

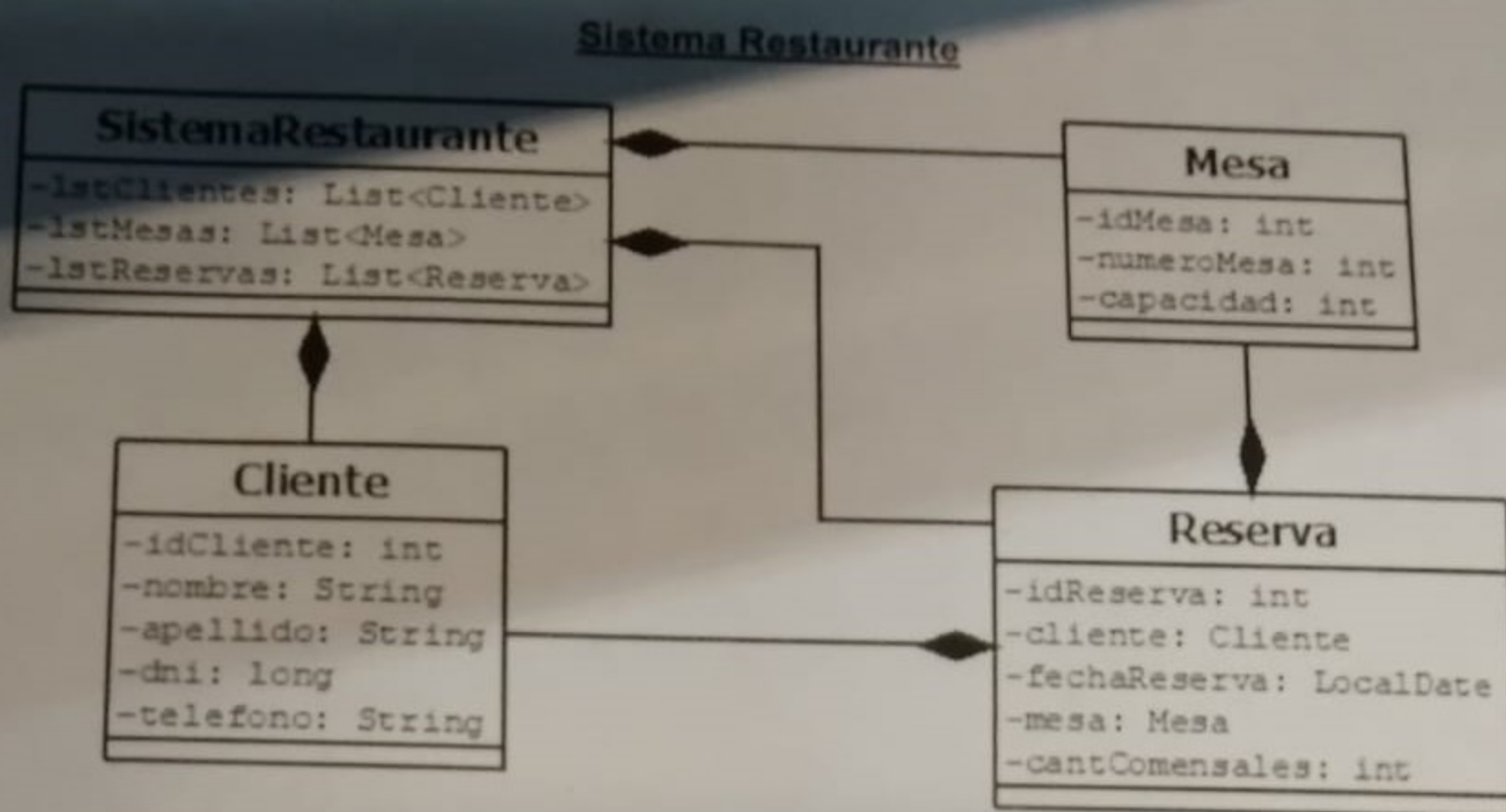


OO1 - 2020 - 1er parcial (tema 2)

Orientación a Objetos 1 (Universidad Nacional de Lanús)



Scan to open on Studocu



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

Casos de uso:

1) + traerCliente(long dni) : Cliente

2) + agregarCliente(String nombre, String apellido, long dni, String telefono) : boolean

Lanzar excepción si ya existe un cliente con el mismo dni.

El id se calcula de forma autoincremental, teniendo en cuenta que la lista puede tener altas y bajas de objetos.

3) + traerMesa(int numeroMesa): Mesa

4) + agregarMesa(int numeroMesa, int capacidad) : boolean

Lanzar excepción si ya existe una mesa con el mismo número.

El id se calcula de forma autoincremental, teniendo en cuenta que la lista puede tener altas y bajas de objetos.

5) - validarComensales(): boolean *→ int cantComensal*

Lanzar excepción si la cantidad de comensales de la reserva es superior a la capacidad de la mesa.

6) + traerReserva(int numeroMesa, LocalDate fechaReserva): Reserva

7) + agregarReserva(Cliente cliente, LocalDate fechaReserva, Mesa mesa, int cantComensales) : boolean

Lanzar excepción si la mesa seleccionada ya está reservada para esa fecha.

El id se calcula de forma autoincremental, teniendo en cuenta que la lista puede tener altas y bajas de objetos.

8) + traerReservasPorFecha(LocalDate fechaDesde, LocalDate fechaHasta) : List<Reserva>

El método retorna una lista de todas las reservas que se encuentren entre la fechaDesde y la fechaHasta

(Incluyendo extremos)

9) + traerMesasLibresPorFecha(LocalDate fecha) : List<Mesa>

El método retorna una lista de todas las mesas que no tengan una reserva para el día indicado

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

Test.java

1) Agregar los siguientes Clientes:

```
Cliente [nombre=Rodrigo, apellido=Sanchez, dni=32694053, telefono=+541179456274],  
Cliente [nombre=Adriana, apellido=Diaz, dni=31953256, telefono=+541109462742],  
Cliente [nombre=Marcos, apellido=Garcia, dni=34096528, telefono=+541188345186]
```

2) Agregar las siguientes Mesas:

```
Mesa [numeroMesa=1, capacidad=2],  
Mesa [numeroMesa=2, capacidad=2],  
Mesa [numeroMesa=3, capacidad=4],  
Mesa [numeroMesa=4, capacidad=6],  
Mesa [numeroMesa=5, capacidad=4],  
Mesa [numeroMesa=6, capacidad=2]
```

3) Agregar las siguientes Reservas:

```
Reserva [cliente=Cliente [nombre=Rodrigo, apellido=Sanchez, dni=32694053,  
telefono=+541179456274], fechaReserva=2023-08-15, mesa=Mesa [numeroMesa=1,  
capacidad=2], cantComensales=2],
```

```
Reserva [cliente=Cliente [nombre=Adriana, apellido=Diaz, dni=31953256,  
telefono=+541109462742], fechaReserva=2023-08-15, mesa=Mesa [numeroMesa=4,  
capacidad=6], cantComensales=4],
```

```
Reserva [cliente=Cliente [nombre=Marcos, apellido=Garcia, dni=34096528,  
telefono=+541188345186], fechaReserva=2023-08-25, mesa=Mesa [numeroMesa=5,  
capacidad=4], cantComensales=4]
```

4) Traer e imprimir todas las reservas entre las fechas 2023-08-15 y 2023-08-20

5) Traer e imprimir todas las mesas libres para la fecha 2023-08-15

6) Intentar agregar nuevamente al siguiente cliente:

```
Cliente [nombre=Adriana, apellido=Diaz, dni=31953256, telefono=+541109462742]
```

7) Intentar agregar nuevamente la siguiente mesa:

```
Mesa [numeroMesa=3, capacidad=4]
```

8) Intentar agregar la siguiente reserva:

```
Reserva [cliente=Cliente [nombre=Rodrigo, apellido=Sanchez, dni=32694053,  
telefono=+541179456274], fechaReserva=2023-08-25, mesa=Mesa [numeroMesa=5,  
capacidad=4], cantComensales=4]
```

9) Intentar agregar la siguiente reserva:

```
Reserva [cliente=Cliente [nombre=Adriana, apellido=Diaz, dni=31953256,  
telefono=+541109462742], fechaReserva=2023-08-17, mesa=Mesa [numeroMesa=6,  
capacidad=2], cantComensales=4]
```