Styling Web Pages with CSS Libraries

# 1 Introduction

CSS is a powerful language allowing complex Web page styling and layout. Various patterns and best practices have appeared overtime in industries that have become popular. Some of these have been collected into commercial and open source libraries such as:

* [Bootstrap](https://getbootstrap.com/)
* [Foundation](https://get.foundation/)
* [Tailwind](https://tailwindcss.com/)
* [Material Design](https://material.io/design)
* [Bulma](https://bulma.io/)

All these libraries define a set of CSS rules that you can readily apply to achieve a professional look and feel, powerful layouts, and responsive designs. Using a library consists of becoming familiar with the CSS rules and applying them appropriately to your HTML to achieve a particular purpose. For this course we are going to be using Bootstrap throughout our assignments. You are free to use other libraries for your final project.

Assignments in this course contain three main sections: ***Labs***, ***Tuiter***, and ***Challenge***. The ***Labs*** section in this assignment will give you an opportunity to practice the concepts described in this assignment, i.e., Bootstrap. Once you've had a chance to practice with Bootstrap, in the ***Tuiter*** section you'll be asked to apply what you've learned to build a Website called ***Tuiter*** inspired on a popular social network. The ***Challenge*** section explores additional, more challenging requirements that are optional for undergraduates, but required for graduate students.

Create a new branch called ***a4*** and do all your work there. When done, add, commit and push the branch to GitHub. Deploy the new branch to Netlify and confirm it's available in a new URL based on the branch name.

# 2 Learning objectives

* Layout and responsive design with Bootstrap
* Laying out Webpages with Bootstrap
* Use Fontawesome icons

# 3 Labs (50pts)

This section presents exercises of how to use the ***Bootstrap*** CSS library to style and layout Web pages. Use the same project you worked on last assignment. After you work through the exercises you will apply the skills to create ***Tuiter*** on your own. Using ***IntelliJ***, open the project you created in the previous assignment. From within IntelliJ, use ***File***, ***Open Project***, and navigate to the project directory, (***tuiter-react-web-app***), and click ***Open*** or ***OK***. From within IntelliJ, on the ***Project*** tab, open the ***tuiter-react-web-app*** directory, and then the ***public*** directory. Do all your work under the ***public*** directory of your project.

Under the ***public/labs*** directory, create a new directory called ***a4*** and create ***index.html*** under ***public/labs/a4***. For this assignment, do all your work in ***public/labs/a4/index.html***. To make it easy on the TAs, add a link to this new ***index.html*** file in ***public/index.html***.

## 3.1 Installing bootstrap (2pts)

Bootstrap can be installed in a couple of ways. You can [download the latest version from their download Website](https://getbootstrap.com/docs/5.1/getting-started/download/), unzip the zip file, and copy the content under their ***dist*** folder into your project. A common place to put libraries you download is under a ***vendors*** directory. Below is an example of having copied the content of the ***dist*** folder into a directory under ***public/vendors/bootstrap***. This alternative is best for software developers where they might have local copies of various versions of libraries they use during the development effort. Local copies of libraries simplify the development process. Your actual location may differ so fix as needed.

| <**link href="../../vendors/bootstrap/css/bootstrap.min.css" rel="stylesheet"**/> | |
| --- | --- |

Alternatively you can reference a CSS file hosted from any number of ***CDN***s. CDN stands for ***Content Delivery Network*** and are remote servers dedicated to hosting common files and resources used by a community. This alternative is best for production environments when the Website is live. Your URL may differ so fix as needed.

| <**link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.0/dist/css/bootstrap.min.css"**  **rel="stylesheet"**/> | |
| --- | --- |

Feel free to use either of the alternatives described above.

## 3.2 Laying out content with containers and grids (8pts)

### 3.2.1 Containers (2pts)

[Bootstrap containers](https://getbootstrap.com/docs/5.0/layout/containers/) establish the root of your HTML document providing a basis of default styles such as the overall margins, paddings, and font of your document. There are two main classes that control container elements: ***.container*** and ***.container-fluid***. The ***.container*** class centers the content with margins on either side and defines several responsive design thresholds. The ***.container-fluid*** class just defines a constant thin margin all around the document.

To practice with containers, copy the HTML code below to the end of ***index.html***, and save. Refresh the browser and confirm it looks similar to image shown. The heading is not flush with the left hand side and the font is not the default browser font.

| ***index.html*** | | ***Browser*** |
| --- | --- | --- |
| <**body**>  <**div class="container"**>  <**h1**>Bootstrap</**h1**>  <**p**>Welcome to Bootstrap!</**p**>  *<!-- main content goes here -->*  </**div**>  </**body**> | | Bootstrap Welcome to Bootstrap! |

### 3.2.2 Grid system (2pts)

It's easy to break a page vertical in HTML with the ***p*** and ***div*** tags. It's a little harder to layout content horizontally. Some resort to HTML tables to layout content horizontally using table ***rows*** and ***columns***, but this is generally considered a bad practice. HTML tables should be used for displaying tabular content only, not laying out HTML content. Nevertheless, laying out content like a table is convenient, so to achieve the same functionality as tables, but without tables, we can use CSS instead. Bootstrap provides classes such as ***.row*** and ***.col*** to layout content in a ***grid***. To practice with ***Bootstrap grids***, copy the HTML code below to the end of ***index.html***, and save. Refresh the browser and confirm it looks similar to image shown.

| ***index.html*** | | ***Browser*** |
| --- | --- | --- |
| <**h2**><**a href="https://getbootstrap.com/docs/5.1/layout/grid/"**>Grid system</**a**></**h2**>  <**div class="row"**>  <**div class="col bg-danger text-white"**>  <**h3**>Left half</**h3**></**div**>  <**div class="col bg-primary text-white"**>  <**h3**>Right half</**h3**></**div**></**div**>  <**div class="row"**>  <**div class="col-4 bg-warning"**>  <**h3**>One thirds</**h3**></**div**>  <**div class="col-8 bg-success text-white"**>  <**h3**>Two thirds</**h3**></**div**></**div**>  <**div class="row"**>  <**div class="col-2 bg-dark text-white"**>  <**h3**>Sidebar</**h3**></**div**>  <**div class="col-8 bg-secondary text-white"**>  <**h3**>Main content</**h3**></**div**>  <**div class="col-2 bg-info"**>  <**h3**>Sidebar</**h3**></**div**></**div**> | | *<!-- a row containing*  *two columns applying class col with default*  *width evenly spread over all columns*  *another row with*  *two columns applying classes col-4 and col-8*  *where 4 + 8 = 12, the total number of columns*  *so 4/12 is ⅓ and 8/12 is ⅔ of the screen*  *-->* |

### 3.2.3 Responsive grids (2pts)

Bootstrap grids can adapt to the size of the screen, that is, they can be responsive. We can achieve this by applying more than one ***.col*** class which only applies for a given window size. To practice with ***Bootstrap responsive*** grids, copy the HTML code below to the end of ***index.html***, and save. Refresh the browser and confirm it looks similar to image shown. Resize the browser and confirm that the grid shows 4 columns, then 2 and then just 1.

| ***index.html*** | | ***Browser*** |
| --- | --- | --- |
| <**h2**>Responsive grid system</**h2**>  <**div class="row"**>  <**div class="col-12 col-md-6 col-xl-3 bg-warning"**>  <**h3**>Column A</**h3**>  </**div**>  <**div class="col-12 col-md-6 col-xl-3 bg-primary text-white"**>  <**h3**>Column B</**h3**>  </**div**>  <**div class="col-12 col-md-6 col-xl-3 bg-danger text-white"**>  <**h3**>Column C</**h3**>  </**div**>  <**div class="col-12 col-md-6 col-xl-3 bg-success text-white"**>  <**h3**>Column D</**h3**>  </**div**>  </**div**> | | Wide browser window shows 4 columns    Moderate width browser window shows 2 columns    Thin browser window shows only 1 column |

Let's try a more dramatic example by adding more content spread over more columns and rows. Copy the HTML code below to the end of ***index.html***, and save. Refresh the browser and confirm that the column layout changes as you resize the browser window.

| ***index.html*** | | ***Browser*** |
| --- | --- | --- |
| <**h2**>Responsive grid system</**h2**>  <**div class="row"**>  <**div class="col-12 col-sm-6 col-md-4 col-lg-3 col-xl-2 col-xxl-1 bg-warning"**>  <**h4**>1</**h4**></**div**>  <**div class="col-12 col-sm-6 col-md-4 col-lg-3 col-xl-2 col-xxl-1 bg-primary text-white"**>  <**h4**>2</**h4**></**div**>  <**div class="col-12 col-sm-6 col-md-4 col-lg-3 col-xl-2 col-xxl-1 bg-danger text-white"**>  <**h4**>3</**h4**></**div**>  <**div class="col-12 col-sm-6 col-md-4 col-lg-3 col-xl-2 col-xxl-1 bg-success text-white"**>  <**h4**>4</**h4**></**div**>  <**div class="col-12 col-sm-6 col-md-4 col-lg-3 col-xl-2 col-xxl-1 bg-warning"**>  <**h4**>5</**h4**></**div**>  <**div class="col-12 col-sm-6 col-md-4 col-lg-3 col-xl-2 col-xxl-1 bg-primary text-white"**>  <**h4**>6</**h4**></**div**>  <**div class="col-12 col-sm-6 col-md-4 col-lg-3 col-xl-2 col-xxl-1 bg-danger text-white"**>  <**h4**>7</**h4**></**div**>  <**div class="col-12 col-sm-6 col-md-4 col-lg-3 col-xl-2 col-xxl-1 bg-success text-white"**>  <**h4**>8</**h4**></**div**>  <**div class="col-12 col-sm-6 col-md-4 col-lg-3 col-xl-2 col-xxl-1 bg-warning"**>  <**h4**>9</**h4**></**div**>  <**div class="col-12 col-sm-6 col-md-4 col-lg-3 col-xl-2 col-xxl-1 bg-primary text-white"**>  <**h4**>10</**h4**></**div**>  <**div class="col-12 col-sm-6 col-md-4 col-lg-3 col-xl-2 col-xxl-1 bg-danger text-white"**>  <**h4**>11</**h4**></**div**>  <**div class="col-12 col-sm-6 col-md-4 col-lg-3 col-xl-2 col-xxl-1 bg-success text-white"**>  <**h4**>12</**h4**></**div**>  </**div**> | |  |

### 3.2.4 Hiding and showing responsive content (2pts)

As users shrink or widen the browser window, there may be more or less space to show some content. Bootstrap can configure content to show or hide depending on the width of the screen. As described earlier in [Responsive grids](#_30mjss205fve), Bootstrap breaks up the screen into 5 sizes: ***extra small***, ***small***, ***medium***, ***large***, ***extra large***, and ***extra extra large***. Use [Bootstrap display](https://getbootstrap.com/docs/5.1/utilities/display/) classes as shown below that illustrates hiding and showing content as you resize the screen. Resize the window and confirm that the content only shows at certain widths.

| ***index.html*** | |  |  |
| --- | --- | --- | --- |
| <**h2**>Hiding and showing responsive content</**h2**>  <**div class="d-block d-sm-none fa-2x"**>XS</**div**>  <**div class="d-none d-sm-block d-md-none fa-2x"**>S</**div**>  <**div class="d-none d-md-block d-lg-none fa-2x"**>M</**div**>  <**div class="d-none d-lg-block d-xl-none fa-2x"**>L</**div**>  <**div class="d-none d-xl-block d-xxl-none fa-2x"**>XL</**div**>  <**div class="d-none d-xxl-block fa-2x"**>XXL</**div**> | | Browser displays  XS  S  M  L  XL  XXL | @ window width  <576px  ≥576px  ≥768px  ≥992px  ≥1200px  ≥1400px |

## 3.3 Tables and lists (8pts)

### 3.3.1 Styling tables (2pts)

Bootstrap provides several classes that enhance the look and feel of common HTML widgets such as tables, lists, and form elements. Let's start with tables. To practice with styling ***HTML tables***, copy the HTML code below to the end of ***index.html***, and save. Refresh the browser and confirm it looks similar to image shown.

| ***index.html*** | | |
| --- | --- | --- |
| <**h2**>Tables</**h2**>  <**table class="table"**>  <**thead**>  <**tr class="table-dark"**><**th**>Quiz</**th**><**th**>Topic</**th**><**th**>Date</**th**><**th**>Grade</**th**></**tr**>  </**thead**>  <**tbody**>  <**tr class="table-warning"**><**td**>Q1</**td**><**td**>HTML</**td**><**td**>2/3/21</**td**><**td**>85</**td**></**tr**>  <**tr class="table-danger"**><**td**>Q2</**td**><**td**>CSS</**td**><**td**>2/10/21</**td**><**td**>90</**td**></**tr**>  <**tr class="table-primary"**><**td**>Q3</**td**><**td**>JavaScript</**td**><**td**>2/17/21</**td**><**td**>90</**td**></**tr**>  </**tbody**>  <**tfoot**>  <**tr class="table-success"**><**td colspan="3"**>Average</**td**><**td**>90</**td**></**tr**>  </**tfoot**>  </**table**> | | |
|  | | |

### 3.3.2 Making tables responsive (2pts)

In general it is a good practice to minimize the number of scrollbars shown at any one time in a browser screen. In browsers large amounts of content extends vertically beyond the height of the window, and then scrollbars allow you to access that extra content. Sometimes it is necessary to use additional scrollbars when appropriate such as tables that might be too wide to fit horizontally. Bootstrap provides tables that can show scrollbars when they don't fit in their parent window. To practice with ***Bootstrap responsive tables***, copy the HTML code below to the end of ***index.html***, and save. Refresh the browser and confirm it looks similar to image shown. Resize the window and confirm that the table shows a horizontal scroll bar when the screen is too small to fit the table comfortably.

| ***index.html*** | | |
| --- | --- | --- |
| <**h2**>  <**a href="https://getbootstrap.com/docs/5.1/content/tables/#responsive-tables"**>  Responsive tables</**a**></**h2**>  <**div class="table-responsive"**>  <**table class="table"**>  <**thead**>  <**tr**><**th**>Very</**th**><**th**>long</**th**><**th**>set</**th**><**th**>of</**th**><**th**>columns</**th**>  <**th**>Very</**th**><**th**>long</**th**><**th**>set</**th**><**th**>of</**th**><**th**>columns</**th**>  <**th**>Very</**th**><**th**>long</**th**><**th**>set</**th**><**th**>of</**th**><**th**>columns</**th**>  </**tr**>  </**thead**>  <**tbody**>  <**tr**>  <**td**>Very</**td**><**td**>long</**td**><**td**>set</**td**><**td**>of</**td**><**td**>columns</**td**>  <**td**>Very</**td**><**td**>long</**td**><**td**>set</**td**><**td**>of</**td**><**td**>columns</**td**>  <**td**>Very</**td**><**td**>long</**td**><**td**>set</**td**><**td**>of</**td**><**td**>columns</**td**>  </**tr**>  </**tbody**>  </**table**>  </**div**> | | |
|  | |  |

### 3.3.3 Styling lists (2pts)

Another set of Bootstrap classes can make simple HTML lists look more presentable. The ***.list-group*** and ***.list-group-item*** classes can be applied to ***ul*** and ***li*** tags correspondingly to make list stand out. The ***.active*** class can be applied to an ***li*** tag to highlight it. To practice with ***Bootstrap lists***, copy the HTML code below to the end of ***index.html***, and save. Refresh the browser and confirm it looks similar to image shown.

| ***index.html*** | | ***Browser*** |
| --- | --- | --- |
| <**h2**>  <**a href="https://getbootstrap.com/docs/5.1/components/list-group/"**>  Favorite movies</**a**>  </**h2**>  <**ul class="list-group"**>  <**li class="list-group-item active"**>Aliens</**li**>  <**li class="list-group-item"**>Terminator</**li**>  <**li class="list-group-item"**>Blade Runner</**li**>  <**li class="list-group-item"**>Lord of the Ring</**li**>  <**li class="list-group-item disabled"**>Star Wars</**li**>  </**ul**> | |  |

### 3.3.4 Styling list of hyperlinks (2pts)

The same ***.list-group*** and ***.list-group-item*** Bootstrap classes can be applied to ***div*** and ***a*** tags to implement a list of anchor links. To practice with ***Bootstrap hyperlink lists***, copy the HTML code below to the end of ***index.html***, and save. Refresh the browser and confirm it looks similar to image shown and that the links work.

| ***index.html*** | | ***Browser*** |
| --- | --- | --- |
| <**h3**>  <**a href="https://getbootstrap.com/docs/5.1/components/list-group/#links-and-buttons"**>  Favorite books</**a**>  </**h3**>  <**div class="list-group"**>  <**a href="https://en.wikipedia.org/wiki/Dune\_(novel)" class="list-group-item list-group-item-action active"**>  Dune</**a**>  <**a href="https://en.wikipedia.org/wiki/The\_Lord\_of\_the\_Rings" class="list-group-item list-group-item-action"**>  Lord of the Rings</**a**>  <**a href="https://en.wikipedia.org/wiki/The\_Forever\_War" class="list-group-item list-group-item-action"**>  The Forever War</**a**>  <**a href="https://en.wikipedia.org/wiki/2001:\_A\_Space\_Odyssey\_(novel)" class="list-group-item list-group-item-action"**>  2001 A Space Odyssey</**a**>  <**a href="https://en.wikipedia.org/wiki/Ender%27s\_Game" class="list-group-item list-group-item-action disabled" tabindex="-1" aria-disabled="true"**>Ender's Game</**a**>  </**div**> | |  |

## 3.4 Styling forms (12pts)

### 3.4.1 Basic form styling (1pts)

Bootstrap has tons of classes to style form elements especially to make them friendly for mobile Web applications. To practice with ***Bootstrap form classes***, copy the HTML code below to the end of ***index.html***, and save. Refresh the browser and confirm it looks similar to image shown.

| ***index.html*** | | ***Browser*** |
| --- | --- | --- |
| <**h2**>  <**a href="https://getbootstrap.com/docs/5.1/forms/form-control/"**>  Forms  </**a**>  </**h2**>  <**div class="mb-3"**>  <**label for="input1"**  **class="form-label"**>  Email address</**label**>  <**input type="email"**  **class="form-control"**  **id="input1"**  **placeholder="name@example.com"**>  </**div**>  <**div class="mb-3"**>  <**label for="textarea1"**  **class="form-label"**>  Example textarea</**label**>  <**textarea class="form-control"**  **id="textarea1"**  **rows="3"**></**textarea**>  </**div**> | |  |

### 3.4.2 Styling dropdowns (2pts)

Dropdowns can also be styled professionally. To practice with ***Bootstrap dropdown styling***, copy the HTML code below to the end of ***index.html***, and save. Refresh the browser and confirm it looks similar to image shown.

| ***index.html*** | | ***Browser*** |
| --- | --- | --- |
| <**h3**>  <**a href="https://getbootstrap.com/docs/5.1/forms/select/"**>  Dropdowns</**a**>  </**h3**>  <**select class="form-select"**>  <**option selected**>Open this select menu</**option**>  <**option value="1"**>One</**option**>  <**option value="2"**>Two</**option**>  <**option value="3"**>Three</**option**>  </**select**> | |  |

### 3.4.3 Styling switches (2pts)

Checkboxes can be styled to look like switches with Bootstrap classes ***.form-check*** and ***.form-switch***. To practice with ***Bootstrap form switches***, copy the HTML code below to the end of ***index.html***, and save. Refresh the browser and confirm it looks similar to image shown.

| ***index.html*** | | ***Browser*** |
| --- | --- | --- |
| <**h3**>  <**a href="https://getbootstrap.com/docs/5.1/forms/checks-radios/#switches"**>Switches</**a**>  </**h3**>  <**div class="form-check form-switch"**>  <**input class="form-check-input"**  **type="checkbox"**  **id="switch1"**>  <**label class="form-check-label" for="switch1"**>  Default switch checkbox input</**label**>  </**div**>  <**div class="form-check form-switch"**>  <**input class="form-check-input"**  **type="checkbox"**  **id="switch2"**  **checked**>  <**label class="form-check-label"**  **for="switch2"**>  Checked switch checkbox input</**label**>  </**div**>  <**div class="form-check form-switch"**>  <**input class="form-check-input"**  **type="checkbox"**  **id="switch3"**  **disabled**>  <**label class="form-check-label"**  **for="switch3"**>  Disabled switch checkbox input</**label**>  </**div**>  <**div class="form-check form-switch"**>  <**input class="form-check-input"**  **type="checkbox"**  **id="switch4"**  **checked disabled**>  <**label class="form-check-label"**  **for="switch4"**>  Disabled checked switch checkbox input</**label**>  </**div**> | |  |

### 3.4.4 Styling range and sliders (1pts)

Range input fields render as sliders. To practice with ***Bootstrap sliders***, copy the HTML code below to the end of ***index.html***, and save. Refresh the browser and confirm it looks similar to image shown.

| ***index.html*** | | ***Browser*** |
| --- | --- | --- |
| <**h3**>  <**a href="https://getbootstrap.com/docs/5.1/forms/range/#steps"**>  Range  </**a**>  </**h3**>  <**label for="range1"**  **class="form-label"**>  Example range</**label**>  <**input type="range"**  **class="form-range"**  **min="0"**  **max="5"**  **step="0.5"**  **id="range1"**/> | |  |

### 3.4.5 Styling addons (2pts)

Addons decorate input felds to give some context on the type and formate of the information to type in the input field. To practice with ***Bootstrap addons***, copy the HTML code below to the end of ***index.html***, and save. Refresh the browser and confirm it looks similar to image shown.

| ***index.html*** | | ***Browser*** |
| --- | --- | --- |
| <**h3**><**a href="https://getbootstrap.com/docs/5.1/forms/input-group/"**>Addons</**a**></**h3**>  <**div class="input-group mb-3"**>  <**span class="input-group-text"**>$</**span**>  <**span class="input-group-text"**>0.00</**span**>  <**input type="text" class="form-control"**>  </**div**>  <**div class="input-group"**>  <**input type="text" class="form-control"**>  <**span class="input-group-text"**>$</**span**>  <**span class="input-group-text"**>  0.00</**span**>  </**div**> | |  |

### 3.4.6 Making forms responsive (3pts)

Forms can be configured to display either horizontally or vertically depending on the size of the containing element. To practice with Bootstrap responsive forms, copy the HTML code below to the end of ***index.html***, and save. Refresh the browser and confirm it looks similar to image shown. Resize the window to show how the form changes layout as the window resizes.

| ***index.html*** | | ***Browser*** |
| --- | --- | --- |
| <**h3**><**a href="https://getbootstrap.com/docs/5.1/forms/layout/#horizontal-form"**>Responsive forms</**a**></**h3**>  <**div class="mb-3 row"**>  <**label for="email1"**  **class="col-sm-2 col-form-label"**>  Email</**label**>  <**div class="col-sm-10"**>  <**input type="text"**  **class="form-control"**  **id="email1"**  **value="email@example.com"**>  </**div**>  </**div**>  <**div class="mb-3 row"**>  <**label for="password1"**  **class="col-sm-2 col-form-label"**>  Password</**label**>  <**div class="col-sm-10"**>  <**input type="password"**  **class="form-control"**  **id="password1"**>  </**div**>  </**div**>  <**div class="mb-3 row"**>  <**label for="textarea1"**  **class="col-sm-2 col-form-label"**>  Bio</**label**>  <**div class="col-sm-10"**>  <**textarea class="form-control"**  **id="textarea1"**  **rows="3"**></**textarea**>  </**div**>  </**div**> | | **Responsive forms**    **Responsive forms** |

Here's another example. Copy the HTML code below to the end of ***index.html***, and save. Refresh the browser and confirm it looks similar to image shown. Resize the window to show how the form changes layout as the window resizes.

| ***index.html*** | | ***Browser*** |
| --- | --- | --- |
| <**h3**><**a href="https://getbootstrap.com/docs/5.1/forms/layout/#horizontal-form"**>Responsive forms</**a**>  </**h3**>  <**form**>  <**div class="row mb-3"**>  <**label for="r1" class="col-sm-2 col-form-label"**>  Email</**label**>  <**div class="col-sm-10"**>  <**input type="email" class="form-control"**  **id="r1"**/>  </**div**>  </**div**>  <**div class="row mb-3"**>  <**label for="r2" class="col-sm-2 col-form-label"**>  Password</**label**>  <**div class="col-sm-10"**>  <**input type="password" class="form-control"**  **id="r2"**/>  </**div**>  </**div**>  <**fieldset class="row mb-3"**>  <**legend class="col-form-label col-sm-2 pt-0"**>  Radios</**legend**>  <**div class="col-sm-10"**>  <**div class="form-check"**>  <**input class="form-check-input" type="radio"**  **name="gridRadios" id="r3"**  **value="option1" checked**/>  <**label class="form-check-label" for="r3"**>  First radio</**label**>  </**div**>  <**div class="form-check"**>  <**input class="form-check-input" type="radio"**  **name="gridRadios" id="r4"**  **value="option2"**/>  <**label class="form-check-label" for="r4"**>  Second radio</**label**>  </**div**>  <**div class="form-check disabled"**>  <**input class="form-check-input"**  **type="radio"**  **name="gridRadios" id="r5"**  **value="option3" disabled**/>  <**label class="form-check-label" for="r5"**>  Third disabled radio</**label**>  </**div**>  </**div**>  </**fieldset**>  <**div class="row mb-3"**>  <**div class="col-sm-10 offset-sm-2"**>  <**div class="form-check"**>  <**input class="form-check-input"**  **type="checkbox" id="r6"**/>  <**label class="form-check-label" for="r6"**>  Example checkbox</**label**>  </**div**>  </**div**>  </**div**>  <**button type="submit" class="btn btn-primary"**>Sign in</**button**>  </**form**> | |  |

### 3.4.7 Overriding Bootstrap style (1pts)

Bootstrap makes it easy to apply professional styles, but sometimes we will need override some of the styles declared by Bootstrap. The way to achieve this is by loading additional stylesheets after loading the Bootstrap library. To practice with overriding Bootstrap styles, copy the HTML and CSS code below to the end of ***index.html*** and ***index.css***, and save. Refresh the browser and confirm it looks similar to image shown.

| ***index.html*** | | ***index.css*** |
| --- | --- | --- |
| <**html lang="en"**>  <**head**>  <**link href="../../vendors/bootstrap/css/bootstrap.min.css"**  **rel="stylesheet"**/>  <**link href="index.css" rel="stylesheet"**/>  </**head**>  <**body**>  <**h3**>Overriding Bootstrap style</**h3**>  <**button class="btn btn-primary"**>  Regular Bootstrap Button  </**button**>  <**br**/>  <**button class="btn btn-primary override-bs"**>  Overriden Button  </**button**>  </**body**>  </**html**> | | .**btn-primary**.**override-bs** {  **background-color**: **#a90dfd**;  **border-color**: **#fd0d51**;  **border-width**: 10**px**;  **border-radius**: 20**px**;  **border-style**: **dashed**;  **font-size**: 2**em**;  **font-weight**: **bolder**;  } |

## 3.5 Navigating with tabs, pills and cards (6pts)

### 3.5.1 Navigating with tabs (2pts)

Bootstrap provides several common navigation widgets such as tabs, menus, and pills. Let's take a look at tabs first. To practice with Bootstrap tabs, copy the HTML code below to the end of ***index.html***, and save. Refresh the browser and confirm it looks similar to image shown.

| ***index.html*** | | |
| --- | --- | --- |
| <**h2**><**a href="https://getbootstrap.com/docs/5.1/components/navs-tabs/"**>Tabs</**a**></**h2**>  <**ul class="nav nav-tabs"**>  <**li class="nav-item"**>  <**a class="nav-link active" href="#"**>Active</**a**>  </**li**>  <**li class="nav-item"**>  <**a class="nav-link" href="#"**>Link</**a**>  </**li**>  <**li class="nav-item"**>  <**a class="nav-link" href="#"**>Link</**a**>  </**li**>  <**li class="nav-item"**>  <**a class="nav-link disabled" href="#" tabindex="-1"**>Disabled</**a**>  </**li**>  </**ul**> | | |
|  | | |

### 3.5.2 Navigating with pills (2pts)

Pills are another navigation widget listing several links. To practice with Bootstrap pills, copy the HTML code below to the end of ***index.html***, and save. Refresh the browser and confirm it looks similar to image shown.

| ***index.html*** | | |
| --- | --- | --- |
| <**h3**>  <**a href="https://getbootstrap.com/docs/5.1/components/navs-tabs/#pills"**>  Pills  </**a**>  </**h3**>  <**ul class="nav nav-pills"**>  <**li class="nav-item"**>  <**a class="nav-link active" href="#"**>Active</**a**>  </**li**>  <**li class="nav-item"**>  <**a class="nav-link" href="#"**>Link</**a**>  </**li**>  <**li class="nav-item"**>  <**a class="nav-link" href="#"**>Link</**a**>  </**li**>  <**li class="nav-item"**>  <**a class="nav-link disabled" href="#" tabindex="-1"**>Disabled</**a**>  </**li**>  </**ul**> | | |
|  | | |

### 3.5.3 Navigating with cards (2pts)

Cards combine images, titles, paragraphs and buttons into a reusable component that can quickly summarize a topic. To practice with ***Bootstrap cards***, copy the HTML code below to the end of ***index.html***, and save. Refresh the browser and confirm it looks similar to image shown. Use an image of your own, save it in ***public/images***, and reference it from the ***img*** tag.

| ***index.html*** | | ***Browser*** |
| --- | --- | --- |
| <**h2**>  <**a href="https://getbootstrap.com/docs/5.1/components/card/"**>  Cards  </**a**>  </**h2**>  <**div class="card" style="width**: 18**rem**;**"**>  <**img src="../../images/starship.jpg"**  **class="card-img-top" alt="..."**>  <**div class="card-body"**>  <**h5 class="card-title"**>Card title</**h5**>  <**p class="card-text"**>  Some quick example text to build on the card  title and make up the bulk of the card's content.</**p**>  <**a href="#" class="btn btn-primary"**>Go somewhere</**a**>  </**div**>  </**div**> | |  |

## 3.6 Decorating documents with Fontawesome icons (14pts)

Fontawesome can be installed in a couple of ways. You can [download the latest version from their download Website](https://use.fontawesome.com/releases/v5.15.4/fontawesome-free-5.15.4-web.zip), unzip the zip file, and copy the unzipped folder to ***public/vendors/fontawesome*** in your project. This alternative is best for software developers where they might have local copies of various versions of libraries they use during the development effort. Local copies of libraries simplify the development process. Add the following link tag to your HTML document.

| <**link href="../../vendors/fontawesome/css/all.css" rel="stylesheet"**/> | |
| --- | --- |

Alternatively you can reference a CSS file hosted from any number of CDNs. CDN stands for Content Delivery Network and are remote servers dedicated to hosting common files and resources used by a community. This alternative is best for production environments when the Website is live.

| <**link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.4/css/all.min.css"**  **rel="stylesheet"**/> | |
| --- | --- |

Feel free to use either of the alternatives described above.

### 3.6.1 Using simple icons (2pts)

To practice with Fontawesome icons, copy the HTML and CSS code below to the end of ***index.html*** and save. Refresh the browser and confirm it looks similar to image shown.

| ***index.html*** | | ***Browser*** |
| --- | --- | --- |
| <**h2**>Simple icons</**h2**>  <**i class="fa fa-cog"**></**i**>  <**i class="fa fa-heart"**></**i**>  <**i class="fa fa-share"**></**i**>  <**i class="fa fa-comment"**></**i**><**br**/>  <**i class="fa fa-image"**></**i**>  <**i class="fa fa-chart-bar"**></**i**>  <**i class="fa fa-calendar"**></**i**>  <**i class="fa fa-smile"**></**i**> | |  |

### 3.6.2 Resizing icons (2pts)

To practice with Fontawesome icons, copy the HTML and CSS code below to the end of ***index.html*** and save. Refresh the browser and confirm it looks similar to image shown.

| ***index.html*** | | ***Browser*** |
| --- | --- | --- |
| <**h2**>Resizing icons</**h2**>  <**i class="fa fa-cog fa-2x"**></**i**>  <**i class="fa fa-heart fa-2x"**></**i**>  <**i class="fa fa-share fa-3x"**></**i**>  <**i class="fa fa-comment fa-3x"**></**i**><**br**/>  <**i class="fa fa-image fa-3x"**></**i**>  <**i class="fa fa-chart-bar fa-4x"**></**i**>  <**i class="fa fa-calendar" style="font-size**: 5**em"**></**i**>  <**i class="fa fa-smile" style="font-size**: 3.5**em"**></**i**> | |  |

### 3.6.3 Coloring icons (2pts)

To practice with Fontawesome icons, copy the HTML and CSS code below to the end of ***index.html*** and save. Refresh the browser and confirm it looks similar to image shown.

| ***index.html*** | | ***Browser*** |
| --- | --- | --- |
| <**h2**>Coloring icons</**h2**>  <**i class="fa fa-cog fa-2x" style="color**: **tomato"**></**i**>  <**i class="fa fa-heart fa-2x" style="color**: **tomato"**></**i**>  <**i class="fa fa-share fa-3x" style="color**: **#5ec15b"**></**i**><**br**/>  <**i class="fa fa-comment fa-3x" style="color**: **#19d49e"**></**i**>  <**i class="fa fa-image fa-3x" style="color**: **rgb**(234,123,12)**"**></**i**>  <**i class="fa fa-chart-bar fa-4x" style="color**: **rgb**(123,234,123)**"**></**i**> | |  |

### 3.6.4 Brand icons (2pts)

To practice with Fontawesome icons, copy the HTML and CSS code below to the end of ***index.html*** and save. Refresh the browser and confirm it looks similar to image shown.

| ***index.html*** | | ***Browser*** |
| --- | --- | --- |
| <**h2**>Brand icons</**h2**>  <**i class="fab fa-twitter fa-2x"**></**i**><**br**/>  <**i class="fab fa-facebook fa-2x"**></**i**><**br**/>  <**i class="fab fa-apple fa-2x"**></**i**><**br**/> | |  |

### 3.6.5 Stacking icons (2pts)

To practice with stacking Fontawesome icons, copy the HTML and CSS code below to the end of ***index.html*** and ***index.css***, and save. Refresh the browser and confirm it looks similar to image shown.

| ***index.html*** | | ***Browser*** |
| --- | --- | --- |
| <**h2**>Stacking icons</**h2**>  <**span class="fa-stack fa-2x"**>  <**i class="fas fa-square fa-stack-2x"**></**i**>  <**i class="fab fa-twitter fa-stack-1x fa-inverse"**></**i**>  </**span**><**br**/>  <**span class="fa-stack fa-2x"**>  <**i class="fas fa-circle fa-stack-2x"**></**i**>  <**i class="fas fa-flag fa-stack-1x fa-inverse"**></**i**>  </**span**><**br**/>  <**span class="fa-stack fa-2x"**>  <**i class="fas fa-camera fa-stack-1x"**></**i**>  <**i class="fas fa-ban fa-stack-2x" style="color**:**Tomato"**></**i**>  </**span**><**br**/> | |  |

### 3.6.6 Rotating icons (2pts)

To practice with rotating Fontawesome icons, copy the HTML and CSS code below to the end of ***index.html*** and ***index.css***, and save. Refresh the browser and confirm it looks similar to image shown.

| ***index.html*** | | ***Browser*** |
| --- | --- | --- |
| <**h2**>Rotating icons</**h2**>  <**div class="fa-4x"**>  <**i class="fas fa-snowboarding"**></**i**>  <**i class="fas fa-snowboarding fa-rotate-90"**></**i**>  <**i class="fas fa-snowboarding fa-rotate-180"**></**i**><**br**/>  <**i class="fas fa-snowboarding fa-rotate-270"**></**i**>  <**i class="fas fa-snowboarding fa-flip-horizontal"**></**i**>  <**i class="fas fa-snowboarding fa-flip-vertical"**></**i**><**br**/>  <**i class="fas fa-snowboarding fa-flip-both"**></**i**><**br**/>  </**div**> | |  |

### 3.6.7 Animating icons (2pts)

To practice animating Fontawesome icons, copy the HTML and CSS code below to the end of ***index.html*** and ***index.css***, and save. Refresh the browser and confirm it looks similar to image shown.

| ***index.html*** | | ***Browser*** |
| --- | --- | --- |
| <**h2**>Animating icons</**h2**>  <**div class="fa-3x"**>  <**i class="fas fa-spinner fa-spin"**></**i**>  <**i class="fas fa-circle-notch fa-spin"**></**i**>  <**i class="fas fa-sync fa-spin"**></**i**><**br**/>  <**i class="fas fa-cog fa-spin"**></**i**>  <**i class="fas fa-spinner fa-pulse"**></**i**>  <**i class="fas fa-stroopwafel fa-spin"**></**i**>  </**div**> | |  |

## 3.7 Exercises (50pts)

Complete the following exercises

1. Install bootstrap as described in [section 3.1](#_d508mp9zhu0m) (2pts)
2. Implement the content layout exercises as described in [section 3.2](#_2ajkv47gyzau) (8pts)
3. Implement the table and list exercises as described in [section 3.3](#_emlsn3gv9hsa) (8pts)
4. Implement the form exercises as described in [section 3.4](#_yvc881g2qkvq) (12pts)
5. Implement the navigation exercises as described in [section 3.5](#_t8mr68pw1wyi) (6pts)
6. Implement the icon decoration exercises as described in [section 3.6](#_bew4c6y40usg) (14pts)

# 4 Tuiter (50pts)

For the tuiter section of this assignment you are going to continue implementing the ***Tuiter*** website started in previous assignments. You will use the ***Bootstrap*** and ***Fontawesome*** CSS libraries to reimplement the ***Explore*** screen. Do all your work in ***public/tuiter/explore/index.html*** and ***public/tuiter/explore/index.css***.

Tuiter's explore screen displays a list of random tuits organized into several categories accessible by tabs. An illustration is shown in the ***Extra extra large*** section that follows. The screen, as most ***Tuiter*** screens we'll work on, have three columns. The column on the left is a navigation sidebar with links to various screens throughout ***Tuiter***. The column in the middle contains the main content, and the column on the right is another side bar with links to other users you might want to follow.

The list of tuits in the main content, rendered in the center column, are categorized by ***For You***, ***Trending***, ***News***, ***Sports***, and ***Entertainment***. The default selected tab is ***For You***. The tabs point to ***for-you.html***, ***trending.html***, ***news.html***, ***sports.html***, and ***entertainment.html*** respectively. You don't need to implement those Webpages. The goal of this section is to implement the ***explore.html*** Webpage so that it looks and behaves as close as possible to how it is described here and displayed in the sections that follow. As you work on explore.html, the following is a few things worth pointing out

1. The navigation on the left has both icons and text
2. The link to the current page is highlighted, e.g., ***# Explore***
3. There is a small margin at the top of the page, below the navigation links on the left, below the search field, below the tabs
4. The width of the Tuit button is the same as the navigation bar above it
5. The main content is centered with equal margins on either side
6. There's an equal spacing between the three columns
7. The ***Follow*** buttons on the ***Who to follow*** side bar on the right are vertically centered
8. The search field has a magnifying glass rendered within its border, and the text and placeholder start to the right of the icon
9. The image at the top of the middle column has a large white text overlayed on top of it, but positioned in the bottom left corner of the image
10. The cog on the right of the search field is vertically aligned with the search field
11. The search field has rounded corners
12. The borders of the avatar images under ***Who to follow*** are circles. Do not use circle images. Use rounded corners instead
13. Use images similar to the ones shown. Use the text shown as a hint to find images on the Web
14. Use the exact text shown

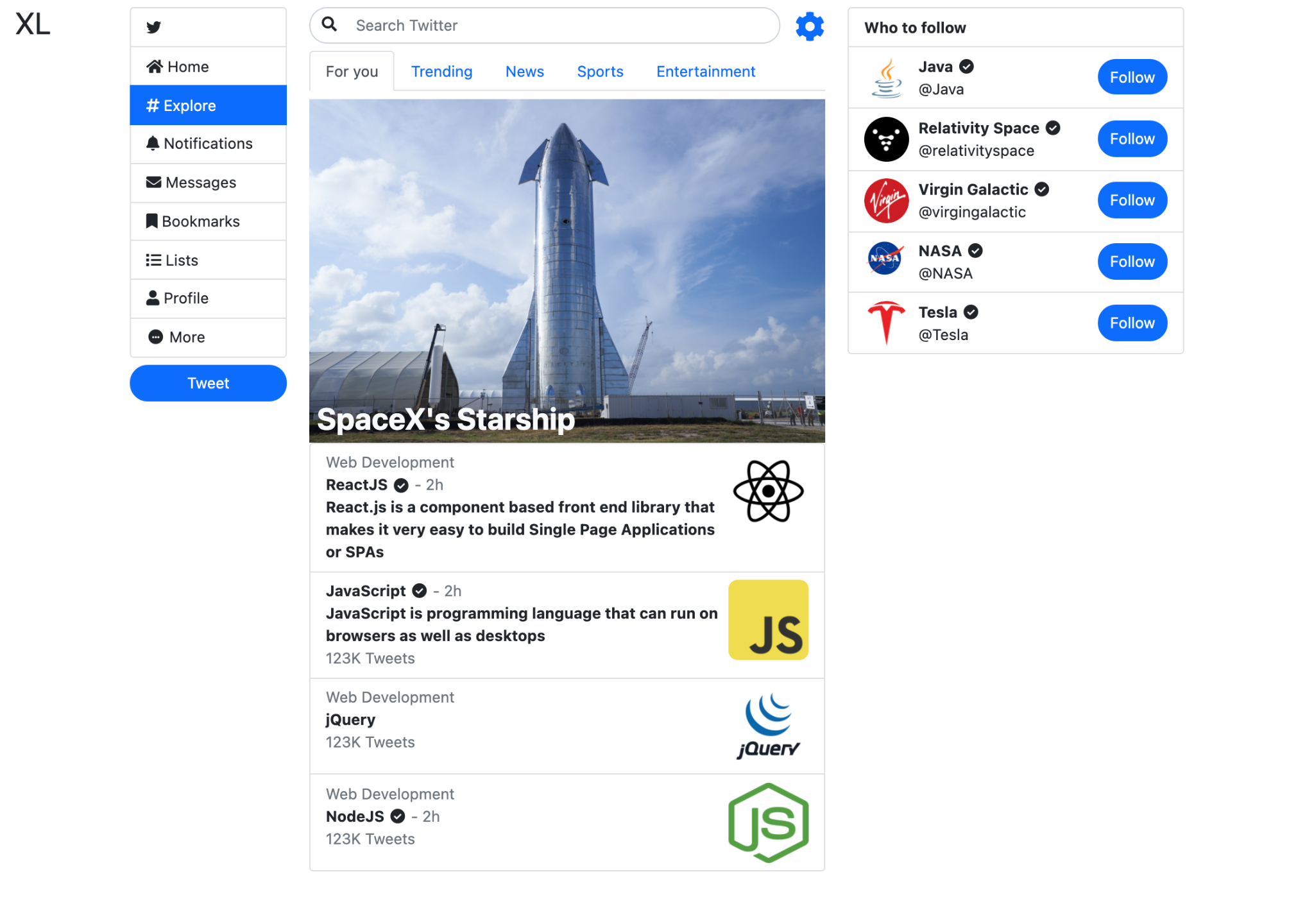
## 4.1 Extra extra large (10pts)

When the browser window is at its widest, the layout displays as shown below. The three columns are at their widest. The XXL on the top left is only for reference.



## 4.2 Extra large (10pts)

As you shrink the browser window, the column widths shrink, expand, or stay the same to favor the column in the middle that contains the main content. The text on the left hand sidebar and in the center column wraps as the window shrinks. The text never overlaps the icons on their right.



Make sure your screen looks as close as possible as shown in this section. Use the same text, colors, background colors, paddings, and margins as shown. Look for icons and images as close to those used in the examples in this document.

## 4.3 Bootstrap positions, margins and paddings

Instead of using CSS to style [positions](https://getbootstrap.com/docs/5.1/utilities/position/), [margins and paddings](https://getbootstrap.com/docs/5.1/utilities/spacing/), use Bootstrap utility classes as shown below. ***m*** stands for ***margin***, ***p*** stands for ***padding***, ***t*** for ***top***, ***b*** for ***bottom***, ***s*** for ***start***, and ***e*** for ***end***. Integers are sizes.

| Positions | Margins | Paddings |
| --- | --- | --- |
| position-absolute, position-relative,  position-fixed, position-sticky  top, bottom, start, end,  top-0, top-50, top-100,  bottom-0, bottom-50, bottom-100,  start-0, start-50, start-100,  end-0, end-50, end-100, | mt-0 mb-0 ms-0 me-0  mt-1 mb-1 ms-1 me-1  mt-2 mb-2 ms-2 me-2  ... ... ... ...  mt-5 mb-5 ms-5 me-5  mt-auto ms-auto  mb-auto me-auto | pt-0 pb-0 ps-0 me-0  pt-1 pb-1 ps-1 me-1  pt-2 pb-2 ps-2 me-2  ... …. ... ...  pt-5 pb-5 ps-5 me-5  pt-auto ps-auto  pb-auto pe-auto |

| 4.4 Large (10pts) When the window is shrunk even further, the left column is too skinny to fit both the icons and the text, so the text hides leaving only the icons visible. The other two columns shrink, widen, or stay the same favoring the center column with the main content.  The Tuit button under the navigation sidebar on the left can wider than the list of icons above it. |  |
| --- | --- |

| 4.5 Medium (10pts) Shrinking the window further makes the window to skinny to fit the right column. Hide the column so the other two columns can fit. The remaining columns need to shrink, widen, or stay the same to fit the remaining icons such as the ones on the left column and the ones on the bottom list.  Make sure your screen looks as close as possible as shown in this section. Use the same text, colors, background colors, paddings, and margins as shown. Look for icons and images as close to those used in the examples in this document. |  |
| --- | --- |

| 4.6 Small (10pts) In its skinniest width, the last tab doesn't fit. Hide the last tab so the rest of the tabs don't wrap. The text in the list of articles wraps so it does not overlap the icons on the right. The width of text and icons shrink or widen as needed.  Make sure your screen looks as close as possible as shown in this section. Use the same text, colors, background colors, paddings, and margins as shown. Look for icons and images as close to those used in the examples in this document. |  |
| --- | --- |

## 4.4 Exercises (50pts)

Implement the Explore page as described above. The Web page should be responsive as described in the following screen sizes

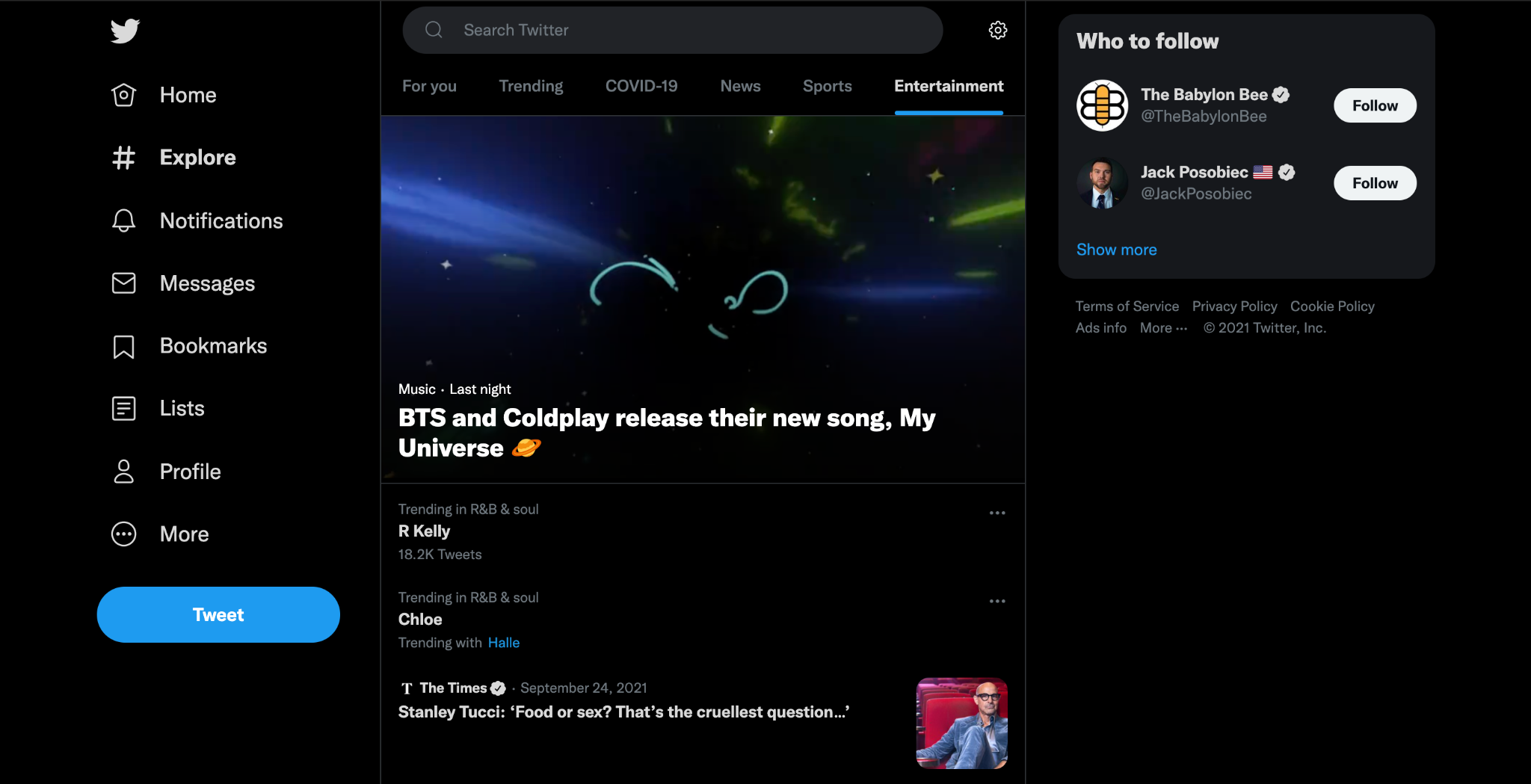
1. Extra extra large (10pts)
2. Extra large (10pts)
3. Large (10pts)
4. Medium (10pts)
5. Small (10pts)

# 5 Challenge (50pts required for graduate students)

For the challenge section restyle the ***explore/index.html*** Webpage to look like the wireframe shown below. Implement style overrides in ***explore/index.css***. The resulting screen doesn't have to be pixel perfect like the wireframe, but use the following as a guideline on what to focus on.

1. ***Borders***:colors, widths and styles
2. ***Tab***: highlighted, unhighlighted, underlined, spacing
3. ***Font and text***: white vs gray color, size, justification
4. ***White spacing***: padding, margins, borders, and corners. General distance between elements and surrounding context
5. ***Colors***: background, foreground, and borders
6. ***Icons***: color, corner rounding, padding, sizing
7. ***Forms and input fields***: rounded corners, background colors, placeholders, button colors and sizes
8. ***Positioning and sizing***: white text and icon overlaid over large image
9. ***Responsive***: screen should be responsive as described in the ***Tuiter*** section
10. ***General***: general look and feel of Web page should look and feel as shown below

Use the screen capture below as a guide of what the resulting screen should look like. Instead of the little bird icon on the top left, use the T icon used in the exercises.



# 6 Delivery

1. In the same React.js application created in earlier assignments, ***tuiter-react-web-app***, complete all the exercises described in this document
2. In a new branch called ***a4***, add, commit and push the source code of the React.js application ***tuiter-react-web-app*** to the same remote source repository in ***GitHub.com*** created in an earlier assignment. Here's an example of how to add, commit and push your code

| **$ git checkout -b a4**  **$ git add .**  **$ git commit -am "a4 bootstrap fall 2022"**  **$ git push** |
| --- |

1. Deploy the ***a4*** branch to the same ***Netlify*** project created in an earlier assignment. Netlify should be configured to deploy all branches to separate URLs. From your Netlify's dashboard go to ***Site settings > Build & deploy > Branches > Branch deploys*** and select ***All***. Now each time you commit to a branch, the application will be available at a URL that contains the name of the branch
2. As a deliverable in ***Canvas***, submit the following URLs
   1. The source repository in GitHub.com. It should look something like  
      [**https://github.com/jannunzi/tuiter-react-web-app.git**](https://github.com/jannunzi/tuiter-react-web-app.git)
   2. The React.js application running on Netlify. It should look something like  
        
      [**https://a4-loving-torvalds-effde8.netlify.app/**](https://a3-loving-torvalds-effde8.netlify.app/)  
        
      Note that the URL contains the name of the branch