



**UAI**

**Universidad  
Abierta  
Interamericana**

# **PROGRAMACIÓN ORIENTADA A OBJETOS**

***GUÍA DE EJERCICIOS PRÁCTICOS***

Mg. Christian A. Parkinson

---

## **EJERCICIO Nº 1**

Una librería necesita un programa que le permita gestionar **sus ventas** de sus respectivos productos.

De cada producto se registran las siguientes características:

- Código de Barras
- Nombre del producto
- Descripción
- Precio

Desarrollar una aplicación Orientada a Objetos utilizando C# que permita:

- 1) Recuperar bajo demanda la recaudación total de la librería.
- 2) Mostrar el subtotal de la venta. **Subtotal de venta de la clase venta**
- 3) Controlar la carga de datos.

## **EJERCICIO Nº 2**

Una estación de servicio necesita un programa que le permita administrar sus ventas, para ello cuenta con 3 surtidores, cada surtidor en su interior estará cargado con 1 tipo de nafta.

Los tipos de naftas junto con sus tarifas por litro son:

- Normal: \$17.20
- Súper: \$18.85
- Premium: \$21.30

1 litro => Precio

Venta(Nafta)

cantidad

Tipo

precio = cant \* precio.

Desarrollar una aplicación Orientada a Objetos utilizando C# que permita:

- 1) Obtener la recaudación total de la estación de servicio. ●
- 2) Obtener la recaudación total de un surtidor. ●
- 3) Obtener el surtidor que más recaudó.
- 4) Obtener el surtidor que menos recaudó.
- 5) Obtener el surtidor que más clientes tuvo.
- 6) Calcular el porcentaje de venta de cada tipo de nafta.
- 7) Calcular el porcentaje de recaudación de cada tipo de nafta.
- 8) Recargar el surtidor cuando éste haya quedado vacío.
- 9) Mostrar el surtidor con mayor cantidad recargas.
- 10) Calcular el promedio de ventas por surtidor.
- 11) Calcular el promedio de recaudación por surtidor.

NaftaDisponible

## **EJERCICIO Nº 3**

Una cafetería cuenta con varias de máquinas de preparar café.

Cada máquina puede hacer 1 solo tipo de café. Los tipos de café con sus precios por litro son:

- Arábigo \$50.
- Robusto: \$58.

El café se puede servir en diferentes vasos:

- Vaso Chico: 50 cm<sup>3</sup>
- Vaso Mediano: 100 cm<sup>3</sup>
- Vaso Grande ¼ litro.

Desarrollar una aplicación orientada a objetos utilizando C# que permita:

- 1) Obtener la recaudación total del negocio.
- 2) Obtener la recaudación por máquina de café.
- 3) Mostrar la máquina que más café sirvió.
- 4) Mostrar la máquina que menos café sirvió.
- 5) Mostrar la máquina que más dinero recaudó.
- 6) Mostrar la máquina que menos dinero recaudó.
- 7) Mostrar el café que más veces se sirvió
- 8) Mostrar el café que menos veces se sirvió.
- 9) Mostrar el café que dejó mayor recaudación
- 10) Mostrar el café que menos recaudación dejó.
- 11) Mostrar el porcentaje de venta y recaudación de cada café.
- 12) Mostrar el porcentaje de venta y recaudación de cada máquina.
- 13) Informar y recargar cada vez que una máquina se encuentre vacía.
- 14) Mostrar la máquina que mayor cantidad de veces se recargó.

## **EJERCICIO Nº 4**

Un after office vende diferentes tipos de cerveza tiradas de barril, las mismas se describen a continuación:

- Rubia de origen nacional, cuyo valor por litro es de \$50.
- Negra de origen nacional, cuyo valor por litro es de \$58.
- Rubia de origen extranjero cuyo valor por litro es de \$65.

Cada barril puede tener un solo tipo de cerveza, y dada las dimensiones del negocio, el sistema debe contemplar el agregado de barriles en función del tiempo.

Cada venta se ve afectada por un cálculo de incremento de valor de la cerveza en un 5%, que depende del tipo de vaso donde se sirve:

- Chopp de 3/4 litros
- Vaso de 1/2 litro
- Jara de 3 litros

Desarrollar una aplicación Orientada a Objetos utilizando C# que permita:

- 1) Calcular la recaudación total del after.
- 2) Obtener el barril más y menos cerveza sirvió, y mostrar las cantidades.
- 3) Obtener el barril que más dinero recaudó.
- 4) Mostrar la cerveza más vendida.
- 5) Mostrar la cerveza que generó más ganancia.
- 6) Obtener la cerveza que se vendió mayor cantidad de litros.
- 7) Obtener la cerveza que se vendió menor cantidad de litros.
- 8) Obtener el tipo de vaso que más se solicitó.
- 9) Calcular el porcentaje de venta de cada tipo de cerveza.
- 10) Calcular el porcentaje de recaudación de cada tipo de cerveza.
- 11) Calcular el porcentaje de venta de cada barril.
- 12) Calcular el porcentaje de recaudación de cada barril.

## **EJERCICIO Nº 5**

Un polideportivo necesita un sistema que le permita administrar los alquileres de sus diferentes canchas. Cada cancha posee un valor de base por hora, y en algunos casos pueden tener agregados, se detalla a continuación:

- Canchas de tenis:
  - Valor: \$200.
  - Opcional: 1 árbitro, adiciona \$100 al valor de la cancha y la ganancia del juez es de \$50.
- Canchas de fútbol 5:
  - Valor: \$500
  - No posee adicionales.
- Canchas de fútbol 7:
  - Valor: \$650.
  - Opcional: 1 árbitro, adiciona \$100 al valor de la cancha y la ganancia del juez es de \$50.
- Canchas de fútbol 11:
  - Valor: \$800.
  - Opcional 1: 1 árbitro, adiciona \$100 al valor de la cancha y la ganancia del juez es de \$50.
  - Opcional 2: Agregando el “opcional 1” se puede agregar 2 jueces de línea al alquiler, incrementando \$90 al costo total, y la recaudación de cada juez de línea es de \$35.

De cada juez se registra, el nombre, apellido y el legajo.

Desarrollar una aplicación Orientada a Objetos utilizando C# que permita:

- 1) Obtener la recaudación total del polideportivo.
- 2) Obtener la ganancia total del polideportivo (Descontando los gastos de los jueces).
- 3) Obtener la cancha que más se alquiló.
- 4) Obtener la cancha que más recaudó.
- 5) Obtener el juez que más partidos dirigió.
- 6) Obtener el juez que con mayor recaudación.
- 7) No permitir que un juez pueda dirigir 2 partidos en simultáneo.
- 8) No permitir que los alquileres de las canchas se superpongan, administrarlos por horarios.

## **EJERCICIO Nº 6**

Un hotel turístico requiere de un programa que le permita administrar y gestionar las reservas de las diferentes habitaciones que maneja.

De cada habitación se conoce el número, los diferentes artefactos que posee (televisión, aire, frigo bar, etc.), y se categorizan por:

- Habitación Simple: Posee una cama de una plaza y tiene un costo de \$200 por Noche.
- Habitación Doble Matrimonial: Tiene un costo de \$350 por noche y posee una cama de dos plazas.
- Habitación Triple: Tiene un costo de \$550 por noche y posee una cama matrimonial plus y una cama extra.
- Habitación Cuádruple: Tiene un costo de \$700 por noche y posee una cama matrimonial y cama-cucheta.

Hay habitaciones que poseen vista al mar que carecen en un 15% el valor de la misma, el adicional de la reserva de la cuna añade \$50 por noche.

Desarrollar una aplicación Orientada a Objetos utilizando C# que permita:

- 1) Registrar una reserva por habitación con cada uno de los integrantes, de los integrantes se conoce el nombre, apellido, la fecha de nacimiento, el DNI y la edad.
- 2) Registrar la ocupación de la habitación (Primero se realiza la reserva y después tiene que tener otro mecanismo que concrete dicha reserva).
- 3) Cargar el depósito de la reserva, el mismo será a partir como mínimo un 10% del total del paquete contratado.
- 4) Formulario de cancelación de reserva teniendo en cuenta los siguientes puntos:
  - a) Si la cancelación de la reserva es inferior a los dos días el valor del depósito se pierde (sólo el 10% de mínimo) el resto se devuelve.
  - b) Si se encuentra dentro de la semana, se reintegra el 50% del mínimo más el excedente de ese depósito.
  - c) Si es mayor de una semana se reintegra el total del valor depositado.
- 5) Cada reserva debe registrar la fecha de checkin, la fecha del checkout, la habitación, los pasajeros y los adicionales (Por ejemplo: la cuna, el consumo del frigobar, otros consumos).
- 6) Los pagos de todos los totales se imputan a la hora de registrar el checkout.
- 7) Mostrar que habitación se solicitó más en el lapso de un periodo de fechas.
- 8) Mostrar el pasajero que se hospedó mayor cantidad de veces en el hotel.
- 9) Obtener la recaudación total del hotel filtrada por periodos de fechas.
- 10) Obtener la habitación que se ocupó mayor cantidad de veces total y filtrada por fechas.
- 11) Controlar que las reservas no se superpongan.

## **EJERCICIO Nº 7**

Un museo dedicado al estudio de animales prehistóricos necesita un sistema que le permita gestionar las diferentes atracciones y las visitas.

Las atracciones se dividen en:

- 1) Sector de animales acuáticos
- 2) Sector de animales terrestres
- 3) Sector de animales de aire.

A cada atracción se le debe cargar los animales que corresponda.

De cada animal se registrará el nombre, el lugar de hábitat, junto con la alimentación, en el caso de tratarse de un animal carnívoro el sistema deberá permitir seleccionar animales previamente cargados, también se registrará el período de existencia expresado en años.

En el caso de los animales terrestres, se deberá registrar si es bípedo, cuadrúpedo, si posee exoesqueleto, y en el caso de las aves, se tiene que poder registrar también, la distancia de visión y la velocidad de vuelo.

El costo base de cada atracción es de \$50, cada entrada posee al menos una atracción, en caso de un pasaporte con dos atracciones se aplicara un 10% de descuento y en el caso, de adquirir todas las atracciones se aplica un 30% de descuento del valor de compra total.

Cada pasaporte pertenecerá a una persona y es intransferible obligando a registrar el nombre apellido y fecha de nacimiento de la persona registrada.

En caso de jubilados (mayores de 65 años) y de menores de 12 años se le aplicara un 50% de descuento del total a abonar por la entrada.

Desarrollar una aplicación Orientada a Objetos utilizando C# que permita:

- 1) Realizar la venta de entradas con sus respectivos descuentos.
- 2) Calcular el total recaudado por el museo.
- 3) Obtener la atracción con mayor cantidad de visitantes.
- 4) Obtener el Sector con mayor recaudación.
- 5) Recuperar la cantidad de ventas de cada atracción y su total recaudado.
- 6) Recuperar que cantidad de animales posee cada atracción
- 7) Generar un formulario de consulta que permita:
  - a. Buscar un animal por nombre.
  - b. Mostrar toda la información correspondiente al animal seleccionado.
  - c. Editar características del animal, (alimentación, período, etc.)
  - d. El nombre y el tipo de animal no debe poder editarse.
  - e. Buscar todos los animales comprendido entre un período de años.
  - f. Buscar todos los animales comprendido entre un período de años discriminado por sector.



## **EJERCICIO Nº 8**

Una empresa ferroviaria necesita un sistema que permita administrar los viajes que realiza en a diferentes destinos. En cada viaje se registra el origen, el destino, la distancia que existe entre ambos, contemplando la posibilidad que pueden existir destinos intermedios entre el origen y el destino final.

El valor de cada viaje en principio estará calculado en un valor fijo, el mismo sufrirá un descuento de un 2% por cada estación intermedia del recorrido. El tiempo de viaje deberá calcularse en función de la distancia entre origen y final, y adicionando 10 minutos por cada estación intermedia, y teniendo en cuenta la velocidad promedio del tren.

De cada estación se conoce, el nombre, la localidad y la provincia, y se sabe que cada localidad puede tener varias estaciones. En cuanto a los pasajeros se debe poder registrar el nombre, apellido, fecha de nacimiento, DNI, y en el caso de menores de edad si cuentan con la autorización de los padres para poder viajar.

Los menores de 3 años no ocupan asiento, deben registrarse pero sin generar costo alguno.

En cuanto a las formaciones, cuentan con diferentes vagones, con butacas numeradas y clasificados por "turista", "pullman" y "ejecutivo". Sabiendo que la categoría "pulman" es un 10% más caro que la "turista" y el "ejecutivo" es un 7% más caro que "pullman".

Cada pasaje que se emite mostrará la información completa del viaje, del pasajero, la categoría y butaca en la que viaja, y en caso de viajar con menores de 3 años también deberá figurar. Además el costo total del pasaje.

Desarrollar una aplicación Orientada a Objetos utilizando C# que permita:

- 1) Obtener la recaudación total de la empresa.
- 2) Obtener la recaudación discriminado por rutas y/o por categorías.
- 3) Obtener la cantidad total de pasajeros.
- 4) Obtener la cantidad de pasajeros discriminado por rutas y/o por categorías.
- 5) Permitir la venta de pasajes para formaciones que se encuentren con lugares disponibles en su respectivo viaje.
- 6) Ningún elemento puede estar hardcodeado.

## **EJERCICIO Nº 9**

Un observatorio astronómico necesita un sistema para poder registrar los diferentes cuerpos celestes encontrados. Los diferentes elementos observados son: estrellas, planetas y satélites naturales.

Cada registro debe dejar asentado: fecha del encuentro, persona que lo observó, objeto encontrado y la distancia expresada en años luz desde la tierra. Los objetos a registrar se describen a continuación:

- Estrellas:
  - Masa.
  - Temperatura expresada en grados Celsius.
  - Edad.
  - Diámetro.
  - Nombre.
  - Color (Roja, Naranja , Amarilla, Blanca , Azul)
  - Tipo de estrella (Enana - Gigante)
- Planetas:
  - Masa.
  - Edad.
  - Temperatura.
  - En caso de pertenecer a un sistema planetario, pueden contener satélites.
- Planetas dentro de un sistema planetario simple:
  - Órbita alrededor de 1 estrella.
  - Distancia en años luz a con respecto a la estrella que orbita.
  - Satélites.
  - Determinar si se encuentra en la zona “Ricitos de oro”
  - Determinar si en función de la temperatura, elementos de la atmosfera y la zona es potencialmente habitable.
- Planetas dentro de un sistema planetario Binario:
  - Órbita alrededor de 2 estrellas.
  - Determina también la distancia con respecto a la segunda estrella.
- Satélites:
  - Nombre.
  - Indicar si posee acoplamiento de marea.
  - Masa.
  - Edad.

Cada sistema de estrellas se deberá agrupar dentro de una constelación.

Del descubridor del cuerpo celeste se registrará el nombre, apellido, fecha de nacimiento y por último el legajo en caso de ser personal del observatorio.

Desarrollar una aplicación Orientada a Objetos utilizando C# que permita:

- 1) Carga y edición de los diferentes cuerpos celestes.
- 2) Convertir dinámicamente años luz en kilómetros y viceversa.
- 3) Convertir dinámicamente las temperaturas de Grados Celsius a Fahrenheit y viceversa.
- 4) Buscar planetas por:
  - a. Nombre.
  - b. Estrella en la que orbita.
  - c. Potencialmente habitables.
- 5) Buscar estrellas por:
  - a. Tipo basado en el color y Tipo.
  - b. Temperatura.
  - c. Constelación.
- 6) De cada búsqueda mostrar toda la información con de dicho cuerpo celeste.

## **EJERCICIO Nº 10**

Un jardín maternal necesita un sistema para poder administrar las inscripciones y el cobro de las cuotas de los chicos que concurren. El jardín cuenta con las salas siguientes salas

- Lactarios: concurren bebés de 0 a 9 meses, cupo 12 bebés y 2 maestras.
- Deambuladores: concurren niños de 9 meses a 1 año y medio, cupo 15 niños y 2 maestras.
- Deambuladores II: son para chicos de 1 año y medio a 2 años, cupo 15 chicos, y 1 maestra.
- Sala de 2: para niños de 2 años, cupo 18 niños y 1 maestra.
- Sala de 3: para niños de 3 años, cupo 20, y una docente.

De las maestras se conoce, el nombre, apellido, el legajo y el sueldo en bruto siendo este la mitad para el caso de las maestras no recibidas.

En cuanto a las cuotas varían dependiendo del tipo de inscripción:

- Cuota base: corresponde a un turno específico (mañana o tarde).
- Cuota doble: corresponde al alumno que asiste a doble jornada, la misma se encarece en un 75%.
- Cuota doble con comedor: incluye un monto correspondiente al almuerzo.

Para el caso de los alumnos cuenten con un hermano en el establecimiento la cuota recibirá una bonificación del 40% sobre el valor total.

Desarrollar una aplicación Orientada a Objetos utilizando C# que permita:

- Gestionar la inscripción y bajas de alumnos.
- Informar por medio de un evento cuando se desee inscribir a un alumno en una sala sin cupo.
- Gestionar las maestras y las salas.
- Gestión mensual de cobro de cuotas
- Calcular el total mensual de recaudación
- Calcular la ganancia mensual del jardín
- Imprimir cuota de un alumno seleccionado
- Imprimir la lista de alumnos ordenada por apellido en formulario independiente

Se requiere:

- Desarrollar con Windows Application y Dll
- Uso de eventos
- Uso de interfaces

## **EJERCICIO Nº 11**

Un hospital privado necesita un sistema que le permita gestionar las historias clínicas de sus pacientes. El hospital está dividido en diferentes especialidades, donde cada especialidad tiene diferentes estudios que el paciente puede realizarse. Cada estudio tiene un costo, y debe estar acompañado de la orden médica con la firma médico fue el que lo solicitó, además el cupo para realizar los estudios es limitado por día.

Cada médico clínico tiene un costo, y este costo se encarece en un 15% por cada especialización que éste posea. También los médicos tienen agendas de pacientes, para lo cual, se debe reservar el turno con antelación y evitar que los turnos se superpongan. Los médicos de guardia son los únicos que pueden atender demanda espontánea.

Desarrollar una aplicación Orientada a Objetos utilizando C# que permita:

- 1) Carga de especialidades, médicos y pacientes.
- 2) Carga de derivaciones y estudios.
- 3) Registrar los resultados de los estudios.
- 4) Generar la receta médica correspondiente al diagnóstico del médico.
- 5) Listar La historia clínica de un paciente ordenada de forma descendiente por fecha.
- 6) Listar la historia clínica de un paciente filtrada por especialidad.
- 7) Obtener el costo total que el paciente debe abonar por su tratamiento.
- 8) Obtener la ganancia total del hospital.
- 9) Listar de forma ascendente las especialidades ordenada por ganancia acumulada.
- 10) Listar de manera descendente las especialidades ordenadas por cantidad de pacientes.

Se requiere:

- Desarrollar con Windows Application y Dll
- Uso de eventos
- Uso de interfaces

## **EJERCICIO Nº 12**

Una biblioteca necesita un sistema para poder gestionar los préstamos de sus respectivos ejemplares. La gestión del préstamo consta del registro de la persona, los ejemplares que desea llevarse, como 1 máximo en caso de ser su primer préstamo, caso contrario 3, se pacta una fecha de devolución no superior a los 5 días hábiles, tiempo en el que debe ser devolverse cada uno de los libros prestados.

En el caso de excederse con las devoluciones se aplicarán diversas sanciones:

- La primera vez se suspenderá al cliente por 1 semana corrida.
- La segunda vez se suspenderá durante 30 días hábiles.
- La tercera se suspenderá durante 30 días mínimo hasta que abone una multa de \$50 por día de retraso.

Los ejemplares se categorizan por:

- Libros
- Libros con material digital
- Revistas

De los libros se conoce el título, autor (nombre, apellido, fecha de nacimiento, y en el caso que corresponda fecha de defunción), el año, la editorial, y la edición.

De los libros con material digital, se conoce además la cantidad de cds que acompañan.

Y de las revistas se registras el nombre, el año, el número, y el tipo de entrega (mensual / quincenal / diario).