Drop Non dominants is another way that we simplify big O notation.

So to explain this bring up some code here.

This is our nested for loop that we saw from before.

What I'm going to add to this is an additional for loop that comes after the nested for loop.

Let's take a look at this in DevTools.

So there's our code and we're going to run this with the number ten.

Okay, I'm going to scroll to the top to begin with.

So we have output here is zero zero and this goes through two.

Nine nine.

This is our nested for loop, that second for loop that follows it goes from zero.

To nine.

So let's take a look at this over here.

That nested for loop ran.

O of n squared times.

The second one ran o of n times.

When we add these together.

This becomes o of n squared plus n.

So if you think about this in terms of if n was 100.

In squared would be 10,000, where the single end that's added to it in this equation is only 100 more.

It's not really affecting the number of operations.

N squared is the dominant term, and n by itself is the non-dominant term.

So we just remove it.

So.

We drop non dominance.