



## Reto 5 - Estadio

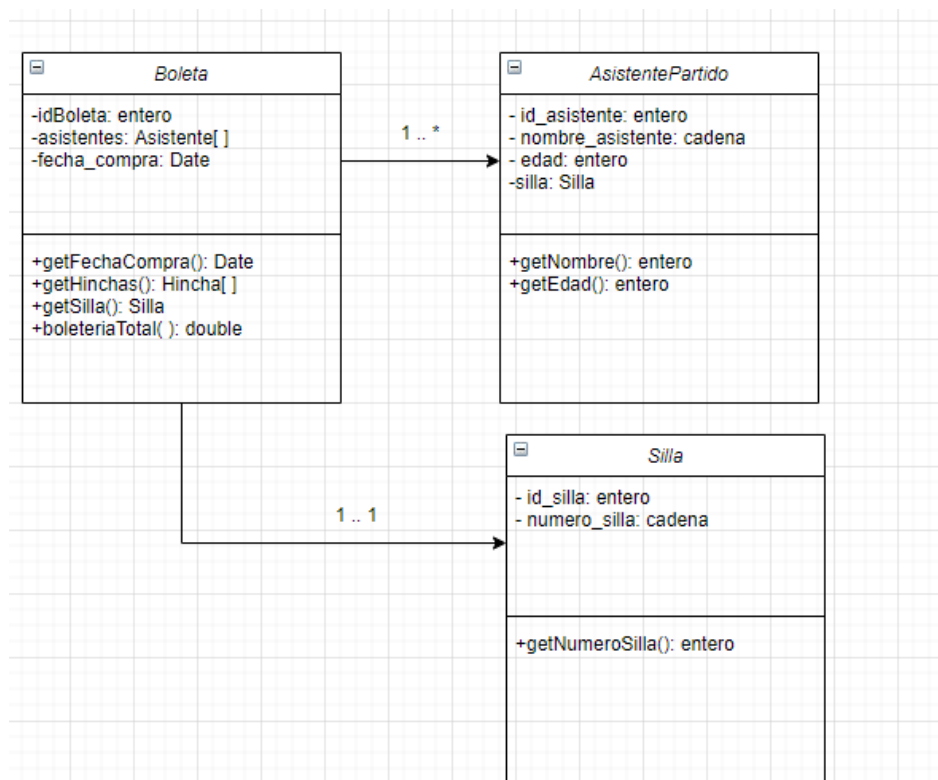
### Objetivo

El objetivo de este reto es que el estudiante reconozca y aplique los elementos básicos del paradigma de la programación orientada a objetos en un escenario abstraído de la cotidianidad.

### Contexto

El estadio Nemesio Girardot requiere un sistema que agilice el sistema de cambio a la hora de ingresar al recinto deportivo, permitiendo un ingreso más rápido y bioseguro. Por ende, se necesita un programa que le ayude al cajero de la taquilla del estadio, para realizar la entrega del cambio de una forma más adecuada.

Considere el siguiente diagrama de clases para la implementación de las clases necesarias **Boleta.java**, **AsistentePartido.java**, **Silla.java**.





**NOTA:** Las clases deben llamarse **Boleta.java**, **AsistentePartido.java** y **Silla.java**

## Reto

Implemente una función nueva llamada `estadioDevuelta()`, la cual recibirá como parámetro el total a pagar por la boleta de entrada al estadio adquiridos por parte del hincha (**ver Reto 3**). Por otra parte, tener en cuenta que la clase `Boleta` tendrá un atributo adicional nuevo, el cual se llamará `valor_cancelado`, en el cual se almacenará el valor con el cual el cliente está pagando por la boleta.

La función debe retornar la cantidad de billetes y monedas que el cajero debe emplear para entregar el cambio, siguiendo estas condiciones:

- a) En caso de que el número de la silla esté en el rango de 1 al 100, se le debe devolver al usuario con billetes de 5000 y billetes de 1000.
- b) Por otra parte, si el número de la silla está en el rango de 101 al 200, se le debe devolver al usuario con billetes de 10000 y billetes de 2000.
- c) Por otra parte, si el número de la silla está en el rango de 201 al 300, se le debe devolver al usuario con billetes de 20000 y billetes de 5000.

Adicionalmente, use la siguiente imagen como referencia para la construcción de las clases con sus atributos y métodos necesarios.



```
import java.sql.Date;

public class Boleta {
    private int id_boleta;
    private AsistentePartido[] asistentes;
    private Date fecha_compra;
    private int valor_cancelado;

    public Boleta(int id_boleta, AsistentePartido[] asistentes, Date fecha_compra, int valor_cancelado) {
    }

    public double boleterisfotal() {
        double suma=0;

        return suma;
    }

    public String estadoDevuelta(int boletaTotal) {
        String mensaje = "";

        return mensaje;
    }

    public Boleta() {
        // 1000 Auto-generated constructor stub
    }

    int getId_boleta() {
        return id_boleta;
    }

    void setId_boleta(int id_boleta) {
        this.id_boleta = id_boleta;
    }

    AsistentePartido[] getAsistentes() {
        return asistentes;
    }

    void setAsistentes(AsistentePartido[] asistentes) {
        this.asistentes = asistentes;
    }

    Date getFecha_compra() {
        return fecha_compra;
    }

    void setFecha_compra(Date fecha_compra) {
        this.fecha_compra = fecha_compra;
    }

    int getValor_cancelado() {
        return valor_cancelado;
    }

    void setValor_cancelado(int valor_cancelado) {
        this.valor_cancelado = valor_cancelado;
    }

    public class AsistentePartido {
        private int id_asistente;
        private String nombre_asistente;
        private int edad;
        private Silla silla;

        public AsistentePartido(int id_asistente, String nombre_asistente, int edad, Silla silla) {
            // 1000 Auto-generated constructor stub
        }

        public AsistentePartido() {
            // 1000 Auto-generated constructor stub
        }

        String getNombre_asistente() {
            return nombre_asistente;
        }

        void setNombre_asistente(String nombre_asistente) {
            this.nombre_asistente = nombre_asistente;
        }

        int getEdad() {
            return edad;
        }

        void setEdad(int edad) {
            this.edad = edad;
        }

        Silla getSilla() {
            return silla;
        }

        void setSilla(Silla silla) {
            this.silla = silla;
        }

        int getId_asistente() {
            return id_asistente;
        }

        void setId_asistente(int id_asistente) {
            this.id_asistente = id_asistente;
        }
    }
}
```



```
public class Silla {  
    private int id_silla;  
    private int numero;  
  
    public Silla(int id_silla, int numero) {  
  
    }  
  
    public Silla() {  
        // TODO Auto-generated constructor stub  
    }  
  
    int getId_silla() {  
        return id_silla;  
    }  
  
    void setId_silla(int id_silla) {  
        this.id_silla = id_silla;  
    }  
  
    int getNumero() {  
        return numero;  
    }  
  
    void setNumero(int numero) {  
        this.numero = numero;  
    }  
}
```



## Casos de prueba

Finalmente, para verificar el funcionamiento del programa se sugiere considerar los siguientes casos de prueba:

# CASO DE PRUEBA	DATO DE ENTRADA	SALIDA ESPERADA								
1	<div><div><div>Boleta</div><div>id_boleta: 0 asistentes: Asistente fecha_compra: 7/21/2021 valor_cancelado: 158.000</div><table><tr><th>ID</th><th>Nombre</th><th>Edad</th><th>Silla</th></tr><tr><td>0</td><td>Juan Sebastián Zuluaga</td><td>29</td><td>Id_silla: 0 Numero: 44</td></tr></table></div><div>En este caso, el valor total a pagar por la función implementada en reto 3 debe ser 120.000</div></div>	ID	Nombre	Edad	Silla	0	Juan Sebastián Zuluaga	29	Id_silla: 0 Numero: 44	El cajero de taquilla debe devolverle al hincha 7 billetes de 5000 y 3 billetes de 1000
ID	Nombre	Edad	Silla							
0	Juan Sebastián Zuluaga	29	Id_silla: 0 Numero: 44							
2	<div><div><div>Boleta</div><div>id_boleta: 1 asistentes: Asistente fecha_compra: 7/21/2021 valor_cancelado: 308.000</div><table><tr><th>ID</th><th>Nombre</th><th>Edad</th><th>Silla</th></tr><tr><td>0</td><td>Pedro Hernandez</td><td>23</td><td>Id_silla: 0 Numero: 122</td></tr></table></div><div>En este caso, el valor total a pagar por la función implementada en reto 3 debe ser 150.000</div></div>	ID	Nombre	Edad	Silla	0	Pedro Hernandez	23	Id_silla: 0 Numero: 122	El cajero de taquilla debe devolverle al hincha 15 billetes de 10000 y 4 billetes de 2000
ID	Nombre	Edad	Silla							
0	Pedro Hernandez	23	Id_silla: 0 Numero: 122							

3

Boleta			
id_boleta: 1			
asistentes: Asistente			
fecha_compra: 7/21/2021			
valor_cancelado: 485.000			
ID	Nombre	Edad	Silla
0	Armando Gonzales	29	Id_silla: 0 Numero: 210

En este caso, el valor total a pagar por la función implementada en reto 3 debe ser 200.000

El cajero de taquilla debe devolverle al hincha 14 billetes de 20000 y 1 billetes de 5000

### Entrega:

1. Suba a la plataforma un archivo con el nombre de **Boleta.java, AsistentePartido.java y Silla.java** este nombre debe de respetarse, dado que, si no se nombre de dicha manera no se tendrá en cuenta para la calificación del reto.
2. **Importante:** Los métodos deben de llamarse **exactamente igual** a como se muestra en el ejemplo de la estructura del código.
3. **Importante:** Las salidas deben ser tal cual se muestran en los casos de pruebas. De lo contrario, el sistema no lo reconocerá.