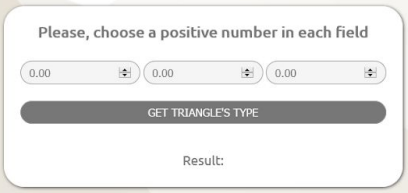
Triangle Challenge



by Kameliya Panayotova

Link: https://github.com/KameliyaPanayotova/triangle.git

Reflection

Design and implementation

The choices for my design are made with the purpose to create a good user experience, as much as possible. Therefore, I have used design principles such as contrast, alignments and symmetry.

I have made use of animation in order to influence the user's eyes and lead where they should focus. Also, animation is a way of keeping the user interested.

Since good user experience is not only about how it looks but also how it works, I have tried to make my solution easy to understand and thus help the user to act upon it. For instance, I have displayed a main text on the top of the inputs in order to give a hint and encourage the user to do something. In this case, the user's purpose it to choose a number in each field and click the button in order to display the result.

In addition, I have validated the inputs, so it helps the user enter the necessary type of data. This also aims at helping the user know what is going on in every single moment of the interaction.

Regarding my code solution, I have used HTML with input fields, button and text field for the interface, CSS for the style and JS for the interaction.

In the JS file, first I created a global variable and assigned it to the id of the HTML button element. It is a global variable because it needs to be accessed globally. Then I created an event listener for the button which calls the getTriangle() function. The

getTriangle() function creates local variables for the inputs and assigns them to the users input from the HTML file. The function also creates a variable for the result and assigns it to the id of the given HTML element. These variables don't need to be accessed globally but only when the button is clicked. The getTriangle() function also checks if the data type from the input fields is correct. If the data type is not correct or empty then the function assigns the result variable to a warning text that gives instruction. But if the data type is ok, then the getTriangle() functions calls another function named getTypeTriangle() which compares the numbers from the input and based on the comparison, it assigns the given triangle type to the result variable. Finally, I used a jQuery library to animate the main content.