One of the most powerful properties of integer optimization is the ability to add what we call logical constraints.

These use binary variables to implement different restrictions.

Let's look at a few examples.

Suppose we want to add the constraint that A and B can't play in both weeks 3 and 4.

We can do this by adding the constraint x_AB3 + x_AB4 less than or equal to 1.

What are feasible values for these two variables, given this constraint?

Both variables can be 0, or one or the other can be 1.

A solution with both variables equal to 1 would be infeasible.

So this constraint models the restriction that teams A and B can't play in both weeks 3 and 4.

Note that this only works because the variables only take values 0 or 1.

Let's look at another example.

Suppose we want to add the restriction that if A and B play in week 4, then they must also play in week 2.

We can do this by adding the constraint x_AB2 greater than or equal to x_AB4.

Let's consider feasible solutions to this constraint.

If x_AB4 equals 1, then x_AB2 must also equal 1.

But if x_AB4 equals 0, then x_AB2 can be equal to either 0 or 1.

So this constraint exactly models the restriction we wanted.

Lastly, suppose that teams C and D must play in week 1 or 2 -- they can't play both games in weeks 3 and 4.

We can model this constraint with x_CD1 + x_CD2 greater than or equal to 1.

Feasible solutions are both variables equal to 1, or one variable equal to 1.

Both variables can't equal 0.

These are just a few examples of logical constraints that we can implement using binary variables.

Let's go into LibreOffice now and add these constraints to our model.

In our spreadsheet, let's scroll down to the bottom of our constraints list.

We wand to add in three new constraints.

The first is that teams A and B can't play in both weeks 3 and 4.

The second is that if teams A and B play in week 4, then they have to play in week 2.

And the third is that team C and D have to play in weeks 1 and/or 2.

So, to add the first constraint, our left hand side should be equal to the sum of the variables for A and B in weeks 3 and 4.

Our sign is less than or equal to and our right hand side is 1.

Our second constraint, the left hand side, is equal to the decision variable for teams A and B in week 2.

The sign is greater than or equal to, and the right hand side is equal to the decision variable for teams A and B in week 4.

For our third constraint, the left hand side is equal to the decision variables for teams C and D in weeks 1 and 2, the sign is greater than or equal to, and the right hand side is 1.

We can see here that our current optimal solution violates every single one of these constraints.

So, let's go ahead and resolve our model to get a new solution.

So go to the Tools menu and select Solver.

Now, let's add in our new constraints.

So in the third cell reference box, go ahead and select the left hand side for the first constraint.

Make sure the operator is less than or equal to, and select the right hand side for the value.

Then in the fourth cell reference box, select both of the left hand sides for the other constraints because they're both greater than or equal to constraints, change the operator to greater than or equal to, and for the value select the two right hand sides.

And go ahead and click Solve.

The solving result says "Solving successfully finished.

Result 20." Select Keep Result.

Let's look at our new solution.

The objective value is now 20 which is less than it was before.

But now we have teams A and B playing in weeks 2 and 4 and teams C and D playing in weeks 2 and 4.

So, depending on your preferences as a decision-maker, you might prefer the previous solution which had a slightly higher objective or this solution which is a little more balanced.

In the next video, we'll discuss how additional constraints often make the problems harder to solve and how large integer optimization problems are solved in practice.