COMP1022Q
Introduction to Computing with Excel VBA

Making Random Numbers

Gibson Lam and David Rossiter

Outcomes

- After completing this presentation, you are expected to be able to:
  1. Use VBA to generate random numbers in a particular range
  2. Use VBA to change the sequence of random numbers so it is not always the same

Random Numbers in VBA

- Random numbers can be generated in VBA using the `Rnd` function
- The `Rnd` function generates a real number smaller than 1 and bigger than or equal to 0, i.e. [0 , 1)
- For example, the following line of code puts a random number in a variable `MyNumber`

```
MyNumber = Rnd()
```

These brackets are optional for `Rnd` – you can ignore them, if you want to

An Example of Making Random Numbers

- Let's make some code which creates a simple math addition game using random numbers
- In the code, two random integers in the range 1 to 100 are generated
- The user is then asked what the sum of those two numbers is, like this:

Generating Random Integers 1/2

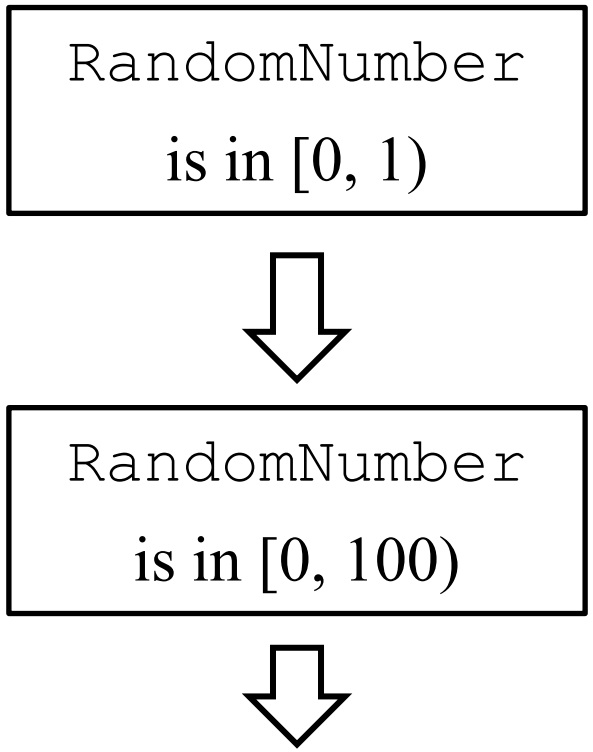
- To make the code, we need two random numbers
- To generate one random integer in the range of 1 to 100 you do these steps:

1. Generate a number between 0 to 0.99999 using the `Rnd` function

```
RandomNumber = Rnd()
```

2. Multiply the generated number by 100

```
RandomNumber = Rnd() * 100
```



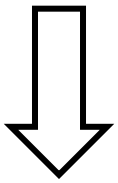
Continued on the next slide…

Generating Random Integers 2/2

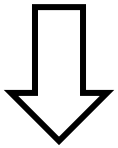
Continued from the previous slide…

3. Dump the decimal place using the `Int` function
   (it doesn't do any rounding)

```
RandomNumber = Int(Rnd() * 100)
```



`RandomNumber` is in [0, 99]



4. Add 1 to the number

`RandomNumber` is in [1, 100]

```
RandomNumber = Int(Rnd() * 100) + 1
```

Storing Things

- For the following game we will use variables that store text
- And we will also use variables that store integer numbers
- For example:

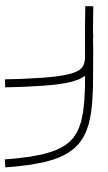
```
Dim MyFavouriteText As String 'stores text
Dim MyFavouriteNumber As Integer 'stores an integer
```

- Here's some examples of using the variables:

```
MyFavouriteText = "you are a silly sausage"
MyFavouriteNumber = 8888
```

Math Game 1/2

- Here are two examples of how to create two variables in one line of VBA code



```
Dim Number1 As Integer, Number2 As Integer
Dim Answer As String, Guess As String

' Create the first number in the range 1 to 100
Number1 = Int(Rnd() * 100) + 1

' Create the second number in the range 1 to 100
Number2 = Int(Rnd() * 100) + 1

' Calculate the answer and store it as string
Answer = Number1 + Number2
```

Continued on the next slide…

Math Game 2/2

Continued from the previous slide…

```
' Execute the loop at least once
Do
    ' Ask the question
    Guess = InputBox("What is " & Number1 _
                     & " + " & Number2 & "?")

' Check the answer at the end of the loop
Loop While Answer <> Guess

MsgBox "Excellent, you have got the " _
       & "correct answer!"
```

Keep doing the loop while this is true

Randomness of the Rnd Function

- Unfortunately, you will find that every time you run your code you will get the same series of random numbers!
  - For example,
    - The first time your program asks a random math question:

1st time you run it:

What is 75 + 71?

    - Later you run the program the second time it will ask the same question again!

2nd time you run it:

What is 75 + 71?

- That means any game which uses the random numbers will be the same every time you play it!
- To change this, you need to use `Randomize`

Ensuring the Sequence is Different Each Time

. . . The code for creating the variables is the same as before . . .

`Randomize` ← You need to use Randomize to ensure that the series of random numbers is different each time

. . . The remaining code is the same as before . . .
' Create the first number, in the range 1 to 100
. . .
' Create the second number, in the range 1 to 100
. . .
' Calculate the answer and store it
. . . and so on . . .

Because *Randomize* has been used, the following sequence of random numbers will be different each time