

Topic — Software



How to generate random numbers with no duplicates in Excel



Sign In



Go Premium

Top Products

AI

More

Table of Contents



[About random functions](#)

We may earn from vendors via affiliate links or sponsorships. This might affect product placement on our site, but not the content of our reviews. See our [Terms of Use](#) for details.

Microsoft Excel can do a lot of cool things. If you need it to generate random numbers, Susan Harkins can show you how.

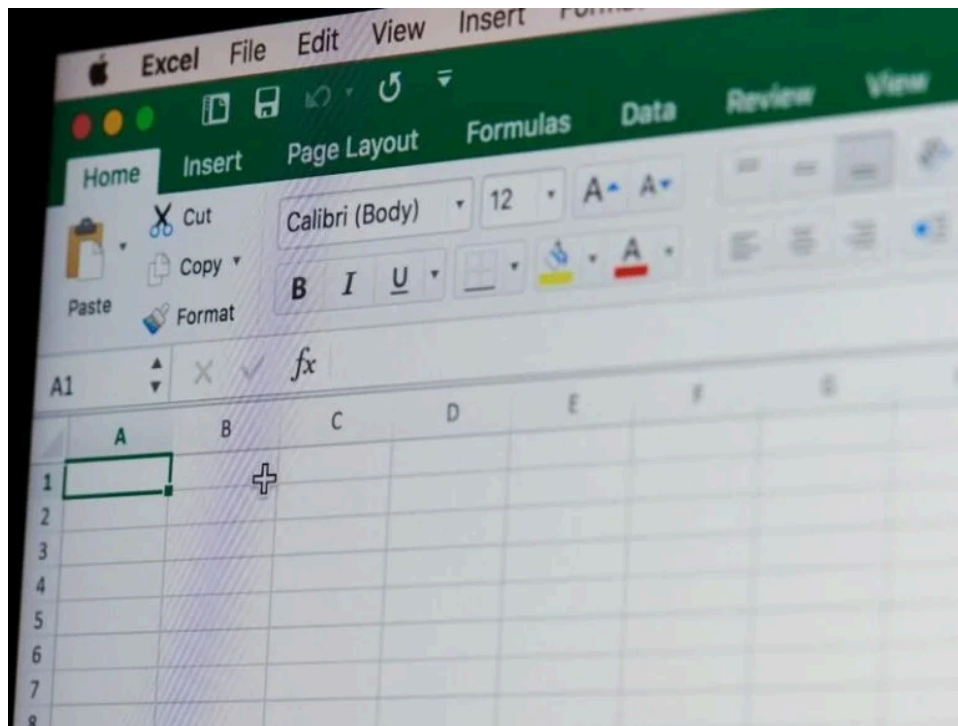


Image: PixieMe/Shutterstock

Over the last few months, I've written about several of Excel's new dynamic array functions. They're powerful and easy to implement—relieving us of some very tiresome expression-building that was necessary before. In this article, we'll combine a few of these new dynamic array functions to generate an array of unique random values. It's easy to generate random values; it's just as easy to limit the bottom and top boundaries for a set of random values. What isn't so easy, as you'll see, is generating an array of unique random values. Fortunately, using dynamic array functions, you only have to jump through a few hoops.

SEE: [83 Excel tips every user should master \(TechRepublic\)](#)

I'm using [Microsoft 365](#) on a [Windows 10](#) 64-bit system. Dynamic array functions are supported by 365, Excel for the Web, Excel 2021, Excel for iPad and iPhone, Excel for Android tablets and phones. There's no demonstration file; you won't need one.

About random functions

Excel has three random value functions: `RAND()`, `RANDBETWEEN()`, and `RANDARRAY()`.

`RAND()` generates random values between 0 and 1, so random decimal values.

`RANDBETWEEN()` lets you specify the bottom and top values. For instance, you might use `RANDBETWEEN(10,1000)`

to generate random integers between 10 and 1,000.

Finally, RANDARRAY() returns an array of random values using the following syntax:

```
RANDARRAY([rows], [columns], [min], [max], [integer])
```

Notice that all of the arguments are optional. These arguments allow you to specify the number of rows and columns to fill, the minimum and maximum values to return, and whether to return integers or decimals. The integer argument is false by default, which returns decimal values. To return integers, you must specify TRUE. In addition, min and max are inclusive values. For example, if these two arguments are 1 and 5, respectively, the function will return 1, 2, 3, 4 and 5, not just 2, 3 and 4.

When considering arguments, keep the following rules in mind:

- RANDARRAY() performs like RAND(), returning a value between 0 and 1 if you omit all arguments.
- RANDARRAY() performs like RAND(), returning a value between 0 and 1 if you omit rows and columns.
- RANDARRAY() will return 0 or 1 respectively, if you omit min and max.
- The min argument must be less than max.
- RANDARRAY() will include decimal values if you don't explicitly set integer to TRUE.
- RANDARRAY() will create the appropriate-sized range to complete the calculations. This range is called the *spill range*. If there aren't adequate cells in this range, RANDARRAY() returns a spill error. To learn more about Excel's spill range, you can read [How to use the spill range in Excel](#)
- RANDARRAY() will update results and resize when the source data changes if the source data is in a Table object.
- Currently, if the data source is in another workbook, both workbooks must be open for RANDARRAY() to function properly.
- To generate a random array of non-duplicate values, we'll use three dynamic array functions: RANDARRAY(), UNIQUE(), and SEQUENCE().

How to generate random values without duplicates, sort of

RANDARRAY() alone has the potential to return duplicate values, as do all the random functions. If you have the time, there's a simple way to achieve random values with no duplicates. Convert the values in the spill range to static values (also replacing the function) and then remove any duplicates using the Remove Duplicates tool on the Data tab. It is easy, but it isn't a reasonable way to work unless what you're doing is a one-time pass.

You can't force RANDARRAY() to return only unique values, but you can combine it with UNIQUE() to get the desired results. For instance, the following function, shown in **Figure A**, will return a single column of five (seemingly) unique values between 1 and 20:

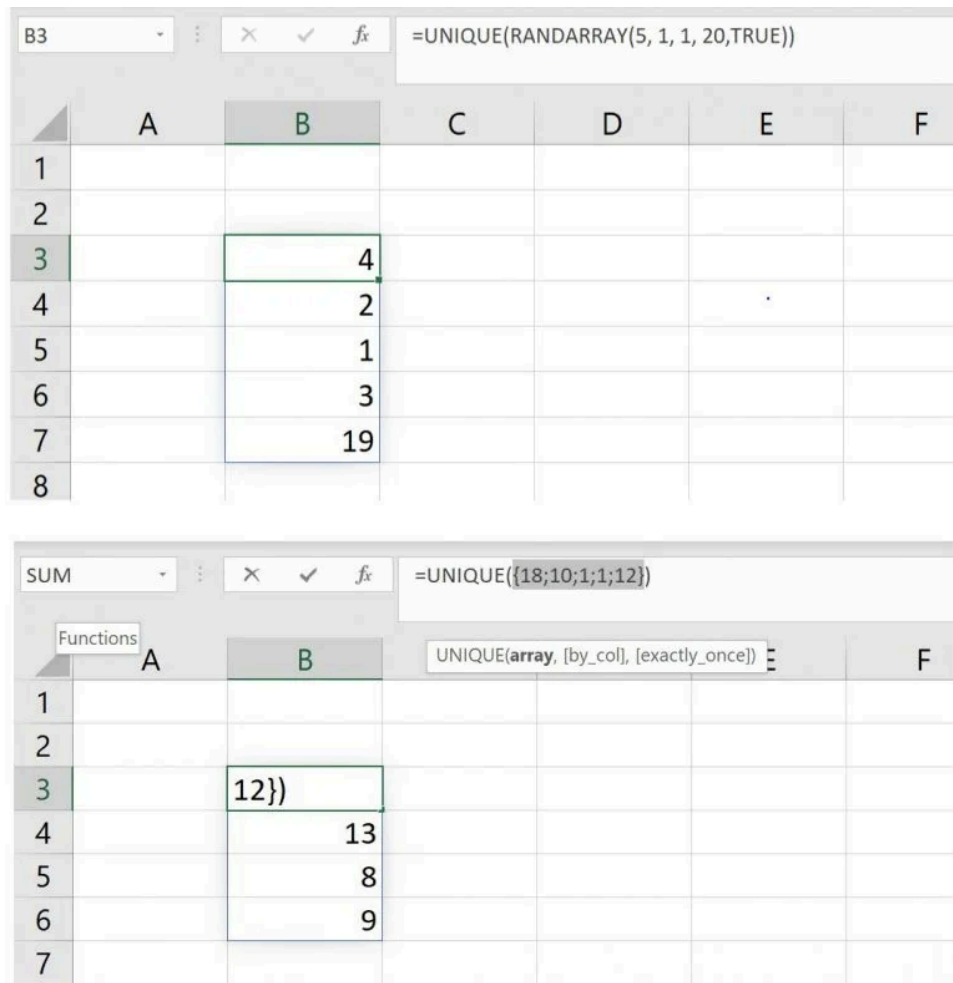
```
=UNIQUE(RANDARRAY(5, 1, 1, 20,TRUE))
```

Two things might happen though: 1.) You see a list of random unique values that is less than you specified; 2.) You see a *spill error* (to the right in **Figure A**).

When RANDARRAY() repeats enough values that the expression can't return five unique values from RANDARRAY()'s pool of random values, it will return what it can—which might be short a value or two. The chances of having so many repeating values in that pool of 20 (the min and max values of 1 and 20, respectively) that you can't return a list of five is small, but not impossible. Press F9 and watch the expression update; eventually you will see an array that is less than five digits.

The shot on the right (**Figure A**) shows the expression in the formula bar where I've manually calculated RANDARRAY() by selecting *only* that function and pressing F9. As you can see, the array *does* repeat a value, the value 1. Hence, the expression can return only four unique values.

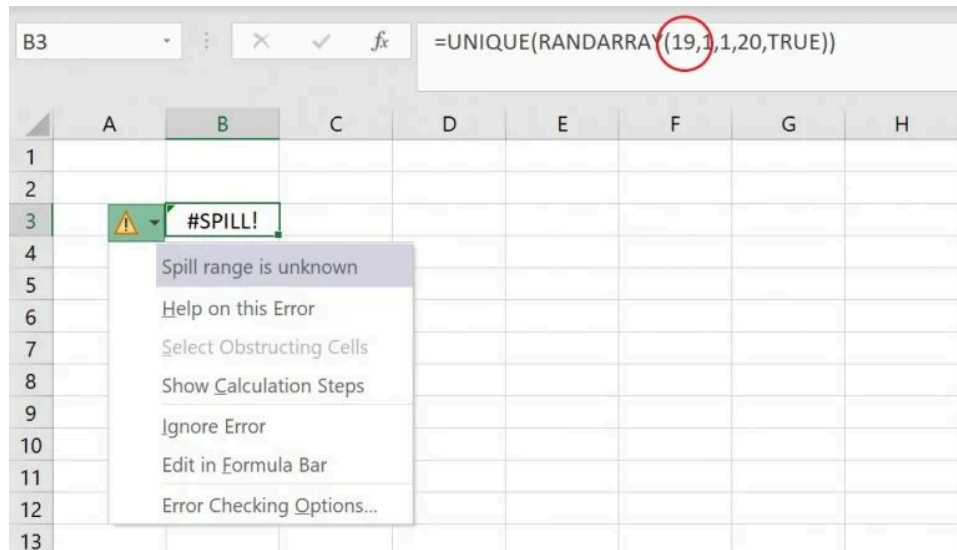
Figure A



You might expect to see five unique values.

Figure B shows a spill error, *spill range unknown*. I've changed the rows argument from 5 to 19 forcing that to happen more often, but if you press F9 often enough, the rows argument of 5 will eventually return a spill error. It has to do with the volatile nature of these functions—sometimes one calculates faster than the other. A *volatile* function recalculates every time a change is made.

Figure B



Spill errors are possible with this simple expression.

One way to ensure that the expression doesn't return fewer values than specified is to create a huge pool of random values. Perhaps the easiest way is to change the rows argument from a specific value to a short expression: n^2 . For example,

```
=UNIQUE(RANDARRAY(5^2, 1, 1, 20, TRUE))
```

The row expression, 5^2 , will return 25 random values, but now the expression returns *up to* 25 values (rows) and you're still susceptible to an occasional spill range error. However, it's unlikely to return fewer than five rows.

If you can live with all these behaviors, you could stop here. Chances are, though, that you'll want something more stable.

How to generate random values without duplicates

In the last section, you learned a lot about RANDARRAY() and UNIQUE() and how they work together, even though the results aren't stable. Your needs will determine whether you need to take this problem to the next level.

Right now, we have an expression that doesn't always return the right number of values and has the potential to return a spill error. To avoid these two problems, you need a more complex expression in the form

```
=INDEX(UNIQUE(RANDARRAY(rows^2, columns, min, max, TRUE)), SEQUENCE(rows))
```

It looks horrible, but don't worry. Once you understand how it works, it will make good sense, and you'll have little trouble applying it to your own work.

RANDARRAY() generates an array of random values based on min and max. The rows argument to the power of 2 generates a large pool of random values—much larger than just the rows value itself. UNIQUE() removes all duplicate values and returns an array of unique values. We got this far in the last section.

The difference this time, is that UNIQUE() doesn't return the array to the sheet. Once UNIQUE() removes duplicates, INDEX() uses the number of rows expressed in SEQUENCE() to determine how many unique random values to return as an array to the sheet.

Figure C shows this expression using our earlier argument values:

```
=INDEX(UNIQUE(RANDARRAY(5^2,1,1,10,TRUE)), SEQUENCE(5))
```

First, RANDARRAY() returns a single-column array of 25 random integers between 1 and 10. UNIQUE() removes all the duplicates from those results. INDEX() then returns the top five unique and random values, as determined by SEQUENCE(5).

Figure C

	A	B	C	D	E	F	G	H	I
1									
2									
3		2		6					
4		16		9					
5		10		10					
6		1		5					
7		13		3					
8		20							
9		9							
10		3							

You need a more stable expression.

Keep in mind that min and max still matter. For instance, if min and max are 1 and 5, respectively, you can't return an array of 10 values—there aren't that many unique values between 1 and 5. The expression will return five unique values, but the remaining cells in the spill range will display errors. That's an over-simplification, but it makes the point.

In addition, large arrays take time. Consider a rows value of 2000^2 . Internally, Excel is calculating 4,000,000 random numbers! You probably don't need that large of a pool. Instead, multiply the rows argument by another value, such as 10. That returns 20,000 random values, which might be enough, dependent on the min and max values.

You'll have to experiment to determine whether a pool is large enough. There's no true silver bullet because of the relationship: the chance of returning an incomplete spill range increases as the number of random numbers passed to `UNIQUE()` decreases.

It's confession time. This is a creative and powerful expression that takes advantage of Excel's newer dynamic array functions. I'd like to claim full credit for designing this expression, but it has been around the internet now for a while. If I could credit the creative genius who came up with it the first time, I would.

To learn more about dynamic array functions, read the following articles:

- [How to use the newish XLOOKUP\(\) dynamic array function in Excel](#)
- [How to generate random text using the RANDARRAY\(\) function in Excel](#)
- [How to use the FILTER\(\) dynamic array function in Excel](#)
- [How to use the spill range in Excel](#)

Subscribe to the Developer Insider Newsletter

From the hottest programming languages to commentary on the Linux OS, get the developer and open source news and tips you need to know. Delivered Tuesdays and Thursdays

Email Address

By signing up to receive our newsletter, you agree to our [Terms of Use](#) and [Privacy Policy](#). You can unsubscribe at any time.

Share Article



Susan Harkins

Susan Sales Harkins is an IT consultant, specializing in desktop solutions. Previously, she was editor in chief for The Cobb Group, the world's largest publisher of technical journals.

[See all of Susan's content](#)

