**4COSCO11C.2 Web Design and Development**

**Tutorial 6**

### Task 1 - Create your Web Page content

You might want to copy your previous JavaScript solution (tutorial 5) to a new file and save it on your home **h:** drive as **tutorial6.html**.

**Important Notes:**

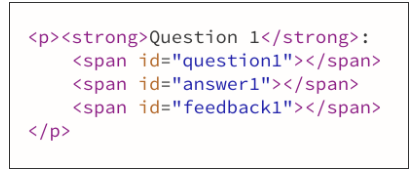
* No space in the filename
* Your file must have the .html extension
* If you save your work outside of your h: drive, it will be deleted

Your Web Pages should follow the following specifications:

#### **HTML (identical to tutorial 5)**

Your Web Pages should follow the following specifications.

* Tutorial 6 for the title.
* 'A Simple Maths Game' as the page heading.
* Create the placeholder on your page for one maths question, followed by the placeholder for the user's answer to that question, and then another place holder for the feedback for that question: For example, use a span with the id="question1", a span with the id="answer1" and a span with the id="feedback1".
* You may put all this inside a separate paragraph, so that one question block looks like this:



* Repeat this 10 times, and update the id so that they match the question numbers (question2, answer2 and feedback2 for the second question, etc.).

#### **CSS (identical to tutorial 5)**

You will declare rules for 2 classes: correct and incorrect as follow:

* the correct class will display text in green colour and bold font
* the incorrect class will display text in red colour and bold font

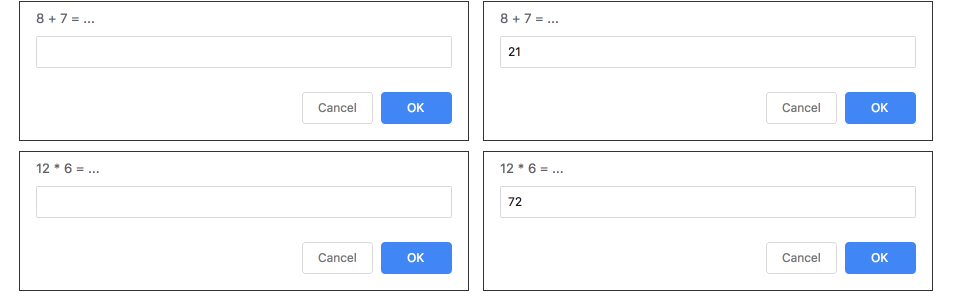
#### **JavaScript**

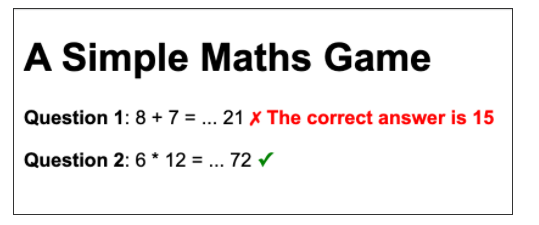
##### **Overview**

The overview is identical to tutorial 5 except that this time the questions are generated randonmly see Process section

* Using JavaScript, you will prompt the user to answer 10 maths questions. You should prompt the user 10 times with a different question each time.
* Record the answers and compare them to the correct answers.
* After each answer or once the user has answered all 10 questions, display on the page each question in the question placeholders, the users's answer in the answer's placeholders and a message saying whether the answer was correct or not in the feedback's placeholders. In addition, use a tick symbol (✓) using the &checkmark; sequence if the answer is correct, and a cross symbol (✗) using the &cross; sequence if it is incorrect
* In case of an incorrect answer, also display what the correct answer is.
* The symbols and feedback to the user need to be in a different colours. Use 2 CSS classes that you declared earlier for this.

See the images below to get an idea of the sequence:





##### **Process**

We now want to generate the questions randomly. The questions are composed of 2 random numbers, and 3 possible operators (+, -, \*).  
**hint 1:**let x = Math.floor(Math.random()\*101); will declare a variable x and initialise it with a random number between 0 and 100.

1. Let's start with 1 question...
   1. Remove all but one question from the JavaScript code you wrote for tutorial 5.
   2. We will need two random numbers for that question. Let's call them x and y
   3. Identify in your code where x and y will be used. Let's take a pen and paper for this. Write the code for your 1st question and annotate it with where the variables will need to replace fixed numbers.
   4. Prompt the user with your first question, and store their answer in a variable.
   5. You will now need to calculate what the correct answer is
   6. Display the text of that question in the question1 span, and the user's answer in the answer1 span.
   7. Compare the answer with the correct answer
   8. If the answer was correct, add a tick symbol (&checkmark;) in the feedback1 span and add the class "correct" to that span (you can use document.getElementById("feedback1").classList.add("correct") for this

. If the answer was incorrect, add the cross symbol (&cross;) in the feedback1 span and add the class "incorrect" to that span

1. Test your code once with a correct answer and once with an incorrect answer
2. If your code works as expected, ideally, we would want to repeat this 10 times... But let's only repeat it once for now, so we cover 2 questions...
3. As for tutorial 5, calculate the score and add it to the score placeholder and test your code a number of times.
4. Now let's change the background colour of the page based on the score. For example, change the background to light red in case of a score below 20.  
   **hint 3:** you will need to change the background within JavaScript using the HTML DOM Style Object. E.g. document.body.style.backgroundColor="purple"
5. If you are struggling with the code, then just repeat this for your 10 questions. If you feel confortable with the code and you're after some challenges, move on to the additional challenges instead where we will use a loop instead. In this case remove your javaScript code for the second question, and start the "Additional Challenges"

**Additional Challenges** - Consider generating the questions and associated display within a loop. However, you will need to experiment with how you will generate the different operators (+, -, \*) within the loop:

1. Actually it will prove complicated to create 10 separate variables for each of the ones that are needed for one question. It will make a lot more sense to use arrays for these, so got ahead and replace your variables with arrays (you can declare an empty array using const questionText = [].
2. Likewise, we need to use arrays for the elements on the page. So we will use classes instead of ids for all the spans that relate to questions. So go ahead and replace id="question1" with class="question", etc.
3. Now, instead of using getElementById to access an element, we will use getElementsByClassName to retrieve an array of elements with the same class name.
4. Whereever you use arrays, you can use the array methods or use the index used as part of the for loop.
5. Test your code with a combination of correct and incorrect answers, then all correct and then all incorrect.
6. Add a button to your webpage and allow the user to start the quiz by clicking the form button.  
   **hint 4:** you will need to place your JavaScript code into a function.