

Nombre Cristian Villalpando Arreola	Matricula 07092081
Curso: Programacion orientada a objetos	Nombre del profesor Hector Antonio Aguilar Mogollan
Modulo:	Evidencia: Actividad IIII
Fecha: 19/02/2025	

Actividad 4:

En una escuela, los directivos necesitan crear un programa para control escolar que permita manejar profesores, alumnos, cursos y materias.

Descripcion:

En esta actividad, diseñarás y desarrollarás un programa para control escolar, el cual debe incluir clases para manejo de alumnos, profesores y materias que se relacionen entre sí.

Objetivo:

Aplicar los conocimientos adquiridos sobre relaciones avanzadas entre clases, creando un programa que incluya algunas relacionadas entre sí mediante composición y herencia.

Clases:

Profesores

-nombre: String
-matricula: String
-numero de nomina: int
-sueldo x hora: float
-materia: String

+calculo_de_nomina() float

Alumnos

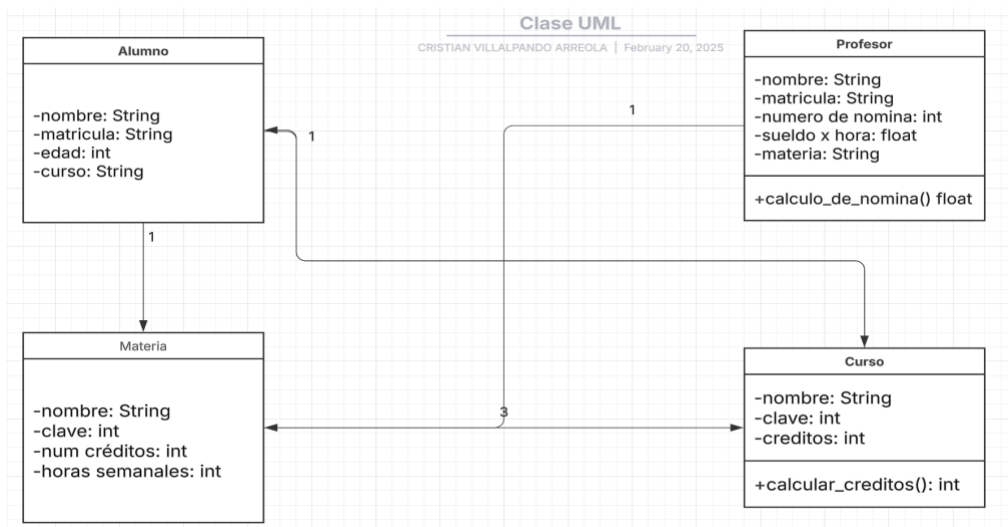
-nombre: String
-matricula: String
-edad: int
-curso: String

Materia

-nombre: String
-clave: int
-num créditos: int
-horas semanales: int

Curso

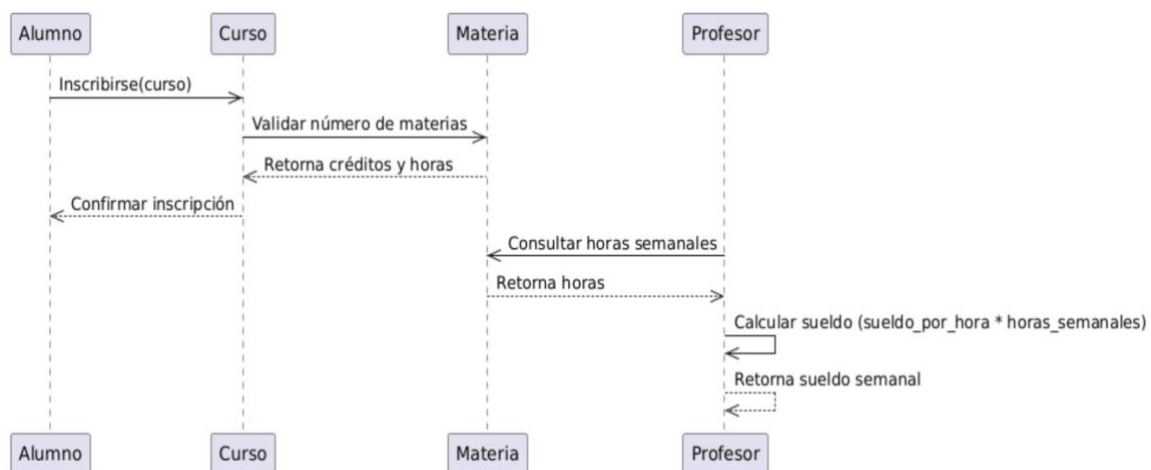
-nombre: String
-clave: int
-creditos: int
+calcular_creditos(): int



Describe el tipo de relación que existe entre las diferentes clases: composición, agregación o dependencia.

Relación	Tipo
Curso → Materia	Composición
Alumno → Curso	Agregación
Profesor → Curso	Agregación
Profesor → Materia	Dependencia

Diagrama de secuencia:



Codigo:

```
import java.util.Arrays;
import java.util.*;
public class Actividad4POB {

    static class Profesor{
        String nombre;
        String matricula;
        int numero_nomina;
        Double sueldox_hora;
        String materia;
        int horas_semanales;

        public Profesor(String nombre, String matricula, int
numero_nomina, Double sueldox_hora, String materia, int
horas_semanales){
            this.nombre= nombre;
            this.matricula= matricula;
            this.numero_nomina= numero_nomina;
            this.sueldox_hora= sueldox_hora;
            this.materia= materia;
            this.horas_semanales= 0;
        }//constructor de profesor
        public void asignar_materia (String materia, int horas){
            this.materia = materia;
            this.horas_semanales= horas;
        }//metodo para asignar materias

        public Double calcular_sueldo_semanal(){
            return sueldox_hora * horas_semanales;
        }//metodo calculo de sueldo
    }//clase profesor

    static class Alumno{
        String nombre;
        String matricula;
        int edad;
        String curso;

        public Alumno(String nombre, String matricula, int edad,
String curso){
            this.nombre= nombre;
            this.matricula= matricula;
            this.edad= edad;
            this.curso= curso;
        }//constructor alumno
    }//clase alumno

    static class Materia{
        String nombre;
        String clave;
        int num_creditos;
        int horas_semanales;
```

```
        public Materia(String nombre,String clave, int num_creditos,
int horas_semanales){
            this.nombre= nombre;
            this.clave= clave;
            this.num_creditos= num_creditos;
            this.horas_semanales= horas_semanales;
        }//constructor de materia
    }//clase materia

    static class Curso{
        String nombre;
        List <Materia> materia;

        public Curso(String nombre, Materia material, Materia
materia2, Materia materia3){
            this.nombre= nombre;
            this.materia = Arrays.asList(material, materia2,
materia3);
        }//constructor curso

        public int obtenerTotalCreditos() {
            return materia.stream().mapToInt(m -> m.num_creditos).sum();
        }//metodo para obtener creditos de la materia
    }//clase curso

    public static void main(String[] args) {
        Materia programacion = new Materia("programacion", "PR0101",
5, 4);
        Materia sistemas = new Materia("sistemas", "SIS102", 4, 3);
        Materia ingles = new Materia("ingles", "ING103", 3, 3);

        Curso cursoDesarrolloSoftware = new Curso("Desarrollo De
Software", programacion, sistemas, ingles);

        Profesor profesor1 = new Profesor("Juan P\@rez", "P001", 1234,
200.0, "Programaci\>n", 10);

        profesor1.asignar_materia("Programaci\>n", 10);

        Alumno alumno1 = new Alumno("Cristian Villalpando", "A001" ,
20, cursoDesarrolloSoftware.nombre);

        System.out.println("Profesor: " + profesor1.nombre + " -
Sueldo semanal: " + profesor1.calcular_sueldo_semanal());
        System.out.println("Alumno: " + alumno1.nombre + ", "
+alumno1.matricula+ " - Curso: " + alumno1.curso);
        System.out.println("Cr\>ditos totales del curso: " +
cursoDesarrolloSoftware.obtenerTotalCreditos());

    }//interaccion
}///clase actividad 4
```

Resultado final:

```
J --- exec:3.1.0:exec (default-cli) @ ACTIVIDAD4P0B ---  
Profesor: Juan Pérez – Sueldo semanal: 2000.0  
Alumno: Cristian Villalpando, A001 – Curso: Desarrollo De Software  
· Créditos totales del curso: 12
```

Explicacion de codigo:

Aquí lo que estamos haciendo es crear las diferentes clases (Profesor, Alumno, Materia, Curso) definimos sus atributos y solo en 2 clases tenemos con función (Profesor y curso), también podemos ver el constructor de cada clase

```
11 static class Profesor{  
12     String nombre;  
13     String matricula;  
14     int numero_nomina;  
15     Double sueldox_hora;  
16     String materia;  
17     int horas_semanales;  
18  
19     public Profesor(String nombre, String matricula, int numero_nomina, Double sueldox_hora, String materia, int horas_semanales){  
20         this.nombre= nombre;  
21         this.matricula= matricula;  
22         this.numero_nomina= numero_nomina;  
23         this.sueldox_hora= sueldox_hora;  
24         this.materia= materia;  
25         this.horas_semanales= 0;  
26     } //constructor de profesor  
27     public void asignar_materia (String materia, int horas){  
28         this.materia = materia;  
29         this.horas_semanales= horas;  
30     } //metodo para asignar materias  
31  
32     public Double calcular_sueldo_semanal(){  
33         return sueldox_hora * horas_semanales;  
34     } //metodo calculo de sueldo  
35 } //clase profesor  
36  
37  
38 static class Alumno{  
39     String nombre;  
40     String matricula;  
41     int edad;  
42     String curso;  
43  
44     public Alumno(String nombre, String matricula, int edad, String curso){  
45         this.nombre= nombre;  
46         this.matricula= matricula;  
47         this.edad= edad;  
48         this.curso= curso;  
49     } //constructor alumno  
50 } //clase alumno  
51  
52 static class Materia{  
53     String nombre;  
54     String clave;  
55     int num_creditos;  
56     int horas_semanales;  
57  
58     public Materia(String nombre, String clave, int num_creditos, int horas_semanales){  
59         this.nombre= nombre;  
60         this.clave= clave;  
61         this.num_creditos= num_creditos;  
62         this.horas_semanales= horas_semanales;  
63     } //constructor de materia  
64 } //clase materia  
65  
66 static class Curso{  
67     String nombre;  
68     List <Materia> materia;  
69  
70     public Curso(String nombre, Materia materia1, Materia materia2, Materia materia3){  
71         this.nombre= nombre;  
72         this.materia = Arrays.asList(materia1, materia2, materia3);  
73     } //constructor curso  
74  
75     public int obtenerTotalCreditos() {  
76         return materia.stream().mapToInt(m -> m.num_creditos).sum();  
77     } //metodo para obtener creditos de la materia  
78 } //clase curso  
79
```

En esta parte del código tenemos la función main, donde estamos asignando valor a: materia, alumno y profesor y en las últimas líneas tenemos la salida de información

```
80 public static void main(String[] args) {
81     Materia programacion = new Materia("programacion", "PR0101", 5, 4);
82     Materia sistemas = new Materia("sistemas", "SIS102", 4, 3);
83     Materia ingles = new Materia("ingles", "ING103", 3, 3);
84
85     Curso cursoDesarrolloSoftware = new Curso("Desarrollo De Software", programacion, sistemas, ingles);
86
87     Profesor profesor1 = new Profesor("Juan Pérez", "P001", 1234, 200.0, "Programación", 10);
88
89     profesor1.asignar_materia("Programación", 10);
90
91     Alumno alumno1 = new Alumno("Cristian Villalpando", "A001", 20, cursoDesarrolloSoftware.nombre);
92
93     System.out.println("Profesor: " + profesor1.nombre + " - Sueldo semanal: " + profesor1.calcular_sueldo_semanal());
94     System.out.println("Alumno: " + alumno1.nombre + ", " + alumno1.matricula + " - Curso: " + alumno1.curso);
95     System.out.println("Créditos totales del curso: " + cursoDesarrolloSoftware.obtenerTotalCreditos());
96
97 } //interaccion
98 } //clase actividad 4
```

Link Github:

<https://github.com/Cristian05105/ProgramacionOB.git>