Lesson plan with instructor notes sample 03/08/2018
Topic: HTML, CSS, JavaScript. Putting them all together
Product: Crypto currency price and alert site. CryptoTicker

Prerequisite: Students are familiar with HTML and CSS styling from previous classes.

Time	Outline	Topics and Notes	Links, Tools and Q&A
	Intro – Understand the problem/ client needs	Lesson Topics: • Lists: ordered and unordered lists • Create buttons to interact with JS file. • id and span • Putting HTML and JavaScript together: 　◦ Use JS to add functionality to buttons 　◦ onclick [get const values from JS file] 　◦ document.getElementById 　◦ Declare constant values 　◦ Conditional statement Extensions/Next class: • CSS = styling the look of the site.	Lesson Link: https://github.com/thelandlords/DC-SampleLesson Ordered list and ordered list types https://www.w3schools.com/html/html_lists.asp Use JS functions to change HTML https://www.w3schools.com/js/js_statements.asp JS - style HTML objects https://www.w3schools.com/Jsref/dom_obj_style.asp
	Basics – Planning the overall big picture	Pseudocode/diagram the program: • Title • place to show BTC and ETH prices • place to ask user for alert settings [high and low for both coins] • get price button and • save alert info button	What essential info needs to be displayed? What should we use to compare the user value to the current price? Conditional statement if
	Demonstrate – details of turning	Display essentials [HTML] *Use forks to show and type each stage of the code.*	Fork 1 = HTML https://codepen.io/thelandlords/pen/wmvJBo

	pseudocode into actual code.	1st: What any HTML needs: doctype, head, body, and usually needs h1...h6,
		
		2nd: The details our program needs:
		ordered lists
		unordered lists
		nested lists
		getting user input <input =... id=...>
		buttons {get price and store user input}
		
		3rd: JavaScript is needed for the conditional statement:
		    if{price>=alertHigh}
		        { {alert the user by changing the fields background color} }
			Fork 2 = JS
			https://codepen.io/thelandlords/pen/QmbvXY
			
			Fork 3 = CSS
			https://codepen.io/thelandlords/pen/zWGwgv
			
			GitHub – all code
			https://github.com/thelandlords/DC-SampleLesson/tree/master/code

	End Product – testing, and launching	Make sure it meets client requirements and has been end user error proofed.	How to improve and add features for the next version? Final product http://cryptoticker.glitchdoctor.net/
	Extensions/ Next class topics	• Style the site using CSS • Add ETH price and alerts option • Get data from API instead of using the const testing values. • Show and compare two or more cryptocurrencies. • Add a graph to chart one or more cryptocurrencies. • Give user alternate alert modes – sound, email, image, etc.	What are some mistakes a user may make when interacting with the app? How can we error proof the code so that a users mistake would not result in unintended app behavior?