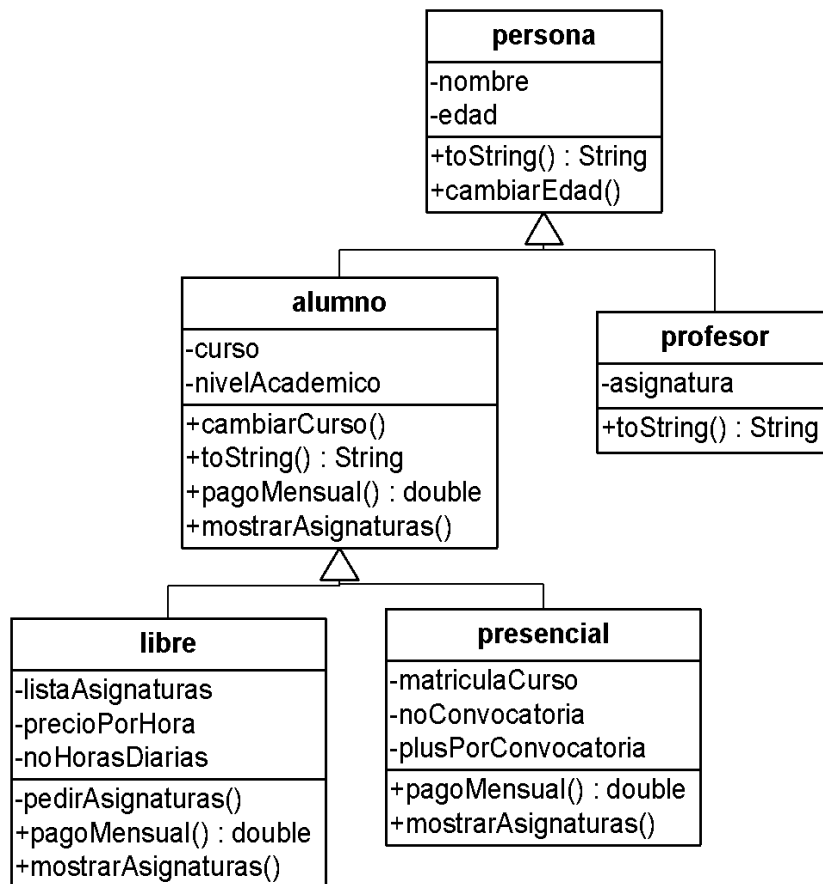


El código dado, implementa el siguiente diagrama de clases:



Y, además, las siguientes especificaciones:

Clase persona:

- ✓ nombre String
- ✓ edad int
- ✓ public String toString() { //devuelve cadena con nombre y edad}
- ✓ public void cambiarEdad(int e) { //asigna edad}

Clase alumno:

- ✓ Es abstracta
- ✓ Hereda persona.
- ✓ curso int
- ✓ nivelAcademico String
- ✓ public void cambiarCurso(int c) { //asigna curso}
- ✓ public String toString() { //redefine el método, devuelve cadena con nombre + edad + curso + nivelAcademico}
- ✓ double pagoMensual(); **//es abstracto**
- ✓ String getAsignaturas(); **//es abstracto**

Clase libre:

- ✓ Extiende alumno.

- ✓ listaDeAsignaturas String []//asignaturas de las que se matricula
- ✓ precioPorHora, es un valor constante tipo float inicializado a 10 Euros.
- ✓ noHorasDiarias int
- ✓ void pedirAsignaturas() { // se inicializa listaDeAsignaturas }
- ✓ double pagoMensual() { //devuelve lo que se paga al mes que se calcula como precioPorHora*noHorasDiarias*30 }
- ✓ String mostrarAsignaturas() { //devuelve una cadena con el nombre de todas las asignaturas que cursa }

Clase presencial:

- ✓ Extiende alumno.
- ✓ matriculaCurso es double
- ✓ plusPorConvocatoria es double
- ✓ noConvocatoria es int
- ✓ double pagoMensual() { //devuelve lo que se paga al mes que se calcula como (matriculaCurso+plusPorConvocatoria*noConvocatoria)/12 }
- ✓ String mostrarAsignaturas() { //devuelve una cadena con el contenido: **“Todas las asignaturas del curso” + curso** }

Clase profesor:

- ✓ Hereda persona.
- ✓ asignatura es tipo String.
- ✓ public String toString() { //redefine el método, devuelve cadena con nombre + edad + asignatura }

Clase prueba:

- 1) Tenemos los siguientes pasos realizados:
 - a) Crea un vector de profesores con dimensión 5
 - b) Muestra el contenido del vector profesores.
 - c) Crea un vector de alumnos con dimensión 5, en este vector se darán de alta 3 alumnos presenciales y dos libres.
 - d) Muestra un listado con las asignaturas que cursa cada alumno.
 - e) Muestra un listado con lo que paga al mes cada alumno y el total cobrado al mes.
- 2) Daremos valores a los datos anteriores para poder realizar el próximo punto.
- 3) Tras haber realizado los puntos anteriores, depuraremos el proyecto ejecutando las líneas paso a paso, añadiremos las variables que consideremos importantes a los watches, y definiremos los siguientes breakpoints:
 - ☐ Tipo línea, por ejemplo al pasar por una salida de pantalla.
 - ☐ Tipo clase, al cargar alguna clase.
 - ☐ Tipo campo, al acceder a algún campo de alguna clase.
 - ☐ Tipo método, al acceder a algún método de alguna clase.
- ✓ En al menos alguno, en las propiedades de los breakpoints definiremos:
 - ☐ Condiciones para que el programa se detenga al alcanzar las variables ciertos valores.
 - ☐ Se alcance un número de veces (hints), determinado.

Condición de entrega:

Realiza captura de los breakpoints solicitados y de la aplicación de sus propiedades.