Instructions

* Add a "lesson-5" folder to your "assignments" folder.
* Within the "lesson-5" folder, add the standard "images" and "css" folders to contain your image objects and css files respectively.
* Add a new HTML file to the "lesson-5" folder named "preston-5.html"
* Copy your HTML markup in your "town-template.html" from the previous week into the "preston-5.html" file. Note any feedback received from the instructor.
* Delete the CSS file link in the <head> from the previous week.
* Add a "normalize.css" file link to your "css" folder and be sure to list it before any other CSS file links. It is OK to use a CDN.
* Add two separate CSS files for small and large views named "small-5.css" and "large-5.css" and store these files in the "css" folder.
* Copy your CSS markup from the previous week's css file into the "small-5.css" file.
* Add links to these two new CSS files after your "normalize.css" reference in your "preston-5.html" page <head>.  
  Most of your CSS markup will be in the small CSS file and necessary changes will be made to the large view.   
  Use a media query to contain the rules in the large CSS file using a reasonable breakpoint. The media query indicates at which point and above changes need to be made to declarations inside your rules. Do not repeat declarations. Only make the necessary changes. See this [example using a media query to change the navigation menu (Links to an external site.)Links to an external site.](https://codepen.io/blazzard-jason/pen/gmGxaj).

Discuss the media query breakpoint value between the small and large view in the [Developer Forum](https://byui.instructure.com/courses/50036/modules/items/3954732). For this project, think of the small CSS file as the main CSS file. The small CSS file rules and declarations are not encased in a media query. The large view should be considered to be anything larger than a phone view at this point in the course.

* Ensure that the HTML head contains all the required content as presented to this point in the course including meta tags (charset, viewport, description) and [a proper title (Links to an external site.)Links to an external site.](https://moz.com/learn/seo/title-tag).
* Continue to use a Google Font API reference or equivalent that meets the specifications of your site plan typography design.
* Use this [CSS Navigation Bar tutorial (Links to an external site.)Links to an external site.](https://www.w3schools.com/css/css_navbar.asp) *(see the Horizontal Navigation Bar Examples section)*for creating a horizontal menu bar for the main menu in the large view.   
  The small view should hide all the navigation links and introduce a menu link only *(see the example layout)*.  
  The horizontal bar should only be applied to the **large** view as per the sketches.   
  *You will work on the responsive menu for the small view in the next week.*
* Design a hover effect on the navigation items <a> tags to change in some way conducive to your design to highlight the current menu item that is being hovered over with the mouse pointer.
* You should continue to employ a wayfinding technique to visually note which page is currently loaded in the navigation items. Create a ***active***class and assign it to the list item <li> for the Preston town page. Give that class a different but aesthetically congruent look and feel.
* As a development specification, do **not** show the five-day forecast in the small view.
* You will need to fill in the content areas with viable content. With images, it is OK to use [free stock photos (Links to an external site.)Links to an external site.](https://www.pexels.com/) or your own high-quality photos.
* Provide proper markup indentation to the markup for human readability.
* Remember to follow the PARC design and typography principles in your design and layout.