Tutorial 3

In this tutorial, you are going to practice **JavaScript** coding based on what you have learned in the last week. Some of the challenges may be quite difficult as they are deliberately included here. Overcoming obstacles is a part of learning journey. You can do it if you are persistent in working on.

Activity 1

Task 1

Download the **Provisions** zip file from MyLO. Within the zip are a **scripts.js** file and a **tutorial.html** file. You will write your JavaScript for this tutorial in scripts.js and execute it within tutorial.html, which contains a series of answer boxes (`div` elements) that you will need to populate.

1. Open tutorial.html and view its source to familiarise yourself with the page structure.
2. Your first task is to include the script.js file in the tutorial.html file. To do this, you will need to use a `<script>` element.
3. Once this is done, save the page and reopen it (or refresh) in Chrome. Right click on the page, choose **Inspect** and then select the **Console** tab. Confirm that you see the *'Hello from script.js'* message, which will validate that you have included the script file successfully.

Activity 2

There are two gymnastics teams, the **Kangaroos** and **Gumtrees**. They compete against each other on four (4) occasions. The team with the highest overall average score wins the annual trophy.

Task 1

1. Calculate the average score for the Kangaroos using the data in the table below and store the result in a variable. Repeat for the Gumtrees team.
2. Compare both team's average scores to determine which is the winner of the competition and output the result in the associated answer box. If there is a draw, both teams win the trophy.
    a. ***Hint**: you will need to access the associated element using one of JavaScript's DOM functions, e.g., `document.getElementBy`…*

Team	Result 1	Result 2	Result 3	Result 4
Kangaroos	97	109	89	90
Gumtrees	89	92	92	111

Task 2

The rules used to determine a winner have changed. Now a team wins only if they have a higher score than the other team and their average is at least 100 points. As before, output the result in the associated answer box.

Team	Result 1	Result 2	Result 3	Result 4
Kangaroos	98	113	102	100
Gumtrees	109	96	124	98

Task 3

A new rule has been applied. In the event of a draw, both teams must have scored an average of at least 100 points. Otherwise, there is no winner of the trophy.

Output the result in the associated answer box.

Team	Result 1	Result 2	Result 3	Result 4
Kangaroo	98	112	102	100
Gumtree	110	96	107	99

Task 4

A fifth competition has been added to the calendar. Each team now competes 5 times and then the average of the 5 results is calculated (one average score per team). A team only wins if it has at least **double** the average score of the other team. Otherwise, there is no winner.

1. Create function called `calculateAverageA` to calculate the average of the 5 results.
2. Use the function to calculate the average for both teams.
3. Create a function called `determineWinner` that has two formal parameters of `avgTeam1` and `avgTeam2`.
4. Output the winner in the associated answer box along with the avg score for each team. For example: *Gumtrees win! (100 vs. 20)*
5. For this task, you do not need to consider a draw.

Team	Result 1	Result 2	Result 3	Result 4	Result 5
Kangaroo	44	95	71	85	100
Gumtree	21	54	49	23	34

Activity 3

Sophie wants to build a very simple tax calculator for her clients. The base tax rate is 35%. However, if the taxable income is greater than $75,000, the tax rate rises to 40%, and if the taxable income is greater than $100,000, the tax rate rises to 45%.

Person	Income
Henry	$65,000
Taylor	$85,000
Amanda	$101,000

Task 1

Write a function called `calcTax` that takes any taxable income value as an input and returns the corresponding tax amount, calculated based on the rule above. Test the function using a taxable income value of 40,000 and confirm that it outputs '14000'.

***Hint**: use `console.log()` to test your function. To access the console in Chrome, refer to Activity 1.*

Task 2

For each person in the table above, use the `calcTax` function to calculate their tax and then output the following to the associated answer boxes, replacing **xxx** with the income, **yyy** with the tax amount, and **zzz** with the income after tax.

*Taxable income was $**xxx**, tax amount was $**yyy**, and the income after tax is $**zzz**.*

Use box 2a for Henry, 2b for Taylor, and 2c for Amanda.

Activity 4

We are going to improve the Sophie's tax calculator.

Task 1

Define an array called `taxableIncomes` containing the values below.

35000, 45000, 50000, 62000, 70500, 82500, 97000, 101000, 132000, 150000

Task 2

Create two empty arrays for the tax and net income amounts called `taxes` and `netIncomes`.

Task 3

1. Use the `calcTax` function you wrote before to calculate the tax and income after tax (taxable income minus tax) for every value in the `taxableIncomes` array. You will need to use a loop for this.
    a. ***Hint**: use the `push()` function to add values to an array[1].*
2. Output the **sixth** value from the `taxes` and `netIncomes` array (comma separated) to the associated answer box.

Task 4

1. Create a function called `calculateAverageB` that has a single formal parameter called `arr`. Complete the function so that it calculates the average of all numbers in the given array (`arr`).
    a. ***Hint**: you will need to use a loop to iterate over all values in the array and add each one to a running total. Once you have looped through all values, you will be able to calculate the average. You can get the number of elements in an array from its `Length` property.*
2. Using the function, output the average of the `netIncomes` array to the associated answer box.

Submission

Show your tutorial.html page and scripts.js file to your tutor in your registered tutorial in **week 5** and be ready to answer any questions on your code. Tutors will expect to see and mark your work at the **beginning of the tutorial**, so please come prepared so that the marking process can be completed as efficiently as possible.

[1] See this W3 Schools page for more info on using arrays.