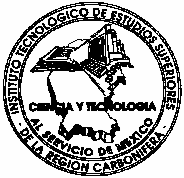
****

M.C. Adriana Ramírez Hernández

Presento:

Manual de Prácticas de la Asignatura Fundamentos de Programación

(Agosto-Dic 2017)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **INSTITUTO TECNOLÓGICO DE ESTUDIOS SUPERIORES**  **DE LA REGIÓN CARBONÍFERA**  **Formato de Prácticas** | | | TEMA: | UNIDAD: | PRÁCTICA: |
| MATERIA: | | ESPECIALIDAD: | |  | **Herencia** | **Splash Screen** |
| Programación Orientada a Objetos, Programación I | | Ing. En Sistemas, Lic. En Informática | |
| NOMBRE: | | TIPO DE PRACTICA: | DURACIÓN: |
| **Splash Screen** | | Guiada | 1 horas |
| OBJETIVOS: | | CONOCIMIENTOS PREVIOS: | |
| * Crear un formulario Splash Screen. | | * Formularios * Control Timer | |
| COMENTARIOS / SUGERENCIAS: | | | |
| * Explicar paso a paso el procedimiento crear la Splash Screen, aprovechar para explicar las propiedades básicas de los formularios. * Explicar la diferencia entre los métodos Show y ShowDialog mandando llamar al formulario Splash de las dos maneras. | | | |
| |  |  | | --- | --- | | PLANTEAMIENTO: | | | 1. Crear un formulario que sirva como Splash Screen para una aplicación en C#. | | | RESULTADOS: | | | Splash Screen: Es una imagen que aparece mientras una aplicación de computadora está cargando. Las splash screen regularmente no cubren la pantalla completa si solamente un rectángulo cerca del centro. Las splash screen son típicamente utilizadas por aplicaciones grandes mientras el programa se encuentra en proceso de carga.  Las splash screen regularmente sirven para mejorar el aspecto de una aplicación y ofrecer una mejor apariencia que puede incluso incluir animaciones y sonido.   1. Crear un formulario llamado frmPrincipal.    1. Cambiar el titulo de la ventana por “Ventana Principal”. 2. Crear un segundo formulario llamado frmSplash.    1. Cambiar la propiedad StartPosition a CenterScreen.    2. Cambiar la propiedad ShowInTaskbar a False.    3. Cambiar la propiedad TopMost a True.    4. Cambiar los valores de la propiedad Size a 500 de Width y 200 de Height.    5. Cambiar la propiedad FormBorderStyle a None.    6. Cambiar la propiedad BackColor a Red.    7. Agregar un control Panel al formulario, llamarlo pnlCentro    8. Cambiar el BackColor del control pnlCentro a White.    9. Cambiar la propiedad Dock del pnlCentro a Fill.    10. Cambiar la propiedad Padding del formulario: All a 10. 3. Agregar un Label dentro del panel y llamarlo lblNombreAplicación.    1. Cambiar el Texto del lblNombreAplicación a Mi Aplicación    2. Cambiar el Tipo, Tamaño y estilo de Font para uno mayor.    3. Cambiar el color de primer plano ForeColor a Blue. | 1. Agregar otro control Label dentro del panel y llamarlo lblAutor    1. Cambiar el Texto del lblAutor por tu nombre.    2. Cambiar el Tipo, Tamaño, Estilo y Color de Fuente del lblAutor. 2. Agregar un control ProgressBar con el nombre pgbProgreso. 3. Agregar un control Timer al formulario y llamarlo tmrAvance.    1. Cambiar la propiedad Interval del tmrAvance a 10.    2. Dar doble clic en el Timer y programar el método de la siguiente manera:   private void tmrAvance\_Tick(object sender, EventArgs e)  {  pgbProgreso.Value++;  if (pgbProgreso.Value == pgbProgreso.Maximum)  {  tmrAvance.Stop();  this.Close();  }  }   1. Programar el método frmSplash\_Load iniciando para que inicie el Timer: tmrAvance.Start() 2. Finalmente, en el constructor del frmPrincipal instanciar un objeto de la clase frmSplash y mostrarlo usando ShowDialog(). | | RETROALIMENTACIÓN: | | | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **INSTITUTO TECNOLÓGICO DE ESTUDIOS SUPERIORES**  **DE LA REGIÓN CARBONÍFERA**  **Formato de Prácticas** | | | TEMA: | UNIDAD: | PRÁCTICA: |
| MATERIA: | | ESPECIALIDAD: | |  | **Herencia** | **Persianas** |
| Programación Orientada a Objetos, Programación I | | Ing. En Sistemas, Lic. En Informática | |
| NOMBRE: | | TIPO DE PRACTICA: | DURACIÓN: |
| **Persianas** | | Guiada | 1 horas |
| OBJETIVOS: | | CONOCIMIENTOS PREVIOS: | |
| * El alumno aprenderá la aplicación de la herencia | | * Formularios | |
| COMENTARIOS / SUGERENCIAS: | | | |
|  | | | |
| |  |  | | --- | --- | | PLANTEAMIENTO: | | | Una tienda se dedica a la venta de persianas y requieren un programa que les permita dar cotizaciones a los clientes.  Para cotizar una persiana solicitan al cliente las dimensiones de la ventana donde colocaran la persiana. El costo por metro cuadrado es de $149.00.  La tienda vende dos tipos de persianas: horizontales y verticales. El costo de las persianas verticales es un 10% más alto que el de las persianas horizontales.  Además de sus ventas locales, la tienda realiza ventas a través de internet, incluyendo en el costo los gastos de envió.  El proveedor del servicio de envíos de la tienda le cobra 100 pesos por los primeros 5kg y 10 pesos el kilogramo excedente en sus envíos.  (En 5.6 kg cobraría 6kg).  Se sabe que cada metro cuadrado de persiana tiene un peso de 1200 gramos. | | | RESULTADOS: | | |  |  | | RETROALIMENTACIÓN: | | | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **INSTITUTO TECNOLÓGICO DE ESTUDIOS SUPERIORES**  **DE LA REGIÓN CARBONÍFERA**  **Formato de Prácticas** | | | TEMA: | UNIDAD: | PRÁCTICA: |
| MATERIA: | | ESPECIALIDAD: | |  | **Herencia** | **Agencia de viajes** |
| Programación Orientada a Objetos, Programación I | | Ing. En Sistemas, Lic. En Informática | |
| NOMBRE: | | TIPO DE PRACTICA: | DURACIÓN: |
| **Agencia de viajes** | | Libre | 1 horas |
| OBJETIVOS: | | CONOCIMIENTOS PREVIOS: | |
| * El alumno aprenderá la aplicación de la herencia | | * Formularios | |
| COMENTARIOS / SUGERENCIAS: | | | |
|  | | | |
| |  |  | | --- | --- | | PLANTEAMIENTO: | | | Una empresa ofrece viajes a la Ciudad de México a sus clientes con un costo de $300.00 por persona entre semana y $450 pesos los fines de semana. La empresa realiza un descuento del 10% a sus clientes frecuentes solamente en viajes de fin de semana.  Si el cliente desea un viaje con una duración mayor a un día, la empresa cobra un 20% adicional diario por el viaje.  Realizar un programa que permita realizar reservaciones de boletos para viajes de uno o varios días a partir del número de boletos requeridos y las fechas del viaje y me indique el costo total a pagar. | | | RESULTADOS: | | |  |  | | RETROALIMENTACIÓN: | | | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **INSTITUTO TECNOLÓGICO DE ESTUDIOS SUPERIORES**  **DE LA REGIÓN CARBONÍFERA**  **Formato de Prácticas** | | | TEMA: | UNIDAD: | PRÁCTICA: |
| MATERIA: | | ESPECIALIDAD: | | **Lanzamiento de Excepciones** | **1. Excepciones** | **Renta Automóvil** |
| Programación Orientada a Objetos, Programación I | | Ing. En Sistemas, Lic. En Informática | |
| NOMBRE: | | TIPO DE PRACTICA: | DURACIÓN: |
| **Renta Automóvil** | | Diagnóstico-Guiada | 1 hora |
| OBJETIVOS: | | CONOCIMIENTOS PREVIOS: | |
| * Realizar un diagnostico inicial a los alumnos para identificar el nivel de dominio que poseen de los temas de la materias de fundamentos de programación. * A partir de lo resuelto por los alumnos, introducir el nuevo tema. | | Materia: Fundamentos de Programación   * Clases y objetos * Propiedades * Condiciones | |
| COMENTARIOS / SUGERENCIAS: | | | |
| * Dejar que los alumnos intenten realizar solos la primera parte del ejercicio sin ayuda del docente la primera media hora de clase e identificar problemas que tengan al tratar de resolverlo. * Resolver el ejercicio para que aquellos alumnos que no hayan podido realizarlo por si solos conozcan la forma de solucionarlo. * Guiar al alumno en el lanzamiento y captura de excepciones. | | | |
| |  | | --- | | PLANTEAMIENTO: | | Una compañía se dedica a rentar automóviles y realiza sus cobros de la siguiente manera:  Al momento de que el cliente solicita el automóvil, toman nota del kilometraje que tiene el automóvil en ese momento.  Cuando el cliente regresa a entregar el automóvil rentado, el encargado de la compañía revisa el kilometraje nuevamente y obtiene el kilometraje recorrido.  Si recorre un total de 120 kilómetros o menos, se le cobra una tarifa fija de $ 550.00. Si el kilometraje es mayor, le cobran $ 550.00 que corresponden a la tarifa fija y $ 25.00 por kilómetro excedido.  Crear una clase que permita calcular el monto a cobrar por la renta del automóvil.  NOTA:  El kilometraje inicial debe ser menor que el kilometraje final.  El kilometraje final debe ser mayor a 0. | | RESULTADOS: | | [Respuesta o Resultados esperados] | | RETROALIMENTACIÓN: | |  | | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **INSTITUTO TECNOLÓGICO DE ESTUDIOS SUPERIORES**  **DE LA REGIÓN CARBONÍFERA**  **Formato de Prácticas** | | | TEMA: | UNIDAD: | PRÁCTICA: |
| MATERIA: | | ESPECIALIDAD: | | **Lanzamiento de excepciones** | **1. Excepciones** | **Días Mes** |
| Programación Orientada a Objetos, Programación I | | Ing. En Sistemas, Lic. En Informática | |
| NOMBRE: | | TIPO DE PRACTICA: | DURACIÓN: |
| **Días Mes** | | Guiada | 1 hora |
| OBJETIVOS: | | CONOCIMIENTOS PREVIOS: | |
| * Comprender el concepto de excepción. * Aprender a lanzar excepciones mediante la instrucción **throw**. * Capturar las excepciones mediante un bloque **try**. | | * Clases y objetos * Propiedades | |
| COMENTARIOS / SUGERENCIAS: | | | |
|  | | | |
| |  | | --- | | PLANTEAMIENTO: | | **1. Crear una clase que reciba el número correspondiente a un mes (1 para Enero, 2 para Febrero, etc…) y que me indique cuantos días tiene ese mes y su nombre. (Suponga que Febrero siempre tiene 28 días y que son solo 12 meses)**   |  |  |  |  | | --- | --- | --- | --- | | **Enero** | **31** | **Julio** | **31** | | **Febrero** | **28** | **Agosto** | **31** | | **Marzo** | **31** | **Septiembre** | **30** | | **Abril** | **30** | **Octubre** | **31** | | **Mayo** | **31** | **Noviembre** | **30** | | **Junio** | **30** | **Diciembre** | **31** | | | RESULTADOS: | | [Respuesta o Resultados esperados] | | RETROALIMENTACIÓN: | | [Preguntas de retroalimentación] | | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **INSTITUTO TECNOLÓGICO DE ESTUDIOS SUPERIORES**  **DE LA REGIÓN CARBONÍFERA**  **Formato de Prácticas** | | | TEMA: | UNIDAD: | PRÁCTICA: |
| MATERIA: | | ESPECIALIDAD: | | **Manejo de excepciones** | **1. Excepciones** | **Estaciones del Año** |
| Programación Orientada a Objetos, Programación I | | Ing. En Sistemas, Lic. En Informática | |
| NOMBRE: | | TIPO DE PRACTICA: | DURACIÓN: |
| **Estaciones del año** | | Libre | 1 hora |
| OBJETIVOS: | | CONOCIMIENTOS PREVIOS: | |
| * Comprender el concepto de excepción. * Aprender a lanzar excepciones mediante la instrucción **throw**. * Capturar las excepciones mediante un bloque **try**. | | * Clases y objetos * Condiciones, switch | |
| COMENTARIOS / SUGERENCIAS: | | | |
| Dejar a los alumnos trabajando solos y observar el avance que tienen para poder determinar si se tiene que reforzar algo del tema. | | | |
| |  | | --- | | PLANTEAMIENTO: | | 1. **Crear una clase que reciba el número correspondiente a un día, y el numero de un mes (1 para Enero, 2 para Febrero, etc…) e indicar en qué estación del año cae dicha fecha. Toma en cuenta los meses de 1 a 12 y que febrero siempre tiene 28 dias.**   **Primavera: 21 Marzo – 20 Junio**  **Verano: 21 Junio - 20 Septiembre**  **Otoño: 21 Septiembre – 20 de Diciembre**  **Invierno: 21 Diciembre – 20 de Marzo** | | RESULTADOS: | |  | | RETROALIMENTACIÓN: | |  | | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **INSTITUTO TECNOLÓGICO DE ESTUDIOS SUPERIORES**  **DE LA REGIÓN CARBONÍFERA**  **Formato de Prácticas** | | | TEMA: | UNIDAD: | PRÁCTICA: |
| MATERIA: | | ESPECIALIDAD: | | **Eventos** | **4. Eventos** | **1. Cuenta Bancaria** |
| Programación Orientada a Objetos, Programación I | | Ing. En Sistemas, Lic. En Informática | |
| NOMBRE: | | TIPO DE PRACTICA: | DURACIÓN: |
| **Cerradura Electrónica** | | Guiada | 2 horas |
| OBJETIVOS: | | CONOCIMIENTOS PREVIOS: | |
| * Practicar el uso de delegado y eventos. * Controlar el estado de la interfaz de usuario a través de eventos de la clase. * Uso de la clase DateTime | | * Eventos * Delegados | |
| COMENTARIOS / SUGERENCIAS: | | | |
| * Asegurarse que el alumno comprenda la teoría de delegados y eventos antes de aplicar el ejercicio. * Incluya y explique el control Listbox al ejercicio. | | | |
| |  |  | | --- | --- | | PLANTEAMIENTO: | | | 1. Crear un programa que permita simular una cuenta bancaria. La cuenta permitirá realizar dos tipos de movimientos: Depósitos y Retiros. El saldo de la cuenta se mostrará en todo momento y también una lista de movimientos realizados incluyendo fecha y hora del movimiento. | | | RESULTADOS: | | | public class CuentaBancaria  {  private decimal saldo;  public decimal Saldo  {  get { return saldo; }  } | public delegate void Movimiento(string informacion);  public event Movimiento MovimientoRealizado;  public void Depositar(decimal cantidad)  {  if (cantidad < 1)  throw new ArgumentException("La cantidad especificada para el deposito es incorrecta.");  saldo += cantidad;  if (MovimientoRealizado != null)  {  MovimientoRealizado(cantidad.ToString("Deposito realizado por la cantidad de $ 0.00"));  }  }  public void Retirar(decimal cantidad)  {  if (cantidad < 1)  throw new ArgumentException("La cantidad especificada para el retiro es incorrecta.");  if (cantidad > saldo)  throw new ArgumentException("Saldo insuficiente para realizar el retiro.");  saldo -= cantidad;  if (MovimientoRealizado != null)  {  MovimientoRealizado(cantidad.ToString("Retiro realizado por la cantidad de $ 0.00"));  }  }  } | | RETROALIMENTACIÓN: | | | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **INSTITUTO TECNOLÓGICO DE ESTUDIOS SUPERIORES**  **DE LA REGIÓN CARBONÍFERA**  **Formato de Prácticas** | | | TEMA: | UNIDAD: | PRÁCTICA: |
| MATERIA: | | ESPECIALIDAD: | | **Eventos** | **2. Eventos** | **3. Cerradura Electrónica** |
| Programación Orientada a Objetos, Programación I | | Ing. En Sistemas, Lic. En Informática | |
| NOMBRE: | | TIPO DE PRACTICA: | DURACIÓN: |
| **Cerradura Electrónica** | | Guiada | 2 horas |
| OBJETIVOS: | | CONOCIMIENTOS PREVIOS: | |
| * Practicar el uso de delegado y eventos. * Controlar el estado de la interfaz de usuario a través de eventos de la clase. * Enlazar eventos de los controles a un mismo método. * Uso del parámetro **Sender** | | * Eventos * Delegados * Boxing y Unboxing | |
| COMENTARIOS / SUGERENCIAS: | | | |
| * Todos los botones de caracteres (0...9, \* y #) enlazarlos al mismo método y explicar el uso del parámetro Sender para diferenciar objetos. | | | |
| |  |  | | --- | --- | | PLANTEAMIENTO: | | | 1. Crear una clase que sea capaz de simular una cerradura electrónica de cinco caracteres. La cerradura permitirá el acceso al usuario al introducir la clave \*1234 y generara un aviso que lo deje entrar a la aplicación. En caso de que introduzcas una clave incorrecta la clase avisara del error. Si suceden cinco errores consecutivos, la cerradura se bloqueara. El contador de errores se reiniciara al introducir la clave correcta. 2. Crear un formulario para probar la clase de la cerradura que contenga una pantalla y botones con los números del 0 al 9 y los símbolos de asterisco (\*) y numeral (#). Y botones de ENTRAR y LIMPIAR. Cuando ocurra un error en el acceso, se generará un sonido y se mostrará ERROR en la pantalla. Si la cerradura está bloqueada, todos los controles se deshabilitarán y la pantalla mostrará el mensaje BLOQUEADO. | | | RESULTADOS: | | | public class CerraduraElectronica  {  private string codigo="\*1234";  private byte intentos=0; | public delegate void Resultado();  public event Resultado InicioExitoso;  public event Resultado InicioError;  public event Resultado Bloqueado;  public void Verificar(string codigo)  {  if (codigo == this.codigo)  {  intentos = 0;  if (InicioExitoso != null)  InicioExitoso();  }  else  {  if (intentos > 5)  {  if (Bloqueado != null)  Bloqueado();  }  else  {  intentos++;  if (InicioError != null)  InicioError();  }  }  }  } | | RETROALIMENTACIÓN: | | | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **INSTITUTO TECNOLÓGICO DE ESTUDIOS SUPERIORES**  **DE LA REGIÓN CARBONÍFERA**  **Formato de Prácticas** | | | TEMA: | UNIDAD: | PRÁCTICA: |
| MATERIA: | | ESPECIALIDAD: | | **Constructores** | **5. Sobrecarga** |  |
| Programación Orientada a Objetos, Programación I | | Ing. En Sistemas, Lic. En Informática | |
| NOMBRE: | | TIPO DE PRACTICA: | DURACIÓN: |
| **Renta Automóvil** | | Diagnóstico-Guiada | 1 hora |
| OBJETIVOS: | | CONOCIMIENTOS PREVIOS: | |
| * Realizar un diagnostico inicial a los alumnos para identificar el nivel de dominio que poseen de los temas de la materias de fundamentos de programación. * A partir de lo resuelto por los alumnos, introducir el nuevo tema. | | Materia: Fundamentos de Programación   * Clases y objetos * Propiedades * Condiciones | |
| COMENTARIOS / SUGERENCIAS: | | | |
| * Dejar que los alumnos intenten realizar solos la primera parte del ejercicio sin ayuda del docente la primera media hora de clase e identificar problemas que tengan al tratar de resolverlo. * Resolver el ejercicio para que aquellos alumnos que no hayan podido realizarlo por si solos conozcan la forma de solucionarlo. * Guiar al alumno en el lanzamiento y captura de excepciones. | | | |
| |  | | --- | | PLANTEAMIENTO: | | Una compañía se dedica a rentar automóviles y realiza sus cobros de la siguiente manera:  Al momento de que el cliente solicita el automóvil, toman nota del kilometraje que tiene el automóvil en ese momento.  Cuando el cliente regresa a entregar el automóvil rentado, el encargado de la compañía revisa el kilometraje nuevamente y obtiene el kilometraje recorrido.  Si recorre un total de 120 kilómetros o menos, se le cobra una tarifa fija de $ 350.00. Si el kilometraje es mayor, le cobran $ 350.00 que corresponden a la tarifa fija y $ 15.00 por kilómetro excedido.  Crear una clase que permita calcular el monto a cobrar por la renta del automóvil.  NOTA:  El kilometraje inicial debe ser menor que el kilometraje final.  El kilometraje final debe ser mayor a 0. | | RESULTADOS: | | [Respuesta o Resultados esperados] | | RETROALIMENTACIÓN: | |  | | | | | | | |