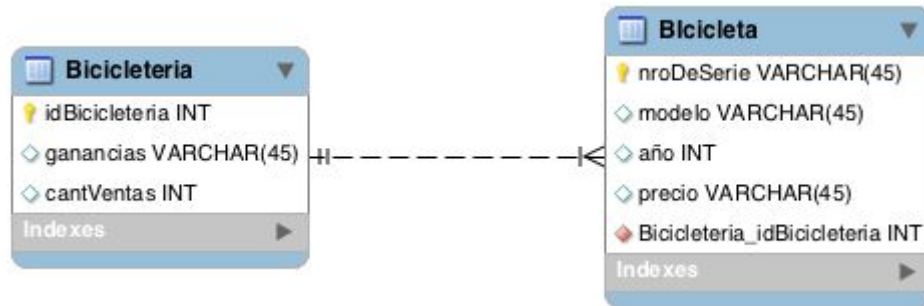
```

public float calcularPrecioTotalDeBicicletasEnVenta(){
    float ret = 0;
    for (int i = 0; i < bicicletas.size(); i++) {
        ret = ret + bicicletas.get(i).getPrecio();
    }
    return ret;
}

```

Ejercicio 4. Interpretación de DER.

A partir del DER, responda Verdadero (V) o Falso (F) a las siguientes afirmaciones (asignar a cada afirmación un valor V o F. Todas las afirmaciones tienen que tener un valor asignado. Puede haber varios V y varios F):



Entre las tablas Bicicleteria y Bicicleta existe una relación 1 a 1	F
Entre las tablas Bicicleteria y Bicicleta existe una relación 1 a N	V
Entre las tablas Bicicleteria y Bicicleta existe una relación N a N	F
La clave primaria de la tabla Bicicleta es nroDeSerie	V
La clave foranea de la tabla Bicicleta es nroDeSerie	F
La tabla Concesionaria no tiene clave primaria	F
La tabla Concesionaria no tiene clave foránea	V

Ejercicio 5. Consultas SQL

Dado el diagrama de entidad-relación presentado en el ejercicio anterior, escriba una consulta SQL que liste la cantidad de ventas y los números de serie de las bicicletas en venta de la bicicletería cuyo idBicicleteria = 1. Ordene los resultados de acuerdo a la cantidad de ventas en forma descendente.

```
SELECT c.cantVentas, a.nroDeSerie FROM Bicicleteria c INNER JOIN Auto a ON
(c.idBicicleteria=a.Bicicleteria_idBicicleteria) WHERE c.idBicicleteria=1 ORDER BY c.cantVentas DESC
```