Interactive Multimedia Presentation

Name:

Class:

Teacher:

Microsoft Powerpoint

FEATURING Microsoft Visual Basic for Applications

Developed by Tim Gamble

From your last presentation, you learnt how to add *interactivity* to your presentation. You used action buttons, hyperlinks and hotspots to allow the user to control the slide sequence, navigate through the program and participate in a game. Incorporating interactivity, as you've probably realised, makes presentations a lot more interesting and enjoyable.

For this next piece of coursework, you are once again going to create an Interactive Multimedia Presentation. This time however, you are going to introduce interactivity through the use of a powerful programming language – *VBA (Visual Basic for Applications)*

Visual Basic for Applications is a 'tool' that can be used as an alternative to the 'tools' available to you through the normal menus. MS PowerPoint provides you with a range of tools (e.g. action buttons) already, but what if you want to do something different and it isn't there in the menu-tools options? This is where **VBA** comes in. It will let you go further and do lots of things that will make your presentation all the more unique! It then becomes *Advanced* Interactive Multimedia Presentation.

At this level, you do not need to learn **VBA**; but you will be given examples of program code that you can easily adapt for your own presentation.

…whose purpose is to be used as an **Aid for Learning**

For a **topic** of your choice…

…and **age group** of your choice.
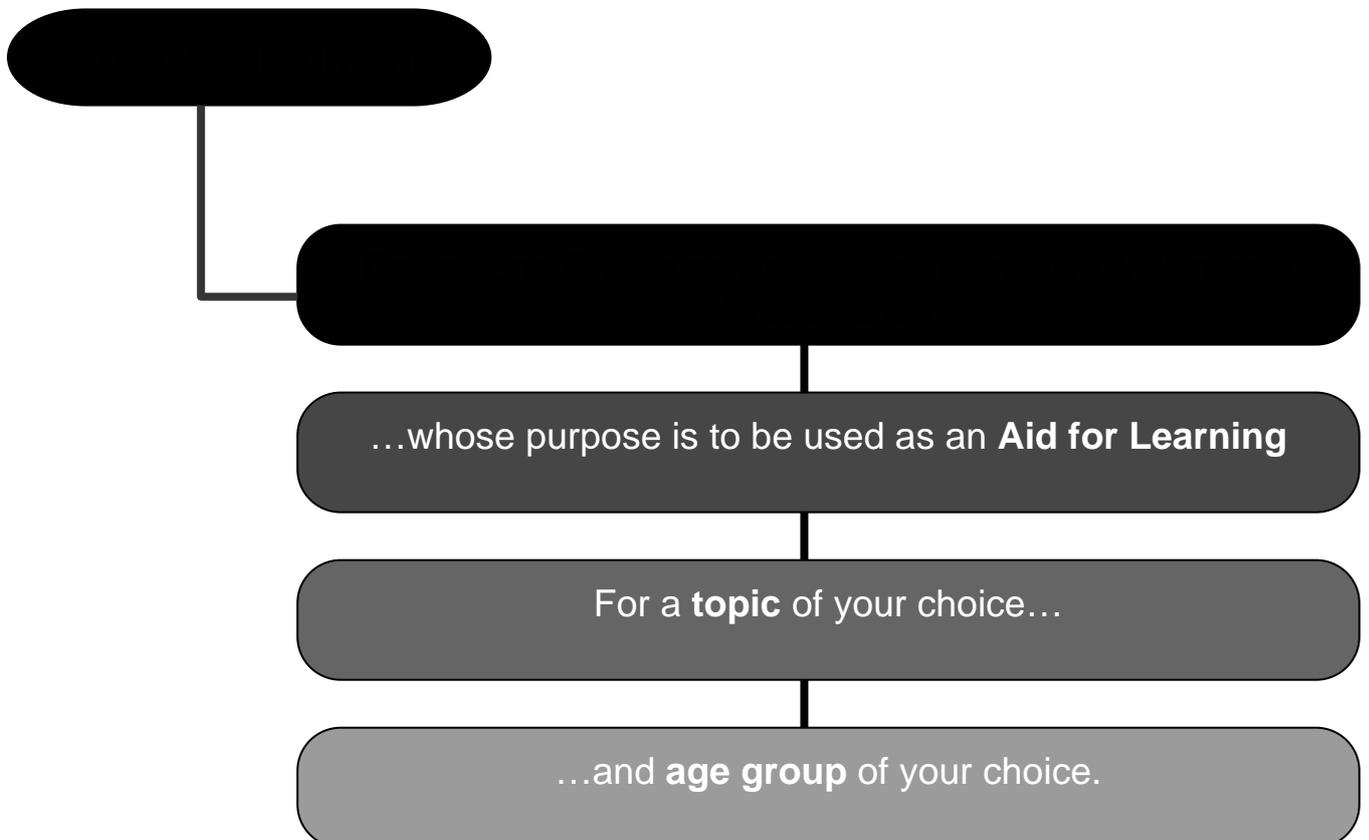
To help you get started, look at the following example.

Topic Chosen: Counting – teaching a child how to count to 10.

Age Group / Audience: Young infants 5 yrs (P1/P2)
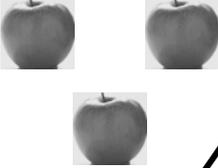
Slide 1

Slide 2

Child enters answer in here.

Afterwards, the child clicks here to see if answer is right.

*Slide 2* (after 'Check answer' is clicked)
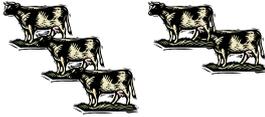
Slide 3

If the pupil got the answer wrong, then obviously the Message Box would inform them and they would have to try again. Otherwise, a correct answer would mean the next slide in the sequence would appear.

This idea could be used for lots of topics! Just think, you could teach someone letters of the alphabet, days of the week, basic French or even identifying colours. The possibilities are endless and the VBA programming code is very similar for each.

You are now going to create your own copy of the Advanced Interactive Multimedia Presentation shown already.   Once you've gone through this, you can then begin to form your own ideas for your own assignment.

- Open up Microsoft PowerPoint and select **BLANK Presentation** from the Slide Layout options.

- Now insert a pale yellow background colour.

- Copy the text displayed on Slide 2, shown on the previous page, using ordinary textboxes. <u>Don't do the Start button yet</u>.

- Go to **View** Menu, select **Toolbars**, and then select **Visual Basic** option.

- <u>A toolbar like shown will appear. Now click on the **Toolbox** icon.</u>

- Click on the Command Button option
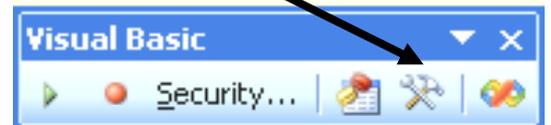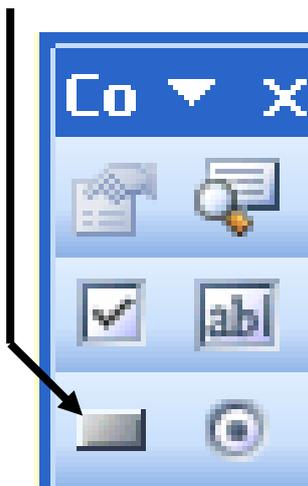
- Draw the Command Button near the bottom of the screen, like slide 2 shown on the previous page.

- Now **Right-Click** the Command Button and select **Properties**.

- A list of **Properties** (characteristics) for the Command Button appears. Find the Property called **Caption** and change it from "*Command Button 1*" to "*Check my answer*".

- The Command Button should now look like this:
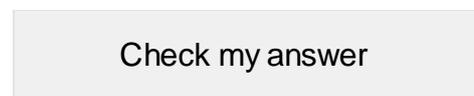
- Now go back to the Toolbox and select the **Textbox** icon. Draw a textbox above the Command Button.

Textbox
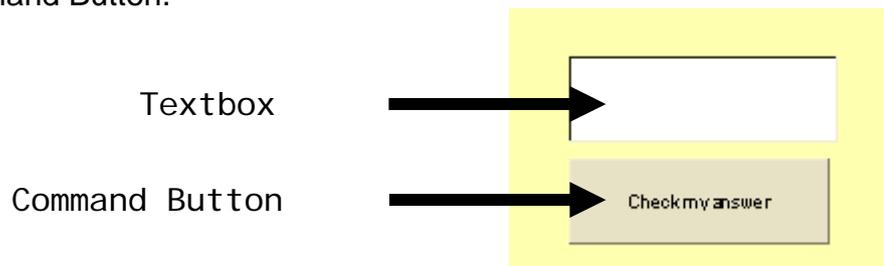
Command Button

- Now **press F5** on your keyboard to run the presentation. Try entering text into the textbox, and click the Command Button. Anything happen?

- Of course not! That's why we now have to use VBA to add program code to the Command Button so that when clicked, the program will check the user's answer to see if it's right or wrong.

- **Double-Click** the Command Button. You shall now find yourself in the *Microsoft Visual Basic Programming environment.* It is here where all coding is entered.

- Your cursor should be flashing between two lines of code, shown here. **It is in here that you shall type in the following Visual Basic code.**

```
Private Sub CommandButton1_Click()

End Sub
```

- Type in the following code **shown in BOLD**, exactly as shown:

```
Private Sub CommandButton1_Click()

If TextBox1.Value = 3 Then
     MsgBox "Correct. Well done!"
     SlideShowWindows(1).View.Next
Else
     MsgBox "Wrong answer. Try again."
End If

End Sub
```

- The code or **Syntax** is interesting if you try to understand it. The computer reads it line by line from the top and carries out or **executes** each instruction. Do YOU understand what each line is telling the computer to do?

- Now click on the **Save** icon near the top. Save your presentation in your own user space as " ***Learning to Count*** ".

- **Exit** out of the Microsoft Visual Basic Environment. You should now return to PowerPoint design mode.

- **Insert a second slide** into your PowerPoint Presentation and give it the title *Question 2*. Don't worry about inserting anything else at this stage.

███

- **Press F5.**   In the textbox insert a wrong answer first e.g. 10. Then click on "Check my answer" to see what happens. Afterwards, try it with the correct answer. Has it worked? It should progress onto slide 2 (Question 2) if you have been successful.

If your program isn't working, check the Syntax (code)

- You have just created an Advanced Interactive Multimedia presentation! Now you're going to change the question a little.

- Go back to PowerPoint and **insert an extra apple**. You should now have 4.

- Now, **double-click the Command Button** so you return to the Microsoft Visual Basic Environment.

- Since the answer to the question has changed, the program code also needs to change. **Can you guess or identify which parts ?**

- Easy! You just have to change the part…

  If TextBox1.Value $= 3$ Then   to   If TextBox1.Value $= 4$ Then
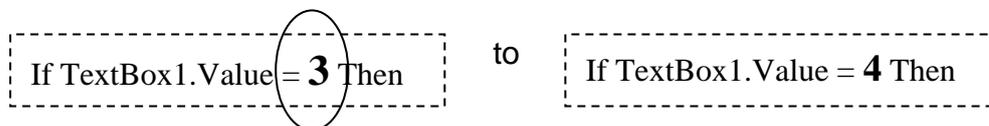
- Click on **Save**, then **Exit** out of Microsoft Visual Basic Environment and **Press F5** to run the presentation. **Has it worked ?**

```
You have just seen how easy it is to change bits of program code
and make it adaptable for different questions.  You shall do
this a lot when creating your own advanced presentation.
```

- On slide 2, design a new question (e.g. how many cows are there?) with corresponding pictures, with a **textbox** and also with a **command button** (C*aption = "Check my answer"*)

- When you have finished with the layout, **double-click the new Command Button** and **copy the VB code** (same code as before) into the same place (between `Private Sub`... and `End Sub`)

- Remember this time, you need to change the answer value from 3 to 5 (because there are 5 cows in my example)

```
If textbox1.value = 5 Then
    Same
    Same...
```

- Remember to click **Save** and **EXIT out of VB,** and then **Run (F5)** the presentation to see if it works.

Add a third question to your presentation to build up your confidence.

Or   Option Buttons

So far, you have seen how a textbox and command button can be used to add interactivity to a slide. There are many other tools or **controls** that can be used too. Another useful control available is the option button.

How many dogs can you see?

☐ 5

☐ 6

◉ 4
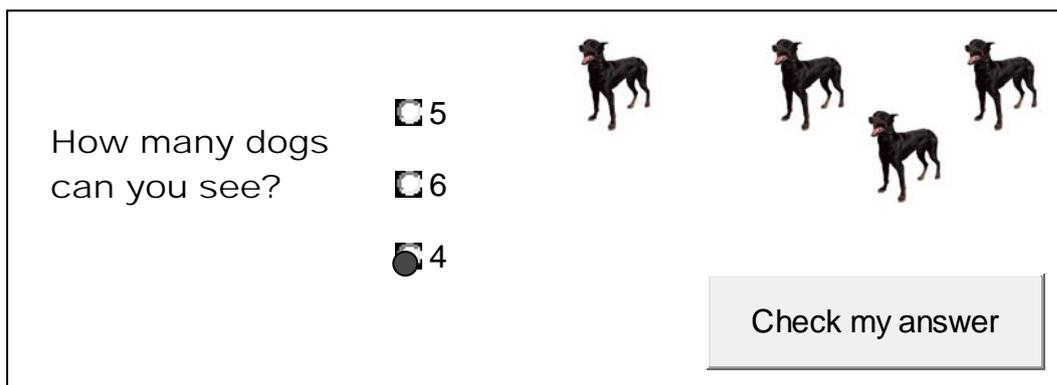
Check my answer

Here is an example of how Option buttons might be used. **The user is given a choice of answers this time, and they can only select one answer.** After they select an option, they then click on *Check my answer* to see if they're right.

- Insert a new slide and copy the question shown in the previous diagram. ('How many dogs can you see?')

- **Insert 3 Option buttons** from the Control Toolbox.

- **Right-click** the first option button and select **Properties**.

- Change the **Caption** property of the first option button to '**4**'.

- **Change all the Captions** to read '**4**', '**5**' and '**6**.'

- **Press F5** to run the presentation and see the option buttons working.

- Now it is time to add the Visual Basic code to get it all working. Once again, **double-click** the **command button** which will take you into the Microsoft Visual Basic Editor Environment.

- Add the following code **shown in BOLD** exactly as you see it:

```
Private Sub CommandButton1_Click()

If OptionButton1.Value = -1 Then
    MsgBox "That is correct. Well done!"
    SlideShowWindows(1).View.Next
Else
    MsgBox "Sorry, that is not right. Try again"
End If

End Sub
```

- Do you recognise any similarities? It is very similar to the coding shown before, only this time rather than using `textbox1.value` ... we use `optionbutton1.value` ...

- Also, try adding a different message to the message boxes. **Just make sure that whatever you want to say is inside the "quotation marks."**

- See if it works by clicking on **Save**, then **Exit** from the VB Environment and **Press F5** once you have returned to PowerPoint.

Working? If it is, design another question using Option Buttons.

Once you have gained confidence with these techniques, you are now ready to begin with your own

Look back at the objectives for this assignment shown on page 2.

My topic is:

(What is the purpose of your presentation?)

My audience/ age group is: _____

Slide Planner
Slide 1 *(Welcome slide)*

Slide 2

Slide 3

Slide 4

Slide 5

Slide 6

Slide 7

Slide 8

(Insert extra loose pages if you need) 11

What colour is the triangle?
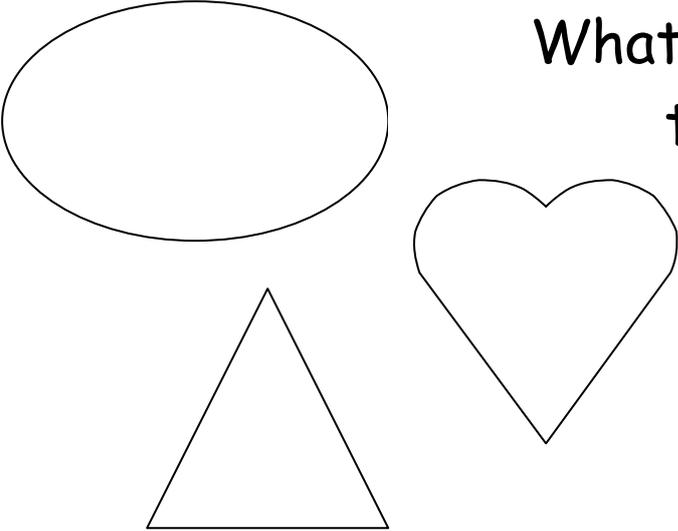
- ◉ Red
- ☐ Blue
- ☐ Green

Check my answer

This could help primary pupils identify colours
OR
identify shapes
(e.g. What is the name of the shape shown?)
a) Square b) Rectangle c) Rhombus

What is the name of the shape below?
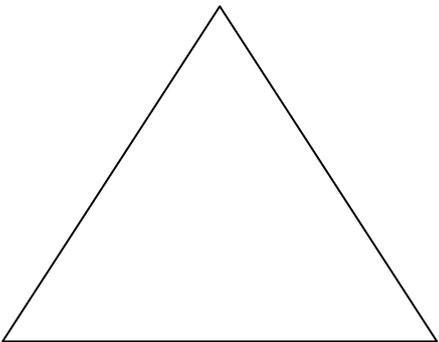
- ☐ Rhombus
- ☐ Traingle
- ☐ Square

Check my answer

Developed by Tim Gamble

Type the name of the animal
in French in the box provided

Chien

Check my answer

Do you know which letter will
complete the following…?

a P P L E

Check my answer

Pupil types in letter here
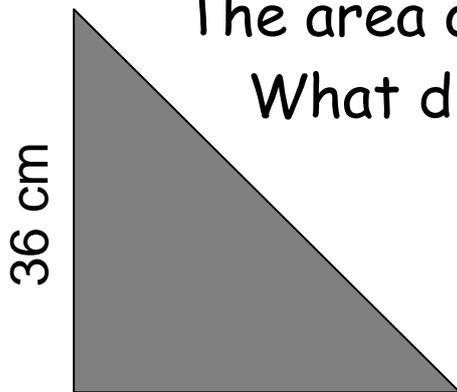
The area of the triangle is 18cm².
What dimension is the base?

36 cm

Check my answer

Biology?

Label all parts of the human heart

Check my answers

The possibilities are endless...

14