

Splice to slice and dice arrays in Perl

[splice](#)[array](#)[Prev](#)[Next](#)

After learning about [pop](#), [push](#), [shift](#), and [unshift](#), students sometimes ask me how to remove an element from the middle of an array.

I usually don't have time to explain it. There are other things to teach them, that seem to be more important than `splice()` in the limited time we have, but usually I at least point them in the right direction.

This time it is much easier as you, the reader, can decide if you'd like to invest the extra time.

How to remove an element from the middle of an array in Perl?

```
1. my @dwarfs = qw(Doc Grumpy Happy Sleepy Sneezzy Dopey Bashful);  
   splice @dwarfs, 3, 2;  
   print "@dwarfs";      # Doc Grumpy Happy Dopey Bashful
```

As you can see the 4th and 5th elements of the array were removed. That's because the second parameter of **splice** is the offset of the first element to be removed, and the third parameter is the number of elements to be removed.

How to insert an element in the middle of an array in Perl?

```
1. my @dwarfs = qw(Doc Grumpy Happy Sleepy Sneezzy Dopey Bashful);  
   splice @dwarfs, 3, 0, 'SnowWhite';  
   print "@dwarfs";  
   # Doc Grumpy Happy SnowWhite Sleepy Sneezzy Dopey Bashful
```

In this case we used `splice` to insert an element. Normally the second parameter (the offset) defines where to start the removal of elements, but in this case the third parameter - the number of elements - was 0 so `splice` has not removed any elements. Instead, the offset is used as the position to insert something new: the value passed as the 4th parameter to `splice`.

That's how SnowWhite ended up among the seven dwarfs.

How to insert a list of values in an array in Perl?

Inserting one element is actually just a special case of inserting several elements.

```
1. my @dwarfs = qw(Doc Grumpy Happy Sleepy Sneezzy Dopey Bashful);  
   splice @dwarfs, 3, 0, 'SnowWhite', 'Humbert';  
   print "@dwarfs";  
  
5. # Doc Grumpy Happy SnowWhite Humbert Sleepy Sneezzy Dopey Bashful
```

In this case after the 3rd parameter we have several values (2 in this case). They get all inserted in the array.

How to insert an array in the middle of another array in Perl?

The same would have happen if we passed an array as the 4th parameter:

```
1. my @others = qw(SnowWhite Humbert);  
   my @dwarfs = qw(Doc Grumpy Happy Sleepy Sneezzy Dopey Bashful);  
   splice @dwarfs, 3, 0, @others;  
   print "@dwarfs";
```

Replace part of an array with some other values

You can also add and remove elements in a single command:

```
1. my @others = qw(SnowWhite Humbert);  
   my @dwarfs = qw(Doc Grumpy Happy Sleepy Sneezzy Dopey Bashful);  
   splice @dwarfs, 2, 4, @others;  
   print "@dwarfs\n";  
  
5. # Doc Grumpy SnowWhite Humbert Bashful
```

In this case we removed four of the dwarfs and replaced them by two full-sized people: SnowWhite and Humbert the Huntsman.

splice

Splice is the ultimate function to change arrays in Perl. You can remove any section of the array and replace it with any other list of values. The number of elements removed and added can be different,

and either of those can be 0 as well.

The general syntax the function has the following parameters, though all the parts (well, except of the array itself) are optional:

```
1. splice ARRAY, OFFSET, LENGTH, LIST
```

The OFFSET and LENGTH define the section in the ARRAY that will be removed. They are both integers. The LIST is a list of values that will be inserted in place of the section that was removed. If the LIST is not provided, or empty, then splice will only remove items but not insert any.

Return values

In **LIST context** splice returns the elements removed from the array.

```
1. my @others = qw(SnowWhite Humbert);  
   my @dwarfs
```