Explain each line in the following code and its effect in details:

|  |
| --- |
| <script type="text/javascript"> 'use strict'; function init() {  const canvas = document.getElementById('webgl-canvas');  // Ensure we have a canvas  if (!canvas) {  console.error('Sorry! No HTML5 Canvas was found on this page');  return;  }  const gl = canvas.getContext('webgl2');  // Ensure we have a context  const message = gl  ? 'Hooray! You got a WebGL2 context'  : 'Sorry! WebGL is not available';  alert(message);  } // Call init once the document has loaded window.onload= init; </script>  **ANSWERS** |
|  |

<script type="text/javascript"> **means** that in XHTML, all special characters have to be encoded or in CDATA section: <script type="text/javascript">

function init() start by the kernel during the execution process; the kernel panic **will** occur if the kernel **is** unable to start it.Also contain the whole data.

const canvas = document.getElementById('webgl-canvas') the **document** method getElementById('webgl-canvas') returns an element object representing the element whose id property matches the specified string 'webgl-canvas'

if (!canvas)this check if, canvas not found

console.error('Sorry! No HTML5 Canvas was found on this page') returns a message to the user when the searched item canvas is not found.  
 Return

const gl = canvas.getContext('webgl2') it returns a drawing context on the canvas, or [null](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/null) when the context identifier is ('Sorry! WebGL is not available') or the canvas has already been set to another context mode.

const message = gl depend on the const gl = canvas.getContext('webgl2') and check when it is available or not.

alert(message) gives output based on the const message = gl condition stated,

Output either ? 'Hooray! You got a WebGL2 context if available and 'Sorry! WebGL is not available' if not

window.onload= init will be executed immediately,depend on when the windows is done loading and the result will be assigned to window