



Universidad Diego Portales

## Laboratorio 2 Programación avanzada

Profesor:

Juan Duarte

Ayudantes:

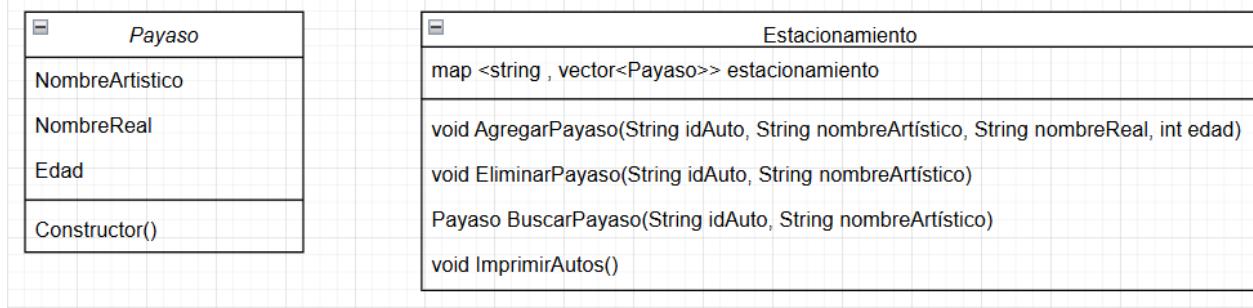
Diego Mena  
Jonas Oviedo

## Instrucciones

Tendrán los siguiente 115 minutos para desarrollar el siguiente laboratorio, el laboratorio será **Individual**, cuenta con 1 ejercicio y deben entregarlo por canvas antes de las **17:20**. Para este laboratorio solo se deberá usar **materia vista en clase o ayudantía** A los códigos se les hará una revisión exhaustiva, en caso de ser detectado **ChatGPT** se evaluará con un 1 sin importar que tan bueno este el código, a continuación los dejo con el problema, suerte :).

## 1. Estacionamiento de Autos de Payasos

El payaso Bob, encargado de organizar un paseo familiar para todos los payasos, necesita llevar un registro de los autos de payasos que se estacionarán. Como Bob no tiene experiencia en la gestión de listas o en la programación, te ha solicitado ayuda para desarrollar un programa que permita gestionar los autos de payasos de forma eficiente. Para lograr esto, deberás programar tres clases: **Payaso** y **Estacionamiento**, siguiendo las especificaciones que se muestran en el siguiente diagrama de clases:



### Clase Payaso

La clase **Payaso** debe modelar la información básica de cada payaso:

#### ■ Atributos:

- **Nombre Artístico**: El nombre con el que el payaso se presenta al público (de tipo `String`).
- **Nombre Real**: El nombre real del payaso (de tipo `String`).
- **Edad**: La edad del payaso (de tipo `int`).

#### ■ Constructor:

- El constructor debe inicializar los atributos con los valores proporcionados por parámetros.

#### ■ GETS

- Ademas debe incluir los gets de **NombreArtistico**, **NombreReal** y **Edad**

### Clase Estacionamiento

La clase **Estacionamiento** es la encargada de gestionar los autos de payasos mediante un mapa:

#### ■ Atributos:

- **Mapa <String, Vector<Payaso> > AutosPorSección**: Donde la clave es el identificador del auto y el valor es un vector de payasos. (ejemplo: "Auto1", "Auto2", "Auto3", etc.)

■ Métodos:

- `void AgregarPayaso(String idAuto, String nombreArtístico, String nombreReal, int edad)`: Agrega un payaso a un auto específico. Crea el auto si no existe.
- `void EliminarPayaso(String idAuto, String nombreArtístico)`: Elimina un payaso de un auto identificado por su nombre artístico.
- `Payaso BuscarPayaso(String idAuto, String nombreArtístico)`: Busca un payaso en un auto y retornar dicho payaso, ademas de imprimir sus datos en pantalla.
- `void ImprimirAutos()`: Imprime todos los autos y sus payasos, mostrando los datos de los payasos junto al auto al que pertenecen.