

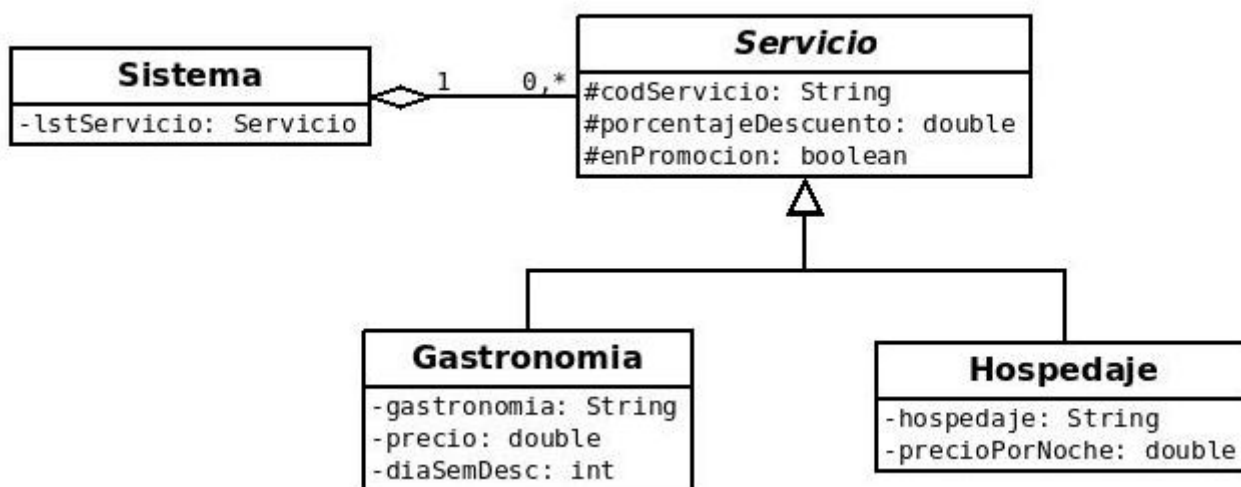
Todos los actos estarán regidos por **la buena fe, la transparencia y la responsabilidad** por parte de las y los estudiantes, docentes, nodocentes y del equipo de gestión.

IDE: Eclipse

* **Proyecto:** Crear una carpeta /tuApellidoNombre/tuNroDni (el nombre del proyecto es tu DNI)

Entrega: Aula Virtual subir la carpeta comprimida con tuApellidoNombre

Sistema de Servicio



Nota: El examen acredita puntos por la resolución del modelo y test de cada CU.

Capa modelo:

Clase **Servicio:**

Superclase abstracta

Atributos:

String codServicio //Identificador único que para que sea válido debe tener una longitud igual a 6, implementar excepción.

double porcentajeDescuento, boolean enPromocion

Metodos:

~ calcularPrecioFinal(LocalDate dia) : double //método abstracto

Clase **Hospedaje:**

Subclase de Servicio

Atributos: String hospedaje, double precioPorNoche

Metodos:

+calcularPrecioFinal(LocalDate dia) : double //Si está en promoción aplica descuento de lunes a viernes

Clase **Gastronomía:**

Subclase de Servicio

Atributos: String gastronomia, double precio, int diaSemDesc

Metodos:

+calcularPrecioFinal(LocalDate dia) : double //Si está en promoción aplica el porcentajeDescuento

Clase **Sistema:**

Atributo: List<Servicio> lstServicio

Metodos:

+ traerServicio (String codServicio) : Servicio

+ traerServicio (boolean enPromocion) : Servicio //retorna una lista de los servicios según parametro

- + traerServicio (boolean enPromocion, LocalDate dia) : Servicio //retorna una lista de los servicios según la fecha parámetros
- + agregarGastronomia(String codServicio, double porcentajeDescuento, boolean enPromocion, String gastronomia, double precio, int diaSemDesc) : boolean //implementar excepción si existe el objeto
- + agregarHospedaje(String codServicio, double porcentajeDescuento, boolean enPromocion, String hospedaje, double precioPorNoche) : boolean //implementar excepción si existe el objeto

Capa test:

Nota: Al comenzar cada test indicar el número a resolver ej: `System.out.println("1-3");` y luego la implementación del mismo.

Test.java

1-1) Intentar crear el objeto Gastronomia: codServicio=4892, gastronomia=Hamburguesa criolla, precio=180.0, diaSemDesc= 4, porcentajeDescuento=15.0, enPromocion=true

1-2) Crear el objeto Gastronomia: codServicio=489235, gastronomia=Hamburguesa criolla, precio=180.0, diaSemDesc= 4, porcentajeDescuento=15.0, enPromocion=true

1-3) Intentar crear el objeto Hospedaje: codServicio=2872, hospedaje=Cabaña 3 personas, precioPorNoche=1500.0, porcentajeDescuento= 10.0, enPromocion=true

1-4) Crear el objeto Hospedaje: codServicio=287282, hospedaje=Cabaña 3 personas, precioPorNoche=1500.0, porcentajeDescuento= 10.0, enPromocion=true

2-1) Calcular e imprimir precio final del servicio:

Gastronomia [489235, porcentajeDescuento=15.0, enPromocion=true, gastronomia=Hamburguesa criolla, precio=180.0, diaSemDesc=4] para el día: 2020-10-28

2-2) Calcular e imprimir precio final del servicio:

Hospedaje [287282, porcentajeDescuento=10.0, enPromocion=true, hospedaje=Cabaña 3 personas, precioPorNoche=1500.0] para el día: 2020-10-27

3) Agregar los siguientes objetos e imprimir la lista de servicios

Gastronomia [858927, porcentajeDescuento=15.0, enPromocion=true, gastronomia=Milanesa con pure, precio=350.0, diaSemDesc=3],

Hospedaje [489259, porcentajeDescuento=10.0, enPromocion=true, hospedaje=Habitación triple, precioPorNoche=2200.0],

Gastronomia [182835, porcentajeDescuento=20.0, enPromocion=false, gastronomia=Gaseosa, precio=120.0, diaSemDesc=3],

Hospedaje [758972, porcentajeDescuento=15.0, enPromocion=false, hospedaje=Habitación simple, precioPorNoche=1000.0]

4-1) Traer todos los objetos Servicios enPromocion=true

4-2) Traer todos los objetos Servicios del dia=2020-10-28 y enPromocion=true