UNIDAD

DIPLOMATURA EN PROGRAMACION JAVA

Proyecto Ventas

Manual de Ejercicios – Capítulos 1 a 5 inclusive

Laboratorio 0

Aplicación Info

Reconstrucción del código de la aplicación Info

Completar los siguientes pasos:

1. Abrir la solución AplicacionInfo (se encuentra en el directorio de instalación de los ejercicios dentro de la carpeta del lenguaje a utilizar. Nombre: AplicacionInfo.sln)
2. Examinar cada uno de los archivos fuente
3. Trabajar en el proyecto MVC_CS o MVC_VB. Establecerlo como proyecto de arranque
4. Completar con el código que sea necesario en cada uno de los fuentes basándose en los diagramas y explicaciones presentados en este módulo para la aplicación Info
5. Notar que se emula el trabajo con eventos al reemplazarlos con llamados a métodos y registrando las correspondientes referencias a los objetos que los contienen
6. Ejecutar la aplicación y verificar su correcto funcionamiento.

Laboratorio 1

Aplicación Info

En este laboratorio se utilizan los conceptos obtenidos en el laboratorio anterior para reemplazar los llamados a métodos por declaraciones y utilización de eventos. Realizar las siguientes acciones:

1. Abrir la solución AplicacionInfo (se encuentra en el directorio de instalación de los ejercicios dentro de la carpeta del lenguaje a utilizar. Nombre: AplicacionInfo.sln)
2. Examinar cada uno de los archivos fuente
3. Trabajar en el proyecto MVC_Eventos. Establecerlo como proyecto de arranque
4. Completar con el código que sea necesario en cada uno de los fuentes basándose en los diagramas y explicaciones presentados en este módulo para la aplicación Info
5. Notar que se debe remplazar con eventos a los métodos antes utilizados para realizar llamadas de callback
6. Ejecutar la aplicación y verificar su correcto funcionamiento.

Laboratorio 2

Creación y uso de la Base de Datos Stock

Ver los videos de creación de la base de datos en el entorno del Visual Studio usando SQLExpress

1. Revisar el contenido de los registros en la tabla Clientes. Para ello, escribir la siguiente consulta:

   SELECT * FROM Cliente

   La cantidad de registros devueltos debe dar como resultado de la consulta 7 filas afectadas (7 row(s) affected)

2. Revisar el contenido de los registros en las tablas Distribución y Stock de la misma manera.

3. Leer de la tabla Cliente el cliente con DNI= 16828325. Para ello, escribir la siguiente consulta:

   SELECT * FROM Cliente WHERE DNI='16828325'

4. Leer de la tabla Distribución todos los registros del cliente con DNI= 16828325.

5. Agregar información propia como si fuera un cliente en la tabla Cliente

   - Para realizar este paso, ingrese su DNI o un número igual que luego recuerde
   - Verificar que en la caja de selección de la barra superior de herramientas figure la tabla Stock. De no ser así, despliegue la lista y selecciónela. Adaptar la siguiente sentencia SQL de inserción para agregar sus detalles en la tabla Cliente:

     1. INSERT INTO Cliente (DNI, NomCliente, Direccion) VALUES *('XXXXXXXX', 'SuNombreYApellido', 'AlgunaDireccion'*)

6. Agregar en la tabla Distribución algunas filas (registros) de distribución con el DNI ingresado en la tabla Cliente

7. Borrar la primera fila de los registros que ingresó en la tabla de Distribución. La siguiente sentencia SQL borra el registro con DNI = 11111111 y ID_Stock = 'BYGUM':

   DELETE FROM DISTRIBUCION WHERE DNI = '11111111' AND ID_Stock = 'BYGUM'

8. Escribir la siguiente sentencia SQL para actualizar el campo nombre del cliente con DNI= 16828325:

   UPDATE Cliente SET NomCliente = 'Alfredo Bardo' WHERE DNI= 16828325

Laboratorio 3

Verificar los proyectos y clases que componen la solución ventas

1. Entender los proyectos que conforman la línea base de la solución
2. Verificar las referencias entre proyectos para entender el acceso a las clases que componen cada proyecto

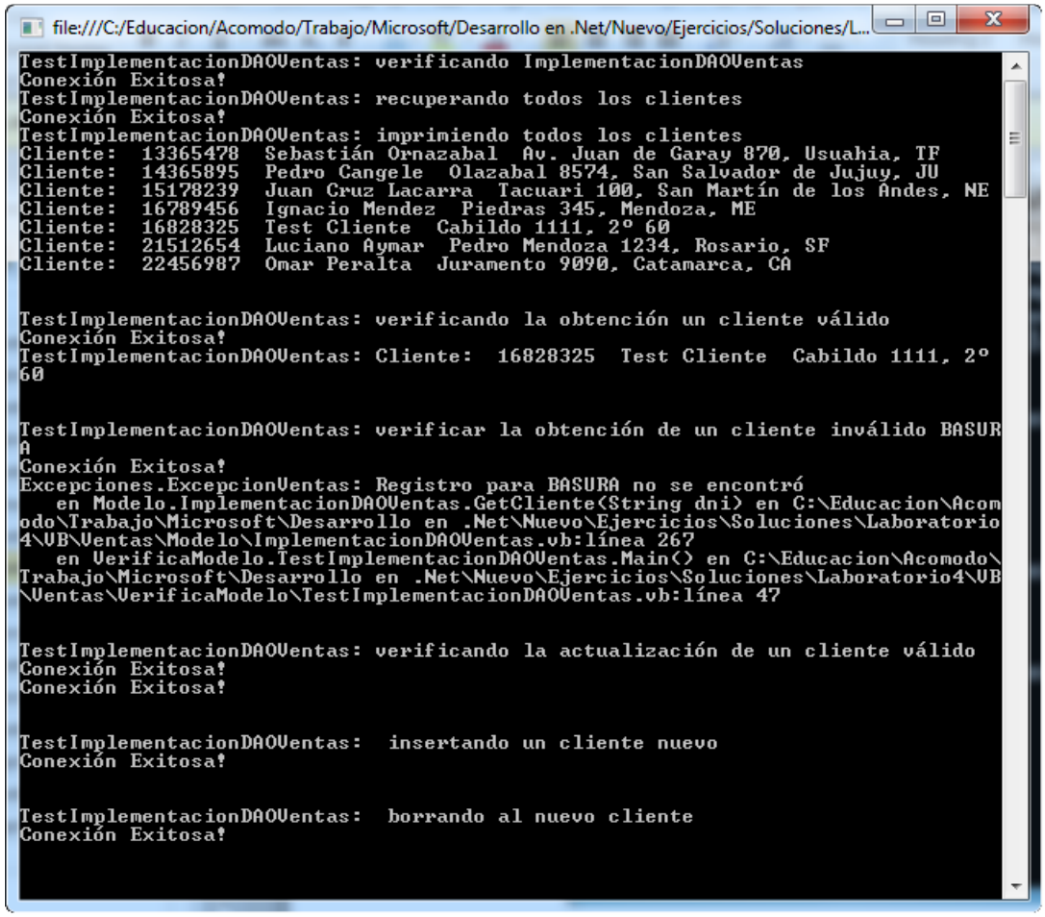
3. Verificar las interfaces dadas para el manejo del patrón de diseño MVC. Identificar las interfaces participantes en la aplicación del patrón de diseño y comprender los diferentes métodos que las componen
4. Examinar el código fuente de las entidades que contendrán la información del modelo del proyecto
5. Verificar que los proyectos no contengan errores de codificación ni de referencias

Laboratorio 4

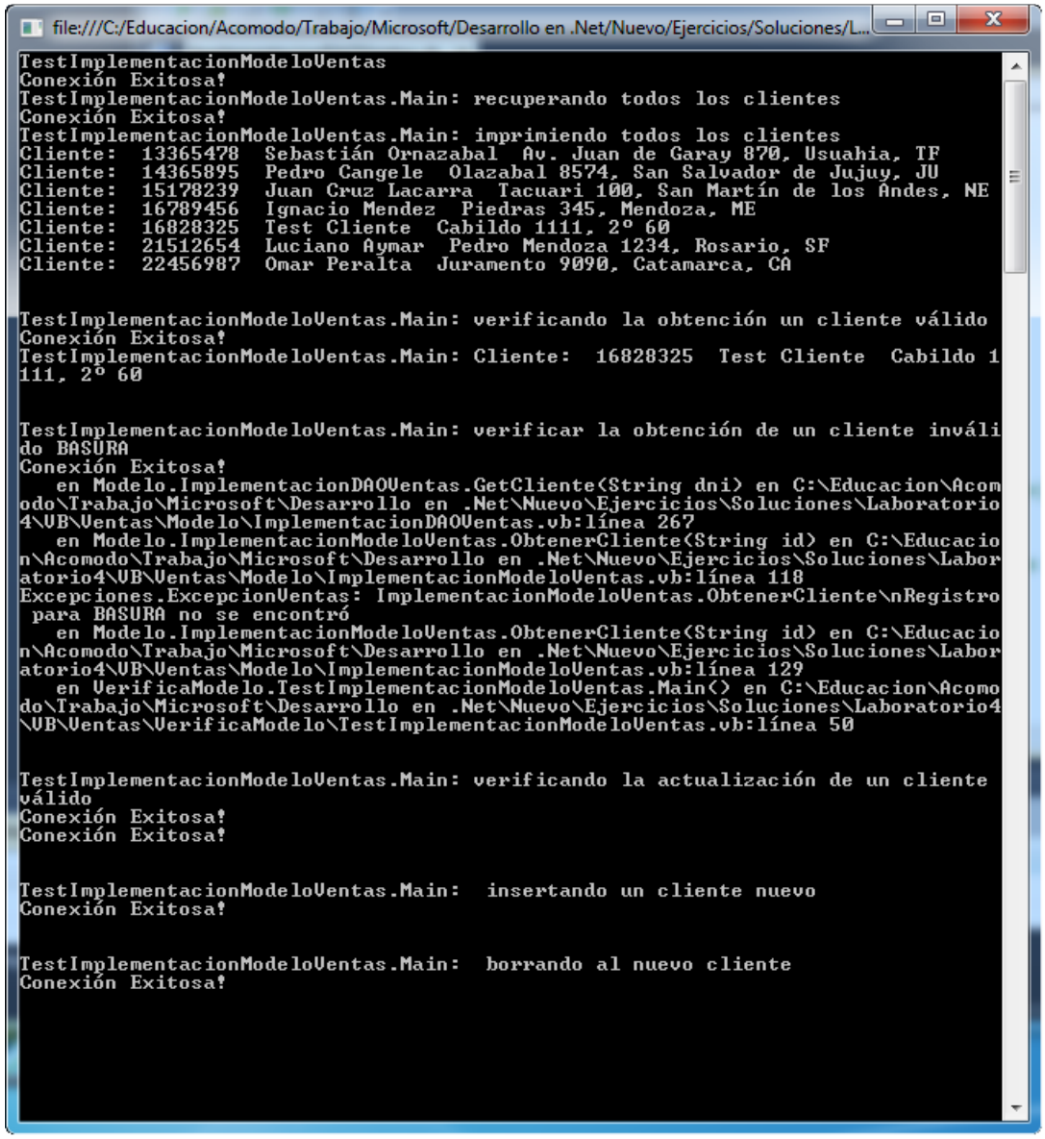
En este ejercicio se deben incorporar dos proyectos a la solución en curso, Modelo y VerificaModelo para completar el código y hacer funcionar el modelo del MVC de la aplicación

1. Ir a la carpeta Laboratorio4 en el directorio donde se encuentran instalados los ejercicios
2. Seleccionar la solución y presionar el botón derecho del mouse para mostrar el menú desplegable
3. Seleccionar Add → Existing Project e incorporar el proyecto Modelo
4. Repetir la acción para el proyecto VerificaModelo
5. **Completar las líneas de código faltantes en las clases ImplementacionDAOVentas e ImplementacionModeloVentas**
6. Seleccionar como proyecto de arranque a VerificaModelo
7. Verificar que existen dos clases con un método Main. Usar ambas clases como comienzo del programa cambiando las propiedades del proyecto.
8. Ejecutar cada programa y corroborar que VerificaModelo finaliza sin errores

Cuando se llama a TestImplementacionDAOVentas, la salida debe ser como la que se muestra en la figura:



Cuando se llama a TestImplementacionModeloVentas, la salida debe ser como la que se muestra en la figura:



Laboratorio 5

Construcción 3

Implementando IVistaVentas

En este ejercicio se crea la clase de vista para Ventas que implementa la interfaz IVistaVentas

1. Incorporar a la solución los proyectos que se encuentran en la carpeta Laboratorio5. Estos son Paneles y Ventas. Para realizar correctamente la operación, borrar el proyecto Ventas creado previamente, tanto de la solución como físicamente desde en el disco rígido.

Incorporar el proyecto del directorio Laboratorio5, reconstruir toda la solución y agregar las referencias a proyectos y espacios de nombres que sean necesarias.

2. Revisar el código que se encuentra en los proyectos
3. Agregar las referencias necesarias a otros proyectos (por ejemplo, la clase Cliente se encuentra en el proyecto Entidades)
4. **Completar, cuando lo indique un comentario dentro de los archivos fuentes, las líneas de código faltantes.** El código que requiere esta acción se encuentra en el control PanelCliente dentro del proyecto Paneles y en la clase ImplementacionVistaVentas
5. Compilar todo el proyecto y verificar que no haya errores
6. Asegurarse de establecer a Ventas como el proyecto inicial.

Laboratorio 6

Construcción 4

Implementar el controlador de Ventas

En este ejercicio se creará el controlador de Ventas que implementa la interfaz IControladorVentas

1. Incorporar al proyecto Ventas la clase que se encuentra en el directorio Laboratorio6. Reemplazar la existente que está creado sólo con el fin que no de errores de compilación.
2. Revisar el código y buscar los comentarios que indican los lugares a completar el código
3. Editar la clase ImplementacionControladorVentas. Revisar el código y seguir las instrucciones que se encuentren en los comentarios para completar la clase
4. Compilar todo el proyecto y verificar que no haya errores
5. Ejecutar el programa y validar la interacción con el controlador

Laboratorio 7

Implementar eventos en la vista de Ventas

En este ejercicio se trabajará con la clase ImplementacionVistaVentas para definir los eventos de la vista al controlador. También se debe analizar las suscripciones a eventos provistas para tomarlas como ejemplo en ImplementacionVistaVentas e ImplementacionModeloVentas

1. Incorporar al proyecto Ventas las clases que se encuentra en el directorio Laboratorio7. La clase ImplementacionVistaVentas debe reemplazar a la existente en el proyecto Ventas y la clase ImplementacionModeloVentas debe reemplazar a la existente en el proyecto Modelo

2. Revisar el código y buscar los comentarios que indican los lugares a completar el código en ImplementacionVistaVentas

3. Editar la clase ImplementacionModeloVentas. Revisar el código y comprender como se dispara el evento que le informa a la vista un cambio en el modelo

4. Compilar todo el proyecto y verificar que no haya errores

Laboratorio 8

Implementar eventos en el controlador de Ventas

En este ejercicio se trabajará con la clase ImplementacionControladorVentas y la interfaz IControladorVentas para definir los eventos del controlador a la vista.

1. Incorporar al proyecto Ventas la clase ImplementacionControladorVentas y la interfaz IControladorVentas al proyecto MVC. Ambas se encuentran en el directorio Laboratorio8

2. Revisar el código y buscar los comentarios que indican los lugares a completar el código en ImplementacionControladorVentas e IControladorVentas. Notar que la invocación al manejador de eventos se hace con un nombre ya definido: eventoDelControladorALaVistaMostrarCliente

3. Compilar todo el proyecto y verificar que no haya errores

4. Ejecutar el programa y validar la interacción entre la vista, el controlador y el modelo