



Reto 3 – 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:

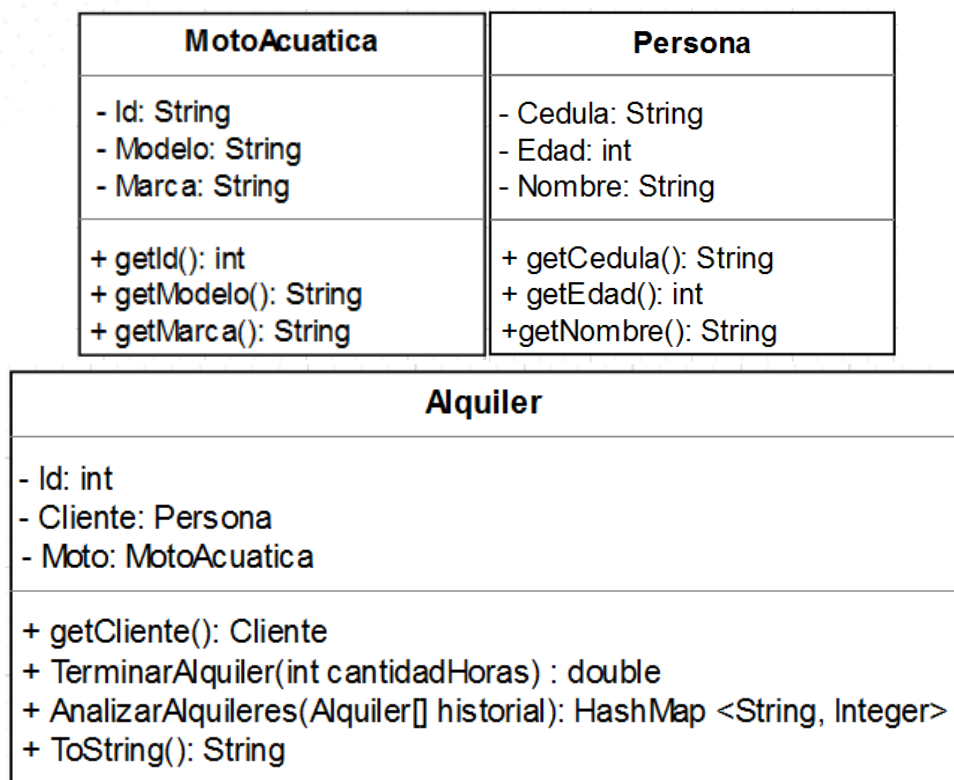
La rentabilidad de las empresas que rentan motos acuáticas se ha disparado desde que implementaron su sistema para cobrar el alquiler de las motos por un tiempo indefinido. Las ganancias han sido tales que han decidido abrir un nuevo departamento de analítica de datos en la empresa para que se encargue de analizar y optimizar los modelos de negocio que se planteen.

Para el primer proyecto de este departamento, se le ha solicitado que modifique el sistema implementado en el reto anterior con una nueva funcionalidad. Las empresas desean conocer cuál es el rango de edad que más renta motos acuáticas, con el fin de crear estrategias de marketing que atraigan más a esta población.



Reto:

1. Implemente las clases que se muestran en el siguiente diagrama:



2. Note que las clases **Persona** y **MotoAcuatica** son iguales a las clases implementadas en el reto anterior. Se recomienda reutilizarlas.



3. La clase **Alquiler** también es igual a la implementada en el reto anterior, pero ahora implementa un **getter ObtenerCliente** que retorna el miembro dato **Persona** de la instancia.
4. **Alquiler** también implementa una nueva función estática **AnalizarAlquileres**, que toma como argumento un arreglo de tipo **Alquiler** que representa un historial de alquileres. La función retorna un **HashMap<String, Integer>**. Recuerde que un [HashMap](#) es una estructura que nos permite representar una asociación de tipo <Llave,Valor>. Para este reto la **Llave** sería una descripción de un rango de edad (String) y el **Valor** es la cantidad de personas encontradas dentro de la respectiva población. Podría representarse como la siguiente tabla:

Rango Edad	Población
18 o menos	25
18 – 30	42
31 - 40	45
41 - 50	35
51 o más	22

La función **AnalizarAlquileres** debe recorrer el historial de alquileres y contar la cantidad de personas encontrada en cada población. Finalmente retorna un **HashMap** como el mostrado anteriormente con los respectivos resultados.

Importante: Para la calificación del reto las **Llaves** (Strings) que componen el HashMap deben ser exactamente iguales a como se muestra en el ejemplo. Preste atención a cómo están escritos, tomando en cuenta minúsculas, mayúsculas, espacios y signos de puntuación.



Casos de Prueba:

Para un historial de alquileres como el siguiente:

IdAlquiler	IdMotoAcuatica	Persona
1	015	Persona
		Cedula: 2155 Edad: 17 Nombre: Carlito
2	014	Persona
		Cedula: 73251 Edad: 19 Nombre: Juan
3	005	Persona
		Cedula: 63743 Edad: 25 Nombre: Valeria
4	021	Persona
		Cedula: 6785 Edad: 33 Nombre: Jose
5	018	Persona
		Cedula: 6785 Edad: 21 Nombre: Miguel
6	014	Persona
		Cedula: 514231 Edad: 48 Nombre: Maria

Nota: Por motivos prácticos solo se muestra un miembro dato de las clases **Alquiler** y **MotoAcuatica** pues para este reto no se hace uso de los datos de estas instancias.



Se espera que el resultado de la función **AnalizarAlquileres** sea un **HashMap** con los siguientes pares llave-valor:

Rango Edad	Población
18 o menos	1
18 – 30	3
31 - 40	1
41 - 50	1
51 o más	0

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 class MotoAcuatica {  
    private String Id;  
    private String Modelo;  
    private String Marca;  
  
    public MotoAcuatica(String id, String modelo, String marca) {  
        //Implementación  
    }  
  
    public String getId() {  
        //Implementación  
    }  
  
    public String getModelo() {  
        //Implementación  
    }  
  
    public String getMarca() {  
        //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) {
        //Implementación
    }

    public Persona getCliente() {
        //Implementación
    }

    public double TerminarAlquiler(int cantidadHoras){
        //Implementación
    }

    public static HashMap<String, Integer> AnalizarAlquileres(Alquiler[] historial){
        //Implementación
    }

    @Override
    public String toString() {
        //Implementación
    }
}
```