

Pīkau Name: Communicating well when programming (CT P0 5)

Video Name: Comparative and Logical Operators Bonus video (EMP10-7b)

Presenter: Tim Bell

Here's another example where I need to use a combination of logical operators to achieve a decision. What I'm doing here is I'm writing a program. Maybe it's going to become an app to tell me the best days for line fishing. This is based on the day of the lunar month. Let's just assume that the user can tell us what day of the month it is. Normally you try and get that from the system's clock or something like that so your app can pop up and say 'today's a good day for line fishing.' There are some maramataka that say the best days for fishing are the first six days of the month or the last seven days or Day 20. So we've got this logical condition or many different values that we want to fit. Here we've got a list of days. I've put them in a comment so I can see them while I'm working on things.

Now I need to put in an 'if' statement to figure out if it's going to be a good day for line fishing. Let's just put in the 'say' already. Say 'It's a good day for line fishing.' The condition here is kind of interesting because for the day to be number 1 - 6 there's many ways we can do that logically. One simple way I could do it is basically find out if the 'lunar day' (notice I've named it 'lunar day' just to be clear to anyone else reading this what sort of day it is) is '< 7'. That is checking really if it is between 1 and 6. Another condition of course will be if the day is '= 20'. So we will just do the 'lunar day = 20'. I'm just putting together the components of my logic. I'm not putting them into the 'if' statement yet. Then finally it's if it's at the end of the month, if the day is after day 23 so 24 onwards (we'll put 'lunar day > 23) then that will be accepted.

Now here's where there is a bit of a twist between English and pure logic because the English says the best days are days 1-6, Day 20 *and* days 24 [and greater] but in pure logic if we say 'is it Day 1 *and* Day 2 *and* Day 20' and so on it will never be Day 1 and Day 20 at the same time. That cannot happen. What's meant here is an 'or'. If it's Day 1 *or* if it's Day 2 *or* if it's Day 20 *or* if it's '> 23'. That's where I need to use my logical operator which is the 'or' operator that will accept if either of these conditions is true. When I say 'either' it's got two possibilities. I've got *three* conditions that I want. This is where we are getting a little beyond the logic that students might need but it's good to know we can actually put an 'or' within an 'or' and that will accept any of them. So we will put the 'lunar day = 20' in there and the 'lunar day > 23' there. What this is saying is that (the left hand side here) 'lunar day < 7' if that is true we'll just say it is a good

day for fishing. Or if that doesn't work let's see if the right hand side is true which is one of these two things can be true and so on. We'll put this up into my 'if' statement. I'm now getting quite a complicated logical decision going for what was a relatively simple rule. Let's run it and see if the cat is good at giving advice about fishing. 'Which day of the lunar month is it?' Let's start off with Day 3. 'It's a good day for line fishing'. Try again with something that's out of range. According to the rules up there Day 7 isn't. If I type in a '7' no result. Let's check out the Day 20 is working okay. 'Good day for line fishing.' Day 24 should be included but let's just check that Day 23 doesn't trigger it so try that. No, no result. Day 24 of the lunar month let's say that. 'It's a good day for line fishing.' So there is a rule that obeys this tradition that those are good days for line fishing. It's embedded in a logical operator.

I've got a very manual system here where you type in the month. That kind of logic would be buried inside an app. Maybe it's an app that automatically starts up each day and figures out the lunar day and puts it through that logic and sends through an alert to tell you that it's time to go line fishing.