

## MISCELÁNEA DE EJERCICIOS EJE TEMÁTICO 5 INTERFACES GRÁFICAS Y BASES DE DATOS JAVA

1. Crear la clase alumno va ha tener las siguientes características:

Nombre: nombres.

Apellidos: apellidos .

añoDeNacimiento: año de nacimiento

númeroPersonal: Identificación del alumnos

grupo: Grupo de clase

turnoHorario: Jornada de estudio es “Mañana” o “Tarde”

Igualmente vamos a tener los siguientes métodos:

cambiaGrupo: Cambia el grupo del alumno.

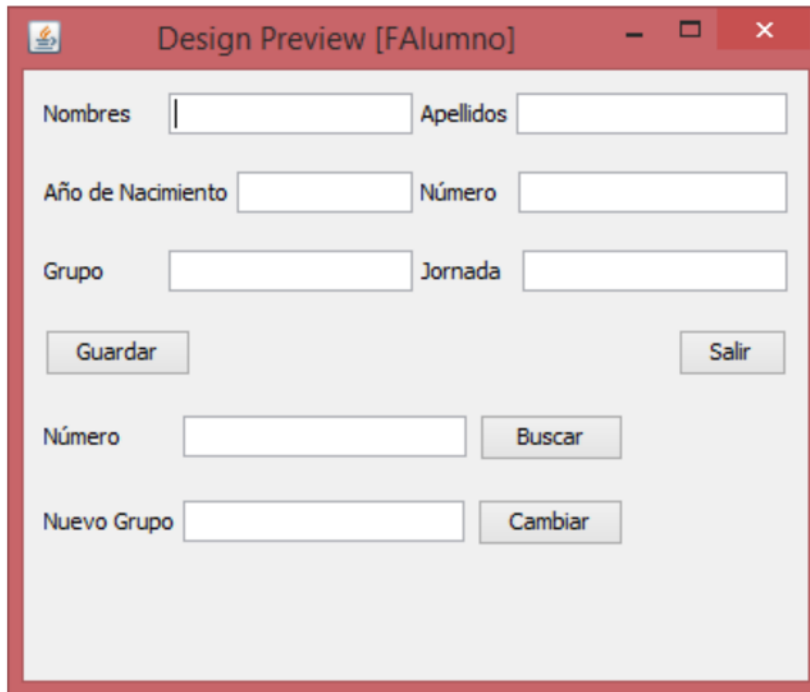
asignaValores: Asigna la información del objeto alumno.

muestraGrupo: Muestra el grupo del alumno.

dameGrupo: Presenta el grupo.

esMayorEdad: Identifica si el alumno es mayor de edad o no.

Ya creado la clase alumno, pasamos a la creación de la forma:



The image shows a Java Swing window titled "Design Preview [FAlumno]". It contains a form with the following fields and buttons:

- Names: "Nombres" and "Apellidos" (two text input fields).
- Birth: "Año de Nacimiento" and "Número" (two text input fields).
- Group: "Grupo" and "Jornada" (two text input fields).
- Buttons: "Guardar" (Save), "Salir" (Exit), "Buscar" (Search), and "Cambiar" (Change).
- Additional fields: "Número" and "Nuevo Grupo" (two text input fields).

En ella vamos a realizar la programación de los botones:

Guardar

Buscar

Cambiar



Design Preview [FPROFESOR]

Nombres	JOSE ANTONIO	Apellidos	DIAZ GOMES
Año de Nacimiento	1990	Número	10
Grupo	E162	Jornada	TARDE

Guardar Salir

Número 10 Buscar

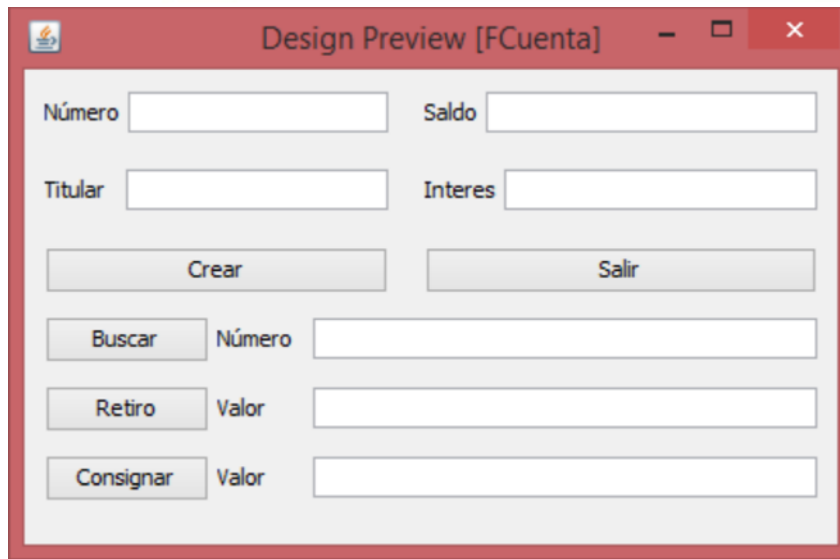
Nuevo Grupo E162 Cambiar

2. Crear una clase llamada profesor que contenga los siguientes atributos: Documento, Nombre, Apellido, Año, Profesión. Adicionar a la clase profesor los siguientes métodos: asignarProfesor y cambiarProfesion.

Design Preview [FPROFESOR]

NOMBRES :		APELLIDOS :	
AÑO :		DOCUMENTO :	
GUARDAR	SALIR	PROFESION:	
CAMBIAR	BUSCAR	NUEVA PROFESION:	
		DOCUMENTO :	

3. Crear un proyecto en donde se crea la clase cuenta con los siguientes atributos: long numero; String titular; float saldo; float interesAnual; además, podamos visualizar la información correspondiente a las siguientes operaciones: retirar, consignar y buscar cuenta.



Design Preview [FCuenta]

Número  Saldo

Titular  Interes

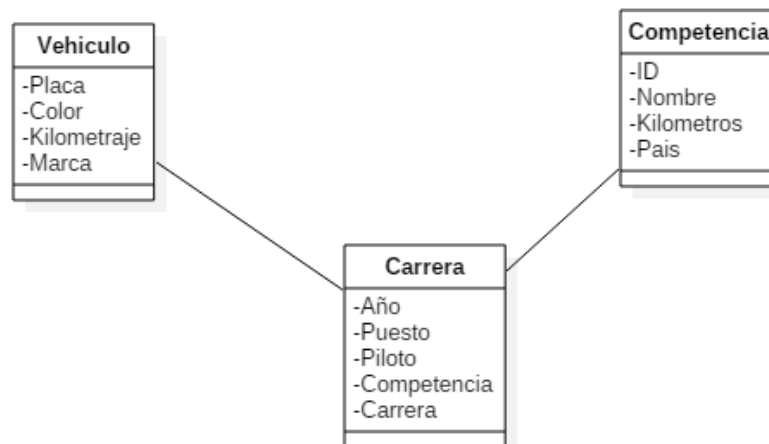
Crear Salir

Buscar Número

Retiro Valor

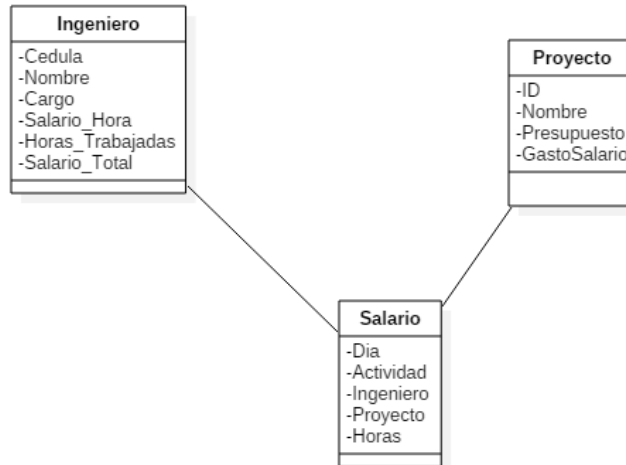
Consignar Valor

4. Se desea implementar el resultado de las competencias automovilísticas, con actualización de los kilómetros recorridos por los autos para ello se implementó el siguiente modelo:

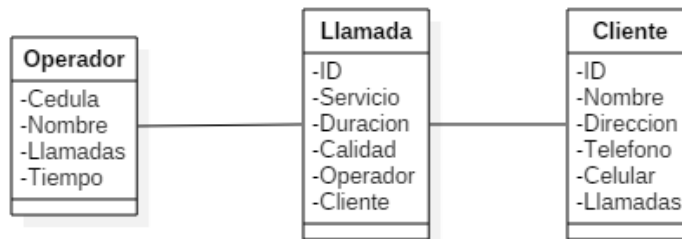


Creación Clase Vehículo  
Creación Clase Competencia.  
Creación Clase Carrera.  
Forma, guardar y buscar Vehículo.  
Forma, guardar y buscar Carrera.  
Forma, guardar y buscar Competencia.  
Actualizaciones.

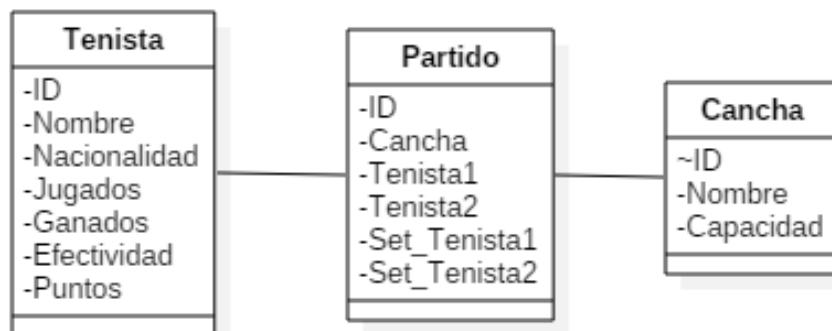
5. Se desea implementar el control de gastos de los salarios gastados en los proyectos de una industria informática, los empleados, tiene un valor de hora y un valor pagado diariamente por proyecto, se debe acumular cuanto se ha ganado el trabajador y el valor total pagado por proyecto si los gastos de los salarios llegan a superar el 50% del presupuesto debe informar que va a llegar al tope y el tope es del 70% del presupuesto.



6. Nos ha pedido llevar un control sobre los operadores, los clientes y las llamadas recibidas en un call center. En donde debemos almacenar la cantidad de llamadas recibidas y el tiempo utilizada para dar solución del operador, así como la cantidad de llamadas realizadas por el cliente. Esta información se debe actualizar cada vez que reciba una llamada. El modelo a implementar es:



7. Se desea almacenar la información del torneo de tenis de Bucaramanga para ello manejaremos información sobre los tenistas, la cancha en donde se juegan los partidos, que son varias, y los partidos en donde se almacenara el resultado del partido el cual nunca debe ser empate y la cantidad máxima de set jugados son cinco en total y máxima por jugador de tres, la efectividad del jugador es el promedio de partidos jugados sobre los partidos ganados y los puntos se calcula por set ganado el cual vale diez puntos. El modelo a implementar es el siguiente:

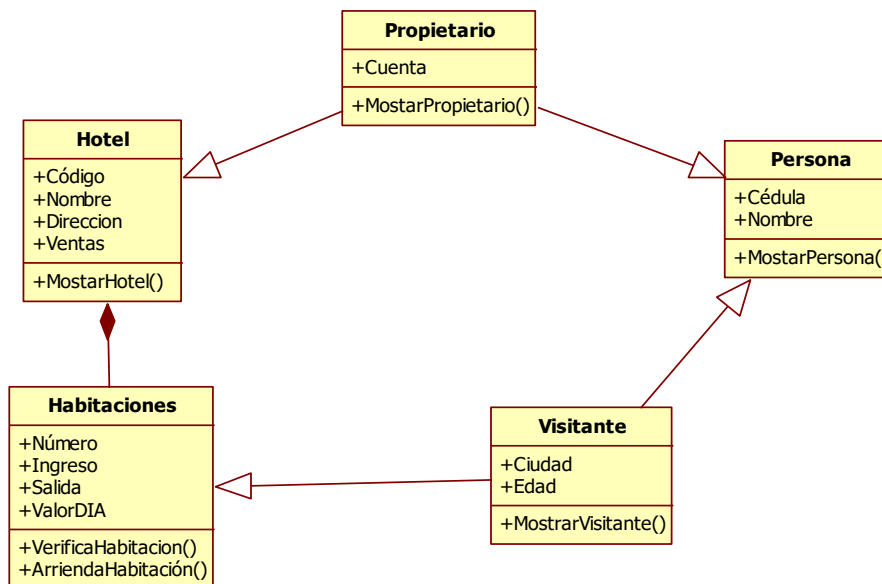


8. Realizar la administración de notas de un alumno por materia. La clase alumno private String nombre,direccion; private int telefono,cedula.  
La clase materia sus atributos son: private long codigo; private String nombre; private Alumno Estudiante; private float credits; private float definitiva; private float porcentajeAcumulado; private LinkedList notas.  
La clase notas sus atributos: Date fecha; double porcentaje; double nota; double definitivaparcial.

NOTA	% Nota	Definitiva	Fecha
4.0	20.0	0.8	Thu Apr 30 16:0...
4.5	30.0	1.3499999999999...	Thu Apr 30 16:0...
5.0	30.0	1.5	Thu Apr 30 16:0...
4.0	20.0	0.8	Thu Apr 30 16:0...

Recuerde que se debe crear primero el alumnos y luego la materia.

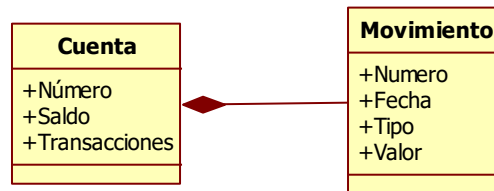
9. Crear un programa que realice lo siguiente de acuerdo al siguiente UML:





El programa de actualizar y mostrar todos sus atributos.  
El hotel tiene un solo dueño y una lista de habitaciones.  
La habitación está libre si no tiene asociado visitante.  
El objeto de tipo hotel es el que se guarda.  
Cuando un visitante se retira del hotel se debe de calcular el valor del ArrendaHabitación que es igual a total de días por el valor de Día y este valor se acumula en ventas del objeto Hotel.  
Crear el proyecto en modo visual.

10. Se desea implementar un proyecto para el manejo de las cuentas de ahorros. Todos los atributos son privados en Modelo de Clases es el siguiente:



En donde se cumple las siguientes condiciones:  
Una Cuenta tiene muchos movimientos.  
Un movimiento puede ser de retiro o de consignación.  
Cada vez que se realice un movimiento se debe actualizar la cuenta en su saldo y en el número de transacciones.  
Crear: Clase Cuenta. Clase Movimiento. Forma Cuenta. Forma Movimiento.

11. Realizar un proyecto para la construcción de objetos a partir de la siguiente clase:

AUTOMOVIL
+PLACA
+COLOR
+VELOCIDAD_MAXIMA
+VELOCIDAD_ACTUAL
+CAPACIDAD_PASAJEROS
+PASAJEROS_ACTUALES
+MARCA
+AUMENTA_VELOCIDAD()
+SUBE_PASAJERO()
+BAJA_PASAJERO()
+FRENO_TOTAL()
+BAJA_VELOCIDAD()

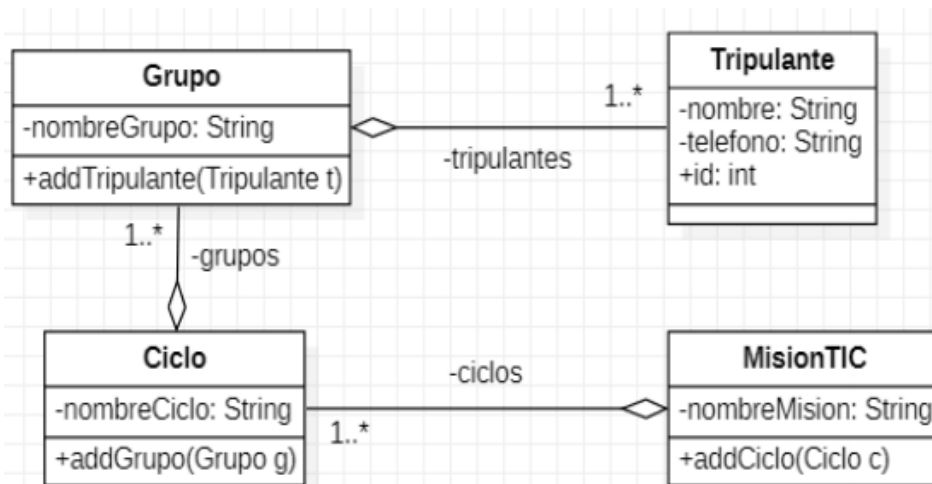
Todos los atributos son privados se debe crear los get y los set y los eventos representados en el diagrama.

Realizar las siguientes tareas:  
Crear un objeto, ejemplo:  
Placa: QRE123  
Color: ROJO  
Velocidad Maxima: 200  
Velocidad Actual: 100

Capacidad Pasajeros: 5  
Pasajeros Actuales: 2  
Marca: MAZDA.

Realizar las siguientes actividades:  
Aumentar la velocidad subiendo 80 km más.  
Se sube un pasajero.  
Aumentar la velocidad subiendo 70 km más.  
Se sube tres pasajeros.  
Se baja un pasajero.

12. Codifique las clases java según el diagrama de clases mostrado a continuación:



Emplee el patrón de diseño MVC.

Conforme al diagrama de clases mostrado implemente:

Una clase conexión que permita realizar un CRUD a la tabla de base de datos mediante el uso de JDBC (Archivo entregado por el formador)  
Una ventana principal mediante el uso de Java Swing para poner en funcionamiento el CRUD de forma visual.

El CRUD que se pide es el siguiente:

- Crear un nuevo contacto
- Eliminar un contacto
- Actualizar un contacto

Los atributos de la Clase contacto corresponden a los campos de la tabla contactos

- Cree los métodos que considere necesarios para cumplir con el ejercicio (Constructores, get, set...).
- Los métodos del CRUD pertenecen a la clase contacto y dependen de la clase ConexionDb
- El funcionamiento de la clase principal depende de la clase Contacto, y será utilizada solo para mostrar en forma de menú de consola o de interfaz gráfica con formulario el funcionamiento del CRUD