So, back in our test class, we're going to implement our first test method. BookingStartsAndFinishesBeforeAnExistingBooking. So, first we need to create a mock object for our repository. So, mock, we set this to a new Mock, of IBookingRepository. now here, we need to program this mock by calling the Setup method. So if we call repository. GetActiveBookings and pass 1 as the ID of the excluding booking, we want this to return a List like this.

A List of Booking, we can initialize this list right here, and look here we have a compilation error, because this method should return an IQueryable object but here we are programming this mock to return a new list. So we need to explicitly convert this list on an iQueryable object, and it's very easy, we simply call .AsQueryable on this list object. Okay? So the compilation error is gone, now in this list, I want to add an existing booking. So let's create an existing Booking. We set the ArrivalDate, let's pick a date, new DateTime, let's say 2017.

January 15. Now, let's imagine this code, is for a hotel reservation.

As you know, in most hotels, the arrival or check in time, is at 2 pm. And Departure, or check out time, is at about 10 am. So here, I'm going to set the arrival to hour 14, minute 0, and second 0.

Now, let's set the departure date we set this to 5 days after, so new date time, 2017, January 20th, at 10 am. Also, we should set the Reference property, let's set this to a simple string like a. And finally, Id property. Now I would rather put Id first, as it's more conventional, so here we have a list of existing bookings, and this is our arrange part.

Now, we need to act. So, we call BookingHelper.Over lappingBookingsExist, we need to give it a booking object, and this Mock repository. So let's create another Booking object, I'm going to set the Id of this to 1.

Because this is the same value we passed. here, to GetActiveBookings, so this Booking is excluded, now we need to set the arrival and departure dates so, to save time, I'm going to copy these two lines here, and then modify them. We have this compilation issue, because we have not passed out mock repository as the second argument to this method. So this pass repository .Object.

Oops, I just realized I that I made a mistake earlier, so I call this object Mock, it's better to call it repository, okay, and here we pass repository.Object. Okay? Now, we need to modify these dates. So the scenario we're testing is where this new booking starts and finishes before an existing booking.

So, I'm going to set this to 10th of January, and the departure date to 14th. Next, we need to get the results. result, and Assert, That result Is Empty. So this means, this is an empty string. Let's run this test, okay, here's our test, and it successfully passed, beautiful. However, as you can see, this test method is very bulky, it's very fat, and we have all these magic values here. So in the next lecture, we're going to refactor this code and make it cleaner.