

CONSOLAS

Interactuan con el usuario

Todos los metodos de las consolas son estaticos

Login

- Usuarios: HashMap<String, Usuario>

- nuevoUsuario: Usuario

ejecutarConsola(): void

guardarUsuarios(): void

Consola_Cliente

sistema: Sistema

+ mostrarOpciones(): void

+ hacerReserva(Auto auto, Cliente cliente, Sede sedeFinal, Sede sedeInicial, String fechaInicial, String fechaFinal, String seguro, Integer rangoHoras): void

+ hacerPagoInicial(): void

Consola_Empleado

sistema: Sistema

+ mostrarOpciones(): void

+ setDisponibilidad(Auto auto, boolean disponible): void

+ mandarALimpieza(Auto auto, boolean limpiar, String fechaDisponible)

+ mandarAMantenimiento(Auto auto, boolean mantenimiento, String fechaDisponible)

+ registrarPagoFinal(): void

+ registrarConductorAdicional(Alquiler alquiler, ArraList<Cliente>): void

Consola_Admin

sistema: Sistema

+ mostrarOpciones(): void

+ agregarAuto(Auto auto): void

+ eliminarAuto(Auto auto): void

+ crearEmpleado(...todos los atributos...): void

+ crearCliente(...todos los atributos...): void

Cliente

- nombre : String

- contacto : String

- fechaNacimiento : String

- nacionalidad : String

- cedulaFoto : File

- licencialD : int

- licenciaFechaExp : String

- licenciaFechaVen : String

- licenciaFoto : File

- tarjetaCredito : String[]

Cliente(... todos los atributos...)

«interface» Usuario

login: String

password: String

getLogin(): String

getPassword(): String

Empleado

- autoridad : String

- sede : String

Empleado(...todos los atributos...)

Auto

- categoria : String

- placa : String

- marca : String

- modelo : String

- color : String

- tipoTransmision: String

- alquilado : Alquiler

- sedeActual : Sede

- mantenimiento: boolean

- limpieza: boolean

- disponible: boolean

- fechaEstimadaDisponible: String

- historialDeAlquileres: ArrayList<Alquiler>

Auto(...todos los atributos...)

setDisponibilidad(boolean disponible): void

setLimpieza(boolean limpiar): void

setMantenimiento(boolean arreglar): void

setFechaEstimadaDisponible(String fecha): void

Asuma que todo atributo privado tiene un metodo "getAtributo()". Como hay demasiados atributos solo se escriben otros metodos mas importantes

Inventario

- carrosTotales : HashMap<String, Auto>

- carroTipo1 : HashMap<String, Auto>

- carroTipo2 : HashMap<String, Auto>

- carroTipoN...

- carrosEnMantenimiento : HashMap<String, Auto>

- carrosEnLimpieza : HashMap<String, Auto>

Inventario(...todos los atributos...)

añadirAuto(Auto auto): void

Sistema

- tarifas : HashMap<String, double>

- seguros : HashMap<String, double>

- inventarioTotal : Inventario

- historial : HashMap<String, Alquiler>

- SedesTotales: HashMap<String, Sede>

+ agregarAuto(Auto auto): void

+ eliminarAuto(Auto auto): void

+ cambiarAutoDeSede(Sede sede, Auto auto): void

+ leerYGuardarDatos(): void

+ nuevoAlquiler(): void

+ agregarNuevoEmpleado(Empleado empleado): void

+ agregarNuevoCliente(Cliente cliente): void

+ haySuficientesAutos(String categoria, String fechaInicial, String fechaFinal): boolean

+ guardarPago(): void

Base_de_Datos

- clientes : File

- autos : File

- pagos : File

- Alquileres : File

- sedes : File

Aunque el Sistema lee datos inicialmente en el arranque de la aplicacoín, otras clases pueden accedera la base de datos para editar y/o agregar informacion

Alquiler

- cliente : Cliente

- estado : String

- sedeInicio : Sede

- sedeFinal : Sede

- fechaInicio : String

- fechaFinal : String

- numeroAlquiler : int

- factura: File

- conductoresAdicionales: ArrayList<String, Cliente>

- bloqueoTarjeta: boolean

Alquiler(...todos los atributos...)

+ añadirConductor(Cliente cliente): void

+ setBloqueoTarjeta(boolean bloq): void

+ generarYGuardarFactura(): void

Sede

- nombre : String

- ubicacion : String

- horarios: String[7]

- adminLocal : Empleado

- inventario : Inventario

- empleados: HashMap<String, Empleado>

Sede(...todos los atributos...)

+ añadirEmpleado(Empleado empleado): void

+ consultarHorarios(): String[7]

ALQUILER Y RESERVAS