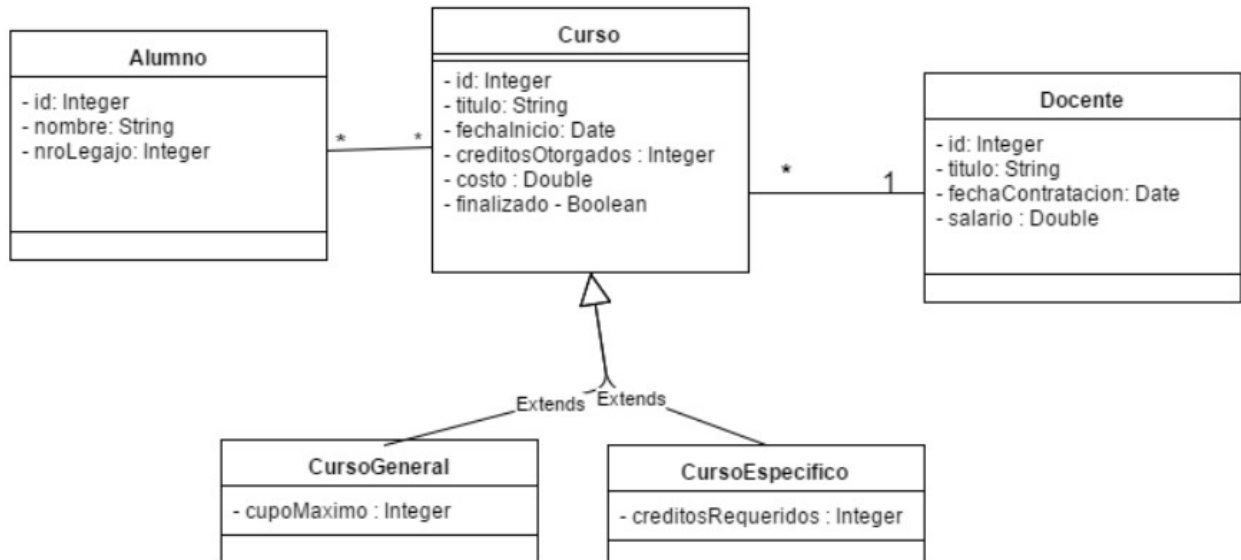


## Exámen Final Programación III

### PARTE I - Consignas:

1. Crear una carpeta llamada final\_07\_mayo\_2019 en el disco D:.
2. Abrir el eclipse seleccionando como workspace la carpeta final\_07\_mayo\_2019.
3. Crear un proyecto con su Apellido\_Nombre\_Parte1
4. **Realizaremos un sistema que intentará administrar las inscripciones de alumnos a distintos cursos y la asignación de docentes a estos cursos. El sistema se puede representar con el siguiente modelo:**



5. Dentro del paquete lógica deberán quedar las siguientes clases correspondientes.
6. Agregar en todas las clases, donde haga falta, los métodos correspondientes que permitan asignar y obtener los valores de los atributos.
7. Se debe tener en cuenta que un docente no puede estar asignado a más de 2 cursos. Se requiere definir una excepción que retorne un string con la frase "El docente **Docente.id** ya cubre el cupo permitido de cursos."
8. Se debe tener en cuenta que un curso no puede tener más de 15 alumnos. Para ello, se requiere definir una excepción que retorne un string con las frases "El cupo del curso **Curso.titulo** esta cubierto. Imposible registrar su inscripción".
9. Generar un programa principal con una clase inicio que arranque la ejecución para poder probar los métodos solicitados en los puntos 7 y 8 (genere los objetos necesarios para satisfacer el requerimiento).

## Exámen Final Programación III

---

### PARTE II - Consignas:

1. Importar el proyecto `exa_11122017` y cambiarle el nombre al proyecto colocándole `Apellido_Nombre_parte2`.
2. **Se trata de implementar un proyecto en java para la gestión de una central de reservas hoteleras muy simplificada. Los clientes de la central de reserva hotelera podrán realizar sus reservas en cualquiera de los hoteles que tengan disponibilidad en las fechas solicitadas.**
3. Dentro del paquete `logica` deberán quedar las siguientes clases: *Usuario*, *Cliente*, *AdministradorHotel*, *Reserva*, *Hotel* y *TipoHabitacion*.
4. Las clases *Cliente* y *AdministradorHotel* deben **extender** de la clase *Usuario*.
  - 4.1. La clase *AdministradorHotel* tiene los siguientes atributos:  
id (int), horario (String), celularAsignado (String), listaHoteles (ArrayList de clase *Hotel*).
5. Agregar a la clase *Reserva* los siguientes atributos:
  - 5.1. cliente (objeto de la clase *Cliente*).
  - 5.2. tipoHabitacion (objeto de la clase *TipoHabitacion*).
6. Agregar a la clase *TipoHabitacion* el siguiente atributo:
  - 6.1. hotel (objeto de la clase *Hotel*).
7. Agregar a la clase *Hotel* los siguientes atributos:
  - 7.1. administradorHotel (objeto de la clase *AdministradorHotel*).
  - 7.2. listaTiposHabitacion (ArrayList de clase *TipoHabitacion*).
8. Declarar un método en la clase *Hotel*, que se llame `obtenerCantidadTiposHabitacion()`, que retorne un entero corto con la cantidad de tipos de habitaciones que posee dicho hotel.
9. Agregar en todas las clases, donde haga falta, los métodos correspondientes que permitan asignar y obtener los valores de los atributos.
10. Dentro del paquete `interfaz` crear una clase visual llamada *PconsultaHabitacionesPorHotel*. Dicha ventana debe permitir consultar las habitaciones con las que cuenta un determinado hotel. Para ello se debe solicitar al usuario que seleccione el hotel y si éste se encuentra, mostrar en la misma pantalla, una tabla con los tipos de habitación que posee el hotel, especificando el nombre del tipo de habitación, la capacidad y el precio en temporada alta. Mostrar también en un cuadro de texto colocado en la parte inferior de la pantalla, la cantidad de tipos de habitación que posee el hotel, invocando el método `obtenerCantidadTiposHabitacion()` declarado en la clase *Hotel*. Finalmente, si el hotel no se encuentra mostrar con un *JOptionPane* un mensaje de aviso.
11. En el paquete `gestores`:
  - 11.1. La clase *Conexión* está completa y debe devolver la conexión a la base “**sisHotel**” de *MySQL*, (revisar nombre de la base de datos, puerto, usuario y contraseña utilizados para realizar la conexión).
  - 11.2. En la clase *Gestor* se debe realizar la codificación del/los métodos necesarios que permitan obtener los datos de la base de datos. Codificar uno o varios métodos a criterio del alumno, que permitan obtener la lista de tipos de habitación de un determinado hotel.
12. Crear en el paquete principal una clase *Inicio* que arranque la aplicación ejecutando la pantalla *PconsultaHabitacionesPorHotel*