

COMP1022Q
Introduction to Computing with Excel VBA

Changing the Appearance of Cells

David Rossiter and Gibson Lam

Outcomes

- After completing this presentation, you are expected to be able to:
 1. Change the background colour of cells using VBA
 2. Change the cell font using VBA
 3. Add borders using VBA

Changing the Appearance of Cells

- Basically, anything you can do by clicking in Excel, you can also do in VBA code
- That includes changing the appearance of cells
- We will do this in some of our lab projects
- Here we look at changing these things:
 - Changing the background colour
 - Changing the font
 - Changing the border

Changing Background Colour

- We will look at these two ways to change the background colour of a cell
 1. Using a simple number
 2. Using colour names

Using a Simple Number

- You can use a simple number to refer to a colour
- In VBA, this number is called the *ColorIndex*
- A ColorIndex number has a range of 1 to 56, so you have 56 colours to choose from
- Here is an example of changing the colour of cell A1 to red
 - red has a ColorIndex of 3:



```
Range("A1").Interior.ColorIndex = 3
```

Showing All ColorIndex Colours

- In the following example, the ColorIndex numbers and their associated colours are displayed in 56 rows, starting at row 5

```
For ColorNumber = 1 To 56
    ' Show the number in column A
    Cells(ColorNumber + 4, 1).Value = ColorNumber

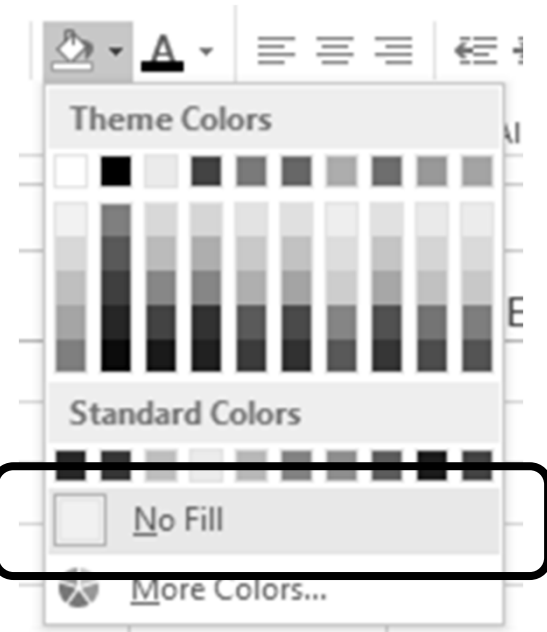
    ' Show the cell colour in column B
    Cells(ColorNumber + 4, 2).Interior.ColorIndex _
        = ColorNumber
Next ColorNumber
```

The Result of the Code

	A	B		A	B		A	B
4	Color Index	Colour	23	19		42	38	
5	1		24	20		43	39	
6	2		25	21		44	40	
7	3		26	22		45	41	
8	4		27	23		46	42	
9	5		28	24		47	43	
10	6		29	25		48	44	
11	7		30	26		49	45	
12	8		31	27		50	46	
13	9		32	28		51	47	
14	10		33	29		52	48	
15	11		34	30		53	49	
16	12		35	31		54	50	
17	13		36	32		55	51	
18	14		37	33		56	52	
19	15		38	34		57	53	
20	16		39	35		58	54	
21	17		40	36		59	55	
22	18		41	37		60	56	

Setting ColorIndex to 0

- If you want to, you can set the ColorIndex of a cell to 0
- This is the same as setting the colour to *No Fill* in the Home ribbon
- Then the cell will have the default (white) background



```
Range("A1").Interior.ColorIndex = 0
```

- This does the same thing:

```
Range("A1").Interior.ColorIndex = xlNone
```


Using Colour Names in VBA

- Sometimes you can use a special VBA word such as 'vbWhite' instead of the colour index number
- There are only a few of those colour names: vbWhite, vbBlack, vbRed, vbGreen, vbBlue, vbYellow, vbCyan and vbMagenta

- For example, this line of code:

```
Range("A1").Interior.ColorIndex = 4
```

does the same thing as this line of code:

```
Range("A1").Interior.Color = vbGreen
```

How to Refer to Every Cell

- You already know you can use `Cells (Row, Column)` to do something with any one cell
- If you don't provide the row and column, `Cells` means all cells i.e. the entire worksheet
- For example, the following code sets the background colour of the entire worksheet, i.e. all the cells, to green:

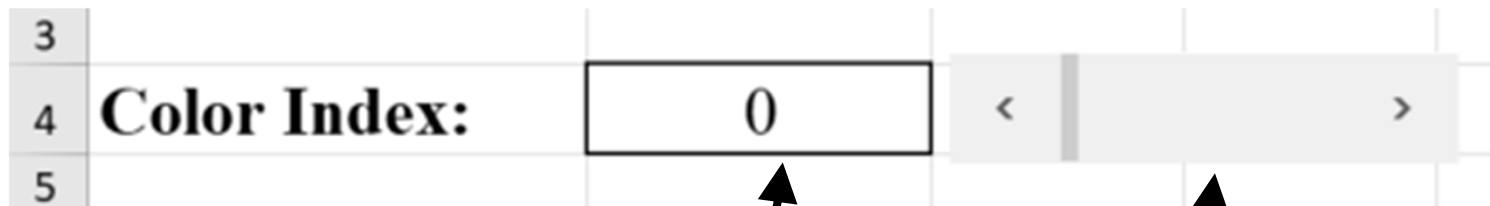
```
Cells.Interior.ColorIndex = 4 ' Green
```



This means the entire worksheet

An Example Which Changes All Cells

- In this example we use a *scrollbar* to change the colour of the entire worksheet



Cell B4 shows the colour number

A scrollbar

- When the scrollbar value is changed the value is automatically put in cell B4, and a macro is executed:

```
Sub HandleScrollbar()  
    Cells.Interior.ColorIndex =  
        Range("B4").Value  
End Sub
```

We will explain more about scrollbar and other controls later in the course, if we have time

Some Examples of Using the Program

3	
4	Color Index: <input type="text" value="5"/> <input type="text" value=" < >"/>
5	

3	
4	Color Index: <input type="text" value="56"/> <input type="text" value=" < >"/>
5	

3	
4	Color Index: <input type="text" value="36"/> <input type="text" value=" < >"/>
5	

3	
4	Color Index: <input type="text" value="4"/> <input type="text" value=" < >"/>
5	

Changing the Font

- The font of a cell can be changed using the `Font` property
- Here are some common things you can do with the font:

```
' Change the font to the 'Broadway' font and  
' its size to 16
```

```
Range("A1").Font.Name = "Broadway"
```

```
Range("A1").Font.Size = 16
```

```
' Change the font to bold
```

```
Range("A1").Font.Bold = True
```



	A
1	Hello!

```
' Change the font colour to red
```

```
Range("A1").Font.Color = vbRed
```

```
' Or, change to red using ColorIndex :
```

```
Range("A1").Font.ColorIndex = 3 ' ColorIndex
```

Changing All Borders

- Sometimes, you may want to change the cell borders
- For example, you can set all borders of cell B2 to thick red borders using this code:

	A	B	C
1			
2			
3			

```
Range ("B2") .Borders.Color = vbRed
```

```
Range ("B2") .Borders.Weight = xlThick
```