

**PROGRAMACIÓN ORIENTADA A OBJETOS – IS210**

**PRACTICA CALIFICADA 2**

# **Sección** : -

# **Profesores** : -

# **Duración** : -

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Instrucciones Parte Práctica**

**ENUNCIADO PARTE PRÁCTICA**

**Pregunta 1:**

Una aerolínea requiere de un programa que le permita gestionar los vuelos que ofrece.

Se sabe que un vuelo tiene un código único, un avión asignado, un origen y un destino. Por su parte, un avión tiene un modelo y una lista de asientos disponibles. Además, un asiento tiene un número, letra y precio que se calcula de la siguiente forma:

Según la letra del asiento se tiene un precio base

|  |  |
| --- | --- |
| Letra | Precio Base |
| A, C, D, F | $ 80 |
| B, E | $ 60 |

Y según el número del asiento se le aplica un factor adicional:

|  |  |
| --- | --- |
| Numero | Factor |
| menor a 7 | 1.5 |
| entre 7 a 16 | 1.3 |
| mayor a 16 | 1.15 |

De forma tal que, por ejemplo, un asiento **14D** le corresponde un precio de **$104.**

Tener en cuenta las siguientes reglas que se deben cumplir:

* Todo avión tiene que tener al menos un asiento y en caso un avión sea retirado del sistema todos los asientos asociados deben ser eliminados también.
* No puede existir un vuelo sin aerolínea, pero una aerolínea puede existir sin vuelos asignados.
* El sistema debe permitir que un vuelo sea registrado sin un avión y luego permitir asignar un avión a un vuelo en específico.

Aplicando los conocimientos vistos en clase, resuelve lo siguiente:

1. Realizar el modelo de clases usando UML. (3 puntos)
2. Implementar las clases necesarias para poder representar el caso presentado.

(3 puntos)

1. Crear y registrar 3 vuelos cada uno con un avión asignado que tenga al menos

2 asientos. (1.5 puntos)

1. Implementar un método que reciba un modelo de avión y retorne una lista de todos los vuelos que tengan asignado un avión con dicho modelo. Además, mostrar en pantalla toda la información de los vuelos, avión y asientos asociados. (3 puntos)
2. Usar una Excepción para mostrar en consola el mensaje de error “Modelo de avión inválido” en caso se ingrese un modelo de avión que no exista en el punto

anterior (D). (1.5 puntos).

**Pregunta 2:**

Una clínica requiere de un programa que le permita gestionar los pacientes que tiene.

Un paciente es registrado con un numero de documento, nombres completos, edad y un historial. Por su parte, el historial de un paciente cuenta con un código y un listado de visitas. Además, cada visita consta de una fecha, motivo y monto gastado.

Un paciente es asignado con una prioridad según el monto total gastado en la clínica de la siguiente forma:

|  |  |
| --- | --- |
| Monto total gastado en la clínica | Prioridad |
| Mayor a S/5,000 | Alta |
| Entre S/ 2,000 y S/5,000 | Media |
| Menor a S/2,000 | Baja |

Tener en cuenta las siguientes reglas que se deben cumplir:

* No puede existir una visita sin historial.
* Un historial puede crearse y luego ser asignado a un paciente.
* No puede existir un paciente sin una clínica, pero una clínica puede existir sin pacientes asignados.
* El sistema debe permitir que un paciente sea registrado sin un historial y luego permitir asignar un historial a un paciente en específico.

Aplicando los conocimientos vistos en clase, resuelve lo siguiente:

* 1. Realizar el modelo de clases usando UML. (3 puntos)
  2. Implementar las clases necesarias para poder representar el caso presentado.

(3 puntos)

* 1. Crear y registrar 3 pacientes con un historial asignado que tenga al menos 2

visitas. (1.5 puntos)

* 1. Implementar un método que reciba una prioridad y retorne una lista de todos los pacientes que sean de dicha prioridad. Además, mostrar en pantalla toda la información de los pacientes, historial y visitas asociadas. (3 puntos)
  2. Usar una Excepción para mostrar en consola el mensaje de error “Prioridad incorrecta” en caso se ingrese un valor incorrecto en el punto anterior (D).

(1.5 puntos).

<

Lima, septiembre del 2021