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

# Enunciado

Una empresa certificadora de calidad de procesos nos pide diseñar un sistema que permita realizar la gestión de incidentes.

El gestor de incidentes poseerá una colección de Tickets que luego podrán ser asignados a Empleados, quienes los realizarán. Los tickets serán guardados según su orden de Prioridad.

Nos indican que hay tres tipos de tickets. Todos los tickets pertenecen a un cliente, del cual se conoce su nombre, la unidad de negocio del cliente afectada (String), la fecha de carga del incidente (String), un título del ticket y el Empleado que reportó el incidente (que puede ser o no el mismo que lo procesará).

Todos los tickets son Priorizables, por lo que deben resolver y devolver su prioridad (Alta, Media o Baja), y además indicar el tiempo de resolución estimado en horas (entero).

Los tickets pueden ser:

* Nueva funcionalidad: poseen una descripción funcional, la estimación de tiempo de desarrollo y una Prioridad.
* Bugs (errores en entornos de Desarrollo): indican si son bugs de test local, integración o UAT; la descripción del error, el nombre de la persona que detectó el problema (String) y el tiempo estimado de resolución.
* Incidentes (errores en entornos de producción): además de poseer una descripción del error y el nombre de la persona que detectó el problema (como los bugs) indican el nombre del host donde está corriendo la aplicación y un boolean que indica si es un error del tipo *stopper* o no. El tiempo de resolución estimado será siempre de 12 horas si el error es *stopper* y de 32 horas si no lo es.

La prioridad de los tickets se calculará de la siguiente forma:

* Para la nueva funcionalidad la prioridad será la dada por el ticket.
* Para los incidentes, si el tiempo de resolución es de 12 horas tendrá prioridad ALTA y si es de 32 horas tendrá prioridad MEDIA.
* Para los bugs se calculará:
  + ALTA si tiene tiempo de resolución de hasta 12 horas;
  + MEDIA si tiene tiempo de resolución menor o igual a 36 horas y mayor a 12 horas;
  + BAJA para el resto de los casos.

El gestor de incidentes tiene además dos funcionalidades claves:

* Realiza la asignación de tickets a los empleados. Esta asignación se hará en forma rotativa, tickets por tickets, uno a cada empleado hasta haber sido asignados todos los tickets, pudiendo ser que algunos empleados tengan asignados más incidentes que otros.
* Genera y retorna un reporte de tiempos promedio de tickets por cada prioridad.

De los empleados sabemos su nombre y apellido y el número de legajo. Los empleados deberán resolver los tickets que les sean asignados en el orden en el cual les fueron asignados.

Basado en el enunciado descripto, realizá:

1. El diagrama de clases que lo modelice, con sus relaciones, atributos y métodos.
2. El método **reporteTiempoPromedioPorPrioridad()**, de la clase que corresponda, que en base a las tareas ingresadas y no asignadas debe calcular y **devolver** (no mostrar por consola) el tiempo promedio estimado para cada prioridad.
3. El método **asignarTickets()** de la clase que corresponda, que debe asignar a los empleados todos los tickets disponibles, dejando la estructura vacía. La asignación debe hacerse como se indicó anteriormente.

# 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, ocultamiento de información y uso de getters y setters sólo cuando corresponda.
* 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.
* Correcta aplicación de TADs vistas en clase

