Tarea 1. Valor 1%

Temas de Estudio

1. Tema 1 Introducción al lenguaje de programación
2. Tema 2 Particularidades del lenguaje C#
3. Tema 3 Manejo de excepciones en C#

Objetivo

Aplicar los conceptos generales del lenguaje de programación C#, con el objetivo de profundizar en la semántica y sintaxis del lenguaje.

Software de Desarrollo

Visual Studio Community 2022 (C#, .Net Framework 4.8 o Net 6.0)

Desarrollo

La empresa **REST-UNED** cuenta con una cadena de renta restaurantes, la cual cuenta con varias sucursales por todo el territorio nacional. Debido a la creciente demanda de servicios se encuentra en una etapa de crecimiento del negocio y cada vez más se expanden por todo el país. Ahora bien, actualmente no cuentan con un sistema de información que les permita realizar una gestión adecuada de los restaurantes, razón por la que le han contratado para que diseñe y programe un sistema de consola en C# que cuente con las siguientes funcionalidades:

El sistema debe contar con un menú principal con las siguientes opciones:

- **Registrar Restaurante**
- **Registrar Categoría Plato**
- **Registrar Plato**
- **Registrar Clientes**
- **Registrar Platos por Restaurante**
- **Consultar Restaurante**
- **Consultar Categoría Plato**

- **Consultar Plato**
- **Consultar Clientes**
- **Consultar Platos por Restaurante**

Registrar Restaurante

Está opción del menú le permite al usuario registrar o agregar los datos relacionados con los restaurantes con que cuenta la empresa. Se debe poder registrar la siguiente información:

- Id de restaurante (int), este valor debe ser único por lo que se debe validar en el sistema que no existan valores repetidos
- Nombre de restaurante (string)
- Dirección (string)
- Estado (bool), activo o inactivo
- Teléfono (string)

Una vez capturada la información se debe agregar a un arreglo de objetos de tipo restaurante. El arreglo debe ser de 20 posiciones

Registrar Categoría de Plato

Debe permitir al usuario registrar las categorías de platos con que cuenta la empresa, pudiendo registrar la siguiente información:

- Id categoría (int), este valor debe ser único por lo que se debe validar en el sistema que no existan valores repetidos
- Descripción (string)
- Estado (bool), activo o inactivo

Una vez capturada la información se debe agregar a un arreglo de objetos de categoría. El arreglo debe ser de 20 posiciones.

Registrar Platos

Debe permitir al usuario registrar los platos con que cuenta la empresa, pudiendo registrar la siguiente información:

- Id Plato (int), este valor debe ser único por lo que se debe validar en el sistema que no existan valores repetidos
- Nombre (string)

- Precio (int)
- Id Categoría (Objeto Categoría) debe almacenar un objeto de tipo Categoría, por lo tanto, debe validar que exista en el arreglo de Categorías.

Una vez capturada la información se debe agregar a un arreglo de objetos de tipo Plato. El arreglo debe ser de 20 posiciones

Registrar Clientes

Debe permitir al usuario registrar los clientes del restaurante, pudiendo registrar la siguiente información:

- Identificación (string) se debe validar que no existan identificaciones repetidas
- Nombre (string)
- Primer apellido (string)
- Segundo apellido (string)
- Fecha de nacimiento (DateTime)
- Género (char)

Una vez capturada la información se debe agregar a un arreglo de objetos de tipo Cliente. El arreglo debe ser de 20 posiciones

Registrar Platos por Restaurante

Funcionalidad que debe permitir al usuario registrar (asociar) los platos que están disponibles en cada restaurante. Para ello se de crear una clase “PlatoRestaurante”, la cual debe contener además de los atributos de id asignación, fecha de afiliación, restaurante, debe contener un arreglo de los platos asignados a un restaurante. Para esto se debe realizar lo siguiente:

- Imprimir en pantalla la lista de restaurantes disponibles.
    - Se debe validar que al menos exista un restaurante registrado para realizarle asignación
    - Para cada restaurante registrado debe mostrar los siguientes datos: Id restaurante, Nombre, Dirección
    - Solo debe mostrar los restaurantes activos
- Solicitar al usuario que digite el ID de restaurante
- Imprimir en pantalla la lista de platos disponibles para asignar al restaurante
    - Se debe validar que al menos exista un plato registrado para ser asociado
    - Para cada plato de la lista se debe mostrar la siguiente información: id plato, nombre,precio

- Solicitar al usuario que digite el Id Plato de plato que desea asociar al restaurante
- Se debe preguntar al usuario si desea agregar más platos al restaurante. Si la respuesta es afirmativa, entonces se vuelve a mostrar la lista de platos para que el usuario nuevamente seleccione e indique un id.
- La clase PlatoRestaurante debe tener un arreglo de 10 posiciones donde se agreguen los platos
- Imprimir en pantalla si la asociación de plato se creó de forma exitosa

Para esta opción se debe poder registrar la siguiente información para PlatoRestaurante:

- Id asignación (int)
- Fecha (DateTime)
- Restaurante (Objeto Sucursal)
- Platos (Arreglo de objeto Platos con 10 posiciones)

La asociación de platos a restaurante se debe almacenar en un arreglo de objetivos tipo PlatoRestaurante de 20 posiciones.

Consultar Restaurantes

Esta opción permite al usuario el imprimir en pantalla todos los restaurantes con que cuenta la empresa, se debe mostrar todos los atributos

Consultar Categoría de Plato

Esta opción permite al usuario imprimir en pantalla las categorías de platos registrados, se debe mostrar todas las propiedades

Consultar Platos

Esta opción permite al usuario imprimir en pantalla todos los platos registrados, para cada plato se debe mostrar todos los atributos

Consultar Clientes

Esta opción permite al usuario imprimir en pantalla todos los clientes, para cada cliente se debe mostrar todos los atributos

Consultar Platos por Restaurante

Esta opción permite consultar los platos que tiene asociados cada restaurante, es decir, consulta los platos que están asignados a un restaurante. Debe imprimir en pantalla la información de cada asociación, es decir, id asignación, fecha afiliación, datos de restaurante (id sucursal, nombre, dirección), la información de los platos (id plato, nombre, precio) que están asignados al restaurante.

Consideraciones técnicas

Debe utilizar POO (Programación orientada a objetos) para resolver el problema.

- Las clases de objetos (Restaurante, Cliente, Categoría Plato, Plato, Plato Restaurante) no deben contener lógica para solicitar información al usuario, solo debe tener la estructura de la clase y sus propiedades.
- Las clases no deben tener métodos vacíos y recuerde utilizar nombres significativos a sus variables.

Uso de arreglos

- No deben ser arreglos de tipo “string” ni “int”, deben ser arreglos de objetos de las clases definidas.
- Cada vez que se agrega un objeto al arreglo se debe preguntar al usuario si desea agregar otro y no obligar al usuario a ingresar los 20 registros de una vez según corresponda el arreglo
- No debe utilizar colecciones, solo se permite el uso de arreglos para esta tarea.

Al salir del sistema debe solicitar confirmación al usuario ¿Está realmente seguro de que desea salir? S/N

Debe implementar el manejo de excepciones

- Si ocurre una excepción, el sistema no debe cerrarse, se debe mostrar un mensaje al usuario y manejar la excepción de forma apropiada.

La aplicación no requiere uso de formularios o GUI, debe realizarse en consola.

Honestidad Académica

https://audiovisuales.uned.ac.cr/play/player/23048

Nota Importante

Cada estudiante es responsable del contenido que entrega, si no es el archivo correcto, no podrá entregarlo posterior a la fecha establecida.

Si el contenido del archivo coincide con algún otro estudiante, o se comprueba que no es de su autoría, se expone a las sanciones indicadas en la plataforma en el documento Lineamientos ante casos de plagio

Indicaciones Importantes

- **Es obligatorio que incluya todo el directorio donde se encuentra < tarea1>.**
- La **Tarea 1** debe estar desarrollado en **Visual Studio Community 2019 o superior** que es la herramienta oficial del curso.
- El programa debe ser modular, utilizando de la mejor manera funciones definidas por usted.
- **Los trabajos deben realizarse en forma individual. Dentro del código del programa debe de indicar la documentación que explique cómo fue realizado el programa.**
- **Si utiliza código de algún ejemplo del libro, del algún profesor o de otra fuente que no sea de su autoría, debe de indicarlo.**
- Comprima todos los archivos en un solo archivo .zip o .rar.
- **Nombre del archivo que envía**: debe ser nombre y primer apellido del estudiante, y nombre de la tarea**. Ejemplo: JuanRojas-tarea1**.
- La entrega de la **<Tarea1>**en las fechas establecidas en la plataforma de aprendizaje en línea Moodle en el apartado que se indique.
- Si no concluyó a tiempo la tarea, debe entregar lo que pudo hacer e incluir una carta explicando las razones por las cuales no finalizó.

Rúbrica de Evaluación

Criterio	Cumple a satisfacción lo indicado en la evaluación	Cumple medianamente en lo indicado en la evaluación	Cumple en contenido y formato pero los aportes no son significantes	No cumple o no presenta lo solicitado
Presentación del código: Correcto uso de variables, nombres de métodos significativos, no presenta métodos vacíos.	5	3	1	0.5
POO y creación de las clases Restaurante, Cliente, Categoría Plato, Plato, Plato Restaurante	10	5	2	0.5
Registrar Restaurante	5	3	1	0.5
Registrar Categoría Plato	5	3	1	0.5
Registrar Plato	10	5	2	0.5
Registrar Clientes	5	3	1	0.5
Registrar Plato por Restaurante	10	5	2	0.5
Consultar Restaurante	5	3	1	0.5
Consultar Clientes	5	3	1	0.5
Consultar Categoría Plato	5	3	1	0.5
Consultar Plato	10	5	2	0.5
Consultar Platos por Restaurante	10	5	2	0.5
Correcto uso del manejo de excepciones	5	3	1	0.5
Interfaz de usuario de consola es fácil de usarse y presenta buen diseño.	10	5	2	0.5
TOTAL	**100**			