**Travel Website Implementation**

Kunika Saxena

Section 002

Contents

[Introduction. 2](#_Toc115266838)

[Web summary 2](#_Toc115266839)

[Website Navigation 3](#_Toc115266840)

[Website (HTML and CSS) Issues 5](#_Toc115266842)

[Future plans to Implement. 6](#_Toc115266843)

[Homework summary 6](#_Toc115266844)

[System implementation 8](#_Toc115266845)

# Introduction.

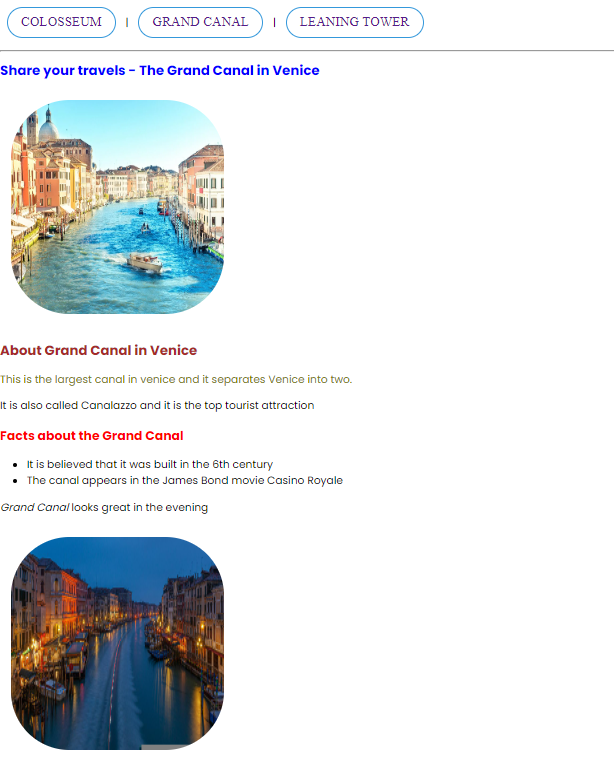
This document talks about the travel website, it gives the information about the website I designed and implemented recently. This is a travel website that gives some information about different tourist attraction sites. Currently the website has three beautifully designed webpages, the one that describes Colosseum which I assumed as my homepage, another one for Leaning tower and the other for Grand canal. This website is purely designed using HTML and CSS which in future plans I will consider improving it using some JavaScript and other front-end languages which I have described it in the future plans.

# Web summary

The website specifically talks about different tourist attractions in Italy. The website is not that big to be more specific, it has only three webpages. I saw it necessary to come up with this website because I saw it will be more important to not only tourists but any other person who will want to travel to the mentioned places in the website. The website vividly describes the Colosseum, Leaning Tower and Grand Canal. The website will educate any person who will want to get some information about the mentioned places. So, for now the web site talks about three places in the three webpages now that each and every page talk about a specific place but with time I will increase the size of the website to cover more areas.

# Website Navigation

The navigation of this website is very simple, now that it has got no logging in a user doesn’t need to be authenticated in order to use the website. If a user opens the website, he or she will land on the landing page, more specifically known as the **homepage.** Many or rather all websites have their homepages named as index, the name of the extension depends on the language used because like for php it will be named index.php. Each and every webpage has got has a navigation link at the top which enables the user to navigate to any of the other two webpages that he or she would want to read about, the user can always move to any of the pages by clicking on the buttons at the top of the webpages. The navigation bar is shown below:



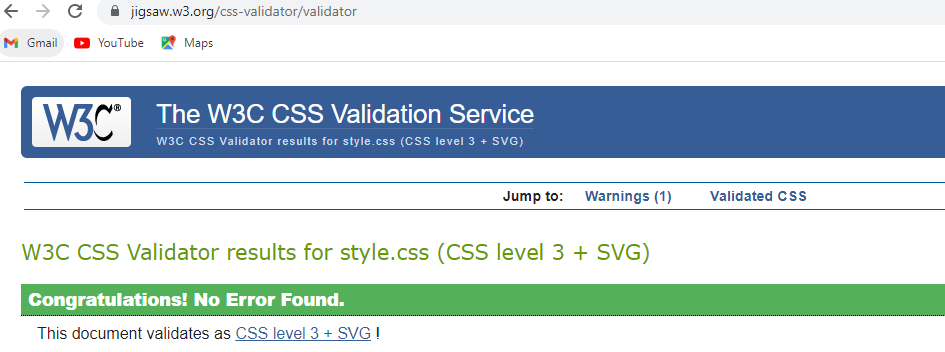
The entire website is contained in one folder known as “travel website” inside the folder there is another folder for images containing the images. There is CSS file known as “style”, index file, grand canal file and leaning tower file. Image folder contains five images. As shown below:

# 

# Website (HTML and CSS) Issues

Personally, I did not find any issue when coming up with this website. I also validated the webpages and I did not find any error or warning. I tested my CSS file and there was no error.

For screen size, the webpages auto adjusted to the new screen size, if reduce or increase the size of the screen the items on the webpages just adjusted to the new screen size.



After adding the CSS style to the website, I tried to increase and decrease the website size, and it is so fascinating that it never affected the website, the arrangements of items and elements adjust themselves to fit in the new screen size. I have not tried it on an android device but in future I will try and make it usable in any device.

# Future plans to Implement.

First of all, I will add user authentication or rather login page to the website so as a user would be required to log in before using the website. Secondly, I have only used html and CSS to design my front end, am planning to come up with some JavaScript to make the website more stylish and more responsive. Thirdly, the website contains only three webpages am planning to expand and make it larger, to accommodate more webpages and for it to contain more information about different tourist attraction sites. Currently I have just used normal CSS to style the site and to make the images responsive, in the future I will use advanced CSS such as flexbox to make entire website responsive to any device a user will use.

# Homework summary

As a beginner and someone who has interest in web development, I found the assignment to be more interesting, I have interacted with different topics in web development especially the front-end part; designing, styling, making different webpages interact and also webpage responsiveness.

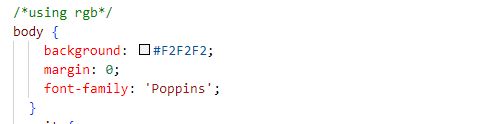
So far from my experience in using CSS, it is very much important in styling the website and making different websites elements responsive in different devices since nobody can use a website which is not attractive, hence it is more important at making website usable by many people because its ability of changing the appearance of the website. CSS is easy to use but challenging, in that, CSS does not give any error at execution, in case of any error it will always run, but the changes you have applied in the webpage will not be seen. CSS is simple in that it uses simple language; it has simple algorithms to be used to design and style a webpage.

Both CSS and HTML are more important in web development, HTML describes the structure and CSS styles them, from my point of view, HTML is somehow difficult as compared to CSS, one has to be more careful with HTML because that is the skeleton to website that one is designing.

# System implementation

1. **Implementing the RGB colors**

I Implemented the RGB using 6 parameter hexadecimals. The first two is for red middle two for green and last two for blue. A good example is as shown below:



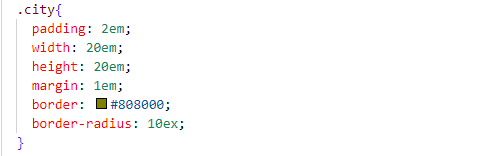
The hexadecimal in the figure above #F2F2F2 represents the RGB color for the body.

1. **Ways to access CSS selectors**

There are 3 ways by which one can access CSS selectors, by class, by id and finally by html TAG.

1. **By class.**

I gave all the images a common class “city” in my CSS I accessed all the images through the class name city and all of the used the same CSS format. As shown below:

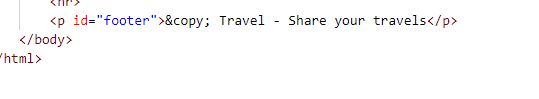


All the IMG used the same class hence they had the same formatting features.

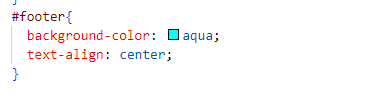
1. **By ID.**

IDs can be given to DIVs and any other element or selectors, IDs help when one wants to use for instance JavaScript, he or she can get the element by the unique ID are accessed using # symbols

I gave the copyright paragraph an id footer then formatted it using CSS

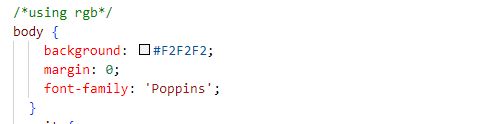


In CSS IDs are accessed using # symbol



1. **By HTML tag**

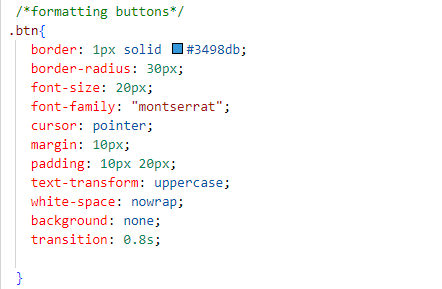
HTML tags like paragraph (p) and hyperlinks (a) can be also used as CSS selectors to format a particular part of a website directly.



The body tag has been used to format the entire body of the website.

1. **Box model styles.**

I used different box model styles. I.e., padding, borders, margin, border-radius, height and width as shown below:



1. **CSS types**

I used the embedded CSS, inline CSS and external CSS

1. **Embedded CSS**

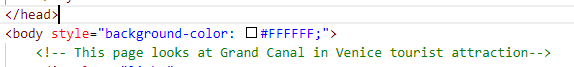
It allows you to define styles for a particular HTML document as a whole in one place. This is done by embedding the <style></style> tags containing the CSS properties in the head of your document.



The style tag is embedded in the head tag so it can access any tag all through the webpage.

1. **Inline CSS**

An inline style may be used to apply a unique style for a single element. To use inline styles, I added the style attribute to the relevant element.



Inline CSS only formats the selected element and it always picks first instance before all the other forms of CSS.

1. **External CSS**

An external style sheet is a separate CSS file that can be accessed by creating a link within the head section of the webpage in the href. Multiple webpages can use the same link to access the stylesheet. The link to an external style sheet is placed within the head section of the page.

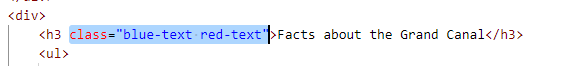


Related webpages can use the same external CSS provided the naming and the way the external CSS has been included is correct.

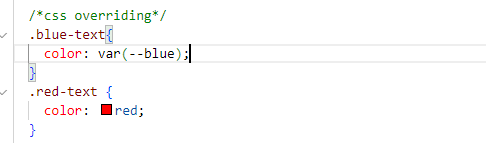
1. **CSS overriding property**

In overriding the most important part is always executed, in CSS the part that comes later is always assumed to be the most important and that is what the browser will pick from CSS as shown below:

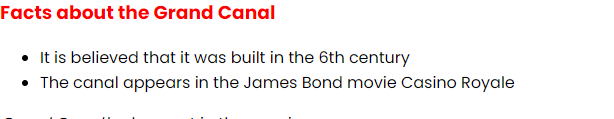
The element is given IDs



Then the IDs is formatted differently in CSS

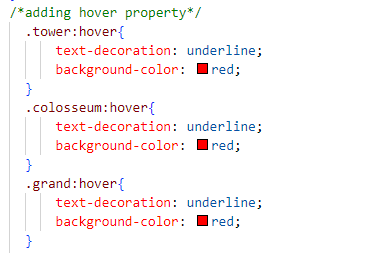


But the first one is overridden and the browser picks the latest one



1. **CSS comments**

Comment is that part of the program which is not executed when the program runs, comments just explain what that part of the program does for any reader to understand what exactly is happening.



The comment explains that the program is actually adding hover property on the buttons, and as you can see, that part is not executed that is why the CSS has shown no error.

1. **Views in different Layouts.**

The site adjusts basing on the size of the screen, as you continue to enlarge the screen size, the site adjusts to fit to the size given and vice versa.

1. **Overflow property.**

The overflow property specifies what should happen if content overflows an element's box. This property specifies whether to clip content or to add scrollbars when an element's content is too big to fit in a specified area. The overflow property only works for block elements with a specified height.

There are **seven** overflow properties:

**Visible** - The overflow is not clipped. It renders outside the element's box. This is default

**Hidden** - The overflow is clipped, and the rest of the content will be invisible. Content can be scrolled programmatically (e.g. by setting scroll Left or scroll To())

**Clip** - The overflow is clipped, and the rest of the content will be invisible. Forbids scrolling, including programmatic scrolling.

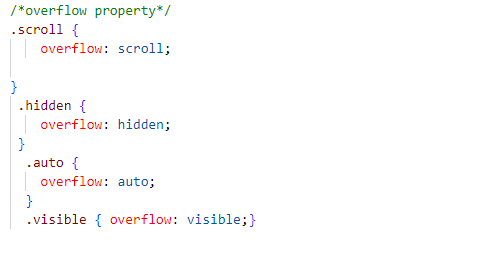
**Scroll** - The overflow is clipped, but a scroll-bar is added to see the rest of the content

**Auto** - If overflow is clipped, a scroll-bar should be added to see the rest of the content

**Initial** - Sets this property to its default value.

**Inherit** - Inherits this property from its parent element.

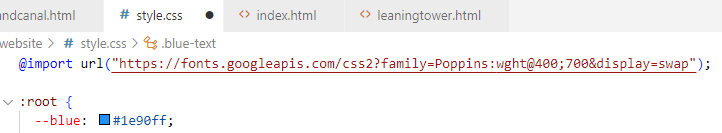
I managed to implement some of them but unfortunately the site does not have a lot of content to overflow.



1. **Font property**

Most of the browsers don’t come with fonts, so at times you may be required to come up with different fonts which may not be available in the browser, for that matter the browser will need to download the font once the webpage is launched.

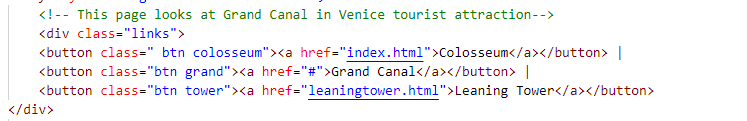
In my CSS file I added a link to download fonts, if the browser cannot load the font it will download it from the link and load it on the site.



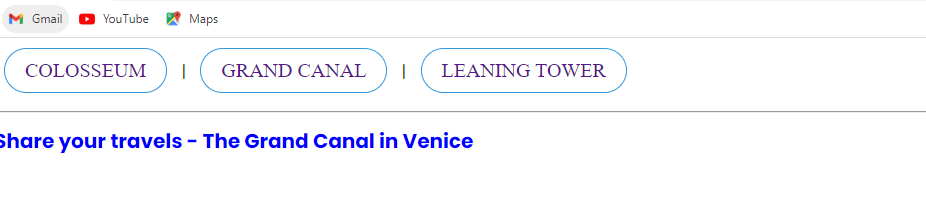
The site loaded the font from the link and as you can see the website uses **Poppins** font family.

1. **Hyperlinks to Buttons.**

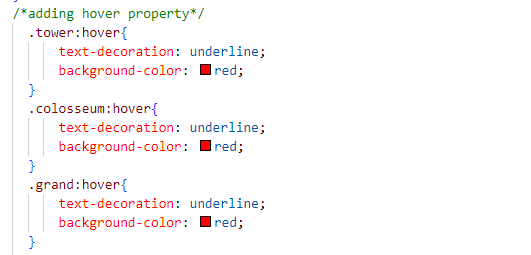
I converted the hyperlinks in to buttons, so once clicked it opens a different webpage.



So, the webpages displayed the hyperlinks in form of a button.

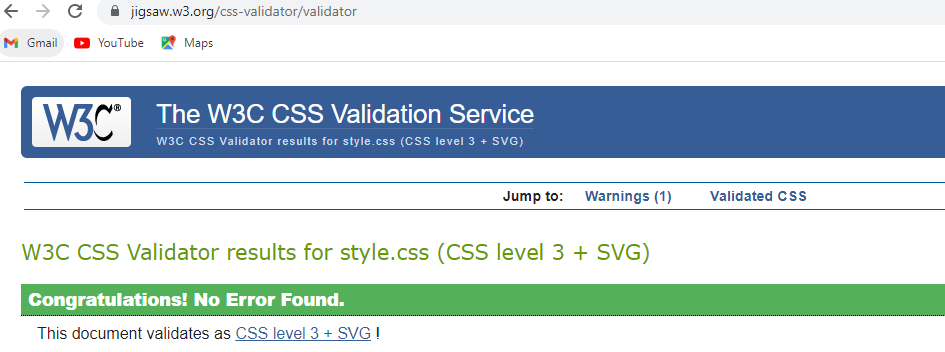


I implemented hover function such as when the mouse pointer hovers on top of the links, the links underlines and color changes to red

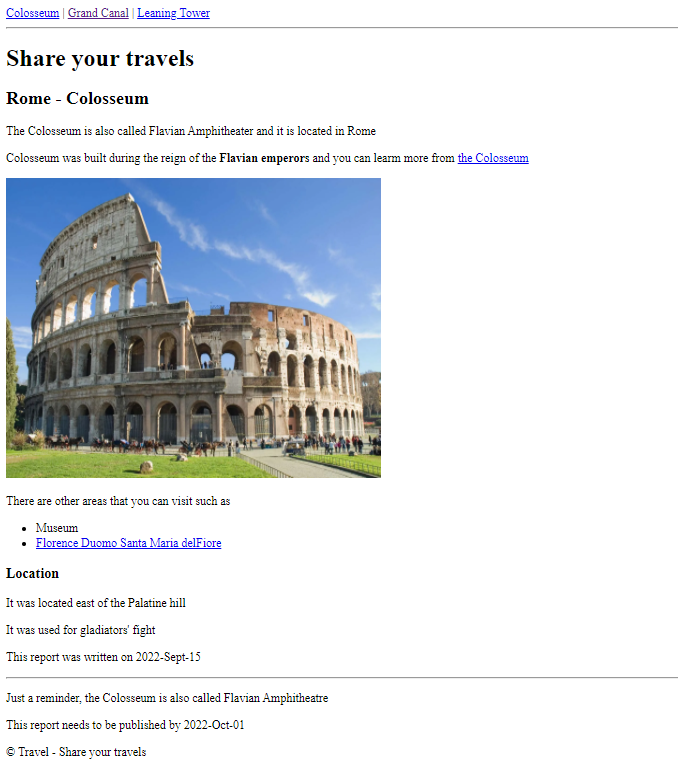


1. **Checking for errors using CSS validator**

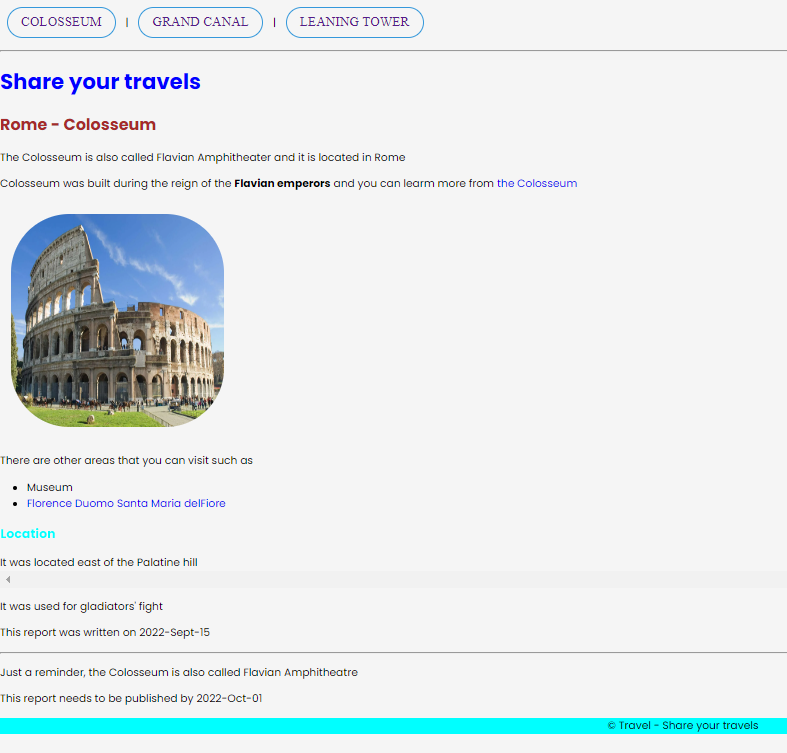
I pasted the CSS file on the link provided and luckily, the CSS file had no even a single error.



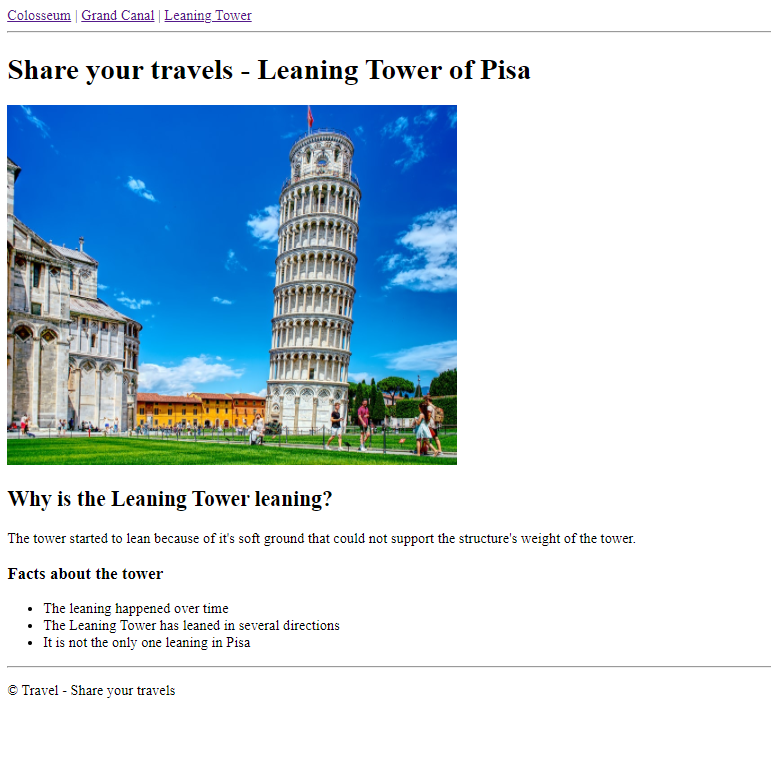
**COLOSSEUM BEFORE**



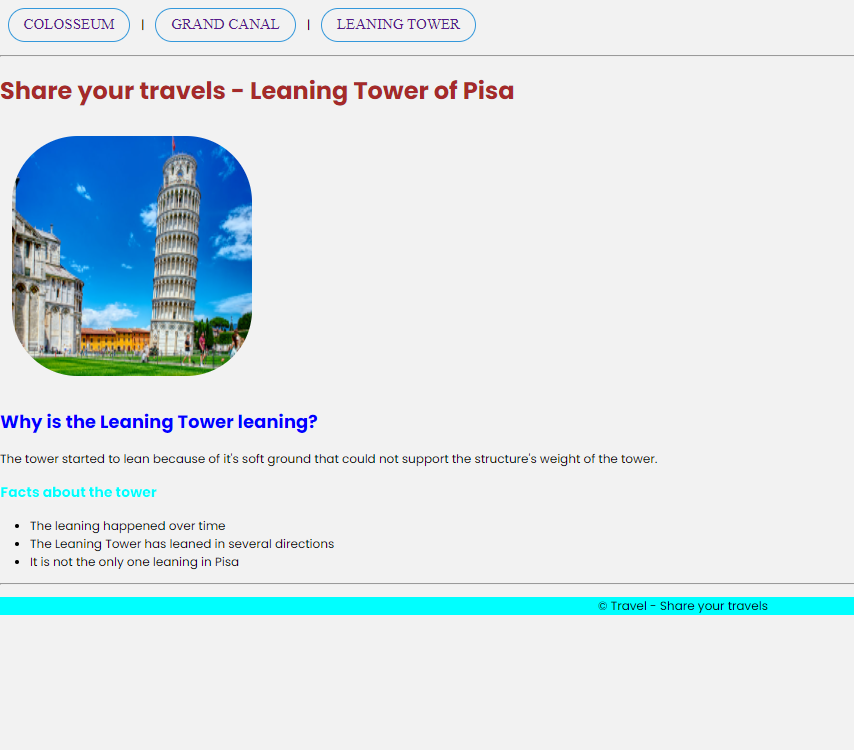
**COLOSSEUM AFTER**



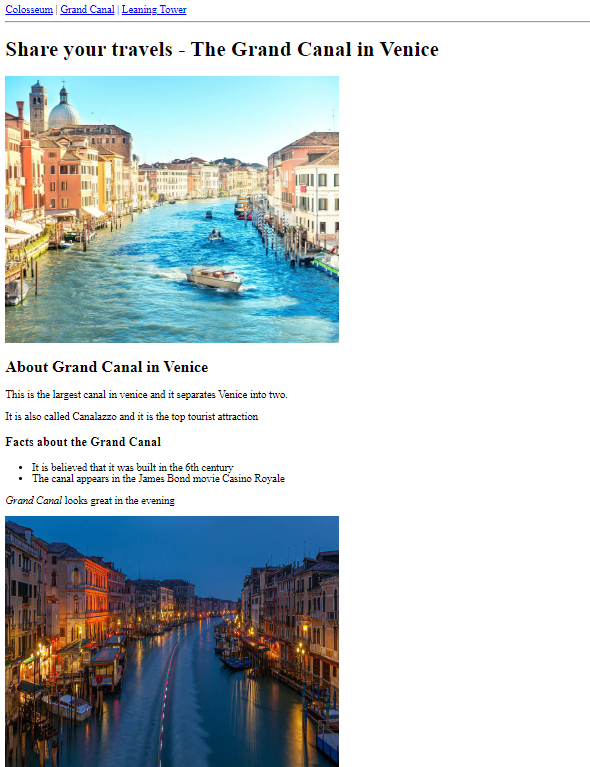
**LEANING TOWER BEFORE**



**LEANING TOWER AFTER**



**GRAND CANAL BEFORE**



**GRAND CANAL AFTER**

