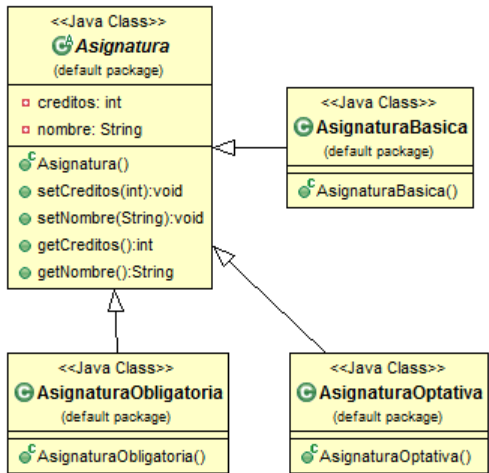


La escuela politécnica ha desarrollado un sistema para gestionar los planes de estudios de las titulaciones que se imparten en el centro. Dentro de este gestor se incorpora un diseño de clases para representar las diversas asignaturas en función de que sean de un tipo u otro. Como se puede ver toda asignatura viene descrita por una cadena de texto con el nombre de la misma y el número total de créditos que tiene asociados. Actualmente se contemplan los tipos que aparecen en el diagrama.



Tipo	Descripción
Básica	La cursan los estudiantes de todas las titulaciones
Obligatoria	La cursan todos los estudiantes de la titulación
Optativa	La cursan algunos estudiantes de la titulación

En el siguiente fragmento de código de la clase **GeneradorPlanEstudios** se puede ver una de las muchas operaciones tipo que contiene el diseño actual. Se sabe que en el futuro pueden aparecer nuevos tipos de asignaturas, desaparecer los actuales o simplemente asumir simplificaciones como que la clase

```

public class GeneradorPlanEstudios {
    //...
    public Asignatura operacionTipo(String tipo, String nombre, int creditos){
        Asignatura a = null;

        // en función del tipo de asignatura
        if (tipo == "basica") {
            a = new AsignaturaBasica();
            // operaciones sobre la asignatura
            a.setNombre(nombre);
            a.setCreditos(creditos);
            // otras operaciones sobre a
            // ...
        }
        else if (tipo == "obligatoria"){
            a = new AsignaturaObligatoria();
            // operaciones sobre la asignatura
            a.setNombre(nombre);
            a.setCreditos(creditos);
            // otras operaciones sobre a
            // ...
        }
        else if (tipo == "optativa"){
            a = new AsignaturaOptativa();
            // operaciones sobre la asignatura
            a.setNombre(nombre);
            a.setCreditos(creditos);
            // otras operaciones sobre a
            // ...
        }
        return a;
    }
    //...
}
    
```

**AsignaturaObligatoria** represente tanto asignaturas “básicas” como “obligatorias”. Teniendo en cuenta que esta clase contiene múltiples métodos similares al presentado, estos cambios suponen mucho trabajo y un riesgo de de inconsistencias si no actualizamos todos y cada uno de estos métodos adecuadamente.

Se desea modificar el diseño actual para solventar estos problemas.

1. ¿Qué patrón habría que utilizar para simplificar la generación y tratamiento de asignaturas? Selecciona el patrón más adecuado para esta tarea. Justifica tu respuesta. Indica además el tipo de patrón, y si el patrón utilizado tiene varias versiones justifica cual utilizarías.(1 punto)
2. Define las clases/interfaces necesarias para aplicar dicho patrón. Representa **y justifica** como quedaría el diagrama de clases.(2 puntos)

3. Reimplementa el método **GeneradorPlanEstudios.operacionTipo()** de acuerdo con el nuevo diseño. **(1 punto)**

La escuela politécnica dispone de otra aplicación para gestionar las prácticas en empresa de los estudiantes. Dentro de esta aplicación se dispone de la clase **PracticaEmpresa** que en cada instancia almacena el número de horas a realizar por el alumno en la empresa, además de otros datos que la escuela considera relevantes. Se desea utilizar esta clase como otro tipo de asignatura más en nuestro generador de titulaciones. Para ello debemos tener en cuenta que la correspondencia entre créditos y horas de prácticas es: 1 crédito = 10 horas. Sabemos además que la clase **PracticaEmpresa** **NO PUEDE SER MODIFICADA**.

```
public class PracticaEmpresa {  
    private int totalHoras;  
    // otros datos ...  
  
    public PracticaEmpresa() { }  
  
    public int getTotalHoras() {  
        return totalHoras;  
    }  
  
    public void setTotalHoras(int totalHoras) {  
        this.totalHoras = totalHoras;  
    }  
}
```

4. ¿Qué patrón habría que utilizar para poder utilizar PracticaEmpresa como otro subtipo de Asignatura? Selecciona el patrón más adecuado para esta tarea. Justifica tu respuesta. Indica el tipo de patrón y si el patrón utilizado tiene varias versiones justifica cual utilizarías. **(1 punto)**

- Define las clases/interfaces necesarias para aplicar dicho patrón. Representa **y justifica** como quedaría el diagrama de clases. **(2 puntos)**
- Incorpora a la operación **GeneradorPlanEstudios.operacionTipo()** el nuevo tipo de asignatura que has creado. Nota: Puedes hacerlo sobre la implementación inicial o tras el rediseño que hayas propuesto en los primeros apartados **(1 punto)**

Ya que las prácticas en empresa están presentes en gran parte de las titulaciones y por normativa han de ser todas iguales (ej: mismo número de horas), aunque con el diseño actual no se hace, se podría reutilizar la misma instancia durante toda la ejecución de la aplicación. Además, debido al coste computacional de esta clase, sería deseable retrasar al máximo posible su inicialización, e impidiendo que se pueda generar más de una instancia de ella.

7. ¿Qué patrón habría que utilizar para poder utilizar PracticaEmpresa con este comportamiento? Selecciona el patrón más adecuado para esta tarea. Justifica tu respuesta ilustrando de que manera cumple los requisitos expuestos en el párrafo anterior. Indica el tipo de patrón y si el patrón utilizado tiene varias versiones justifica cual utilizarías. **(2 puntos)**