Prueba Practica .NET

**Instrucciones**: Realice la siguiente prueba generando los scripts / archivos que aplique para cada uno y súbalo a un **repositorio GIT** en cualquier servicio en la nube. El link deberá ser enviado al correo electrónico de donde recibió la prueba para poderlo descargar y revisar.

1. Escriba el código de una página ASPX que solicite el nombre del usuario, contraseña, y que valide que los campos no vayan vacíos antes de enviar la solicitud al servidor.

Ver archivo adjunto en el repositorio(index.aspx)

1. Escriba un documento XML que permita intercambiar información de tarjetas de crédito entre dos sistemas distintos: nombre del sistema, localización, número de tarjeta, ID único de cliente, etc.

Ver archivo adjunto el repositorio(ScriptXML.xml)

1. Con el siguiente modelo de base de datos:

Diagrama

Descripción generada automáticamente

Descripción de las Tablas:

Catálogos:

* PROYECTO: Listado de proyectos de tarjetas que existen (Ejemplo: Premia, Konmi, Yujule, etc.)
* PRODUCTO: listado de productos de tarjetas que hay. (ejemplo Premia Clásica, Premia Oro, Premia Platinum)
* TIPO: Tipo de mensaje (ejemplo de mensaje de texto, mail, mensaje en el estado de cuenta)
* TIPO\_INFORMACION: Tipo del mensaje. Ejemplo mensaje de bienvenida, mensaje de mora, mensaje de promoción

Otras:

* FORMATO\_MENSAJE: son los formatos de los mensajes existentes.
* MENSAJE: si el mensaje aplica a que proyecto y que producto

1. Escriba la consulta en SQL que devuelva el nombre del proyecto y sus productos correspondientes del proyecto premia cuyo código es 1

Select pr.nombre as PROYECTO, pr.descripcion as PRODUCTO

from proyecto pr

Inner join producto\_proyecto pp on pr.proyecto = pp.proyecto

Inner join producto pd on pp.producto = pd.producto

Where pr.proyecto = 1;

1. Escriba una consulta SQL que devuelva los distintos mensajes que hay, indicando a qué proyecto y producto pertenecen.

Select fm.asunto as MENSAJE, pr.nombre as PROYECTO, pt.descripcion as PRODUCTO

From mensaje mj

inner join producto\_proyecto pp on mj.producto = pp.producto and mj.proyecto =

pp.proyecto

inner join producto pt on pp.producto = pt.producto

Inner join proyecto pr on pp.proyecto = pr.proyecto

inner join formato\_mensaje fm on mj.codformato = fm.codformato;

1. Escriba una consulta SQL que devuelva los distintos mensajes que hay, indicando a qué proyecto y producto pertenecen. Pero si el mensaje está en todos los productos de un proyecto, en lugar de mostrar cada producto, debe mostrar el nombre del proyecto y un solo producto que diga “TODOS”.

Declare @cantProductos int

Set @cantProductos = (select count(\*) from producto)

Select |(count(pp.producto) = @cantProductos, ‘TODOS’,fm.asunto) as MENSAJE,

pp.proyecto

From mensaje mj

Inner join producto\_proyecto pp on mj.producto = pp.producto and mj.proyecto =

pp.proyecto

Inner join producto pd on pp.producto = pd.producto

Inner join proyecto pr on pp.proyecto = pr.proyecto

Inner join formato\_mensaje fm on mj.cod\_formato = fm.cod\_formato

Group by pp.proyecto, fm.asunto

1. Escrba el código de una clase, en sintaxis C# que permita hacer conexiones a una base de datos cualquiera, y que devuelva en uno de sus métodos un Dataset al hacer un SELECT, y en otro de sus métodos el número de filas afectadas al ejecutar un INSERT, UPDATE o DELETE. También debe poder devolver el número de filas que devolvería un SELECT si se ejecutara.

public class conexiónBD

{

Public void abrirConexion

{

String recurso = “sqlserver”;

String catal = “prueba”;

String seguridad= “test”;

String usuario = “user”;

String pass = “pass”;

String conexión = "Data Source =" + recurso

+ "; Initial Catalog =" + catal

+ "; Integrated Security=" + seguridad

+ "; User ID=" + usuario

+ "; Password=" + pass;

try

{

SqlConnection con = new SqlConnection(conexion);

con.Open();

Console.WriteLine("La conexión se ha creado!");

Console.WriteLine("Estado de conexion: " + con.State.ToString());

}

catch(Exception ex)

{

Console.WriteLine("Hay un error conectándose a la base de datos " + ex.Message);

}

}

Public void llenarRegistros

{

string sql = "select nombre from tabla";

SqlCommand comando = new SqlCommand(sql, conexion);

SqlDataReader registros = comando .ExecuteReader();

while (registros.Read())

{

Console.WriteLine("{0}", reader.GetString(0));

}

}

Public void contarRegistros

{

SqlCommand cmd = new SqlCommand(Query, conexion);

Int lineas = cmd.ExecuteNonQuery();

Console.WriteLine(“Numero de líneas es: ” + lineas);

}

}

1. Construya un modelo Entidad Relación que permita llevar el control de Canje de sus puntos acumulados por premios.
   1. El artículo debe tener código, nombre, tipo y subtipo.
   2. Se debe manejar la existencia de artículos por sucursal. Ejemplo, está la sucursal Promerica Majadas y Promerica Columbus y en cada sucursal existen artículos como DVD de Disney, lapiceros, mouse, etc. Y cada sucursal debe saber qué cantidad posee en inventario.
   3. Van a existir varias promociones para canjear y estas van a estar compuestas por varios artículos. Estas promociones poseen rangos de fecha de vigencia y en dichas promociones estará el costo de los puntos y el precio del artículo
   4. El canje debe tener ID del artículo, Cantidad Canjeada, Precio del artículo, la promoción y observaciones.

Imagen de la pantalla de un computador

Descripción generada automáticamente con confianza baja