**😃 Leé por lo menos dos veces el enunciado antes de resolverlo.**

# Enunciado

La dirección general de tránsito de la Ciudad de Buenos Aires nos encarga un sistema para emitir estadísticas sobre los reclamos sobre infracciones de tránsito que procesan a diario.

Cada uno de los reclamos atendidos día a día se almacenan en orden de llegada, cada una contiene información sobre el dni del reclamante, un indicador que muestra si el reclamo fue aceptado o no y la infracción reclamada.

Para esta versión, solo se piden estadísticas de dos tipos de infracciones, Exceso De Velocidad y por Mal Estacionamiento. Las infracciones son caducables, debiendo indicar si la están caducadas o no. Además, cada infracción debe poder indicar su importe, que se calcula en base al valor de la unidad de infracciones (que puede ser actualizado de ser necesario) y a una constante propia de cada tipo de infracción. Todas las infracciones poseen la fecha en la que fueron labradas, del tipo **Fecha\***., Las infracciones por Exceso de velocidad nunca caducan, pero las de mal estacionamiento caducan tras 365 días.

El valor de las infracciones de Exceso de velocidad es 10 veces el valor de la unidad, mientras que el valor de las infracciones por Mal Estacionamiento es de 7 veces el valor de la unidad.

***\* Asumir que la clase Fecha fue provista. No debe ser desarrollada pero sí plasmada en el UML junto a sus dependencias. Toda fecha será capaz de indicar su mes (como enumerado) y cuántos días transcurrieron desde tal hasta el día de hoy (entero).***

Basado en el enunciado descripto, realizá:

1. El diagrama de clases que lo modelice, con sus relaciones, atributos y métodos.
2. El método **informeReclamadasPorMesTipo()** que debe **devolver** (no mostrar por consola) la cantidad de infracciones reclamadas por tipo de infracción y mes.
3. El método **caducada()** de las clases que considere necesario, que debe indicar si la infracción está caducada o no.
4. El método **cantidadMalEstacionamientoCaducadas()** que debe **devolver** (no mostrar por consola) la cantidad de infracciones reclamadas del tipo Mal Estacionamiento que resultaron caducadas.
5. El método **infraccionesReclamadasPorImporte()** que debe **devolver** (no mostrar por consola) un listado de infracciones reclamadas ordenadas de forma ascendente por el importe..

# Criterios

Para considerar aprobado el examen, el mismo debe demostrar la correcta aplicación de los siguientes conceptos de la programación orientada a objetos:

* Correcta definición de clases y asignación adecuada de sus responsabilidades.
* Encapsulamiento y uso de getters y setters sólo cuando corresponda con la visibilidad adecuada.
* Modularización reutilizable y mantenible con uso de métodos con correcta parametrización.
* Correcta aplicación de miembros de instancia y de clase.
* Correcta aplicación de herencia y polimorfismo.
* Correcta aplicación conceptual de las relaciones entre clases.
* Correcto uso de arreglos uni y/o bidimensionales
* Correcto uso de las TADS estudiadas (Pila, Cola y Lista Ordenada)

