

Enunciado:

Con el objetivo de aumentar la seguridad se pide agregar una clave de acceso a una base de datos Access, la clave a utilizar es “**oop**” y el nombre de la base de datos es “**BibliotecaDB.mdb**”.

Para agregar una clave debemos abrir la base de datos en modo exclusivo “**Open Exclusive**”, como se ve en la **Figura 1** y luego establecer la clave con el icono que muestra la **Figura 2**, una vez establecida la clave la siguiente vez que intentemos abrirla con la aplicación Microsoft Access, nos la pedirá tal como muestra la **Figura 3**.

En la **Figura 4**, podemos ver el archivo “**App.config**” de nuestra aplicación, con el atributo **connectionString** codificado en base64, para aumentar la seguridad y que no aparezca la clave “oop” en texto plano de simple lectura. Solo hemos dejado un comentario, color verde, a los fines académicos para que podamos ver el texto decodificado cosa que evidentemente no deberíamos hacer en una aplicación productiva real.

La **Figura 5**, muestra la interfaz de usuario de la aplicación, en ella podemos ver la cadena de conexión a la base de datos escrita en dos formas, **codificada** y **sin codificar**, en el archivo de configuración (“**App.config**”) se escribe solo en formato codificado, siendo la aplicación la responsable de decodificarla, más precisamente la clase “**Codificador**” cuyo código podemos ver en la **Figura 6**.

La clase “**Codificador**” solo tiene dos métodos una que codifica y otro que decodifica, ver **Figura 6**.

Existen varios sitios web que permiten codificar y decodificar base64, uno de ellos es <http://base64encode.net/> pero podría utilizarse cualquier otro para realizar el proceso de codificación de la cadena de conexión y su posterior escritura en el archivo “**App.config**”.

La **Figura 7**, muestra los 64 códigos ASCII, que conforma el código base64 aunque básicamente son a-z, A-Z y 0-9 para los primeros 62 valores. Otras variaciones Base64 comparten la misma propiedad pero utilizan diferentes símbolos en los dos últimos valores.

El resto de la aplicación utiliza la clase “**LibroPersistidor**” para realizar el CRUD (create, read, update and delete), de este modo es transparente o desconocido el tema de la seguridad ya que la clase se encarga de ocultar la implementación de la misma.

Figura 1

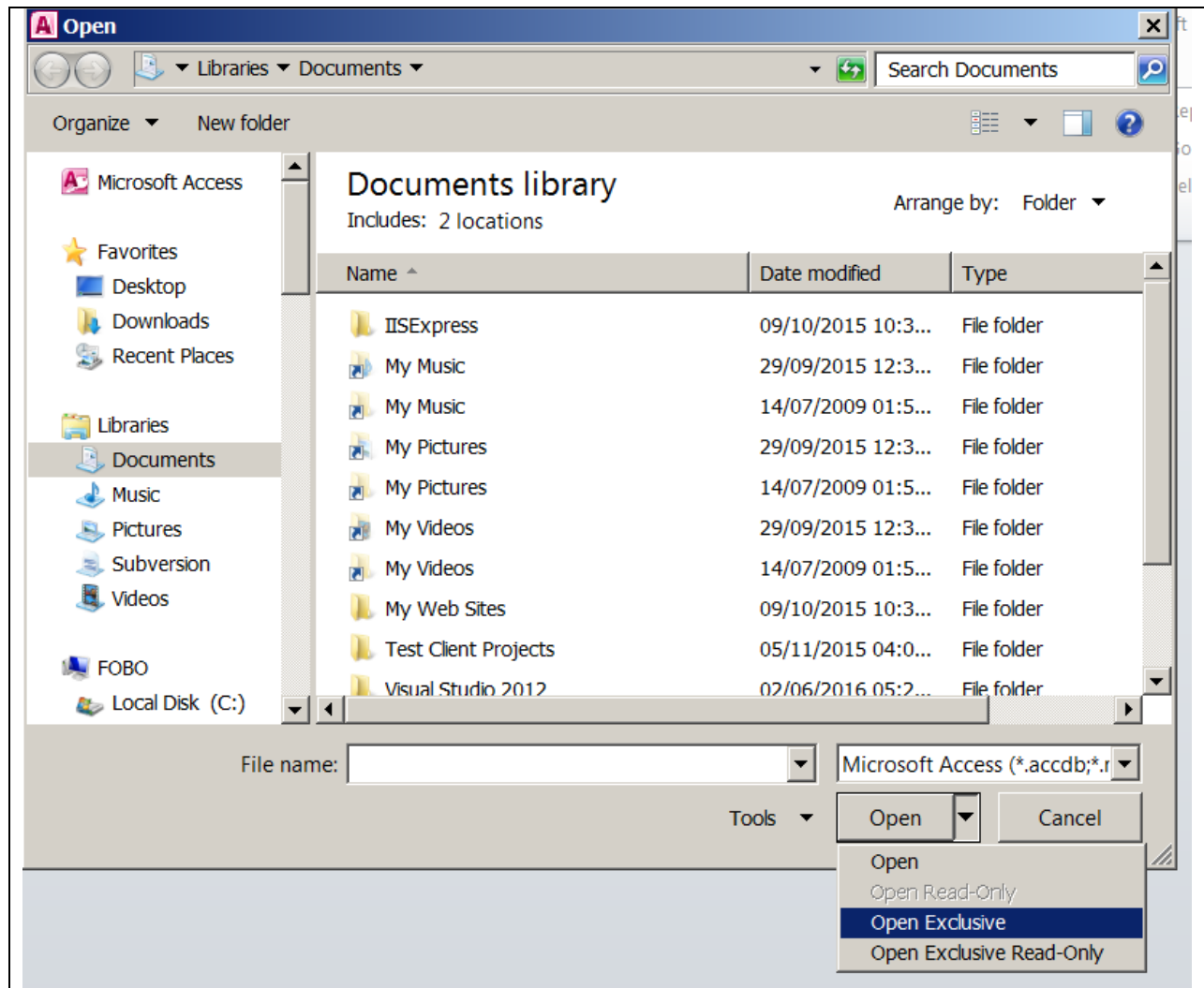


Figura 2

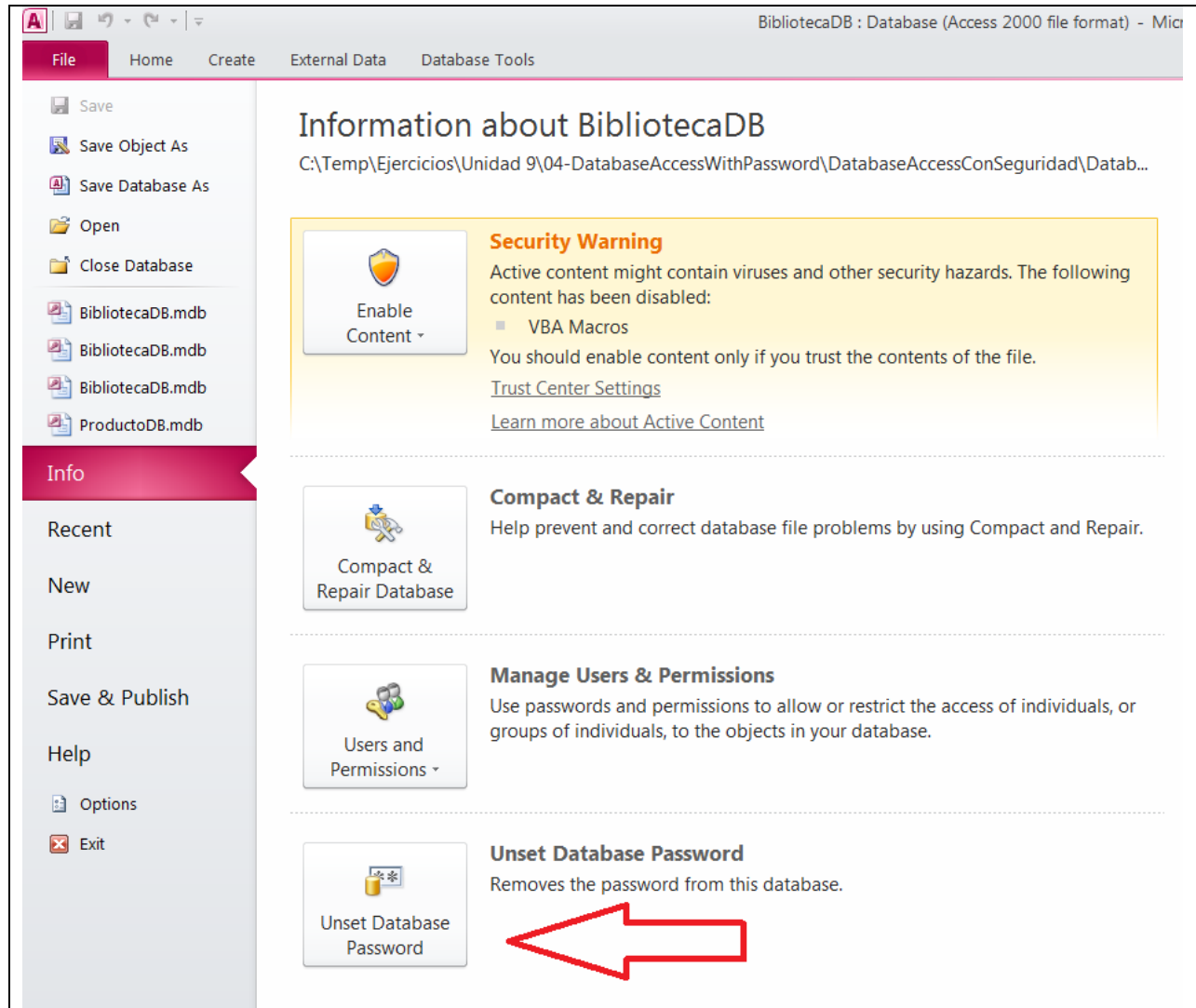


Figura 3

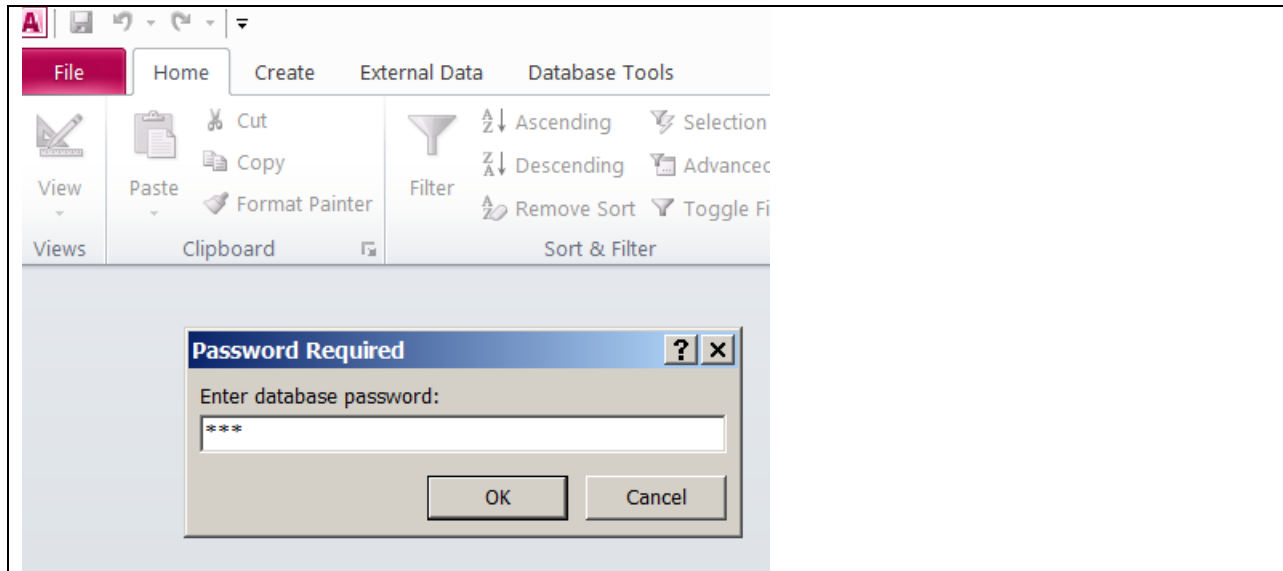


Figura 4

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <connectionStrings>
    <!--
      <add name="cnnString"
        connectionString="Provider=Microsoft.Jet.OLEDB.4.0; Data Source=..\Database\BibliotecaDB.mdb;
        Jet OLEDB:Database Password=oop;"
      />
    -->
    <!-- Cadena de conexión a la base de datos codificada en base64 -->
    <add name="cnnString"
      connectionString="UHJvdmlkZXI9TWljcm9zb2Z0LkpIdC5PTEVEQI40LjA7IERhdGEgU291cmNIPS4uXC4uXERhdGFyYXNIXEJpYmxb3RlY2FEQI5tZGI7IA0KICA"
    />
  </connectionStrings>
</configuration>
```

Figura 5

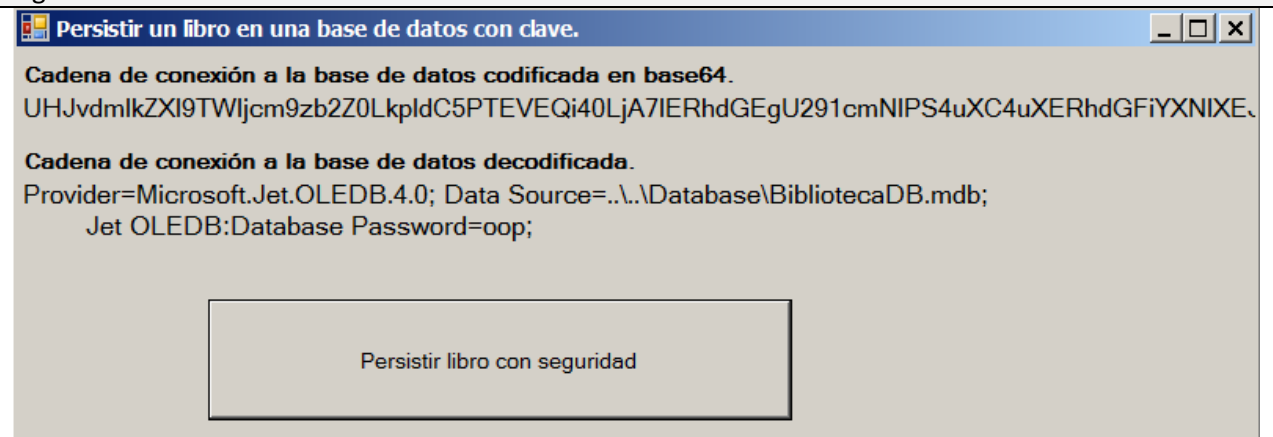


Figura 6

```

11 /// <remarks>
12 /// Puede utilizar el sitio http://base64encode.net/ para codificar y decodificar a y desde base64.
13 /// </remarks>
14 public static class Codificador
15 {
16     /// <summary>
17     /// Codifica un texto a Base64
18     /// </summary>
19     /// <param name="plainText"></param>
20     /// <returns></returns>
21     public static string Encode(string plainText)
22     {
23         var plainTextBytes = System.Text.Encoding.UTF8.GetBytes(plainText);
24         return System.Convert.ToBase64String(plainTextBytes);
25     }
26
27     /// <summary>
28     /// Decodifica un texto escrito en codigo Base64.
29     /// </summary>
30     /// <param name="base64EncodedData"></param>
31     /// <returns></returns>
32     public static string Decode(string base64EncodedData)
33     {
34         var base64EncodedBytes = System.Convert.FromBase64String(base64EncodedData);
35         return System.Text.Encoding.UTF8.GetString(base64EncodedBytes);
36     }
37 }

```

Figura 7

Base64 index table:							
Value	Char	Value	Char	Value	Char	Value	Char
0	A	16	Q	32	g	48	w
1	B	17	R	33	h	49	x
2	C	18	S	34	i	50	y
3	D	19	T	35	j	51	z
4	E	20	U	36	k	52	0
5	F	21	V	37	l	53	1
6	G	22	W	38	m	54	2
7	H	23	X	39	n	55	3
8	I	24	Y	40	o	56	4
9	J	25	Z	41	p	57	5
10	K	26	a	42	q	58	6
11	L	27	b	43	r	59	7
12	M	28	c	44	s	60	8
13	N	29	d	45	t	61	9
14	O	30	e	46	u	62	+
15	P	31	f	47	v	63	/