

EJERCICIOS RESUELTOS TEMA 9

9.1. Partiremos de la base de datos "ejercicio8", que creaste en el ejercicio 8.1. Si no lo habías hecho, crea ahora una nueva base de datos "ejercicio9", que tendrá una única tabla "ordenadores". De cada ordenador se desea guardar un código (que será la clave primaria), una marca (no nula), un modelo y un año de lanzamiento (mira los datos de ejemplo para deducir los tipos de datos necesarios).

```
using System;
using System.Data.SQLite;

namespace _9._1
{
    class Program
    {
        static void Main()
        {
            SQLiteConnection conexion = new SQLiteConnection("Data Source =
            ..\\..\\..\\ejercicio9.sqlite; Version = 3; New = True; Compress = True");
            conexion.Open();

            string creacion = "CREATE TABLE ordenadores (codigo varchar (8)
            PRIMARY KEY, marca varchar (15) " +
            "NOT NULL, modelo varchar (15), lanzamiento numeric (4))";
            SQLiteCommand cmd = new SQLiteCommand(creacion, conexion);
            cmd.ExecuteNonQuery();

            conexion.Close();
            Console.WriteLine("Tabla creada con éxito.");
            Console.ReadLine();
        }
    }
}
```

9.2. Añade los ordenadores:

- ATM, Oric, Atmos, 1984
- ZX80, Sinclair, ZX80, 1980
- VIC20, Commodore, VIC-20, 1981
- VG8235, Philips, VG8235 MSX2, 1985
- C64, Commodore, 64, 1982
- 520ST, Atari, 520ST, 1985

```
using System;
using System.Data.SQLite;

namespace _9._2
{
    class Program
    {
        static void Main()
        {
            SQLiteConnection conexion = new SQLiteConnection("Data Source =
            ..\\..\\..\\ejercicio9.sqlite; Version = 3; New = False; Compress = True");
```

```

        conexion.Open();

        string insertar = "INSERT INTO ordenadores VALUES ('IBM5150', 'IBM',
'PC(5150)', 1981); " +
        "INSERT INTO ordenadores VALUES ('SPEC48', 'Sinclair', 'ZX Spectrum
48K', 1982);" +
        "INSERT INTO ordenadores VALUES ('CPC464', 'Amstrad', 'CPC464',
1984);" +
        "INSERT INTO ordenadores VALUES ('HB55', 'Sony', 'Hit-Bit 55 MSX',
1984);" +
        "INSERT INTO ordenadores VALUES ('QL', 'Sinclair', 'QL', 1984);" +
        "INSERT INTO ordenadores VALUES ('PPC640DD', 'Amstrad', 'PPC640 DD',
1988);" +
        "INSERT INTO ordenadores VALUES ('ATM', 'Oric', 'Atmos', 1984);" +
        "INSERT INTO ordenadores VALUES ('ZX80', 'Sinclair', 'ZX80', 1980);"
+
        "INSERT INTO ordenadores VALUES ('VIC20', 'Commodore', 'VIC-20',
1981);" +
        "INSERT INTO ordenadores VALUES ('VG8235', 'Philips', 'VG8235 MSX2',
1985);" +
        "INSERT INTO ordenadores VALUES ('C64', 'Commodore', '64', 1982);" +
        "INSERT INTO ordenadores VALUES ('520ST', 'Atari', '520ST', 1985);";
        SQLiteCommand cmd = new SQLiteCommand(insertar, conexion);
        cmd.ExecuteNonQuery();

        Console.WriteLine("Datos insertados con éxito. Mostrar tabla: ");

        string mostrar = "SELECT * FROM ordenadores";
        cmd = new SQLiteCommand(mostrar, conexion);
        SQLiteDataReader datos = cmd.ExecuteReader();

        while (datos.Read())
        {
            string codigo = Convert.ToString(datos[0]);
            string marca = Convert.ToString(datos[1]);
            string modelo = Convert.ToString(datos[2]);
            int lanzamiento = Convert.ToInt32(datos[3]);
            Console.WriteLine("Código: {0}, Marca: {1}, Modelo: {2}, Año de
lanzamiento: {3}", codigo, marca, modelo, lanzamiento);
        }

        conexion.Close();
        Console.ReadLine();
    }
}

```

9.3. Muestra la(s) marca(s) de la(s) que tenemos más ordenadores.

```

using System;
using System.Data.SQLite;

namespace _9._3
{
    class Program
    {
        static void Main()
        {
            SQLiteConnection conexion = new SQLiteConnection("Data Source =
..\..\..\ejercicio9.sqlite; " +
                "Version = 3; New = False; Compress = True");

```

```

        conexion.Open();

        string mostrar = @"
            SELECT marca, count(*)
            FROM ordenadores
            GROUP BY marca
            HAVING count(*) =
                (SELECT MAX(cuenta) FROM
                 (SELECT marca, count(*) as cuenta FROM
ordenadores GROUP BY marca));";
        SQLiteCommand cmd = new SQLiteCommand(mostrar, conexion);
        SQLiteDataReader datos = cmd.ExecuteReader();

        while (datos.Read())
        {
            string marca = Convert.ToString(datos[0]);
            int numTotal = Convert.ToInt32(datos[1]);
            Console.WriteLine("La marca de la que tenemos más ordenadores es
{0}, y hay {1} " +
                "ordenadores en la base de datos.", marca, numTotal);
        }

        conexion.Close();
        Console.ReadLine();
    }
}

```

9.4. Muestra los modelos de los ordenadores pertenecientes a la(s) marca(s) de la(s) que tenemos más ordenadores.

```

using System;
using System.Data.SQLite;

namespace _9._4
{
    class Program
    {
        static void Main()
        {
            SQLiteConnection conexion = new SQLiteConnection("Data Source =
..\\..\\..\\ejercicio9.sqlite; " +
                "Version = 3; New = False; Compress = True");
            conexion.Open();

            string mostrar = @"
                SELECT modelo FROM ordenadores WHERE marca IN
                (SELECT marca
                 FROM ordenadores
                 GROUP BY marca
                 HAVING count(*) =
                     (SELECT MAX(cuenta) FROM
                      (SELECT count(*) as cuenta,
marca FROM ordenadores GROUP BY marca)));";

            SQLiteCommand cmd = new SQLiteCommand(mostrar, conexion);
            SQLiteDataReader datos = cmd.ExecuteReader();

            Console.WriteLine("Los modelos de la marca de la que tenemos más
ordenadores son: ");

```

```

        while (datos.Read())
        {
            string modelo = Convert.ToString(datos[0]);
            Console.WriteLine(modelo);
        }

        conexion.Close();
        Console.ReadLine();
    }
}

```

9.5. Muestra los modelos de los ordenadores pertenecientes a la(s) marca(s) a la que pertenece el ordenador más reciente, ordenados alfabéticamente (usando el modelo como criterio de ordenación).

```

using System;
using System.Data.SQLite;

namespace _9._5
{
    class Program
    {
        static void Main()
        {
            SQLiteConnection conexion = new SQLiteConnection("Data Source =
..\\..\\..\\ejercicio9.sqlite; " +
                "Version = 3; New = False; Compress = True");
            conexion.Open();

            string mostrar = @"SELECT modelo FROM ordenadores WHERE marca =
                                (SELECT marca FROM ordenadores WHERE
lanzamiento =                                (SELECT MAX(lanzamiento) FROM
ordenadores)) ORDER BY modelo ASC;";

            SQLiteCommand cmd = new SQLiteCommand(mostrar, conexion);
            SQLiteDataReader datos = cmd.ExecuteReader();

            Console.WriteLine("Los modelos de la marca del ordenador más reciente
son: ");

            while (datos.Read())
            {
                string modelo = Convert.ToString(datos[0]);
                Console.WriteLine(modelo);
            }

            conexion.Close();
            Console.ReadLine();
        }
    }
}

```

9.6. Muestra todos los datos de los ordenadores de la primera marca (alfabéticamente, ordenadas de la 'A' a la 'Z').

```

using System;
using System.Data.SQLite;

```

```

namespace _9._6
{
    class Program
    {
        static void Main()
        {
            SQLiteConnection conexion = new SQLiteConnection("Data Source =
..\\..\\..\\ejercicio9.sqlite; " +
                "Version = 3; New = False; Compress = True");
            conexion.Open();

            string mostrar = @"SELECT * FROM ordenadores WHERE marca =
                (SELECT MIN(marca) FROM ordenadores); ";

            SQLiteCommand cmd = new SQLiteCommand(mostrar, conexion);
            SQLiteDataReader datos = cmd.ExecuteReader();

            while (datos.Read())
            {
                string codigo = Convert.ToString(datos[0]);
                string marca = Convert.ToString(datos[1]);
                string modelo = Convert.ToString(datos[2]);
                int lanzamiento = Convert.ToInt32(datos[3]);
                Console.WriteLine("Código: {0}, Marca: {1}, Modelo: {2},
Lanzamiento: {3}", codigo, marca, modelo, lanzamiento);
            }

            conexion.Close();
            Console.ReadLine();
        }
    }
}

```

9.7. Muestra los modelos de los ordenadores que no pertenezcan a las marcas que comienzan con letra A, usando una subconsulta.

```

using System;
using System.Data.SQLite;

namespace _9._7
{
    class Program
    {
        static void Main()
        {
            SQLiteConnection conexion = new SQLiteConnection("Data Source =
..\\..\\..\\ejercicio9.sqlite; " +
                "Version = 3; New = False; Compress = True");
            conexion.Open();

            string mostrar = @"SELECT modelo, marca FROM ordenadores WHERE marca
IN
                (SELECT marca FROM ordenadores WHERE SUBSTR
(marca, 1, 1) != 'A');";

            //Con esta orden bastaría...
            //SELECT modelo FROM ordenadores WHERE SUBSTR(marca, 1, 1) != 'A';

            SQLiteCommand cmd = new SQLiteCommand(mostrar, conexion);
            SQLiteDataReader datos = cmd.ExecuteReader();

```

```

        while (datos.Read())
        {
            string modelo = Convert.ToString(datos[0]);
            string marca = Convert.ToString(datos[1]);
            Console.WriteLine(modelo + marca);
        }

        conexion.Close();
        Console.ReadLine();
    }
}

```

9.8. Unifica los ejercicios anteriores en la misma clase.

```

using System;
using System.Data.SQLite;

namespace _9._8
{
    class Program
    {
        static SQLiteConnection conexion;

        static void Main()
        {
            EstablecerConexion();
            CrearBBDD();
            Console.WriteLine("Tabla creada con éxito. Pulsa Intro para
continuar...");
            Console.ReadLine();
            InsertarDatos();
            Console.WriteLine("\nDatos insertados con éxito. Pulsa Intro para
continuar...");
            Console.ReadLine();
            Console.WriteLine("\nMostrar la marca que más ordenadores tiene: ");
            PrimeraConsulta();
            Console.WriteLine("Pulsa Intro para continuar...");
            Console.ReadLine();
            Console.WriteLine("\nMostrar los modelos de la marca que más
ordenadores tiene: ");
            SegundaConsulta();
            Console.WriteLine("Pulsa Intro para continuar...");
            Console.ReadLine();
            Console.WriteLine("\nMostrar los modelos de la marca que tiene el
ordenador más reciente: ");
            TerceraConsulta();
            Console.WriteLine("Pulsa Intro para continuar...");
            Console.ReadLine();
            Console.WriteLine("\nMostrar los modelos de la primera marca ordenada
alfabéticamente: ");
            CuartaConsulta();
            Console.WriteLine("Pulsa Intro para continuar...");
            Console.ReadLine();
            Console.WriteLine("\nMostrar los modelos de las marcas que NO
empiezan por 'A': ");
            QuintaConsulta();
            CerrarConexion();
            Console.WriteLine("Pulsa Intro para acabar...");
            Console.ReadLine();
        }
    }
}

```

```

    }

    static void EstablecerConexion()
    {
        conexion = new SQLiteConnection("Data Source =
..\\..\\..\\ejercicio9.sqlite; Version = 3; New = True; Compress = True");
        conexion.Open();
    }

    static void CrearBBDD()
    {
        string creacion = "CREATE TABLE ordenadores (codigo varchar (8)
PRIMARY KEY, marca varchar (15) " +
        "NOT NULL, modelo varchar (15), lanzamiento numeric (4))";
        SQLiteCommand cmd = new SQLiteCommand(creacion, conexion);
        cmd.ExecuteNonQuery();
    }

    static void InsertarDatos()
    {
        string insertar = @"INSERT INTO ordenadores VALUES ('IBM5150', 'IBM',
'PC(5150)', 1981),
('SPEC48', 'Sinclair', 'ZX Spectrum 48K', 1982), ('CPC464',
'Amstrad', 'CPC464', 1984),
('HB55', 'Sony', 'Hit-Bit 55 MSX', 1984), ('QL', 'Sinclair', 'QL',
1984),
('PPC640DD', 'Amstrad', 'PPC640 DD', 1988), ('ATM', 'Oric', 'Atmos',
1984),
('ZX80', 'Sinclair', 'ZX80', 1980), ('VIC20', 'Commodore', 'VIC-20',
1981),
('VG8235', 'Philips', 'VG8235 MSX2', 1985), ('C64', 'Commodore',
'64', 1982),
('520ST', 'Atari', '520ST', 1985));";
        SQLiteCommand cmd = new SQLiteCommand(insertar, conexion);
        cmd.ExecuteNonQuery();
    }

    static void PrimeraConsulta()
    {
        string mostrar = @"
        SELECT marca, count(*)
        FROM ordenadores
        GROUP BY marca
        HAVING count(*) =
        (SELECT MAX(cuenta) FROM
        (SELECT marca, count(*) as cuenta FROM
ordenadores GROUP BY marca));";
        SQLiteCommand cmd = new SQLiteCommand(mostrar, conexion);
        SQLiteDataReader datos = cmd.ExecuteReader();

        while (datos.Read())
        {
            string marca = Convert.ToString(datos[0]);
            int numTotal = Convert.ToInt32(datos[1]);
            Console.WriteLine("La marca de la que tenemos más ordenadores es
{0}, y hay {1} " +
                "ordenadores en la base de datos.", marca, numTotal);
        }
    }

    static void SegundaConsulta()
    {

```

```

        string mostrar = @"
            SELECT modelo FROM ordenadores WHERE marca IN
            (SELECT marca
             FROM ordenadores
             GROUP BY marca
             HAVING count(*) =
              (SELECT MAX(cuenta) FROM
               (SELECT count(*) as cuenta,
marca FROM ordenadores GROUP BY marca)))";

        SQLiteCommand cmd = new SQLiteCommand(mostrar, conexion);
        SQLiteDataReader datos = cmd.ExecuteReader();

        Console.WriteLine("Los modelos de la marca de la que tenemos más
ordenadores son: ");

        while (datos.Read())
        {
            string modelo = Convert.ToString(datos[0]);
            Console.WriteLine(modelo);
        }

        static void TerceraConsulta()
        {
            string mostrar = @"SELECT modelo FROM ordenadores WHERE marca =
                               (SELECT marca FROM ordenadores WHERE
lanzamiento =                               (SELECT MAX(lanzamiento) FROM
ordenadores)) ORDER BY modelo ASC;";

            SQLiteCommand cmd = new SQLiteCommand(mostrar, conexion);
            SQLiteDataReader datos = cmd.ExecuteReader();

            Console.WriteLine("Los modelos de la marca del ordenador más reciente
son: ");

            while (datos.Read())
            {
                string modelo = Convert.ToString(datos[0]);
                Console.WriteLine(modelo);
            }
        }

        static void CuartaConsulta()
        {
            string mostrar = @"SELECT * FROM ordenadores WHERE marca =
                               (SELECT MIN(marca) FROM ordenadores); ";

            SQLiteCommand cmd = new SQLiteCommand(mostrar, conexion);
            SQLiteDataReader datos = cmd.ExecuteReader();

            while (datos.Read())
            {
                string codigo = Convert.ToString(datos[0]);
                string marca = Convert.ToString(datos[1]);
                string modelo = Convert.ToString(datos[2]);
                int lanzamiento = Convert.ToInt32(datos[3]);
                Console.WriteLine("Código: {0}, Marca: {1}, Modelo: {2},
Lanzamiento: {3}", codigo, marca, modelo, lanzamiento);
            }
        }
    }
}

```



```

static void QuintaConsulta()
{
    string mostrar = @"SELECT modelo, marca FROM ordenadores WHERE marca
IN                                     (SELECT marca FROM ordenadores WHERE SUBSTR
(marca, 1, 1) != 'A');";

    SQLiteCommand cmd = new SQLiteCommand(mostrar, conexion);
    SQLiteDataReader datos = cmd.ExecuteReader();

    while (datos.Read())
    {
        string modelo = Convert.ToString(datos[0]);
        string marca = Convert.ToString(datos[1]);
        Console.WriteLine(modelo + marca);
    }

    static void CerrarConexion()
    {
        conexion.Close();
    }
}

```