

## Práctico 4

Asignatura: Programación Orientada a Objetos Fecha: 2 de septiembre de 2016  
Profesor: Juan Carlos Figueroa Durán

Nombre alumno:

---

Implementar en Java la aplicación cuyo diagrama de clases se presenta en la Figura 1.

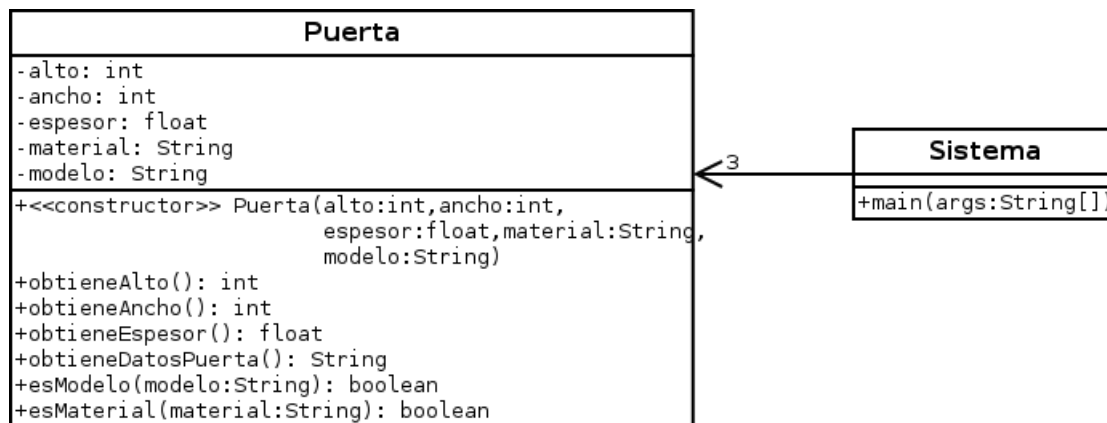


Figura 1: Diagrama de clases Puerta Sistema

Escriba el código equivalente a la definición de la clase Puerta, considerando lo siguiente:

- El método `obtieneDatosPuerta()`, entrega un String con todos los datos de una puerta con el siguiente formato: `< alto > X < ancho > X < espesor > - < modelo >:< material >`.
- El método `esModelo()`, retorna verdadero si el modelo de la puerta es el mismo que se recibe por parámetro. Retorna falso en caso contrario.
- El método `esMaterial()`, retorna verdadero si el material de la puerta es el mismo que se recibe por parámetro. Retorna falso en caso contrario.
- Incluya las precondiciones necesarias.



UNIVERSIDAD DEL BÍO-BÍO

Universidad del Bío-Bío  
Facultad de Ciencias Empresariales  
Departamento de Ciencias de la Computación  
y Tecnologías de Información

ICI  
Programación Orientada  
a Objetos

Escriba el código para la clase Sistema, considerando:

- Cree 3 objetos de la clase Puerta con datos de su invención. Solo use constantes.
- Desplegar los datos de cada puerta e indicar cuantas puestas son de pino.
- Todos los resultados obtenidos deberán ser desplegados en pantalla, con los mensajes pertinentes mediante el uso del objeto System.out y sus métodos print y println.

Implementar en Java la aplicación cuyo diagrama de clases se presenta en la Figura 2.

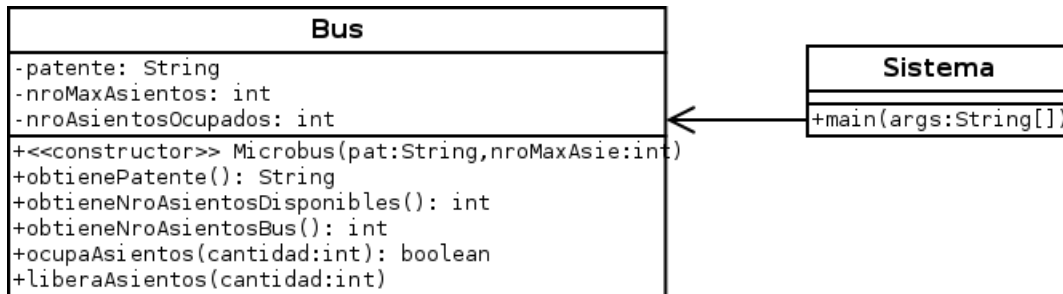


Figura 2: Diagrama de clases Puerta Sistema

Escriba el código equivalente a la definición de la clase Puerta, considerando lo siguiente:

- El atributo *nroMaxAsientos* corresponde al total de asientos de que dispone el Bus.
- El atributo *nroAsientosOcupados* corresponde a los asientos que en un momento determinado tienen pasajeros sentados.
- Cuando se crea un objeto de la clase, se define que todos los asientos del bus están libres.
- El método *ocupaAsientos()*, aumenta la cantidad de asientos ocupados de acuerdo a la cantidad entregada como parámetro. Considere que no se puede aceptar una cantidad de pasajeros mayor a la cantidad de asientos desocupados, ni un valor menor o igual a cero. El método retorna verdadero en caso que sea posible ocupar los asientos y falso en caso contrario.
- El método *liberaAsientos()*, disminuye la cantidad de asientos ocupados de acuerdo a la cantidad entregada como parámetro. Considere que no se puede aceptar una cantidad de pasajeros mayor a la cantidad de asientos ocupados, ni menor o igual a cero. El método debe incluir la precondition adecuada para prevenir errores.
- Incluya las precondiciones necesarias.



Universidad del Bío-Bío  
Facultad de Ciencias Empresariales  
Departamento de Ciencias de la Computación  
y Tecnologías de Información

ICI  
Programación Orientada  
a Objetos

Escriba el código para la clase Sistema, considerando:

- Crear un objeto de la clase Bus con patente “ZD-34-22” y con capacidad para 60 pasajeros.
- Subir y bajar pasajeros del bus de acuerdo con la siguiente secuencia:
  - Suben 4, Suben 2, Sube 1, Baján 2, Suben 3, Baja 1, Suben 6
- Desplegar el número de patente del bus, la cantidad de asientos libres y la cantidad de asientos ocupados.
- Todos los resultados obtenidos deberán ser desplegados en pantalla, con los mensajes pertinentes mediante el uso del objeto System.out y sus métodos print y println.