

# Virtual Dice in Excel



Most computer games are based on the idea that events can be modeled with probabilities. For example, in a football simulation, we can identify from a running backs statistics how many yards they are likely to gain on each carry (average) and how each carry varies from our expected gain (variance).

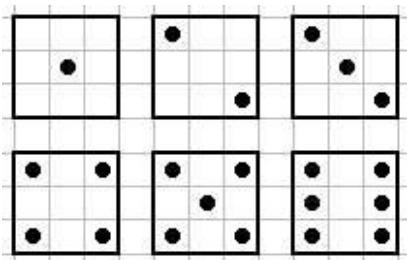
Today's computer game programming is complex. We can apply the concept of modeling events easily in **MS Excel**.

1. Open up a new workbook – we will only need one sheet, so delete any other sheets that you get by default (**right-click** on each tab and select **DELETE**).
2. Rename Sheet 1 as **Virtual Dice** (**right-click** and select **RENAME**).
3. Next, we will use MS Excel's random integer number function (RANDBETWEEN) to create a value for each roll of our virtual dice. Since there are 6 sides on each dice (cube), we need to randomly generate 6 integers. Here's the function:

**=RANDBETWEEN(1,6)**

Carefully enter the above function in the following cell: **A3**. Then enter the same random integer function cell: **F3**,

4. The key to making this project work without using some complex **Visual Basic for Applications** (Microsoft programming language to customize MS Office) is to realize we can use boxes of 9 MS Excel cells to represent each dice. This means we can use a simple **IF function** to assign a dot in each appropriate cell based on the random value of each dice as illustrated below:



5. The dots we will use for the dice will be **character 108** from the font **WINGDING**. We can center each dot in each cell. Then we can quickly create the face of each dice using the rectangular shape that MS Excel cells default to and then resize the width of each cell to match each cell's height – when we are done, we will have square dice.

6. The **top left cell** of each dice needs to show a dot if our random integer function generates 2, 3, 4, 5, or 6. Therefore, the IF statement in cell **B3** is:

**=IF(A3>1,CHAR(108),"" )**

7. The IF statement in cell **G3** is:

**=IF(F3>1,CHAR(108),"" )**

8. The top middle cell of each dice matrix is always blank. There is never a dot in that part of dice and we will **leave C3 and H3 empty**.

9. The top right cell of each dice needs to show a dot if the random integer function generates 4, 5, and 6. Therefore, the IF statement cell **D3** is:

**=IF(A3>3,CHAR(108),"" )**

10. The IF statement in cell **I3** is

**=IF(F3>3,CHAR(108),"" )**

11. In the 2nd row of each dice, the left-most cell only displays a dot if our random integer function generates a 6 (greater than 5 since we are working with integers). the IF statement in cell **B4** is:

**=IF(A3>5,CHAR(108),"" )**

12. The IF statement in cell **G4** is:

**=IF(F3>5,CHAR(108),"" )**

13. In the 2nd row of each dice, the center cell is turned on for 1, 3, or 5. the IF statement in cell **C4** is:

**=IF(ISODD(A3),CHAR(108),"" )**

14. The IF statement in cell **H4** is:

**=IF(ISODD(F3),CHAR(108),"" )**

15. The right column in the second row of each dice only displays a dot if the random integer function generates a 6 (greater than 5 since we are working with integers). The IF statement in cell **D4** is (same as B4):

**=IF(A3>5,CHAR(108),"" )**

16. The IF statement in cell **I4** is:

**=IF(F3>5,CHAR(108),""')**

17. The bottom right cell is in each dice matrix only displays a dot when our random integer function generates 4, 5, and 6. The IF statement in cell **B5** is:

**=IF(A3>3,CHAR(108),""')**

18. The IF statement in cell **G5** is:

**=IF(F3>3,CHAR(108),""')**

19. The bottom middle cell of each dice matrix is always blank. There is never a dot in that part of dice and we will **leave C5 and H5 empty**.

20. The bottom right cell of each dice matrix displays a dot when the random integer function generates anything other than a 1. The IF statement in **D5** is (same as B3):

**=IF(A3>1,CHAR(108),""')**

21. The IF statement in cell **I5** is

**=IF(F3>1,CHAR(108),""')**

22. There! We have programmed a pair of virtual dice! All that's left to do is tidy-up a bit and make the game look better.

23. In cell **A1**, enter **Virtual Dice, by YOUR NAME**. Apply a dark font color (**FONT COLOR button** on the **HOME ribbon**).

24. A **range** is a group or block of cells in a worksheet. We identify the range by the top upper cell on the left and then the bottom cell on the right of the block of cells. (**NOTE: Column ranges or row ranges are identified by the top/bottom or left/right cells respectively**). Highlight **the cell range A1:W32** and apply a light-colored fill to those cells (**FILL COLOR button** on the **HOME ribbon**).

25. Next, we will make our random integer function "invisible" – we can do this by **highlighting columns A and F and coloring the font the same color we have filled the cells with** (**FONT COLOR button** on the **HOME ribbon**).

26. Enter the heading **Total** in cell **K2** and then **center** it in that cell. Use the **BORDER button** on the **HOME ribbon** to put a **BOTTOM BORDER** on cell **K2**.

27. We will display the value of the pair of dice in cell **K4**. We can add the value on dice 1 to the value of dice 2 with the following formula in cell **K4**:

**=A3+F3**

28. **Highlight the cell range K2:K5** and place a **THICK BOX BORDER** around it (**BORDER button** on the **HOME ribbon**). Then, while the cell range K2:K5 is still selected, **add a contrasting light color fill** so that this box stands out against your background fill color (**FILL button** on the **HOME ribbon**).

29. Now we'll tweak our virtual dice so that they are white, square, and stand out against the background fill. Highlight the range B3:D5. Color them **WHITE** with the **FILL button** (on the **HOME ribbon**).

30. While **B3:D5** are still highlighted, add a **THICK BOX BORDER** around it (**BORDER button** on the **HOME ribbon**).

31. Repeat this for the second dice, **range G3:I5**.

32. In cell **A7**, using the same color font you used in cell A1 and using **BOLD**, add the text:

**To recalculate the dice, hit SHIFT+F9**

33. Finally, we will resize the width of each column that makes up each dice to match the height of each row in each dice. Verify **that the row height is 21 pixels** by clicking on the bottom edge of any row in the far right heading of the row numbers. Mouse over the right column line for each of the three columns for each dice (B,C, D for dice 1; G, H, I for dice 2). Using your mouse, resize each column width to 21 pixels.

34. **Viola!** You have just programmed your first game in Excel! Be sure to **test it by hitting SHIFT+F9** several times to verify that it works properly and that your **TOTAL** figure for the 2 dice is accurate.