LESSON TRANSCRIPT

JAVASCRIPT - LESSON 1, TOPIC 5: NUMBERS AND INPUTS

INTRO
DEFINITION OF PROMPTS
CREATING A PROMPT IN JAVASCRIPT

INTRO

00:01 Welcome back. This is lesson one, topic five, numbers and inputs. Start then with a bit of a breakdown of what we're going to cover in this video.

00:11 This video has two main topics. Firstly, what are prompts? And secondly, how do we do mathematical operations in code? Starting with prompts then, let's have a definition.

DEFINITION OF PROMPTS

00:27 Prompts are a method used to display information. Now, this is where a visual might help. So you've probably come across many of these in your time on the Internet.

00:40 And as you can see, prompts are a kind of physical pop-up. And at the top of them, we have We'll see you next week.

00:45 We've got some guidance text to tell our users what we need from them. And then we usually have a little input box where we get them to write a piece of information.

00:55 A name, a username, a password, this kind of thing. So this is what a prompt physically looks like. And bonus points for those of you who are interested. Do you out there who know who Radia Pearlman is.

01:08 Okay, so now let's break down an example of how we go about creating these prompts in JavaScript. As you can see on screen, we have an exchange rate program here.

CREATING A PROMPT IN JAVASCRIPT

01:21 Let's break down the code and see what's happening. So on line 3, we create a constant. We give it the name GBP or Great British Pounds.

01:31 And we set the value of that constant to equal prompt, enter an amount in GBP and 10. So let's look at this part specifically because this is what we're interested in here.

01:45 So in order to create this prompt window that we see on screen, we use this prompt method. So just like with console.log, we have the name of the method prompt, and then we open a set of brackets to contain the information that we need to give it.

02:00 So in this case, our prompt method takes the value of this method. So we're going to use this prompt method to create this method. Two arguments. So we've got the text that we want to be visible in our pop-up to what we need from our users. So in this case, we've gone for entered an amount in GBP.

02:22 And then a comma. And then we have this value 10. So this is optional. You don't have to put one on here. But what this is, is this is a default value. So it's kind of a placeholder to go inside the box and give the user an indication of the form of response that we need. So it just sits in there as some Placeholder text

02:38 Next on line four, we create a const with the name rate and we give it a value of 1.5.

02:46 So every one British pound is going to be worth 1.5 of our future coins. Then on line five, we create a constant to hold the value of the mathematical calculation we're going to do.

03:00 So we create a const result and then we give it the maths we want it to do. So in this case we want you to take whatever value our user inputs in the prompt, GBP, which

we know because we gave it the name GBP on line three, and we want you to times it by our rate.

03:22 So this might look a little bit unusual here. We've seen that we can use pluses in JavaScript, like in mathematics, to add things together or to add strings together in concatenation.

03:33 But when it comes to time-sing, that's where things differ- or multiplication rather, that's where things differ from traditional maths.

03:40 So, you know how we would use an x in maths to denote multiplication? Well, in JavaScript, we use this asterisk symbol.

03:48 Lastly, we are console logging our result and using string interpolation. You'll notice that because we have those backticks and we're using the name of a variable after this dollar sign and inside these curly braces.

04:07 So we are taking this result and we're going to put it in there with string interpolation and we're logging the results of this maths that we've just done for the user.

04:21 So this is a lot of new information. I'm sure you might be sort of thinking, oh, we've got a little bit lost in the middle there.

04:27 Well, don't worry. We're about to practice this together in our walkthrough. So in the example coming up, we're going to be converting an amount in Great British Pounds to a new made-up currency we're creating called EGP.

04:42 Let's have a go at this together.

04:46 OK, so thinking about our aim here, which is to create a currency converter that takes in a value in pounds and tells us what its value would be in our new currency, EGP, let's kind of think through the steps we need to take.

05:03 So I think to start with, let's create a variable, a constant, to hold the value of whatever our user inputs.

05:12 So when we ask them how many pounds would like to convert, we need to store that answer in something.

05:18 So let's create that now. Let's say const GBP and then use our equals sign or equals assignment operator. And now we're going to create that pop-up window.

05:31 So if you recall from our future coin example, we use this word prompt followed by a set of brackets. Now that might look quite familiar because it's a similar structure to our console dot log method, right?

05:46 So we have to remember everything we want to be in that prompt. We need to go inside those sets of brackets.

05:52 And now in here, I'm going to put a string that is whatever text I want to display to my user.

05:58 The instructions, right? So in this case, let's go for please. Always pays to be polite. Enter an amount in GBP.

06:11 But this can be whatever you want, right? And now also, if you recall, we have the option to give our input field, that text box, a default placeholder.

06:22 So it's not just an empty box. So in our future coin, it was 10, let's say 20 this time. Okay.

06:30 So we have our first line here. Now, entirely up to you, but in my daily life as a programmer, I like to call it a day. I can run things semi-regularly so I can make sure that each line is doing what I expect it to.

06:42 So let's give this a run now and make sure that we're getting that pop-up prompt window. Let's have a see.

06:48 Let's just visually confirm that. Brilliant. So we have, please enter an amount in GBP. And our placeholder value of 20 there.

06:57 Great. So now let's think about what other information we need to give this programme. So we've captured what our user's input is.

07:06 So whatever, however many British pounds they enter, and now I need to think about the kind of maths of it all.

07:12 So I'm gonna need a constant to hold the rate, right? And let's go a bit wild this time. Let's say it's an amazing new currency hour.

07:22 It's 38.62 EGP. To the British pound. So I've created this variable here to hold that value. Next, I need to write the maths of it all.

07:36 So let's create another variable for that. So, In our previous example, we called that result. We can call it whatever we want as long as it makes logical sense.

07:46 So in this case, let's say EGP. So this new currency name and we'll say that our value in EGP is equal to whatever the user inputted in line one.

08:00 So in that case, that value is being held in this variable GBP times the rate. Oh! The keyboard's having a funny moment there.

08:12 times the rate. But, as you can see, have you noticed, there's something wrong with the line I just typed? And that is, we don't use this multiplication symbol in code.

08:25 So just a nice reminder that we instead use the asterisk. It means the same thing. So the value of my EGP is equal to whatever the user input times by the rate.

08:38 And now there's one more step I want to do and that's to feedback to my user on what the answer to their question was.

08:45 Is the value of 10 great British pounds in EGP. So to do that we're going to use our friend the console.log method and let's this time do some string concatenation for practice.

09:01 So I'm going to say console.log EGP so the total amount plus so remember when we use concatenation we use our plus symbol and then I'm going to say EGP so this will be a number 300, 400, whatever plus I'm going to say to my user EGP but there's something missing here and I wonder if you've spotted it's

09:29 a tricky one so as a reminder. It won't add the spaces in for us so I'm going to add a space before that to make sure it looks correct when it runs so let's have a look let's

run things please enter an amount in Great British pounds let's say 100 pounds how many EGP does that get me and we've console

09:54 logged our output here so we have the the value we've performed our mathematical equation here and we have that value followed by space e g p and just to test my theory let's take the space away and run it again and see what happens oh yeah we're gonna give it an amount let's say 50 this time it's out

10:20 and edit it directly to the end so just as a great reminder there that we have to account for these spaces ourselves as well fantastic great work

10:30 Let's move now to maths operations and specifically, how can we perform maths operations in code. But first, let's start with a definition.

10:40 So we can use maths operators to do calculations. This might give you a bit of an insight into what this means.

10:49 So in this case, maths operators refer to the things in this column. So in traditional maths, we're talking plus sign, minus, multiplication, division, etc.

11:03 But as you can see from our table, some of the operators that we use in JavaScript differ a little from the ones that we would use in traditional maths.

11:12 So in the first two columns here, addition and subtraction, are much the same. As you would expect. Then we can see when we get to multiply, instead of the cross, as we discovered in a previous walkthrough, we use this asterisk symbol.

11:28 And now division, also different, we use one-forward slash to represent a division symbol. Next, we have an exponent, meaning when you raise something to the power of something else.

11:42 And in this case, we denote that with two asterisks. So what I mean by this is, if we want to say two to the power of five, we would do two, asterisks, asterisks, five.

11:55 Last, we have the modulo operator. This is a slightly tricky one to get your head around, but it becomes clearer with time.

12:02 So, this looks like a kind of percentage symbol and it refers to when we want to get the remainder value after a division has occurred.

12:14 So just bear that in mind for now, but that probably won't be something you'll use for quite a while. Similarly, we have two different types of numbers, both of which are valid in JavaScript, so we can have whole numbers like the one here, but we can also use decimals like 0.5.

12:34 So excitingly, you've almost made it to your first live lab for this lesson. So let's have a little bit of a look at what you're going to cover.

12:44 And just as a reminder, these live labs are where you get a chance to put everything we've been learning into practice and write lots of code yourself.

12:53 So specifically, you're going to be looking at writing a JavaScript program that outputs text using our console dot log method, running your program in the code pen editor, and making sure you've got all your seconds.

13:12 Getting input from the user using a prompt, storing information's constants using const and joining strings of text together. So we learned those two methods.

13:25 We had string interpolation and concatenation. And basically sharing your code for others to try out.

13:32 Only a handful of videos, I hope this list of things that you're going to be programming this week gives you a bit of confidence in how well you're doing.

13:40 In your JavaScript journey. All right, so before you try out your very first live labs, just a reminder here to please complete any outstanding quizzes.