



### **Cuadernillo de examen**

ASIGNATURA	Programas de Aplicación II/III	CÓDIGO	305
CONVOCATORIA	Extraordinaria Febrero de 2000	PLAN DE ESTUDIOS	1994/1996
ESPECIALIDAD	Común	CURSO	3º
TURNO	Tarde	CENTRO	Facultad
CARÁCTER	Anual	CURSO ACADÉMICO	1999/2000
DURACIÓN APROXIMADA	2 horas		

## **Soluciones al examen**

### **Pruebas teórico-prácticas**

1. Tipos de componentes ActiveX. Enumere los distintos tipos de componentes ActiveX, explicando sus diferencias y distintos usos.

*Apuntes de clase*

2. Objetos Recordset. Explique los distintos tipos de objetos Recordset del modelo DAO, indicando sus diferencias en cuanto a comportamiento y métodos de búsqueda que se pueden utilizar en cada uno de ellos.

*Apuntes de clase*

3. Arrays en Visual Basic. Declaración de arrays en Visual Basic. Ámbito de los arrays. Arrays Dinámicos.

*Apuntes de clase*

Utilizando un array dinámico, realice la declaración de una pila e implemente un procedimiento que permita introducir nuevos elementos en la misma.

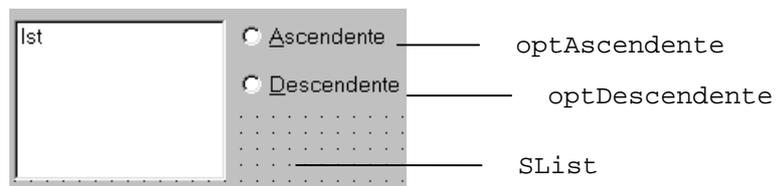
```
Private Pila() As String 'Declara el array dinámico Pila  
Private cima As Long
```

```
Private Sub Insertar(Pila() As String, e As String)  
    cima = cima + 1  
    ReDim Preserve Pila(1 To cima)  
    Pila(cima) = e  
End Sub
```

**Puntuación: 1 punto cada pregunta**

### **Pruebas prácticas**

1. Se desea diseñar un control ActiveX (SList) que permita tener un control ListBox ordenado ascendente o descendientemente. El control constará de un ListBox y dos botones de opción.



Deberá exportar los métodos AddItem, RemoveItem y Clear del control lst. También deberá exportar el evento Click del control ListBox y la propiedad Text del mismo. Cuando el usuario pulse sobre el botón de opción optAscendente el contenido de la lista se ordenara de forma ascendente, cuando el usuario pulse sobre el botón de opción optDescendente lo hará de forma descendente. Además se deberá crear una propiedad SortType que indique el tipo de ordenación actual de la lista. Esta propiedad tomará un valor 0, cuando la lista esté desordenada, 1 cuando esté ordenada ascendientemente y 2 cuando esté ordenada descendientemente.

**Puntuación: 3 puntos**

Option Explicit



### **Cuadernillo de examen**

ASIGNATURA	Programas de Aplicación II/III	CÓDIGO	305
CONVOCATORIA	Extraordinaria Febrero de 2000	PLAN DE ESTUDIOS	1994/1996
ESPECIALIDAD	Común	CURSO	3º
TURNO	Tarde	CENTRO	Facultad
CARÁCTER	Anual	CURSO ACADÉMICO	1999/2000
DURACIÓN APROXIMADA	2 horas		

```
Option Compare Text
'Declaraciones de eventos
Event Click()
'Valores por omisión de las propiedades
Const m_def_SortType = 0
'Propiedades
Dim m_SortType As Integer

Public Sub AddItem(ByVal Item As String, Optional ByVal Index As Variant)
    lst.AddItem Item, Index
End Sub

Public Sub Clear()
    lst.Clear
End Sub

Private Sub optAscendente_Click()
    Dim i As Integer
    Dim j As Integer
    Dim aux As String

    For i = 0 To lst.ListCount - 2
        For j = 0 To lst.ListCount - 2 - i
            If lst.List(j) > lst.List(j + 1) Then
                aux = lst.List(j)
                lst.List(j) = lst.List(j + 1)
                lst.List(j + 1) = aux
            End If
        Next
    Next
    SortType = 1
End Sub

Private Sub optDescendente_Click()
    Dim i As Integer
    Dim j As Integer
    Dim aux As String

    For i = 0 To lst.ListCount - 2
        For j = 0 To lst.ListCount - 2 - i
            If lst.List(j) < lst.List(j + 1) Then
                aux = lst.List(j)
                lst.List(j) = lst.List(j + 1)
                lst.List(j + 1) = aux
            End If
        Next
    Next
    SortType = 2
End Sub

Public Sub RemoveItem(ByVal Index As Integer)
    lst.RemoveItem Index
End Sub
```



**Cuadernillo de examen**

ASIGNATURA	Programas de Aplicación II/III	CÓDIGO	305
CONVOCATORIA	Extraordinaria Febrero de 2000	PLAN DE ESTUDIOS	1994/1996
ESPECIALIDAD	Común	CURSO	3º
TURNO	Tarde	CENTRO	Facultad
CARÁCTER	Anual	CURSO ACADÉMICO	1999/2000
DURACIÓN APROXIMADA	2 horas		

```

Private Sub UserControl_ReadProperties(PropBag As PropertyBag)
    lst.Text = PropBag.ReadProperty("Text", "")
    m_SortType = PropBag.ReadProperty("SortType", m_def_SortType)
End Sub

Private Sub UserControl_WriteProperties(PropBag As PropertyBag)
    Call PropBag.WriteProperty("Text", lst.Text, "")
    Call PropBag.WriteProperty("SortType", m_SortType, m_def_SortType)
End Sub

Private Sub lst_Click()
    RaiseEvent Click
End Sub

Private Sub UserControl_InitProperties()
    m_SortType = m_def_SortType
End Sub

Public Property Get Text() As String
    Text = lst.Text
End Property

Public Property Let Text(ByVal New_Text As String)
    lst.Text() = New_Text
    PropertyChanged "Text"
End Property

Public Property Get SortType() As Integer
    SortType = m_SortType
End Property

Public Property Let SortType(ByVal New_SortType As Integer)
    m_SortType = New_SortType
    PropertyChanged "SortType"
End Property
  
```

2. El Club Deportivo Villamanrique guarda información sobre sus socios en una base de datos Access (Club.mdb) que, entre otras, tiene las siguientes tablas:

Tabla: Socios		
Campo	Formato	Observaciones
DNI	Texto	Clave primaria de nombre PrimaryKey
Nombre	Texto	Apellidos y nombre del socio
CuotaMensual	Númérico	Precio de la cuota que el socio tiene que abonar mensualmente

**Cuadernillo de examen**

ASIGNATURA	Programas de Aplicación II/III	CÓDIGO	305
CONVOCATORIA	Extraordinaria Febrero de 2000	PLAN DE ESTUDIOS	1994/1996
ESPECIALIDAD	Común	CURSO	3º
TURNO	Tarde	CENTRO	Facultad
CARÁCTER	Anual	CURSO ACADÉMICO	1999/2000
DURACIÓN APROXIMADA	2 horas		

**Tabla: Actividades**

Campo	Formato	Observaciones
IDActividad	Numérico	Clave primaria de nombre PrimaryKey
Denominación	Texto	Nombre de la actividad
PrecioMes	Numérico	Precio mensual que cuesta apuntarse a la actividad

**Tabla: SocioActividad (se usa para relacionar las dos tablas)**

Campo	Formato	Observaciones
IDActividad	Numérico	Junto con el campo DNI forma la clave primaria de nombre PrimaryKey
DNI	Texto	Junto con el campo IDActividad forma la clave primaria de nombre PrimaryKey
FechaInscripción	Numérico	Fecha en la que se realizó la inscripción

Se desea hacer un programa en Visual Basic que gestione la inscripción de los socios a las actividades. El formulario de la aplicación presenta el siguiente aspecto:

cmdBuscar

cmdApuntars

cmdTotal

El formulario no contiene ningún otro control. El formulario debe contemplar las siguientes opciones:

- Al cargar el formulario, además de abrir la base de datos y los recordset, se deberán cargar en el combo cmbActividades el nombre de todas las actividades disponibles.

**Puntuación: 0,5 puntos**

```
Option Explicit
Dim db As Database
Dim rsSocios As Recordset
Dim rsActividades As Recordset
Dim rsSocioActividad As Recordset
```

**Cuadernillo de examen**

ASIGNATURA	Programas de Aplicación II/III	CÓDIGO	305
CONVOCATORIA	Extraordinaria Febrero de 2000	PLAN DE ESTUDIOS	1994/1996
ESPECIALIDAD	Común	CURSO	3º
TURNO	Tarde	CENTRO	Facultad
CARÁCTER	Anual	CURSO ACADÉMICO	1999/2000
DURACIÓN APROXIMADA	2 horas		

```
Private Sub Form_Load()  
    'Abrir la base de datos y los recordset  
    Set db = DBEngine.Workspaces(0).OpenDatabase("\\luis\bd\club.mdb")  
    Set rsSocios = db.OpenRecordset("Socios")  
    Set rsActividades = db.OpenRecordset("Actividades", dbOpenDynaset)  
  
    'Cargar el combo con las actividades  
    Do While Not rsActividades.EOF  
        cmbActividades.AddItem rsActividades("Denominación")  
        rsActividades.MoveNext  
    Loop  
  
End Sub
```

- Al pulsar sobre el botón cmdBuscar se buscará el DNI control txtDNI en la base de datos de socios. Si dicho socio no existiera, un mensaje advertiría de la incidencia. Si existe, aparecerá su nombre en el control txtNombre y todas las actividades a las que está apuntado en el ListBox lstActividades. Al pulsar sobre cualquiera de dichas actividades, aparecerá la fecha de inscripción a la misma en el control txtFecha.

**Puntuación: 1,5 puntos**

```
Private Sub cmdBuscarSocio_Click()  
    Dim Qry As String  
  
    'Busca el socio  
    rsSocios.Index = "PrimaryKey"  
    rsSocios.Seek "=", txtDNI  
    If rsSocios.NoMatch Then  
        MsgBox "El socio no existe", vbInformation, _  
            "Club Deportivo Villamanrique"  
        txtNombre = Empty  
        lstActividades.Clear  
        txtFecha = Empty  
    Else  
        txtNombre = rsSocios("Nombre")  
        'Se recuperan las actividades a las que está apuntado  
        Qry = "SELECT * FROM Actividades, SocioActividad "  
        Qry = Qry & " WHERE Actividades.IDActividad = " & _  
            SocioActividad.IDActividad "  
        Qry = Qry & " AND SocioActividad.DNI = '" & txtDNI & "'"   
        Set rsSocioActividad = db.OpenRecordset(Qry)  
        lstActividades.Clear  
  
        'Si no tiene actividades  
        If rsSocioActividad.RecordCount = 0 Then  
            MsgBox "No tiene actividades", vbInformation, _  
                "Club Deportivo Villamanrique"  
        Else  
            'y se carga el ListBox  
            Do While Not rsSocioActividad.EOF  
                lstActividades.AddItem rsSocioActividad("Denominación")  
                rsSocioActividad.MoveNext  
            Loop  
            rsSocioActividad.MoveFirst  
        End If  
    End If
```

**Cuadernillo de examen**

ASIGNATURA	Programas de Aplicación II/III	CÓDIGO	305
CONVOCATORIA	Extraordinaria Febrero de 2000	PLAN DE ESTUDIOS	1994/1996
ESPECIALIDAD	Común	CURSO	3º
TURNO	Tarde	CENTRO	Facultad
CARÁCTER	Anual	CURSO ACADÉMICO	1999/2000
DURACIÓN APROXIMADA	2 horas		

```
        txtFecha = Empty
    End If
End Sub

Private Sub lstActividades_Click()
    rsActividades.FindFirst "Denominación = '" & lstActividades & "'"
    rsSocioActividad.FindFirst "Actividades.IDActividad = " & _
        rsActividades("IDActividad") & _
        " AND SocioActividad.DNI = '" & rsSocioActividad("DNI") & "'"
    txtFecha = rsSocioActividad("FechaInscripción")
End Sub
```

- Al pulsar sobre el botón cmdApuntarse se apuntará al socio en la actividad que esté señalada en el combo cmbActividades, La fecha de inscripción será la fecha actual.. Si el socio ya estuviera apuntado, un mensaje indicaría la incidencia.

**Puntuación: 1 punto**

```
Private Sub cmdApuntarse_Click()
    On Error GoTo ErrorApuntarse

    Dim rs As Recordset

    Set rs = db.OpenRecordset("SocioActividad")
    rsActividades.FindFirst "Denominación = '" & cmbActividades & "'"
    rs.AddNew
    rs("IDActividad") = rsActividades("IDActividad")
    rs("DNI") = txtDNI
    rs("FechaInscripción") = Date
    rs.Update

    cmdBuscarActividades_Click
Exit Sub

ErrorApuntarse:
    If Err = 3022 Then
        MsgBox "El socio ya se ha apuntado a dicha actividad", _
            vbInformation, "Club Deportivo Villamanrique"
    Else
        MsgBox Err.Description
    End If
End Sub
```

- Al pulsar sobre el botón cmdTotal se calculará el total mensual que debe pagar el socio y aparecerá en el control txtTotal. Dicho importe se calculará sumando el precio mensual de cada una de las actividades a las que está apuntado y la cuota mensual del socio.

**Puntuación: 1 punto**

```
Private Sub cmdTotal_Click()
    Dim Qry As String
    Dim rs As Recordset

    'rsActividades.FindFirst "Denominación = '" & lstActividades & "'"

    Qry = "SELECT Sum(PrecioMes) AS tot FROM SocioActividad, Actividades"
```



### **Cuadernillo de examen**

---

ASIGNATURA	Programas de Aplicación II/III	CÓDIGO	305
CONVOCATORIA	Extraordinaria Febrero de 2000	PLAN DE ESTUDIOS	1994/1996
ESPECIALIDAD	Común	CURSO	3º
TURNO	Tarde	CENTRO	Facultad
CARÁCTER	Anual	CURSO ACADÉMICO	1999/2000
DURACIÓN APROXIMADA	2 horas		

---

```
Qry = Qry & " WHERE Actividades.IDActividad = _  
SocioActividad.IDActividad " ' '  
Qry = Qry & " AND DNI = '" & txtDNI & "' "  
  
Set rs = db.OpenRecordset(Qry)  
  
txtTotal = rs("Tot") + rsSocios("CuotaMensual")  
  
End Sub
```