EXCEL VOCABULARY QUIZZER Github.Ver

Not prefer to show my real name and student ID

Outline

1. Memorizing Vocabulary AS-IS
2. Features of Excel Vocab Quizzer
3. See how it works
4. Programming details and explanation

When you studying a new language, you need to memorize **vocabularies**!

Problems you may have:

► You need to install an app on your computer

► You may have to watch Ads

► You can't find an ideal vocabulary book

► You want an easier way to customize your vocabulary book

Now you have Excel Vocabulary Quizzer!

In a .xlsm file with a well-designed program,

which contains:

▶ Simple and user-friendly interfaces

▶ Number of questions customizable

▶ Just enter/paste your Word& Answer to add words in the worksheet "Dictionary"

▶ Score report output with advices

Let's open the file and see!

Programming details and explanation

Structure

- Forms
  - QuizingForm
  - QuizInitForm
- Modules
  - CreateResultReport
  - GlobalVariables
  - OpenQuizModule
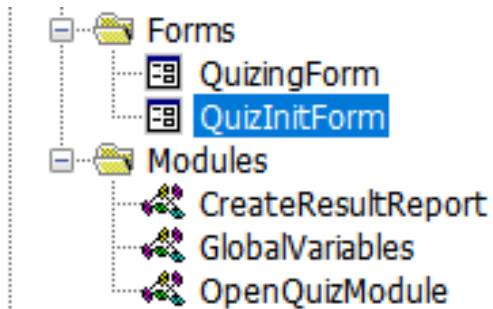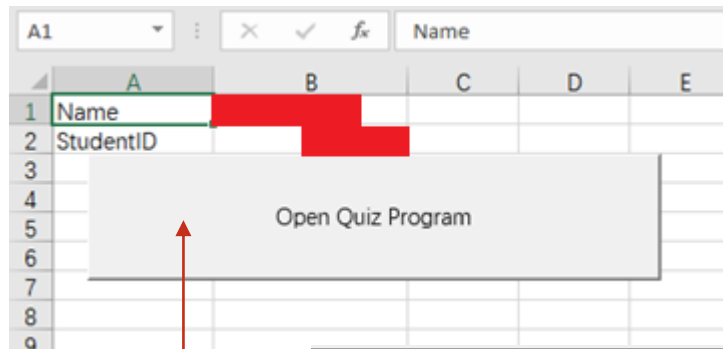
OpenQuizModule

Bind procedure to the button

```
Sub OpenQuiz()
QuizInitForm.Show
End Sub
```

QuizInitForm


```
'Reads the preset. I used this for testing.
'It's almost useless now. LOL
Private Sub UserForm_Initialize()
NumOfQuestions = 3
Me.TbxNumOfQuestions.Value = NumOfQuestions
'Debug.Print "Num of questions: " & NumOfQuestions
End Sub

'Exit
Private Sub BtnExit_Click()
Unload QuizInitForm
End Sub

'Set the number and Start
Private Sub BtnStart_Click()
NumOfQuestions = Me.TbxNumOfQuestions.Value
QuizingForm.Show
End Sub
```

GlobalVariables

Use variables across Modules and Forms, outside the procedure or function

```
Option Explicit
' Declare Global Variable
Public NumOfQuestions As Integer
Public Stopped As Boolean
Public rightAnsPos As Integer
Public tempForFunction As Integer
Public rightNo As Integer
Public totalRows As Integer
Public cardQuestion As String
Public cardAnswer As String
```

```
wrongAnswer1 As String
wrongAnswer2 As String
wrongAnswer3 As String
isFinished As Boolean
CurrentQuestionNum As Integer
Score As Integer
WrongQuestionNumForArray As Integer
WrongQuestionNum() As Integer
```

But I don't recommend this! This leads to tighter COUPLING! Which would cause bugs easily if you are careless!

Programming details and explanation

QuizzingForm

Get a random valid word and few answers from the sheet "Dictionary", by obtaining random numbers.

```vba
' Fresh the QA, determine 1 right and 3 wrong answers
Function RefreshQA(rightNo, totalRows, dict)
    rightNo = RndBetween(2, totalRows)
    cardQuestion = dict.Range("A" + VBA.Conversion.CStr(rightNo)).Value
    Me.LblCardFront.Caption = cardQuestion
    cardAnswer = dict.Range("B" + VBA.Conversion.CStr(rightNo)).Value
    wrongAnswer1 = dict.Range("B" + GetWrongAnswer(rightNo, totalRows)).Value
    wrongAnswer2 = dict.Range("B" + GetWrongAnswer(rightNo, totalRows)).Value
    wrongAnswer3 = dict.Range("B" + GetWrongAnswer(rightNo, totalRows)).Value
    ' Put Answers
    rightAnsPos = RndBetween(1, 4)
    RefreshQA = 0
End Function
```

Quizing

Question 1/n

CardFront

A

B

C

D

Get your result report

Give the correct answer a random position, according to the cases it will be at A, B, C, or D.

```vba
' Fresh the QA UI, setting the caption
Function RefreshQAUI(RightAnswerPosition, cardAnswer, wrongAnswer1, wrongAnswe
    If isFinished = True Then
        tempForFunction = FreezeAndEnableCheckReport()
    Else
        Select Case RightAnswerPosition
            Case 1
                Me.BtnSelectiveA.Caption = cardAnswer
                Me.BtnSelectiveB.Caption = wrongAnswer1
                Me.BtnSelectiveC.Caption = wrongAnswer2
                Me.BtnSelectiveD.Caption = wrongAnswer3
            Case 2
                Me.BtnSelectiveA.Caption = wrongAnswer1
                Me.BtnSelectiveB.Caption = cardAnswer
                Me.BtnSelectiveC.Caption = wrongAnswer2
                Me.BtnSelectiveD.Caption = wrongAnswer3
            Case 3
                Me.BtnSelectiveA.Caption = wrongAnswer1
                Me.BtnSelectiveB.Caption = wrongAnswer2
                Me.BtnSelectiveC.Caption = cardAnswer
                Me.BtnSelectiveD.Caption = wrongAnswer3
            Case 4
                Me.BtnSelectiveA.Caption = wrongAnswer1
                Me.BtnSelectiveB.Caption = wrongAnswer2
                Me.BtnSelectiveC.Caption = wrongAnswer3
                Me.BtnSelectiveD.Caption = cardAnswer
        End Select
        Me.FrmQuestionNum.Caption = "Question " & _
        VBA.Conversion.CStr(CurrentQuestionNum) & "/" & VBA.Conversion.CStr(Nu
    End If
    RefreshQAUI = 0
End Function
```

Freeze when finished

Updating the caption on the frame

If you are not satisfied with the default Rnd function, **create one yourself.**

```vba
' Random value between a-b, including a and b
Function RndBetween(a, b) As Integer
    ' Randomize is EXTREME IMPORTANT or it will output SAME VALUE!!!
    Randomize
    RndBetween = Int(a + (b - a + 1) * Rnd)
End Function
```
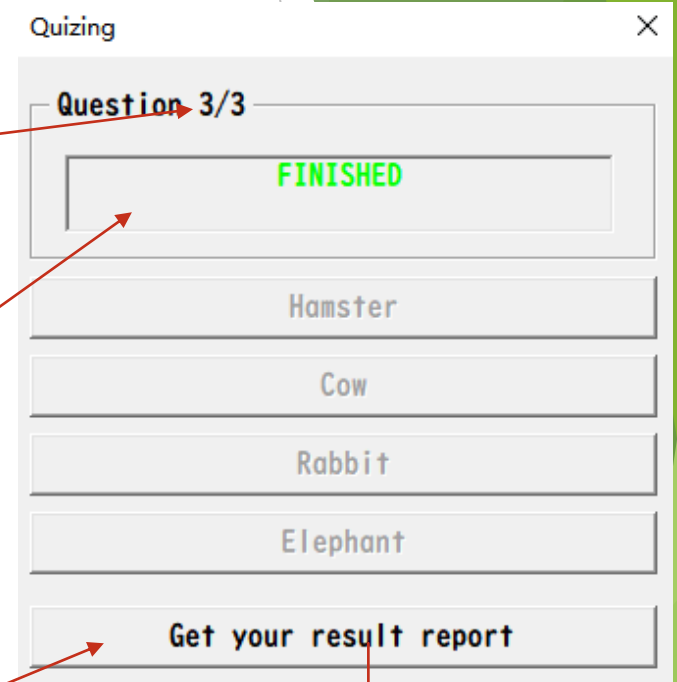
Programming details and explanation

```
' Check whether the quiz is finished and increasing current question number
Function CheckFinishedAndIncrease(dict)
    If CurrentQuestionNum = NumOfQuestions Then
        Debug.Print ("Ended")
        isFinished = True
    Else
        CurrentQuestionNum = CurrentQuestionNum + 1
    End If
    CheckFinishedAndIncrease = 0
End Function
```

```
' Fresh the QA UI, setting the caption
Function RefreshQAUI(RightAnswerPosition, cardAnswer, wrongAnswer1, wrongAnswe
    If isFinished = True Then
        tempForFunction = FreezeAndEnableCheckReport()
    Else
        Select Case RightAnswerPosition
```

Freeze when finished

```
' Freeze 4 selections and allow user click Show Report Button
Function FreezeAndEnableCheckReport()
    Me.LblCardFront.Caption = "FINISHED"
    Me.LblCardFront.ForeColor = &HFF00&
    Me.BtnSelectiveA.Enabled = False
    Me.BtnSelectiveB.Enabled = False
    Me.BtnSelectiveC.Enabled = False
    Me.BtnSelectiveD.Enabled = False
    Me.BtnSeeReport.Enabled = True
    FreezeAndEnableCheckReport = 0
End Function
```

Quizing ✕

Question 3/3

FINISHED

Hamster

Cow

Rabbit

Elephant

Get your result report

📁 Modules
　🎇 CreateResultReport

Programming details and explanation

```vba
' Returns which line the wrong answer in
Function GetWrongAnswer(RightAnsNo, tr) As String
    Dim WrongAns As String
    Do
        WrongAns = VBA.Conversion.CStr(RndBetween(2, tr))
    Loop While WrongAns = RightAnsNo
    GetWrongAnswer = WrongAns
End Function
```

```vba
' Check the answer, add score, and collect mistakes
Function CheckAns(rightAnsPos, pos)
    If rightAnsPos <> pos Then
        Debug.Print ("Wrong")
        WrongQuestionNumForArray = WrongQuestionNumForArray + 1
        ReDim Preserve WrongQuestionNum(WrongQuestionNumForArray)
        WrongQuestionNum(WrongQuestionNumForArray - 1) = rightNo
    Else
        Debug.Print ("Right")
        Score = Score + 1
    End If
    CheckAns = 0
End Function
```

To prevent getting that right answer again, loop till found a wrong answer and return which line it is in at the sheet "Dictionary"

Judge right or wrong by checking the button you click is the right one or not.

Your mistakes will be recorded. When you failed in a question, the number of the right answer will be added to the public array.

The variable WrongQuestionNumForArray determines the border of the public array WrongQuestionNum()

```vba
Public WrongQuestionNumForArray As Integer
Public WrongQuestionNum() As Integer
```

Dynamic array in VBA needs to be "ReDim"ed and preserved every time if you want to make it longer

Programming details and explanation

```
' Fresh the QA UI, setting the caption
Private Sub BtnSelectiveA_Click()
    tempForFunction = CheckAns(rightAnsPos, 1)
    tempForFunction = CheckFinishedAndIncrease(dic
    tempForFunction = RefreshQA(rightNo, totalRows,
    tempForFunction = RefreshQAUI(rightAnsPos, car
End Sub

Private Sub BtnSelectiveB_Click()
    tempForFunction = CheckAns(rightAnsPos, 2)
    tempForFunction = CheckFinishedAndIncrease(dic
    tempForFunction = RefreshQA(rightNo, totalRows,
    tempForFunction = RefreshQAUI(rightAnsPos, car
End Sub

Private Sub BtnSelectiveC_Click()
    tempForFunction = CheckAns(rightAnsPos, 3)
    tempForFunction = CheckFinishedAndIncrease(dic
    tempForFunction = RefreshQA(rightNo, totalRows,
    tempForFunction = RefreshQAUI(rightAnsPos, car
End Sub

Private Sub BtnSelectiveD_Click()
    tempForFunction = CheckAns(rightAnsPos, 4)
    tempForFunction = CheckFinishedAndIncrease(dic
    tempForFunction = RefreshQA(rightNo, totalRows,
    tempForFunction = RefreshQAUI(rightAnsPos, car
End Sub
```

When you click on any of those selective, it will run the function to check the answer, progress, and go to the next question.

I would like to use function even I don't need a value to be returned, just because I got used to it in other languages. Sub procedure works either. No need to set a variable and call the function like me.

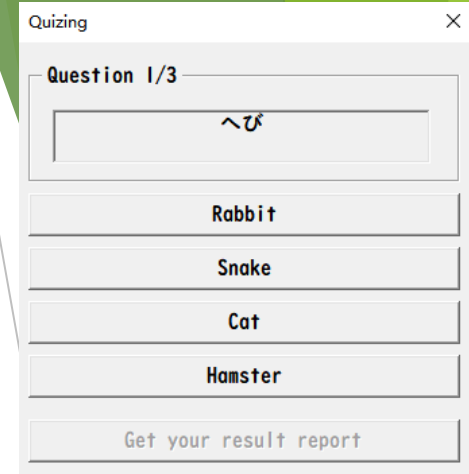Reset variables is very important in this project because there are **TOO MANY PUBLIC VARIABLES!**
Be careful when things get COUPLING!
If I don't clean them, these variables will still affect if user start another round without closing all the forms.

```
Private Sub UserForm_Initialize()
    ' Reset Variables
    isFinished = False
    Score = 0
    WrongQuestionNumForArray = 0
    Erase WrongQuestionNum
    ' Get numbers of questions
    CurrentQuestionNum = 1
    'Next line of code is just for Temporary test
    'NumOfQuestions = 3
    Me.FrmQuestionNum.Caption = "Question " & _
    VBA.Conversion.CStr(CurrentQuestionNum) & "/" & VBA.Conversion.CStr(Nu

    ' Get total number of rows
    Dim dict As Worksheet
    Set dict = Workbooks("FinalProject.xlsm").Worksheets("Dictionary")
    totalRows = dict.Range("A1").End(xlDown).Row
    ' Get Question and Answers for the first time
    tempForFunction = RefreshQA(rightNo, totalRows, dict)
    tempForFunction = RefreshQAUI(rightAnsPos, cardAnswer, wrongAnswer1, w
End Sub
```

Quizing ✕

Question 1/3

へび

Rabbit

Snake

Cat

Hamster

Get your result report

Programming details and explanation   CreateResultReport

```vba
'Create a new worksheet or select exist sheet
Dim ws As Worksheet
Dim Found As Boolean
Found = False
For Each ws In ActiveWorkbook.Worksheets
    If ws.Name = "ScoreReport" Then
        ws.Select
        Found = True
        Exit For
    End If
Next ws
If Found = False Then
    Sheets.Add.Name = "ScoreReport"
    ActiveSheet.Select
End If

' Merging neasasary cells
Range("A1:D1").Select
Selection.Merge
Range("A4:D4") Select
```

```vba
'Putting Score Report value
Dim ScoreFormula As String
ScoreFormula = VBA.Conversion.CStr(Score) & "/" &
Range("A3").NumberFormatLocal = "@"
Range("A3").Value = ScoreFormula
Range("B3").Value = "=" + ScoreFormula
Range("B3").Style = "Percent"

'Criteria
Dim FloatScore As Single
FloatScore = Score / NumOfQuestions
If FloatScore >= 0.9 Then
    Range("C3").Value = "A+"
ElseIf FloatScore >= 0.8 Then
    Range("C3").Value = "A"
ElseIf FloatScore >= 0.7 Then
    Range("C3").Value = "B"
ElseIf FloatScore >= 0.6 Then
    Range("C3").Value = "C"
Else
    Range("C3").Value = "F"
End If

'Add that day's date
Range("D3").Value = Format(Now, "yyyy-mm-dd")
```

. . .Recorded macro part

Programming details and explanation

How many mistake you take, determines how many words you need to review

```
'Add mistaken words to review
For i = 1 To WrongQuestionNumForArray
    Range("A" + VBA.Conversion.CStr(5 + i)).Value = Workbooks("FinalProject.xlsm").Worksheets("Dictionary").Range& _
    ("A" + VBA.Conversion.CStr(WrongQuestionNum(i - 1)))
    Range("A" + VBA.Conversion.CStr(5 + i) + ":" + "B" + VBA.Conversion.CStr(5 + i)).Select
    Selection.Merge
    Range("C" + VBA.Conversion.CStr(5 + i)).Value = Workbooks("FinalProject.xlsm").Worksheets("Dictionary").Range& _
    ("B" + VBA.Conversion.CStr(WrongQuestionNum(i - 1)))
    Range("C" + VBA.Conversion.CStr(5 + i) + ":" + "D" + VBA.Conversion.CStr(5 + i)).Select
    Selection.Merge
Next i
'Back to Cell A1
Range("A1").Select
End Sub
```

In default array,
1st element's index is 0, so the 2nd one will be 1…

Starting from row 5, the first word will be in row 6, the second one will be in 7…

	A	B	C	D
1	Score Report			
2	Score	Percentage	Criteria	Date
3	1/3	33%	F	2024/1/21
4	You should review			
5	Questions		Answers	
6	いぬ		Dog	
7	ハムスター		Hamster	
8				

Dealing with these stuffs needs patience, testing, and a clear mind, better if you have a draft paper.

Summary & Conclusion

- It is a simple quiz card program created by VBA. Users can use it in Excel, practicing, adding custom words, and getting a score report feedback with their mistaken questions to help them do better.

- By making this project, I practiced most of the things I have learned in previous classes, including the basic macro recording, datatypes, statements, the object control, and the GUI, Etc.

- This project not only strengthened my VBA programming skills, but also achieved precious programming good practices especially in dealing with huge amount of code, bugs and how to make my mind clear when facing ambiguous problems.

Thank you