**Chapter 6:**

Intro to Bootstrap

Bootstrap is a Frontend framework/component library

It gives you preset classes which you can use by inserting the class name to your elements.

Responsive – adjusts to view port. Adaptable layout

CDN (content delivery network) = multiple areas to access content. Cuts down on latency to reach a website if it’s hosted in multiple locations. You will access the closest area.

You can load bootstrap onto your website by linking to the bootstrap CDN, by using the starter template from your CDN (easiest), or by copying the bootstrap CSS code into your website (slowest)

**How to create a website:**

Observe other websites for inspiration/ideas

Wireframe – low fidelity (just pen and paper quick sketch)

balsamiq.cloud – website to create wireframes

Mockup – high fidelity (what you see is what you get)

Prototype – create your website

**Columns** are divided in 12 units in bootstrap

You can specify how much units

<div class=”col-(#)”>

If you do not specify a unit #, it will divide the space evenly among your columns

You can specify unit numbers for different viewports

<div class=”col-md-2 col-sm-3 col-12”>

The smallest size column applies to all larger sizes unless otherwise specified.

If you have multiple CSS stylesheets applied, the one coded in LAST will be the one applied to your website. Order matters. Add the generic style sheet first then your more specific style sheets last.

**Class=”container” bootstrap element:**

Centers your container elements. container-fluid does not automatically adjust in sizes as viewport changes in size.

**Chapter 7:**

**Carousel:**

Can have automatic or manual carousels.

Can add buttons to move forward and backward

Can pause if mouse is hovered.

Can specify delay time before moving from one item to another.

Anchor tags can be used as buttons for forward and backward buttons.

**Card:**

Useful to display pricing (contains card-header, card-body, card-footer)

Wrap in card-deck for better formatting on 1 row.

Wrap instead with row/column to display better with different viewports

**Hierarchy:**

Element written last is applied last (final edit)

Element applied lower in code is higher up on Z-axis (more forward)

**Z-index** allows you to customize hierarchy

Only works with position relative, fixed or absolute

Higher numbers are placed further infront (more towards viewer)

Default z-index is 0.

**Media Query:**

Allows different CSS rules to apply given different changes to the media.

@media <type> <feature> {change something}

If the media is of type <type> of at least <feature> then {change something}

@media screen (min-width: 900px) {change something}

You can link to id or website.

**Code refactoring (tidy up your code):**

1. Readability
2. Modularity (organizing code in understandable sections)
3. Efficiency
4. Length (don’t sacrifice readability/modularity)

**Combining selectors:**

Multiple selectors can be selected by separating with a comma

H1, h2, h3, h4, h5, 6h {CSS code}

Can have space between commas.

Hierarchical selectors can be used to specify selectors given a certain condition

Selector1 selector2 {CSS code}

Selector1 is parent, selector2 is child. Must be separated with space

Combined selectors

Selector1.selector2 {CSS code}

Elements with both selector1 and selector2 are affected. Must not have space