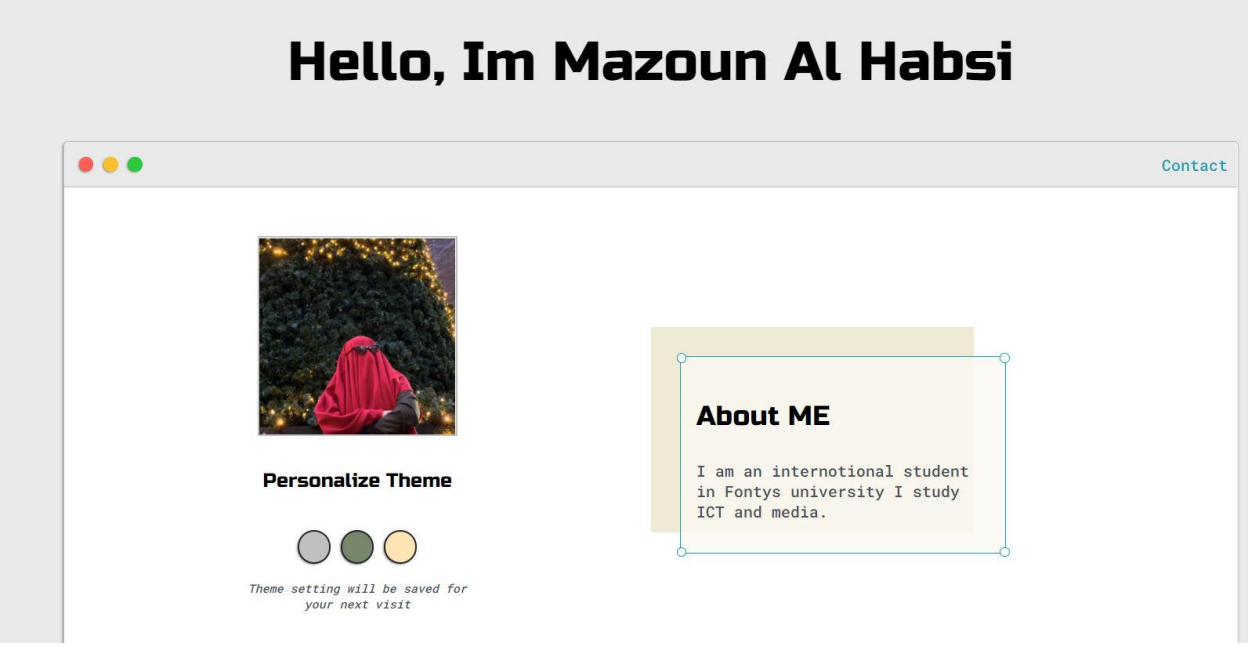
Front End-Develobment

PITI REPORT

Personal Portofolio Website



Introduction

The very first website I made is a personal website about me, my likes and dislikes, hobbies, social accounts, pictures, posters and etc... I followed an online tutorial on how to do a personal portfolio with some Javascript codes. I thought the idea that was the idea.

The second website I made is a gallery website. All images on the website were taken by me. I intended at first to make it as a website of my country, but then I changed the idea to show just galleries. The next website I will be doing will be about my country, and since I like to do two birds one stone on most of my work I will publish the website if it feels great.

Finally, I will make all websites connected to my personal website. However, I will not publish all websites. What's more, I will be updating all the websites in the long run.

W1.Personal Portfolio

I wasn't too sure about this website. I had the wrong idea that we have to do a personal website about ourselves. However, since I haven't done this in the first semester I can say I learned a lot by doing it.

What I have learned on this website:

- Grids and Flexbox
    - Grid: I found this website where it explains a lot about gride columns, gride rows, grid area, and the most important grid template area. The grid template area was a key point on the responsivity of the website, it is easy to specify the locations of the element with this code. The link of the website: https://css-tricks.com/snippets/css/complete-guide-grid/
    - Flex: On the same website I found a good explanation about the flexbox as well. What I understand from these CSS codes that we can adjust items on a freeway, not like the grides. I also learned what's the meaning of the parent element and the child element. The link to the website: https://css-tricks.com/snippets/css/a-guide-to-flexbox/
    - I learned that in order to use Flix or Gride we have to specify the display first as (display: flex;) or (display: grid;).
- JS theme edit and library
    - I learned that we connect the javascript codes as the way we link the CSS.
    - I learned that we can get data from the library of the page and use it. It is much similar to reading an Arduino chip but instead, you read user clicks.
- Shadows, and Drawing on the website
    - I learned that we can make a lot from HTML and CSS codes. At first, I thought it is not that deep and we need to use other languages to do these tricks like the 3 colored circles and the similar apple page.

- Text highlight, Opacity, and Rgba
    - I learned that it is easy to this thing on the website and the apportonites of using Rgba. I learned that the Rgba allows you to choose the color and the opacity at the same time.
- How to make a downloadable document
- How to link social accounts and open them on another window
- Use one CSS file for the whole website
- Variables use

Finally, I only watch one youtube tutorial to learn all of these things. I should say that I learned many things from creating this website. The link to the tutorial: https://www.youtube.com/watch?v=r_hYR53r61M&list=LL&index=33&t=5628s

W2.Gallery website

The idea of this website is to make a fully hover gallery website. I used javascript in every page I made. I followed many youtube tutorials to get the best of this website. I learned many new things in this phase.

What I have learned on this website:

- Connecting css or js or img on a different folder.
    - Using ../foldername/folder name/img.jpj
    - Using background: url(..); on css and so on
- Creative hover Text Scrolling
    - With using js, classes and the overflow codes. What's more, in js I had to spacifiy the needed time for both the inner text and the text that shows on top of the video background.

- 3D Rotating Image
    - I saw a youtube tutorial to do this. However, when playing with some codes and numbers I found that I can make different animations at a specific time and then change the animation speed and direction.
    - This action was possible by using the (@keyframes), specifying the perspective pixels and the rotation degree.
    - I add the image shadow reflection as well using (transform-style: preserve-3d;)
- Responsive Image Card with vanilla tilt js
    - On these steps, I did not code the js but I included a library to change a bit. I fixed the CSS to my needs and add as many images as I wanted. I used the flex display option to align the aliments on the page.
- Use classes on js
    - I learned that we can call the classes from the HTML codes to the js codes by adding a dot in front of the class name. (".calssName")
- Make a glassy layer, a video background and implement it on all pages
    - Specifying the height, the width, the position, and the backdrop filter allowed me to do this step. For the rest of the pages I had to use z-index codes to specify what comes first and what comes last, this can be seen mostly on the Masfat Alabreen page(M.A.html).
- Position
    - When specifying the position of text I understand that sometimes the text comes on the position that I need but it does not start from the place it should therefore I had to use this code (Transform: translate(-50%, -50%);).

- I learned that in CSS we can make all text uppercase or downcase by using this code (text-transform: uppercase;).
- Show different image when clicking to view more
- Flex-wrap
    - I learn that it sets whether flex items are forced onto one line or can wrap into multiple lines.
- I used new things like Section, span, transform, TranslateY, and more on CSS. I used scroll, remove, add, event listener, queryselector, and reading from the window.

Finally, I tried many youtube tutorials however not all of them worked. For the main page, I saw how to add the background video, the glass layer, and the js animation for the glassy layer and the speed transform. The link to this tutorial is: https://www.youtube.com/watch?v=r_hYR53r61M&list=LL&index=34&t=5629s

The link for the 3d rotating images: https://www.youtube.com/watch?v=j1-Ak3WWV_g

The link for the transform scroll effect: https://www.youtube.com/watch?v=LFY3e53wBPw

The link for the vaniel tilt js: https://micku7zu.github.io/vanilla-tilt.js/

The feedpluse document is on my personal portfolio.

W3.Three.Js website

In the third phase, I learned a lot. I made more exercise than in previous phases, did more research, and investigated deeply. In this phase, I made a 3d website where users can navigate it using keyboard keys. I started it by reading the three.js website and following the exercises, then I moved to youtube videos. After which I tried to navigate for the already existing website and try to change the codes to my needs. I decided to follow these plans for many reasons and one of them is that I understood the basics of three.js. What's more, I had to use Wampserver app on most of the websites that I made. I understand that it is for security reasons.

I faced many issues, the biggest one that took most of my time is the one connected to CORS policy. At first, I thought that I had missings codes or errors that did not allow the 3d objects to appear on the pages, I tried to change the codes, change the gltf files, tried different youtube tutorials, looked at many StackOverflow articles and problems, and finally decided to ask the teacher. Finally, I solved the problem by using the Wampserver app, add all the websites to a file called www in the app files on my laptop and finally, all objects appeared and I was able to continue. I learned that the 3d objects on three.js are called mesh, it was easy to search for documents and solutions using this keyword.

Learn progress:

1. I first started with making a line, a text, a cube, and then to the animation and implement it on those exercises.
    a. We have to add these variables in each script:
        i. Scene
        ii. Camera
        iii. Renderer
        iv. Material
2. I then tried to show the gltf objects which I can get from the sketchfab website.
    a. I used different three.js sources to do this:
        i. <script src="three.min.js">
        ii. <script src="OrbitControls.js">
        iii. <script src="GLTFLoader.js">
    b. I also had to use these variables:
        i. Loader (to load the gltf objects)
        ii. Controls
        iii. Abint
3. I moved then to changing the image of the cube. I learned that we can specify if we want to display the image on both sides, on the outside, or on the inside only. However, I did not implement that on the final website because I needed the image to show as an image with frames and located in different areas.
4. Then I moved on to coding the background, I learned that I need many pictures if I want the background to look like it is moving with the user.
5. Next, I moved to coding textures without using the gltf and it worked for some websites without using the local server.
6. Finally, I moved into finding already exciting websites and change their codes to my needs. These steps took a lot of time to understand the codes but I was fast when changing the codes after I understand them.

The links that I looked at to learn :

- Three.js website: http://threejs.org/
- Information about the texture: https://threejsfundamentals.org/threejs/lessons/threejs-textures.html#format
- Change background color in three.js: https://discourse.threejs.org/t/change-background-color/768/2
- Different kind of background (interactive background): https://threejsfundamentals.org/threejs/lessons/threejs-backgrounds.html
- YouTube tutorial (Three.js Tutorial 5 - Textures & Colours) : https://www.youtube.com/watch?v=l77yAZ0E950
- YouTube tutorial (3D Model Loading in Website using Three.js): https://www.youtube.com/watch?v=JUwnSK163zs
- Sketchfab website: https://sketchfab.com/search?sort_by=-relevance&type=models
- Three.js Tutorial – How to Render 3D Objects in the Browser: https://www.freecodecamp.org/news/render-3d-objects-in-browser-drawing-a-box-with-threejs/
- Creating custom three.js shapes: https://stackoverflow.com/questions/17843562/three-js-create-custom-3d-shape
- https://javascript.plainenglish.io/javascript-in-3d-an-introduction-to-three-js-780f1e4a2e6d

W3.Three.Js website-PART2

In this phase, I continued working on the last website or on the last page I made. I made the 3d world look more alike to a museum where I add walls, floor, and a roof all into 2 rooms. I studied positioning more in-depth, I'm able to position mesh elements or any texture wherever I like. I stumbled with a lot of issues with the video, one of them was autoplay police where the user should interact with the website before this is able to work. I solved it by adding a start button to run the video. Here is what I studded and was able to discover during this phase:

1. Mesh elements(floor, roof, blocks)

   I was able to play around with the sizes with this code "new THREE.BoxGeometry(4.3,8,0.1)" it can specify the width, the height, and the depth. I had to create many elements with both negative and positive numbers.

2. Positioning(yxz)

   I learned how to position elements after showing them. To show an element or a mesh object I had to type " scene.add("object");" and to specify the place of the object I had to use the positioning. I will call the name of the object first and then positon and then specify where do I want to place it or to which direction x,y,z. This was made with this code " crate2.position.y += 4;". After positioning I had to fix some walls or roof or floor size where I had to go back to point 1 and again fix the position many times.

3. I learned that I have to specify if I want the objects to receive the shadows and cast them by using these codes " txt2.receiveShadow = true; txt2.castShadow = true;".

4. Alongside moving camera direction I also learned how to move up and down by first finding arrows numbers and then specifying where do I need the camera to go by choosing x, y, or z and then specifying the speed an example of this code is

   " if(keyboard[38]){ // up arrow key

   camera.position.y += Math.sin(camera.rotation.x - Math.PI/2) * player.turnSpeed:} "

5. Ambition light

   I wanted to allow the user to add or remove light, I had the idea after knowing the use of the ambition light and knowing how to specify the value. I tried to do it with white color but it only worked with blue. Why not blue! However, I add the white color as well. To add white color I used this

   " abint = new THREE.AmbientLight(0xffffff, 0.2); scene.add(abint); "

   To add and remove the blue color I used this where I had to contact an expert with three.js through discord. I understand the code and what does it do but I still need to figure one or two things.

```
"let brightness = 0;

const abLight = new THREE.AmbientLight( 0xffffff );

const addLight = () => {

  brightness = Math.min( brightness + 16 , 255 );

  abLight.color.set( brightness, brightness, brightness );

  scene.add( abLight );}

const removeLight = () => {

  brightness = Math.max( brightness - 16 , 0 );

  abLight.color.set( brightness, brightness, brightness );}"
```

6. To add a video and make it run I had to follow the three.js page first but it did not work. I had to do many researches and run many examples to make it work. I was able to do this page for an example: https://sad-newton-e693f8.netlify.app/ .I stumbled with the first issue which I will describe in point 7.Furthermore, I was able to do run a video by adding a button as I described previously. I learned how to add a button and make it do something with js. I used these codes and then was able to add buttons to the lights.

   On the html:

   ```
   <button onClick='startVideo()' style="position:absolute;top:5px;left:500px; ">Start Video</button>

   <div id='PlayVideo'></div>
   ```

   On the JS:

   ```
   const startVideo = () => {

                document.getElementById('PlayVideo').innerHTML = '<video id="video" loop autoplay style="display:none"><source src="2.mp4"></video>';

          //adding video codes }
   ```

   For the buttons alone:

   ```
   Const nameOfTheFunction = () => {

   //Some codes

   }
   ```

   I was able to learn this from an expert with three.js as well. This is because I was not able to figure out why the video does not work. I figured out that my codes weren't wrong but because of the policy. The link to the policy: https://developers.google.com/web/updates/2017/09/autoplay-policy-changes.

7. Adding Video (progress example and policy)

   First of all, I followed a three.js tutorial on how to run a video using three.js. The tutorial: https://threejs.org/docs/#api/en/textures/VideoTexture. This tutorial did not work so I searched for an example on the same page and was able to make this: https://sad-newton-e693f8.netlify.app/ . At first, I was working with youtube videos and this is where I stumbled with the first issue where I had to add /embed/ on the URL to allow me to use youtube videos. Like this(https://www.youtube.com/embed/VIDEO_ID) this was the error **https://forum.freecodecamp.org/t/youtube-refused-to-connect/245262**

8. To display video (iframe.src, I need to specify each frame)
9. object.rotation.y = ry (The result of this transformation is that V (= Rotation axis) is coincident with the z axis.)

   https://www.cs.helsinki.fi/group/goa/mallinnus/3dtransf/3drota.htm

10. Overall, I learned a lot from this project. However, I learned more when working on the team project where I am proud to say that I can work very easily with CSS and media CSS to make websites responsive and play around with object positioning. What's more, In this last phase I sometimes thought that I am good and sometimes i was not doing great. Finally, I was able to acknowledge that I need to learn javascript more in-depth if I want to make a nice 3d world or work with any js libraries.

Some useful links I found:

- Youtube embed to allow to use the videos **https://developers.google.com/youtube/player_parameters/#Manual_IFrame_Embeds**
- **Specifying the ambient light value https://stackoverflow.com/questions/16319096/three-js-change-light-intensity-dynamically**