



Excel 2019 VBA – Módulo II

Gabarito

Teste seus conhecimentos

Capítulo 1	Capítulo 2	Capítulo 3	Capítulo 4
1 - C	1 - A	1 - A	1 - D
2 - B	2 - C	2 - D	2 - C
3 - B	3 - C	3 - A	3 - D
4 - A	4 - C	4 - E	4 - C
5 - E	5 - C	5 - C	5 - E

Capítulo 5	Capítulo 6	Capítulo 7	Capítulo 8
1 - E	1 - E	1 - C	1 - E
2 - D	2 - D	2 - D	2 - D
3 - C	3 - C	3 - A	3 - D
4 - D	4 - B	4 - A	4 - A
5 - D	5 - E	5 - A	5 - B

Capítulo 9	Capítulo 10
1 - B	1 - E
2 - D	2 - D
3 - D	3 - A
4 - D	4 - C
5 - C	5 - D

Mãos à obra

Laboratório do Capítulo 2 (1)

Exercício 1

```
SELECT * FROM FUNCIONARIO WHERE CIDADE<>"SÃO PAULO" AND SALARIO BETWEEN 1000 AND 2000
```

Exercício 2

```
SELECT NOMEF, ADMISSAO, SALARIO FROM FUNCIONARIO WHERE ADMISSAO BETWEEN #1/1/2000# AND #31/12/2006#
```

Exercício 3

```
SELECT NOMEF, NOMEDEPTO FROM FUNCIONARIO INNER JOIN DEPTO ON FUNCIONARIO.CODDEPTO = DEPTO.CODDEPTO WHERE NOMEDEPTO IN ("RH", "OUVIDORIA")
```

Exercício 4

```
select * from projeto where dtinicio>=#1/1/2007#
```

Exercício 5

```
select top 1 nomef, cidade, salario from funcionario where cidade="são paulo" order by 3 desc
```

Laboratório do Capítulo 3 (1)

```
Sub ImportTbl()  
'Esta sub retorna todos os dados contidos na tabela Customers do  
banco de dados Northwind  
    Dim Db As Database  
    Dim rs As Recordset  
    Dim Ws As Object  
    Dim i As Integer  
    Dim N As Integer  
  
    'Esta linha definirá o objeto Ws como Sheets("Planilha1")  
    Set Ws = Sheets("Produtos")  
  
    'Seta o banco de dados  
    Set Db = OpenDatabase(ThisWorkbook.Path & "\Northwind.accdb")  
  
    'Seta o RecordSet para retornar todos os dados da tabela Custom-  
ers  
    Set rs = Db.OpenRecordset("Products")  
  
    'Este trecho de código apagará todos os dados na planilha sele-  
cionará a célula A1  
    Cells.ClearContents  
  
    'Este loop coletará os nomes dos campos do cabeçalho da tabela  
'e os replicará na primeira linha a partir da célula A1  
    For i = 0 To rs.Fields.Count - 1  
        Ws.Cells(1, i + 1).Value = rs.Fields(i).Name  
    Next i  
  
    'Esta linha de código aplicará a formatação negrito ao cabeçalho  
da tabela  
    Ws.Range(Ws.Cells(1, 1), Ws.Cells(1, rs.Fields.Count)).Font.  
Bold = True  
  
    'Esta linha de código inserirá o RecordSet a partir da célula A2  
    Ws.Range("A2").CopyFromRecordset rs  
  
    'O código a seguir irá selecionar todos os dados e auto ajustar  
a largura das colunas  
    Cells.Columns.AutoFit  
  
    'Fecha o banco de dados e o RecordSet  
    rs.Close  
    Db.Close  
  
    'Define o número de registros importados  
    N = Sheets(2).Cells(rows.count, 1).End(xlUp).Row - 1  
  
    'Mensagem de sucesso  
    MsgBox "foram importados " & N & " registros!", vbInformation,  
"Produtos"  
  
End Sub
```

Laboratório do Capítulo 4 (1)

```
Sub RecallBD()  
'Testa captura de dados de base em planilha  
  Dim MeuBD As Database  
  Dim MinhaTabela As Recordset  
  Dim Ssql As String  
  Dim ThisPath As String  
  Dim MeuArq As String  
  
'Define nome e endereço da base XLS  
  ThisPath = ThisWorkbook.Path  
  MeuArq = "\Medicamentos.xlsx"  
  
'Seta a variável objeto  
  Set MeuBD = _  
  OpenDatabase(ThisPath & MeuArq, False, False, "Excel 12.0")  
  
'Monta a SQL  
  Ssql = Empty  
  Ssql = Ssql & "SELECT MEDICAMENTOS, QTDE FROM [Dados$] " & vbCr  
  Ssql = Ssql & "WHERE QTDE > 50 " & vbCr  
  
'Seta o objeto RecordSet  
  Set MinhaTabela = MeuBD.OpenRecordset(Ssql)  
  
'Registra os dados na Planilha  
  Cells.ClearContents  
  Cells(1) = "MEDICAMENTOS"  
  Cells(2) = "QTDE"  
  Cells(2, 1).CopyFromRecordset MinhaTabela  
  
  Cells.Columns.AutoFit  
  
  MeuBD.Close  
  
End Sub
```

Laboratório do Capítulo 5 (1)

```
Private Sub ImportTable()  
    'Base para o exercício  
    Dim cn As ADODB.Connection, rs As ADODB.Recordset, sql As  
String, linha As Long  
    'Define a conexão com o banco de dados Northwind.accdb  
    Set cn = New ADODB.Connection  
    cn.Provider = "Microsoft.ACE.OLEDB.12.0"  
    cn.ConnectionString = "Data Source=" & ThisWorkbook.Path & "\  
Northwind.accdb"  
    cn.Open  
    'define um novo objeto recordset  
    Set rs = New ADODB.Recordset  
    'define a instrução sql  
    sql = "SELECT Pedidos.CodigoDoCliente, Sum(DetalhesPedidos.  
Quantidade), "  
        & "Sum([DetalhesPedidos]![Quantidade]*  
        [DetalhesPedidos]![PrecoUnitario]"  
        & "*(1-[DetalhesPedidos]![Desconto])) "  
        & "FROM Pedidos INNER JOIN DetalhesPedidos "  
        & "ON Pedidos.NumeroDoPedido = DetalhesPedidos.NumeroDoPe-  
dido "  
        & "GROUP BY Pedidos.CodigoDoCliente "  
        & "ORDER BY Sum([DetalhesPedidos]![Quantidade]*  
        [DetalhesPedidos]![PrecoUnitario]"  
        & "*(1-[DetalhesPedidos]![Desconto])) DESC"  
    'gera o recordset para o sql sobre a conexão definida  
    rs.Open sql, cn  
    Cells.Clear  
    'define o cabeçalho das células no excel  
    Range("A1").Value = "CÓDIGO"  
    Range("B1").Value = "QUANT TOTAL"  
    Range("C1").Value = "VALOR TOTAL"  
    Range("A2").CopyFromRecordset rs  
    linha = Cells(Rows.Count, 1).End(xlUp).Row  
    Range(Cells(2, 2), Cells(linha, 2)).NumberFormat = "#,###"  
    Range(Cells(2, 3), Cells(linha, 3)).NumberFormat = "$  
#,##0.00"  
    Columns.AutoFit  
    cn.Close  
End Sub
```

Laboratório do Capítulo 6 (1)

```
Sub Inserir_n_Registros()  
  
Dim cliente As String, produto As String, valor As Currency  
Dim LINHA As Integer  
Conectar  
  
For LINHA = 2 To Cells(Rows.Count, 1).End(xlUp).Row  
  
cliente = Cells(LINHA, 1)  
produto = Cells(LINHA, 2)  
valor = Cells(LINHA, 3)  
  
sql = "INSERT INTO VENDAS (CLIENTE, PRODUTO, VALOR) " & _  
" VALUES ('" & cliente & "', '" & produto & "', " & valor & ");"  
  
cn.Execute (sql)  
Next LINHA  
MsgBox "DADOS INSERIDOS COM SUCESSO", vbInformation, "VENDAS"  
  
Desconectar  
End Sub
```

Laboratório do Capítulo 7 (1)

```
Private Caminho As String
Private NomeArq As String
Private CamiArq As String
Private Arq As String
Private App As Application

Sub Import_Arq_Texto()
'Importa um arquivo texto e insere uma planilha após a primeira
para receber os dados

'Define o objeto application
Set App = Application

'Define os estados de application
App.ScreenUpdating = False
App.DisplayAlerts = False

'Verifica qual arquivo txt será importado
NomeArq = ActiveCell

'Define nome e caminho dos arquivos envolvidos
Arq = ActiveWorkbook.Name
Caminho = ThisWorkbook.Path
CamiArq = Caminho & "\" & NomeArq

'Verifica se a pastas está com uma planilha apenas.
'Se houver mais de uma planilha, elimina as excedentes
n = Sheets.Count

Do Until n = 1
    Sheets(n).Delete
    n = n - 1
Loop

'Abre o arquivo txt
Workbooks.OpenText Filename:=CamiArq, _
    DataType:=xlDelimited, _
    TextQualifier:=xlTextQualifierDoubleQuote, _
    Tab:=True, Semicolon:=False, _
    Comma:=False, Space:=False

'Cria a segunda planilha e transfere os dados
ActiveSheet.Move After:=Workbooks(Arq).Sheets(1)

'Ajusta o tamanho das colunas
Sheets(2).Cells.Columns.AutoFit

'Devolve os estados default de application
App.DisplayAlerts = True
App.ScreenUpdating = True

End Sub
```

Laboratório do Capítulo 8 (1)

```

Sub Preenchendo_Combo()
Set fs = CreateObject("Scripting.FileSystemObject")
Set ff = fs.GetFolder(ThisWorkbook.Path & "\Produtos")
Set fc = ff.Files
Sheets("Planilha1").ComboBox1.Clear
For Each fl In fc
    Sheets("Planilha1").ComboBox1.AddItem _
        Left(fl.Name, Len(fl.Name) - 4)
Next fl
Sheets("Planilha1").Image1.Picture = LoadPicture()
End Sub

Sub Preenchendo_Image()
On Error GoTo yu
Sheets("Planilha1").Image1.Picture = _
LoadPicture(ThisWorkbook.Path & "\Produtos\" & _
    Sheets("Planilha1").ComboBox1 & ".jpg")
Exit Sub
yu:
Sheets("Planilha1").Image1.Picture = LoadPicture()
End Sub

```

Laboratório do Capítulo 9 (1)

```

Private GrafEd As String

Private Sub cmb_Fechar_Click()
    Unload Me
    Sheets(1).Select
End Sub

Private Sub UserForm_Initialize()
    Application.ScreenUpdating = False

    GrafEd = ThisWorkbook.Path & "\grafico1.jpg"

    Sheets(2).ChartObjects(1).Activate
    ActiveChart.Export GrafEd, "jpg"
    Me.image1.Picture = LoadPicture(GrafEd)
    Me.image1.PictureSizeMode = fmPictureSizeModeStretch

    Application.ScreenUpdating = True

End Sub

```



Laboratório do Capítulo 10 (1)

Formulário:

```
Private Sub cmb_Mensagem_Click()  
    Set meuCliente = New cls_Cliente  
  
    NomeDoCliente = Me.tbv_Cliente  
    meuCliente.Name = NomeDoCliente  
    MsgBox "Bem-vindo " & UCase(meuCliente.Name), vbExclamation,  
    "Cliente"  
  
    Unload Me  
End Sub
```

Módulo:

```
Public meuCliente As cls_Cliente  
Public NomeCliente As String  
Public NomeDoCliente As String
```

Botão Formulário:

```
Sub Formulario()  
    ufm_Clientes.Show  
End Sub
```

Classe:

```
Property Let Name(ByVal nomeEntrada As String)  
    NomeCliente = nomeEntrada  
End Property  
Property Get Name() As String  
    Name = NomeCliente  
End Property
```