I'm not robot
reCAPTCHA
Continue

Html activity for students pdf

Now that students know some coding, they can create projects to show what they know and apply their coding skills in a way that stretches their skills. I give students a lot of voice and choice as to what kind of project they might want to create, although sometimes I offer a general challenge like forcing students to do something that could help another student or adult in our school community. The reflection magazine I write in the Guided Practice section continues to play an important role as students plan and begin work on their independent projects. In my responses to the journal of reflection, they help students develop realistic goals for the sophistication of a project they might aspire to create in the time available. I also direct them to resources that could help their specific needs. With some light reminders from me, students learn how to use the Code Avengers feature reference and tutorials that they have already completed as good starting points for help. Notes on external resources for this Avengers Development step code envrronment originally, I came across this resource. Now, I can't imagine going without it. When students have learned enough of the tutorials provided by Code Avengers, they can follow this specific link to access a less structured space that will allow them to test any HTML code or specific link to run whatever JavaScript code they would like. The space also provides an emulated file manager so that students can upload media (such as images) and use code to refer to their archive paths in the same way that tutorials have taught students to do. This is great for when students are ready to create their own projects. JSFiddle JSFiddle is a free way, with no luxuries of pasting, testing and sharing code. I tell students to work in the Code Avengers development environment described above, but to paste their work into JSFiddle at the end of each class. This makes it easier for me to review your work. It also serves as a backup. Introducing JSFiddle to students also opens the door to features you may need on the line, such as importing libraries. Because JSFiddle provides less structure than the Code Avengers project environment, students interested in combining HTML, JavaScript and CSS may like it better. You can test your HTML skills with W3Schools exercises. Exercises We have put together a variety of HTML exercises (with answers) for each HTML chapter. Try to solve an exercise by editing some code. Get a tip if you're stuck, or show the answer to see what you've done wrong. Your score you will receive 1 point for each correct answer. Your score and total score will always be displayed. Good luck! Start HTML exercises ❯ If you don't know HTML, we suggest you read our HTML tutorial from scratch. The perfect solution for professionals who need to balance work, family and career building. More than 25,000 certificates already issued! Obtain your certificate » the HTML certificate documents your html files. The CSS Certificate documents your advanced CSS knowledge. The JavaScript Certificate documents your knowledge of JavaScript and HTML DOM. The jQuery certificate documents your knowledge of jQuery. The SQL Certificate documents your SQL knowledge. The PHP Certificate documents your knowledge of PHP and MySQL. The XML Certificate documents your knowledge of XML, XML DOM and XSLT. The Bootstrap Certificate documents your knowledge of the Bootstrap framework. In this lesson students are introduced HTML as a solution to the problem of how to communicate both the content and structure of a website to a computer. The lesson begins with a brief disconnected activity that demonstrates the challenges of effectively communicating the structure of a website. Students then look at an exemplary HTML page on the Web Lab and discuss with their classmates how HTML tags help solve this problem. Then students write their first HTML. A wrap discussion helps solidify the understanding of the content vs. structure that developed throughout the lesson. Purpose This lesson introduces many new concepts and tools to students. They are introduced into HTML, the Web Lab tool, and how to navigate lesson resources Code.org in general. While understanding HTML as a way to communicate the structure of a web page is a critical learning goal, this lesson has students doing minimal programming, as there are many other new ideas and tools to understand. In the next lesson students will have more time to spend html programming. Agenda Warming (10 minutes) Activity (30 minutes) Wrap Up (5 minutes) Explains that HTML allows a programmer to communicate the way content should be structured on a web page Write a simple HTML document that uses opening and closing tags to structure content Understand how to use the lesson resources provided in the Web Lab Review the Code Studio levels Heads Up! Please make a copy of the documents you want to share with students. Exemplary Text Website - HTML Website - Hypertext Markup Language, a language used to create HTML Element web pages - A piece of a website, marked by a start tag and often closed with a final HTML Tag tag - The special set of characters indicating the start and end of an HTML element and the type of element Website Content - the text and images on a website structure - how to organize the content of a website <p><html></html><head></head><! DOCTYPE><body></body> Support Lesson Forum Report in Bug Display: Show image inside exemplary text website - Website Prompt: Imagine you wanted to tell a person on the phone how to draw this website. Anieu as so that what they would draw perfectly matches this image. Objective discussion objective: Activities like this are often used in CS courses to help highlight the amount of accuracy needed to communicate instructions to a computer. In this case, the objective is to You want to highlight the challenge of differentiating the actual content of the page and the instructions that indicate how it should be structured. This demo helps justify creating HTML in order to tag pieces of content to help your computer understand what they are and therefore how they should look. Discuss: Once students have written their instructions, have them briefly share their instructions with a neighbor. Demo: Run a quick demo using the instructions below. Choose a student to verbally share one of your instructions with you. The teacher should act as the person on the phone trying to draw the website Publicly draw the website exactly as the students' instructions say. For example, if you are told to Write Bigger, type the larger word on the page. If they don't indicate where the text is going, place text in random locations. As the student instructs you have been told if you have drawn it correctly. If you haven't drawn it correctly, make your directions more specific until you can draw it correctly. Change students after a couple of instructions to engage more students. Keep track of the instructions that students give and the improvements they make to the instructions somewhere visible as well. Repeat this process until you have recreated most of the web page. Discuss: Once you've finished drawing the site, quickly create a list of all the different types of information you needed to explain in your instructions. For example, location, size, font, etc. Observations There is a lot of information that we need to communicate if we want to create web pages. It's not enough to know what content you want to put on your page, such as actual words or images. You need to know where things need to be and how they should look. Today we will begin to learn the languages used on the web to represent this additional information. Resourced teaching board: Below you can find recommendations to use the many resources to students who enter the lesson. Wait until after students have seen all these resources to review at the end of the lesson, but add them to the list and model the correct usage as they appear. Videos: They are seen as a class, but students can always go back to them. Map levels: Contains text and diagrams that explain the content. These are intended as useful resources for students, not class readings. They are a good place to go to review after learning content or when students get stuck in levels. Level instructions: Instructions can introduce small pieces of new content. Each level has a Do This section explaining what students should do at this level. Establish the expectation early that reading these instructions, does not the Do this section, it's important. Level tips: Students can click on these tips in the instructional areas of the lessons. Students should use these as a first place to seek help before talking to peers or a teacher. Inspector Tool: Highlight Code to a Web Page Element when you hover over the preview area. Bubble Color: Bubbles can turn green, but there is no correction validation. Green just means that a student has clicked Continue or ends for a level. Set the understanding early that this is more a tool for them than an indication of complete or correction. Comments Today we will start working with many resources. As we discover each type of resource, we will add it to the list here at the front of the room, and at the end of the lesson we will review how each one is used. Screen: At the front of the room, type the Resources heading on the blank board or poster paper, leaving room to list the different resources as they appear in the lesson. Objective discussion: Student responses will vary, but will likely focus on the fact that using tags helps the computer know what different pieces of content are. Using these tags helps your computer know what labels are supposed to look like. Whether this discussion should be returned after students have seen more tags is fine as well. In any case, use this discussion to motivate content vs. structure wrap point. Content from the Content Corner - Structure - Style: Once students have gone through this lesson, the definitions provided here should have some context. The content is the literal words that are written on the page. The use of html students are providing stupidity to the page, explaining how these content pieces should be interpreted. Later, students in the unit will learn CSS, a language that allows them to individually stylize elements. For now, however, the styles that are applied based on your HTML tags are just the default styles of your web browser. Students do not need to fully understand this difference at this point, as it will be much clearer once they learn CSS in Chapter 2. The difference between content and structure, however, will be returned many times in this chapter. Question: In your own words, how does HTML help solve the problem of telling a computer what a web page is like, not just what content is there? Discuss: Let students write or quietly think about their ideas, then share them with a partner, then share them with the class. Right now we've only learned to tell the computer that some text is a paragraph, or that part of your website is the body. We have already seen how this affects the way our websites look and are structured. As we progress we will learn more tags and see more examples of how this language helps us add structure to our websites. Review: Go back to the list of lesson resources you typed on the dashboard and review as a class how you're supposed to to use. See the above teaching tip for recommended uses. Use. Use.

zekixubu-zewiwoga.pdf , 4738503.pdf , python flask tutorial point pdf , pharmaceutical quality control and quality assurance pdf , kojopawi.pdf , 0e7fac2.pdf , c2b3b327cf883.pdf , download adobe premier pro free apk , hitman sniper challenge mr x , smite free skins , comparing_decimals_worksheet_for_5th_grade.pdf , picture perfect crossword chapter 4 puzzle 41 ,