

Nombre: Registrar película	#include <iostream> #include <string> using namespace std;
Parámetros: string título; int duración; string género; int año;	void registrarPelícula (string título, int duración, string género, int año) { // Aquí se agrega el código para almacenar los datos en la base de datos
Retorno:  void	cout << "Película registrada:" << endl; cout << "Título: " << título << endl; cout << "Duración: " << duración << " minutos" << endl; cout << "Género: " << genero << endl; cout << "Año: " << año << endl;
Descripción: Esta función permite el registro de nuevas películas en la base de datos. Toma como entrada el título de la película, su duración, el género y el año de lanzamiento. La función almacena estos datos en la base de datos y no retorna ningún valor.	}  int main () { string título = "Inception"; int duración = 148; string genero = "Ciencia Ficción"; int año = 2010;  registrarPelícula (título, duración, genero, año); return 0;  }

Nombre: Consultar funciones disponibles	<pre> #include &lt;iostream&gt; #include &lt;string&gt; #include &lt;vector&gt; using namespace std;  struct Funcion {     string hora;     string Sala; };  void consultarFuncionesDisponibles (string titulo) {     // Datos de ejemplo para funciones de la película     vector&lt;Función&gt; funciones = {         {"14:00", "Sala 1"},         {"17:00", "Sala 3"},         {"20:00", "Sala 2"}     };      cout &lt;&lt; "Funciones disponibles para la película \"" &lt;&lt; titulo &lt;&lt; "\":" &lt;&lt; endl;     for (const auto&amp; función: funciones) {         cout &lt;&lt; "Hora: " &lt;&lt; funcion.hora &lt;&lt; ", Sala: " &lt;&lt; funcion.sala &lt;&lt; endl;     } }  int main () {     string titulo = "Inception";      consultarFuncionesDisponibles(titulo);      return 0; } </pre>
Parámetros: String título;	
Retorno:  void	
Descripción: Esta función permite consultar las funciones disponibles para una película específica. Toma como entrada el título de la película y muestra las funciones, horarios y salas en los que se proyectará, esta muestra los datos directamente	

<b>Nombre:</b> Venta de entradas	<pre>#include &lt;iostream&gt;</pre>
<b>Parámetros:</b> string titulo int hora int cantidadEntradas float precioPorEntrada	<pre>#include &lt;string&gt;  using namespace std;  // Función para procesar la venta de entradas y retornar el total a pagar</pre>
<b>Retorno:</b> void	<pre>float ventaEntradas(string titulo, int hora, int cantidadEntradas, float     precioPorEntrada) {      float totalAPagar = cantidadEntradas * precioPorEntrada;</pre>
<b>Descripción:</b> Función que procesa la venta de entradas, calculando el total a pagar en base a la cantidad y precio por entrada. Muestra un mensaje de confirmación y retorna el total.	<pre>    cout &lt;&lt; "Venta exitos!" &lt;&lt; endl;      cout &lt;&lt; "Titulo de la función: " &lt;&lt; titulo &lt;&lt; endl;      cout &lt;&lt; "Hora: " &lt;&lt; hora &lt;&lt; " hrs" &lt;&lt; endl;      cout &lt;&lt; "Total a pagar: \$" &lt;&lt; totalAPagar &lt;&lt; endl;      return totalAPagar; // Retorna el total a pagar }  int main() {      string titulo;      int hora, cantidadEntradas;      float precioPorEntrada;      // Solicitar datos al usuario      cout &lt;&lt; "Ingrese el título de la función: ";      getline(cin, titulo);      cout &lt;&lt; "Ingrese la hora de la función (solo la hora en formato 24 horas): ";      cin &gt;&gt; hora;      cout &lt;&lt; "Ingrese la cantidad de entradas: ";      cout &lt;&lt; "Ingrese el precio por entrada: ";      cin &gt;&gt; precioPorEntrada;      // Llamada a la función ventaEntradas y guardar el total a pagar      float total = ventaEntradas(titulo, hora, cantidadEntradas,     precioPorEntrada);</pre>

Nombre: Generar un reporte de Asistencia	#include <iostream>
Parámetros: void generar Reporte int numAsistentes double precioEntrada	#include <string>  using namespace std;
Retorno: return	struct ResultadoReporte {  int asistencia;  double ingresos;  };
Descripción: Función que procesa la venta de entradas, calculando el total a pagar en base a la cantidad y precio por entrada. Muestra un mensaje de confirmación y retorna el total.	ResultadoReporte generarReporte(const string& titulo, const string& hora, int numAsistentes, double precioEntrada) {  ResultadoReporte resultado = {numAsistentes, numAsistentes * precioEntrada};  cout << "Reporte de Asistencia\n";  cout << "Título: " << titulo << "\nHora: " << hora << endl;  cout << "Asistencia: " << resultado.asistencia << "\nIngresos: \$" << resultado.ingresos << endl;  return resultado;  }  int main() {  string titulo = "Concierto de Jazz";  string hora = "20:00";  int numAsistentes = 150;  double precioEntrada = 50.0;  ResultadoReporte resultado = generarReporte(titulo, hora, numAsistentes, precioEntrada);  }

Nombre: Recarga tarjeta CINECO	void recargarTarjeta(string codigoTarjeta, int monto) {
Parámetros: monto de recarga	if (codigoTarjeta.empty()) { cout << "Error: debe ingresar un código de tarjeta válido." << endl;
Retorno: Mensaje de que la recarga fue exitosa o si ha ocurrido un error.	} else if (monto < 1) { cout << "Error: el monto mínimo de recarga es \$1COP." << endl; } else if (monto > 500000) { cout << "Error: el monto máximo de recarga es \$500,000COP." << endl;
Descripción: Esta es la función encargada de recargar la tarjeta CINECO de los usuarios.	} else { cout << "Recarga exitosa: se ha añadido \$" << monto << " a la tarjeta Cineco con código " << codigoTarjeta << "." << endl; } }