Exercise Javascript Manipulating HTML Code

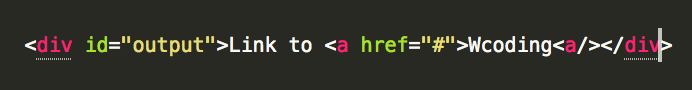
**Evaluated skills :**

*→ Manipulation of HTML Code*

**Question 1 :** *Edit an item*

Within <div id = "output"> there is a #text node and a <a href="#"> link. This link leads nowhere since it is equals to #. It should therefore change its href attribute to point to http://www.wcoding.com.

Write the code that allows you to do that.

file html : 

**Question 2 :** *Modify an attribute generically*

Within <div id = "output2"> there is a <span id = "mascot> hosting Zozor (it's a background image), but Zozor is not in a good mood ... we want him to smile by adding a data-state attribute to <span id = "mascot"> and giving it a *happy* value.

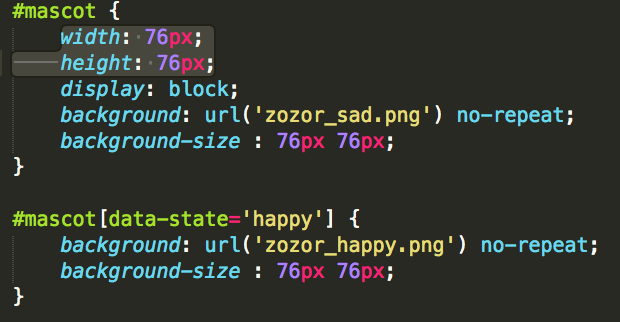
Write the code that allows you to do that.

In order to help you :

file html :



Your CSS code :



**Question 3 :** *Using innerHTML*

Within the <div id = "output3"> is a small quote from Albert Einstein. What it would take is to put this quote in a <blockquote> element, it would be prettier (because I planned a CSS style).

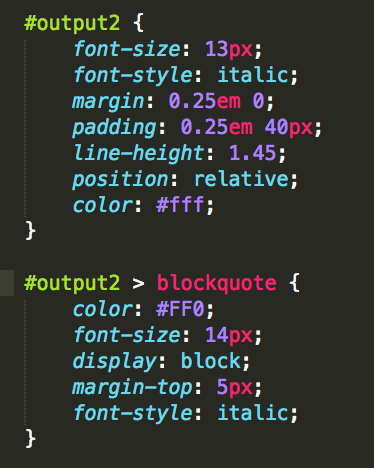
Write the code that allows you to do that.

In order to help you :

html file :



Css file :

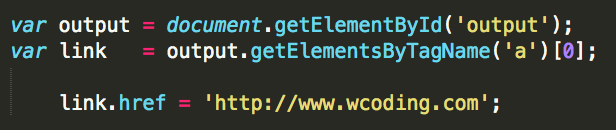


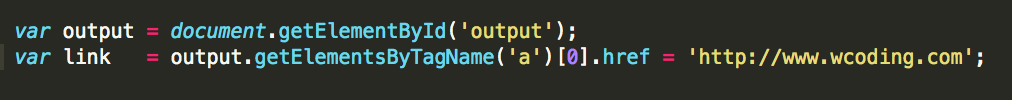
**Correction :**

**Question 1 :**

* With getElementsByTagName () :

One way to do this is to use getElementsByTagName () within div #output:

This code can be reduced in this way:



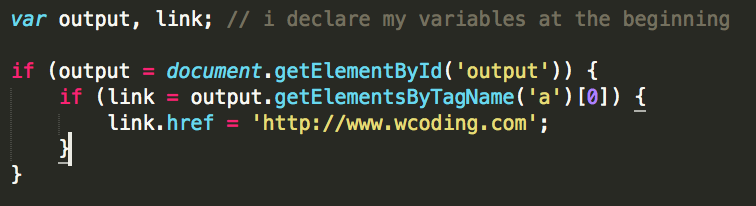
Or even like this, but it becomes unreadable :



* With getElementsByTagName () and be careful of errors

The codes above lack security, and it may be worthwhile to perform tests before issuing instructions. This is necessary if your script is for a userscript for example:

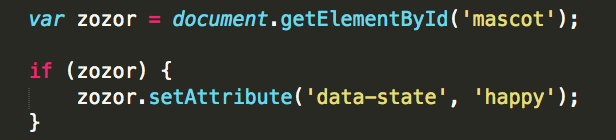
But personally, I prefer that, because I find it pretty clean:



* With querySelector ()

querySelector () can be particularly useful for this type of case, provided that your script is not intended for old browsers:

**Questions 2 :**



**Question 3 :**

