14.3.3 Exit Codes

Click one of the buttons to take you to that part of the video.

Exit Codes 0:00-0:12

In this demonstration, we will discuss exit codes.

To illustrate how exit codes work, we're going to go ahead and run a command successfully.

Successful Exit Code 0:08-0:38

So, if we look in the current directory, we'll find that there is a file named 'file'. It's right there, very plain and simple. If I type in 'ls file', that was a successful command. If we take a look at the exit code, we'll see that it's zero (0). Zero (0) indicates that the program or the procedure ended successfully.

Error Exit Code. 0:39-1:25

Now let's look at an error. If I type 'ls' and then use uppercase letters for 'FILE', it gives us an error telling us that, that doesn't exist. If we do that same test, to take a look at what the exit code was, we see that the number two (2) is there. By default, two (2) means, 'file not found'. Using that, along with scripts, we can define different exit code numbers to mean different things.

I've created a couple of scripts and I've put them in the directory of usr, local, bin. Let's change to that directory temporarily using 'cd /usr/local/bin'. In this directory, you can see I have two files. One is 'demo1' and the other is 'demo2'.

Second Script 1:26-2:04

Inside of 'demo2'--Let's take a look at that. I'll go ahead and 'cat demo2', and we'll see that what I'm doing here is, I'm executing the 'demo1' script. Then I'm testing the exit code. Based on the exit code, I do something. If a '101' comes up, then I print, "Error 101, must be the root user to run." If exit code 102 is the result, then the configuration file does not exist. If I do return a zero, that tells me I've successfully completed the script.

First Script 2:05-2:37

Let's take a look at 'demo1'. I'll 'cat demo1.sh'. Very simply, what it does is, it tests to see if I'm the root user. That's the first test. If I am not the root user, '!= 0', then, I'll exit with the code of 101, saying I'm not the root user. Else, if the file 'exit.conf' does not exist, then I'm going to exit with a 102 error. Otherwise, it will be successful.

Run the Script, Not root User 2:38-3:11

Let me go ahead and clear the screen, and let's go ahead and run 'demo2.sh'. It tells me gives me the error 101, meaning I must be the root user to run this. Let's go ahead and 'sudo su' to the root, type in the password of my current user, and you can tell by the prompt change that I am now the root user. If I look at my present working directory, it tells me I'm root. That's great. If I do a 'whoami', it tells me I am the root user's.

Run the Script, No Configuration File 3:12-3:31

Let's go ahead and run that same program, 'demo2.sh', and now it didn't tell me that I had the error of not the root user. It's telling me, I don't have the configuration file. Okay, well that's fine. I'll go ahead and create the configuration file 'exit.conf'. Great.

Run the Script, Successful 3:32-3:51

Let's go ahead and run demo2.sh again and this time, it successfully completed. So if you remember from the scripts, the first thing you checked for was, "Am I the root user?" Then it checked for a configuration file, and then if those two checks worked, then it was successfully completed.

Summary 3:52-3:55

In this demonstration, we showed you how exit codes work and how to test for exit code completion.
