





Reto 5 – Renta de Motos Acuáticas

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 departamento de analítica de datos de la empresa le comenta que la funcionalidad implementada en el reto anterior les ha dado la información necesaria para comenzar a planificar e implementar diferentes estrategias de marketing para atraer usuarios. Para comenzar con la implementación de estas estrategias quieren empezar por una que sea simple, como descuentos, y les permita analizar si se obtienen los resultados esperados para continuar con estrategias más complejas y que requieren de mayor inversión y esfuerzo.

Dicho esto, le solicitan que implemente una nueva funcionalidad que aplique un pequeño descuento a todos los alquileres que cumplan con ciertas condiciones relacionadas con la edad del cliente.







Reto:

MotoAcuatica	Persona	
- Id: String - Modelo: String	- Cedula: String - Edad: int	
- Marca: String	- Nombre: String	
+ getId(): int	+ getCedula(): String	
+ getModelo(): String + getMarca(): String	+ getEdad(): int +getNombre(): String	

Alquiler

- ld: int
- Cliente: PersonaMoto: MotoAcuatica
- + getCliente(): Cliente
- + TerminarAlquiler(int cantidadHoras, boolean descuento): double
- + AnalizarAlquileres(Alquiler[] historial): HashMap <String, Integer>
- + ToString(): String
- ObtenerDescuento(): int
- 1. Modifique la clase **Alquiler** como se muestra en el diagrama, implemente la función **ObtenerDescuento**, que retorna un número entero.
- 2. La función **ObtenerDescuento** retorna un valor en función a la edad del cliente, de acuerdo a la siguiente tabla:

Rango Edad	Valor Descuento (%)	
Menos de 18	0	
18 – 40	2	
41 o más	5	







3. Refactorice la función **TerminarAlquiler**, note que ahora toma dos parámetros en lugar de uno, el nuevo parámetro es un **boolean** que indica si se aplicará un descuento a la hora de calcular el costo. En caso de que sea verdadero al costo total del alquiler se le aplicará el descuento que indique la función **ObtenerDescuento**, en caso de ser falso no se aplica ningún descuento.







Casos de Prueba:

Caso de Prueba	Datos de Entrada	Salida Esperada
1. TerminarAlquiler(5, false)	Alquiler	250000
	- ld: 123 - Cliente: 10004830 Héctor 20 - Moto: 3 Yamaha VXCruiser	
2. TerminarAlquiler(5, true)	Alquiler	245000
	- ld: 123 - Cliente: 10004830 Héctor 20 - Moto: 3 Yamaha VXCruiser	
3. TerminarAlquiler(5, false)	Alquiler	500000
	- Id: 125 - Cliente: 10008612 Carlos 14 - Moto: 3 Yamaha VXCruiser	
4. TerminarAlquiler(5, true)	Alquiler	500000
	- Id: 125 - Cliente: 10008612 Carlos 14 - Moto: 3 Yamaha VXCruiser	
5. TerminarAlquiler(2, true)	Alquiler	120000
	- Id: 92 - Cliente: 522356 Israel 53 - Moto: 2 Sea-Doo RXT	
6. TerminarAlquiler(2, false)	Alquiler	114000
	- Id: 92 - Cliente: 522356 Israel 53 - Moto: 2 Sea-Doo RXT	







ENTREGA:

- 1. Los archivos que suba a la plataforma para su calificación deben llamarse **exactamente** *Persona.java*, *MotoAcuatica.java* y *Alquiler.java*, de lo contrario no se calificará.
- 2. Los nombres de las clases, miembros dato y funciones deben llamarse exactamente como se muestran en los diagramas mostrados al comienzo del reto, las firmas de sus clases deben ser cómo se muestra en las siguientes imágenes:

```
public class Persona {
    private String Cedula;
    private int Edad;
    private String Nombre;

public Persona(String cedula, int edad, String nombre) {
        //Implementación
    }

public String getCedula() {
        //Implementación
    }

public int getEdad() {
        //Implementación
    }

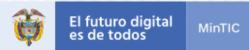
public String getNombre() {
        //Implementación
    }

public String getNombre() {
        //Implementación
    }
}
```













```
import java.util.HashMap;
public class Alquiler {
   private int Id;
   private Persona Cliente;
   private MotoAcuatica Moto;
   public Alquiler(int id, Persona cliente, MotoAcuatica moto) {
   public Persona getCliente() {
   public double TerminarAlquiler(int cantidadHoras, boolean descuento){
   public static HashMap<String, Integer> AnalizarAlquileres(Alquiler[] historial){
       //Implementación
   @Override
   public String toString() {
   private int ObtenerDescuento(){
       //Implementación
```