APPLYING BOOTSTRAP

What is our GOAL for this MODULE?

We learned to make the e-portfolio website responsive while applying bootstrap classes to all the elements of the website thus developing advanced UX design skills.

What did we ACHIEVE in the class TODAY?

- Made the personal e-portfolio website mobile responsive.

Which CONCEPTS/ CODING BLOCKS did we cover today?

- Using Bootstrap to make the portfolio website responsive.
- Using predefined CSS properties and Javascript functions in Bootstrap.
- Building an awesome menu bar.

What did we REVISE today?

- Bootstrap is used for building efficient and responsive websites. Responsive means, the website will look proper in all devices - from bigger screens of the laptop to a small screen of mobile.
- BOOTSTRAP has predefined CSS properties and Javascript functions. We just need to call them inside the class of the element once it is imported into index.html. This is exactly like how we import javascript files.

How did we DO the activities?

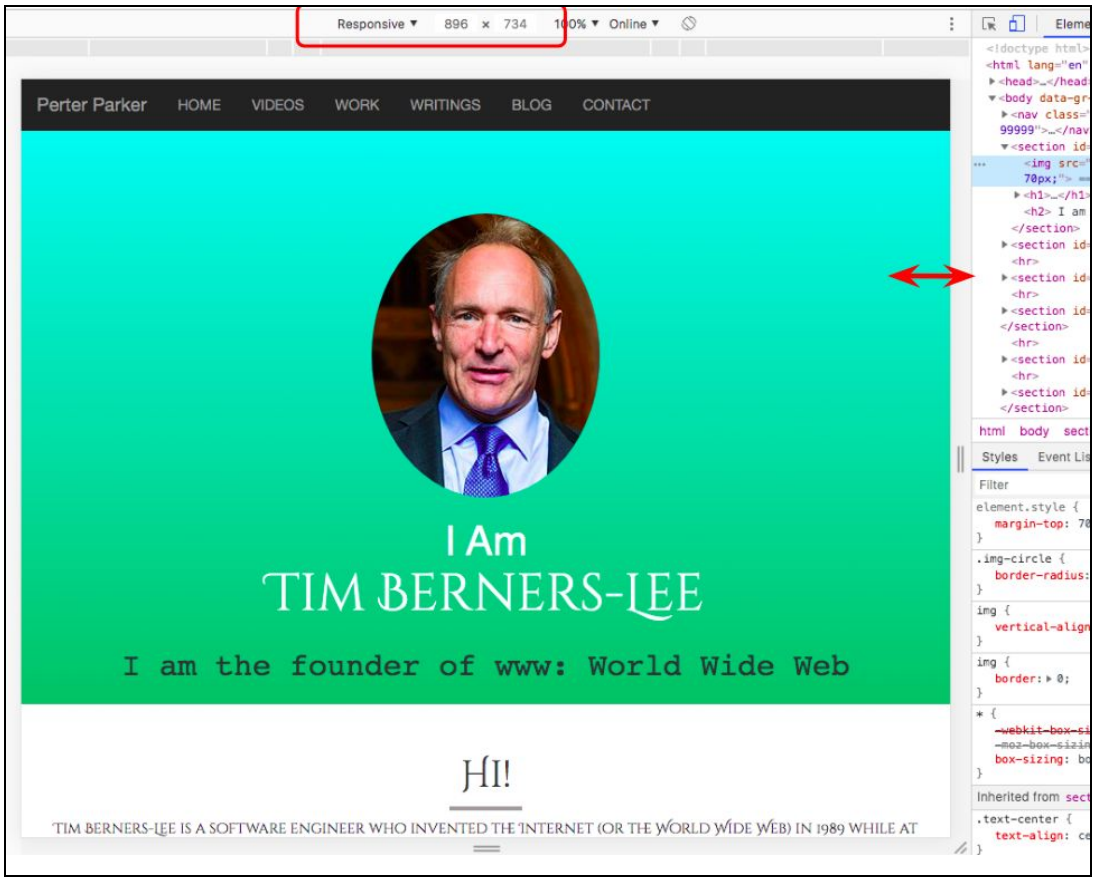
Continue in the website created in lesson 62. Refer to Student activity 3 for the file.

Screen resolution means the number of horizontal and vertical pixels on a display screen. For example:

- On laptop: screen resolution - horizontal 1200px and more
- On tablets: screen resolution - between horizontal 992px to horizontal 786px
- On mobiles: screen resolution less than horizontal 786px.

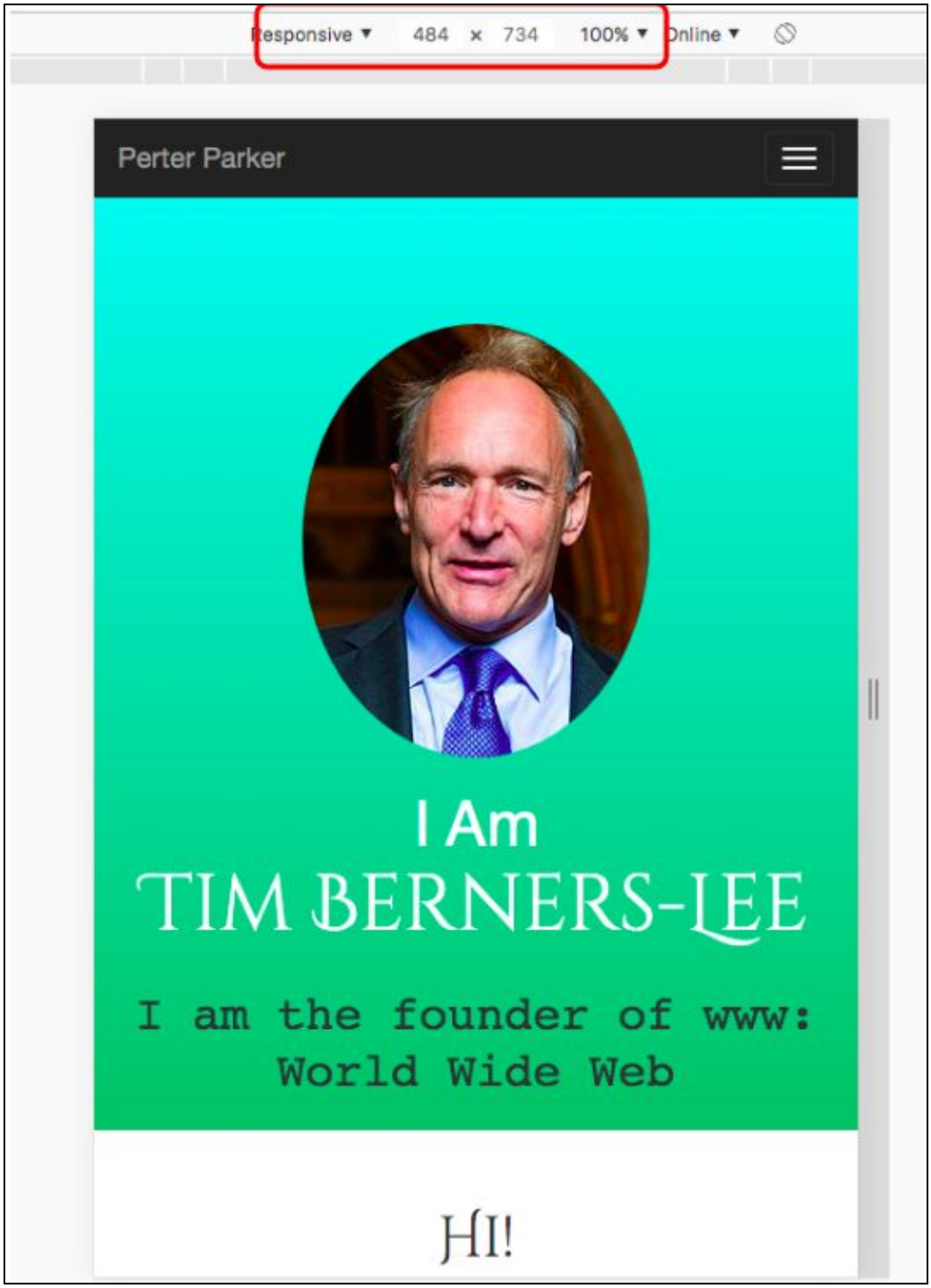
Steps to check how the website looks on different screen resolutions.

1. Open the website in the browser https://jynyhy9vu5r8xz5zy0utww-on.drv.tw/www.ADV-C68.com/ , right click on the screen and click on inspect element.



2. Then this screen is open and you can change the horizontal and vertical pixel's values from the input box as marked on the image or resize the screen as per marked on the image.

For example:



- I have mentioned horizontal value as 484px so you can put any value and see how it looks on what screen resolution .

Use the files from the Student Activities and continue with the following:

Menu Bar

```
<nav class="navbar-inverse my_nav_bar">
    <div class="navbar-header">
      <button type="button" class="navbar-toggle" data-toggle="collapse" data-
      target="#myNavbar">
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
      </button>
      <a class="navbar-brand" href="#">Perter Parker</a>
    </div>

    <div class="collapse navbar-collapse" id="myNavbar">
      <ul class="nav navbar-nav">
        <li><a href="#welcome">HOME</a></li>
        <li><a href="#videos">VIDEOS</a></li>
        <li><a href="#work">WORK</a></li>
        <li><a href="#writings">WRITINGS</a></li>
        <li><a href="#aboutMe">BLOG</a></li>
        <li><a href="#contact">CONTACT</a></li>
      </ul>
    </div>

</nav>
```

```
<nav class="navbar-inverse my_nav_bar">
    <div class="navbar-header">
```

1. <nav> is a tag which is used to wrap all the menu items inside it.
    - It uses **navbar-inverse.** It is a bootstrap css property. We are using it by calling it inside the class of the element. It is used to give the background-color and font color to the menu bar.
    - We have **my_nav_bar.** We can now give some style to **my_nav_bar**

```
.my_nav_bar
{
    position:sticky;
    top:0;
    z-index:99999
}
```

- **position:sticky -** It means that the menu bar will get sticky on the position where it is placed.It will always be on the top of the screen even when you scroll down the page.

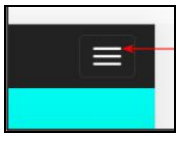
- **top:0 -** This means it will start from the top of the website.
- **z-index:9999** - This means it will always be on the top of any other element. If we scroll, there are images and various other elements. We do not want these other elements to be seen above the menu bar when we scroll.

```
<div class="navbar-header">
```

2. In the div we used class **navbar-header.** It's a bootstrap css property and we are using it by calling it inside the class of the element.
    - The navbar-header is mostly an architectural class for Bootstrap navbar. It allocates approximately 150px to the left of the navbar to wrap the navbar-brand.
    - But the most useful property of the navbar-header is its responsiveness to 100% width till 768px (tablet views). It also helps to get the menu item on the left hand side.

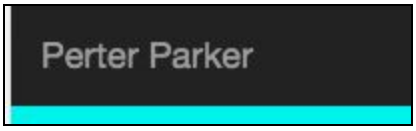
```
<button type="button" class="navbar-toggle" data-toggle="collapse" data-
target="#myNavbar">
  <span class="icon-bar"></span>
  <span class="icon-bar"></span>
  <span class="icon-bar"></span>
</button>
```

3. Button
    - **navbar-toggle:** It is a bootstrap css property and we are using it by calling it inside the class of the element.
        - This helps to style the button like  and it helps to show the button only in tablets and in mobiles. This means the screen resolution is less than 768px horizontally.
    - **data-toggle="collapse"** tells the button that there is something that will be opening when the button is clicked.
    - **data-target="#myNavbar"** tells the button that when the button is clicked, the div with the id = myNavbar should open.
    - Inside the button there is a span with class **icon-bar.** This is also a bootstrap css property and we are using it by calling it inside the class of the element.

      This we use to get these three lines  .

```
<a class="navbar-brand" href="#">Perter Parker</a>
```

4. **Anchor tag** is used so that if someone clicks on the name of the website, it should open the home page. For now we have added **'#'** (hashtag), but the student needs to replace the # with the website link of his/her hosted website after building it.
    - **navbar-brand**: This again is a bootstrap css property and we are using it by calling it inside the class of the element. This is used to give **font color, height, margin, font-size and padding** to the name of our website.



*Note: If you observe, we are just giving the class name and those elements are looking good, CSS properties are added automatically. All this is because of the Bootstrap predefined CSS properties.

```
<div class="collapse navbar-collapse" id="myNavbar">
```

5. **Div** - This is used to hold all the menu items.
    - **collapse**: Another bootstrap css property and we are using it by calling it inside the class of the element. This property asks the div to hide all the items when the screen resolution is in tablet or mobile zone (less than 768px horizontally).
    - **navbar-collapse** - One more bootstrap css property and we are using it by calling it inside the class of the element. It is the whole and soul for the style and display of the menu items. It adds background color, and shows the big menu as this one, when the screen size is above 768px. It hides when the screen is below 768px.

    - **id="myNavbar"** is the identification of the div, that opens this div when the element that has data-target="#myNavbar" is clicked. Currently that element is the button.

```
<ul class="nav navbar-nav">
  <li><a href="#welcome">HOME</a></li>
  <li><a href="#videos">VIDEOS</a></li>
  <li><a href="#work">WORK</a></li>
  <li><a href="#writings">WRITINGS</a></li>
  <li><a href="#aboutMe">BLOG</a></li>
  <li><a href="#contact">CONTACT</a></li>
</ul>
```

6. **ul** is an **unorder list** which we will use for listing the menu items.
    - **nav, navbar-nav -** Bootstrap css properties that we are using by calling it inside the class of the element. These both are used to align and style the list items.

7. Then we have **li** which is a list **item**. These are the menu items with an anchor tag with its respective div.

Contact Section

Html:

```
<section id="contact">
    <iframe src="https://www.google.com/maps/embed?
    pb=!1m14!1m8!1m3!1d15078.353856082635!2d72.8906491!3d19.1257017!3m2!1i1024!2i768!4f13.1!
    3m3!1m2!1s0x0%3A0xacf72f56793d190a!2sWHITEHAT%20EDUCATION%20TECHNOLOGY%20PVT%20LTD!5e0!3
    m2!1sen!2sin!4v1573468987088!5m2!1sen!2sin" height="450" frameborder="0"
     allowfullscreen=""></iframe>

    <div class="contact_1" >
           <a href="mailto:my@gmail.com?Subject=Hi">
               <h5><i class="fa fa-envelope"></i>  mygmailid@gmail.com</h5>
            </a>

           <a href="#">
               <h5><i class="fa fa-instagram" ></i>  myinstagramid</h5>
            </a>
        </div>

    </section>
```

1. For making the social media links responsive we just need to add some bootstrap classes to the contact_1 div and everything of HTML remains the same.

```
<div class="contact_1 col-lg-6 col-md-6 col-sm-6 col-xs-12" >
       <a href="mailto:my@gmail.com?Subject=Hi">
           <h5><i class="fa fa-envelope"></i>  mygmailid@gmail.com</h5>
        </a>

       <a href="#">
           <h5><i class="fa fa-instagram" ></i>  myinstagramid</h5>
        </a>
    </div>
```

2. We have added some bootstrap class to this div for the responsive width. By using bootstrap we can make responsive website layout.
    - Bootstrap divides the screen into 12 columns. That means if 100 is the full width of the screen and 12 is the number of columns used by bootstrap to divide the screen, then if we divide the full width of the screen which is 100 with the bootstrap columns which are 12 the -
      100/12 = 8.33. This means each column value will be 8.33.

3. Various screen resolutions are referred to differently.
    - Screen resolution of 1200px and above is referred to as col-lg (larger).
    - Screen resolution from 1200px to 992px is referred to as col-md (medium).
    - Screen resolution from 992px to 768px is referred to as col-sm(small).
    - Screen resolution from 768px to 0px is referred to as col-xs (extra-small).

	Extra small <768px	Small >=768px	Medium >=992px	Large >=1200px
Class prefix	.col-xs-	.col-sm-	.col-md-	.col-lg-

4. So:
    - If we assign **col-lg-12** to a div, it means in the screen resolution 1200px and above the div will be 100% of the screen.
    - If we assign **col-lg-11** to a div, it means in the screen resolution 1200px and above the div will be 91.63% (8.33*11) of the screen.
    - If we assign **col-lg-10** to a div, it means in the screen resolution 1200px and above the div will be 93.3% (8.33*10) of the screen.

And so on.

5. The same method is followed for all screen sizes.
    - If we assign **col-md-9** to a div, it means in the screen resolution between 1200px and 992px the div will be 74.97% (8.33*9) of the screen.
    - If we assign **col-md-8** to a div, it means in the screen resolution between 1200px and 992px the div will be 66.64% (8.33*8) of the screen;

And so on.

- If we assign **col-sm-7** to a div, it means in the screen resolution between 992px and 768px, the div will be 58.31% (8.33*7) of the screen.
- If we assign **col-sm-6** to a div, it means in the screen resolution between 992px and 768px, the div will be 49.98% (8.33*6) of the screen,

And so on

- If we assign **col-xs-5** to a div, it means in the screen resolution between 768px and 0px, the div will be 41.65% (8.33*5) of the screen.
- If we assign **col-xs-4** to a div, it means in the screen resolution between 768px and 0px, the div will be 33.32% (8.33*4) of the screen;

And so on.

```
<div class="contact_1 col-lg-6 col-md-6 col-sm-6 col-xs-12" >
```

- **Contact_1** has **col-lg-6 col-md-6 col-sm-6 col-xs-12**. This means:
  - **col-lg-6** - Till 1200px the width of contact_1 will be almost 50% of the screen.
  - **col-md-6** - From 1200px till 992px the width of contact_1 will be almost 50% of the screen.
  - **col-sm-6** - Form 992px till 768px the width of contact_1 will be almost 50% of the screen.
  - **col-xs-12** - From 768px till 0px the width of contact_1 will be 100% of the screen.

6. Now we will create a div with class **contact_2**

```
<div class="contact_2 col-lg-6 col-md-6 col-sm-6 col-xs-12">
        <form>
            <span>Name : </span>     <input type="text"
            name="name" required>
            <br><br>
            <span>Comment : </span><input type="text" name="comment" required>
            <br><br>
            <input type="submit" value="submit" class="btn btn-success" >
        </form>
    </div>
```

```
<div class="contact_2 col-lg-6 col-md-6 col-sm-6 col-xs-12">
```

- **Contact_2** has **col-lg-6 col-md-6 col-sm-6 col-xs-12**. This means:
    - **col-lg-6** - Till 1200px the width of contact_2 will be almost 50% of the screen.
    - **col-md-6** - From 1200px till 992px the width of contact_2 will be almost 50% of the screen.
    - **col-sm-6** - Form 992px till 768px the width of contact_2 will be almost 50% of the screen.
    - **col-xs-12** - From 768px till 0px the width of contact_2 will be 100% of the screen.

7. Inside that we will have our **form tag**.
    - Form tag is an important tag. It is used to collect data from the user. So when someone fills this form, data is stored in a database. (Database is a place where data is stored.) This data can be used for various purposes.
      For example: In school when we go for admission, we fill out a form. In this form we give information about us and it is stored in a file.

8. In html everything will remain the same. We will add the bootstrap class same as contact_2.

9. Elements inside the form:

```
<form>
    <span>Name : </span>     <input type="text"
    name="name" required>
    <br><br>
    <span>Comment : </span><input type="text" name="comment" required>
    <br><br>
    <input type="submit" value="submit" class="btn btn-success" >
</form>
```

- **span tag** is used for the label of the input boxes.
- **br tag** is used for line-break.
- **input tag** is used for getting the input value from the user. It has the attribute set as **required**, which means the user cannot proceed the form until he/she has filed the input box.
- **Input type submit** means it is a submit button for the form.
- In the submit button we will add bootstrap class **btn btn-success**
    - **btn** will add proper padding, font-size, display:inline properties and margin to the button.
    - **btn-success** will add the background color green and color white to the button.

- CSS code for form: This will give grey background color to the input element.

```
input
{
    background: grey;
}
```

What's NEXT?

We will enhance the e-portfolio website by adding a responsive blog to it while applying bootstrap classes to all the elements of the website.

EXTEND YOUR KNOWLEDGE

Here are Few Best Reference Links we complied to extend your knowledge about the concept we learnt today in the class. This will help you become pro at coding and creating industry grade tech products

Short Videos: Watch these Short Videos to Understand the Class Concepts Applied in Real World Applications.

1. What is Bootstrap: https://www.youtube.com/watch?v=R-p_V-8k6hY
2. Intro to Responsive Design: https://www.youtube.com/watch?v=bW2FZOtEOqQ
3. Bootstrap: Complete Website Layout: https://www.youtube.com/watch?v=lY-JUYD_H_g&t=43s
4. Bootstrap Navbar: https://www.youtube.com/watch?v=9e6k9uVzQAs

Coding Playground: Try Out These code examples yourself to deepen your knowledge

1. 

:

   https://www.w3schools.com/bootstrap/bootstrap_get_started.asp

2. 

:

   https://www.w3schools.com/bootstrap/bootstrap_navbar.asp