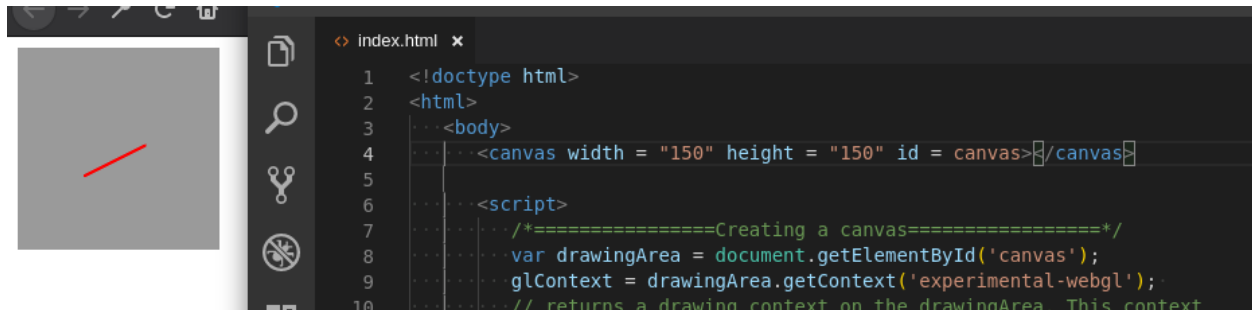


Part B:

TASK 1:

Change canvas width and height.

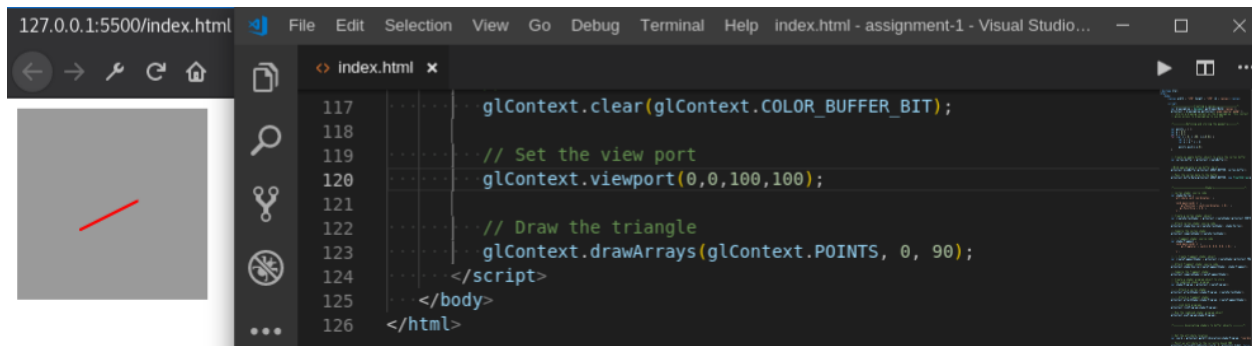
SOLUTION:



TASK 2:

Change the viewport width and height.

SOLUTION:

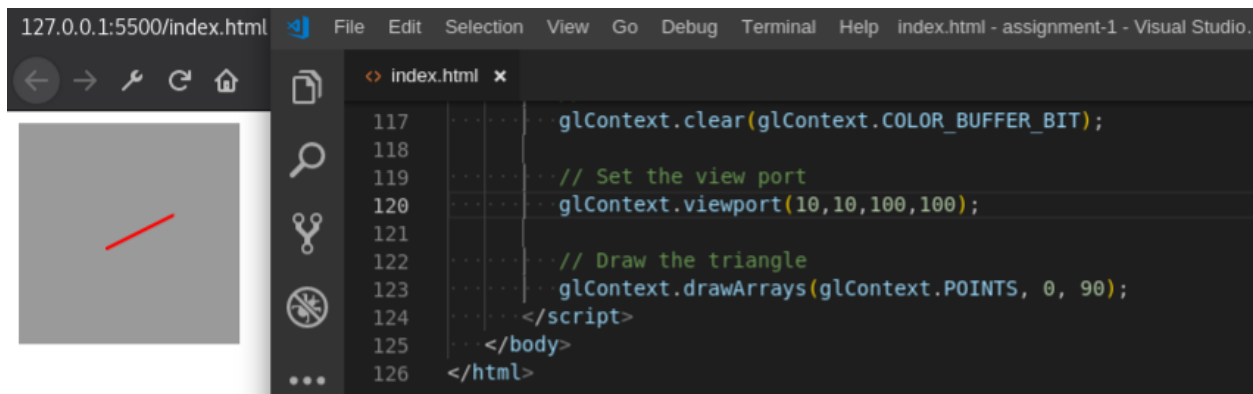


TASK 3:

What will happen if first two parameters in `gl.viewport()` are set to different values?

SOLUTION:

`gl.viewport(x,y,glContext.width,glContext.height)` tells WebGL how to convert from clip space (-1 to +1) back to pixels and where to do it within the canvas. If we change first two parameters of this function then horizontal and vertical coordinate for the lower left corner of the viewport origin will be changed. In the following screenshot, the line moves up a little bit as we set `(x=10,y=10)`. The default values were 0.



Part C:

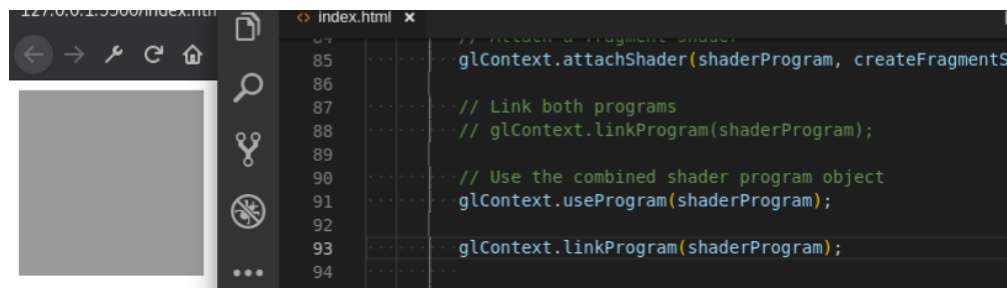
TASK 1:

Change the orders of the lines between line no. 34 to 83, and create 2 situations where -

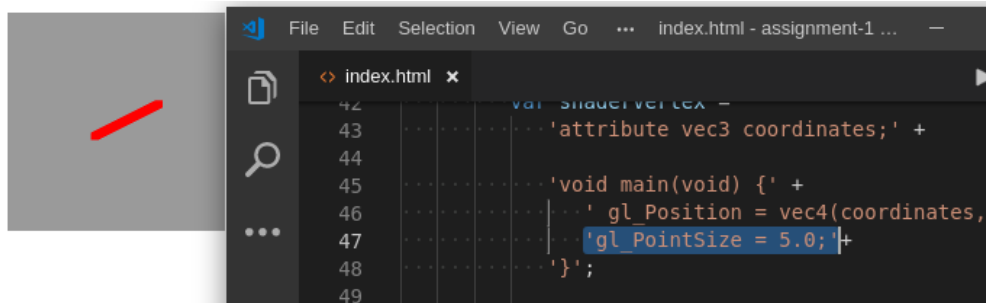
- (a) The codes does not work: explain why?
- (b) The code works: explain why?

SOLUTION:

(a) If we use `glContext.linkProgram(shaderProgram)` after `glContext.useProgram(shaderProgram)`, nothing draws on the canvas because both of the shaders are used to calculate and fill the color on individual pixels and without linking them nothing will be shown.



(b) In vertex shading program, the value of pointsize is increased from 3.0 to 5.0 and this makes the line wider because this built-in variable specifies the diameter of rasterized points.



Part D:

SOLUTION:

The major changes are shown in the following screenshots.

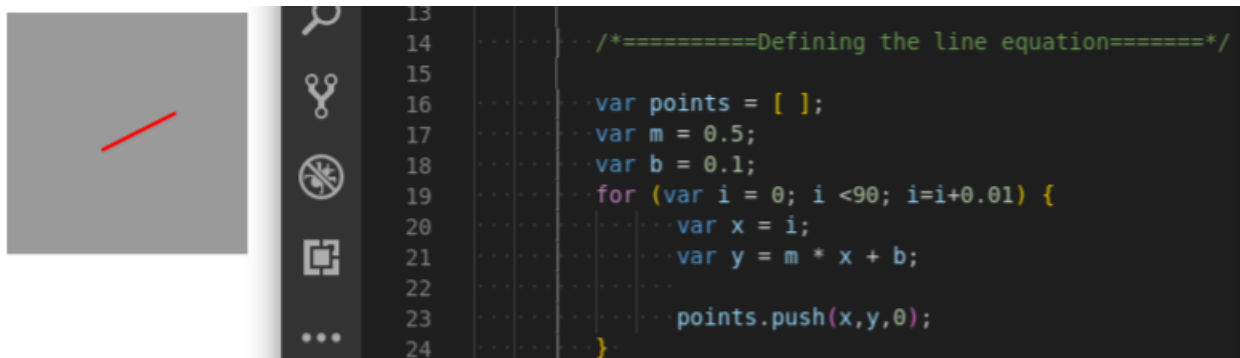


Figure 1: Assigning values to points array

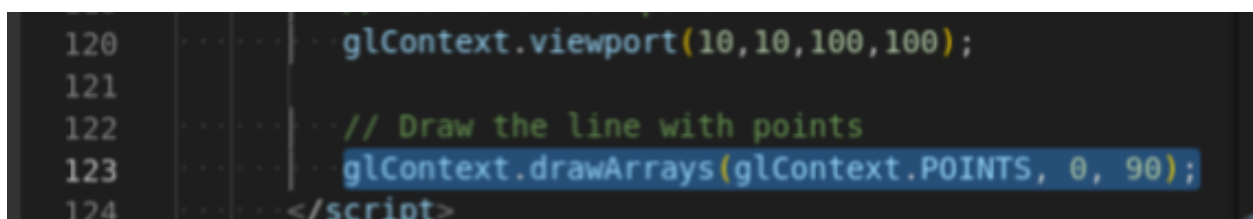


Figure 2: Draw the points