

## Creación de la Interfaz para la Práctica de Laboratorio

Para Trabajar con la Pila

### Instrucciones para el trabajo con la pila:

Crearé una pila, que le permitirá que una vez que haga clic en el botón atender, pueda deshacer la atención, para ello se implementará código para el botón de atención y código para los botones de atender y deshacer atención, primero se crearán los elementos necesarios para la interfaz, luego se les adicionara el código necesario, se crearan las clases y código necesario.

#### Creación de la Interfaz:

1.- Crear un panel, lo renombrará con el name panelAtencion

2.- En dicho panel pondrá insertar varios controles.

- Controles label, para poner los títulos
- Controles TextBox, para la entrada de datos:

#### Para la clínica veterinaria:

- Los títulos serán: títulos: Dueño, Información de Contacto, hora de Llegada y Número de Mascotas
- Los Textbox, serán para capturar los siguientes datos: nombre del dueño, Información de Contacto, hora de Llegada y Número de Mascotas quedando de la siguiente forma:

	Dueño	Info Contacto	Hora Llegada	Cent Mascotas
*				

3.- Adicionará dos controles button, Uno para atender y otro para deshacer atención, quedando el formulario de esta manera, cambiando el campo Text, de cada uno de ellos a Atender y a Deshacer Atención, respectivamente y la propiedad name cambiarla a btnAtender y btnDehacer, respectivamente, quedando el formulario de la siguiente forma:

**Para la Clínica Veterinaria**

**COLA** Fecha: 22/ 5/2025

Cola de Ingreso

Atencion en Consulta

Dueño

Información de Contacto

Hora de llegada

Numero de Mascotas

**COLA DE ATENCION**

	Dueño	Info Contacto	Hora Llegada	Cant Mascotas
*				

Atender

Deshacer Atencion

4.- Adición de código al evento click del boton atención en consulta, o atención, o atención en florería

Adicinará el siguiente código, que pondrá visible el panel de atención y ocultará en panel de la cola, además de poner visible los botones Atender y deshacer atención. Escribirá el código correspondiente:

**PanelCola->Visible = false;**

**panelAtencion->Visible = true;**

**lblTitulo->Text = "ATENCION A CLIENTE";**

**btnAtender->Visible = true;**

---

**btnDehacer->Visible = true;**

\*tomar en cuenta que deberá cambiar los nombres de los controles si es que usted puso otro nombre, es decir, el botón para atender, aquí tiene el name btnAtender, pero si en su proyecto tiene otro nombre en la propiedad name, deberá de ajustarlo a ello.

La propiedad Text, del control lblTitulo, estará en dependencia del trabajo que le toque, para veterinaria, deberá poner: Atención a cliente, para Consultorio: Atención a Paciente, y para la florería: Atención a cliente en la florería; aunque está en la libertad de poner el título que desee.

De forma que al hacer click en el botón de atención, se mostrará de la siguiente forma su formulario:

**Para la Veterinaria**

5.- Deberá adicionar lo contrario en el botón de Cola de ingreso, ya que al hacer click en él habrá que esconder los botones atender y deshacer atención y mostrar la ventana para ingresar datos en la cola, por lo que deberá asociar al botón de cola de atención el siguiente código:

**panelCola->Visible = true;**

**panelAtencion->Visible = false;**

**lblTitulo->Text = "COLA";**

**btnAtender->Visible = false;**

**btnDeshacer->Visible = false;**

---

**Creación de la clase para trabajar con la pila:**

Crearé la clase pila según su trabajo, adicionando dicha clase y su código a la namespace que tiene ya creado, para el trabajo con las clases:

#### Para la veterinaria

```
public ref class pila_veterinaria
{
private:
    Stack <datos^>^ pila; //defino la pila
public:
    pila_veterinaria() {
        pila = gcnew Stack<datos^>();
    }
    void Agregar(datos^ c) {
        pila->Push(c);
    }
    datos^ Eliminar() {
        if (pila->Count == 0) return nullptr;
        datos^ dato = pila->Peek();
        pila->Pop();
        return dato;
    }
    int ObtenerCantidad() {
        return pila->Count;
    }
};
```

#### Trabajando con el objeto pila:

1.- Adicionar la definición del objeto pila

#### Para veterinaria:

```
12     using namespace System::Data;
13     using namespace System::Drawing;
14
15     /// <summary>
16     /// Resumen de MyForm
17     /// </summary>
18     public ref class MyForm : public System::Windows::Forms::Form
19     {
20     public:
21         cola_veterinaria^ cola_clientes = gcnew (cola_veterinaria);
22         pila_veterinaria^ pila_clientes = gcnew(pila_veterinaria);
23
24
25
```

#### Para floreria:

## 2.- Adicionar código al evento clic del botón atender.

**Para la Clinica Veterinaria:**

```
datos^ primero;
// Sacando de la cola, para atender
primero = cola_clientes->EliminarInicio();

//Elimina de la tabla visual
if (dataGridView1->Rows->Count > 0) {
pila_clientes->Agregar(primero); // agrega a la pila, el elemento eliminado de la cola

// Accede a la primera fila del DataGrid
DataGridViewRow^ fila = dataGridView1->Rows[0];
// Guarda los valores en los TextBox
txtCDueño->Text = fila->Cells[0]->Value != nullptr ? fila->Cells[0]->Value->ToString() : "";
txtCContacto->Text = fila->Cells[1]->Value != nullptr ? fila->Cells[1]->Value->ToString() : "";
txtCHora->Text = fila->Cells[2]->Value != nullptr ? fila->Cells[2]->Value->ToString() : "";
txtCCantMascotas->Text = fila->Cells[3]->Value != nullptr ? fila->Cells[3]->Value->ToString() : "";

// Elimina la fila del grid
dataGridView1->Rows->RemoveAt(0);
// Opcional: seleccionar la nueva primera fila si existe
if (dataGridView1->Rows->Count > 0) {
dataGridView1->ClearSelection();
dataGridView1->Rows[0]->Selected = true;
}
}
```

## 3.- Adicionar código al evento clic del botón Desatender.

**Para Veterinaria:**

```
datos^ primero;
primero = pila_clientes->Eliminar(); //se Saca de la pila
if (primero!=nullptr) {
cola_clientes->AgregarInicio(primero); //Se vuelve a poner a la cola
//Ahora hay que adicionarlo al Grid
dataGridView1->Rows->Insert(0, primero->getDueño(), primero->getContacto(), primero->getHora_llegada(), primero->getcantmascotas());
dataGridView1->SelectionMode = DataGridViewSelectionMode::FullRowSelect;
dataGridView1->CurrentCell = dataGridView1->Rows[0]->Cells[0];
clear_atencion();
}
}
```

```
void clear_atencion() {  
    txtCDueño->Text = " ";  
    txtCContacto->Text = " ";  
    txtCHora->Text = " ";  
    txtCCantMascotas->Text = " ";  
}
```

\*Adicione al código del botón atender, al inicio, la invocación a la función `clear_atencion()` , para que deje en blanco todos los textbox de la ventana de atención.