



FULL STACK DEVELOPMENT



FULL-STACK HOME ASSIGNMENT

Congratulations on making it this far!

Some words of reassurance, you've made it to this point in the application process because you showed us that you have the capability to self-study basic technologies and organize your time effectively. We believe you have what it takes to make the most of the training and resources you were given and can reach the ultimate goal of working in the high-tech industry in web development.

Throughout the program we will be learning all about how independently developers in the high-tech industry work and learn how this is a skill you can develop. That independent learning is exactly what you are going to start practicing to complete this assignment.

Some important information before you start:

- Please take into account that the pre-course assignment takes 20-30 hours to complete.
- We have included video resources to help you, feel free to use them and more if needed.
- The assignment portion is what you will find here. Don't worry if it takes you a while! That is normal and exactly why we give you so much time to complete it.
- Once you have completed the assignment, you are ready for the final stage - the interview! You will submit your code and go over it with a technical mentor to make sure you wrote your own code and that you understood it well enough to continue your studies.
- Throughout this assignment, you will be learning HTML, CSS, and basic JavaScript. Each of these topics are equally important and will be evaluated during the interview.
- You will be required to use your problem-solving skills in order to complete the assignment. This means using the resources ITC have provided you, and using Google to your advantage.

If you have any questions, full time students can email the Program Manager Wanda Holsman at wanda@itc.tech, with the subject line PRECOURSE. Recommendation: Try to be specific in your questions and avoid questions like 'which video do I need to watch'.

We wish you lots of luck, and we believe in you!



BEFORE YOU START

- There is useful information spread across the totality of this document. Make sure you read all of it, including the FAQ.
- Make the best out of it.
- The design and implementation of the style in the HTML pages is up to you. You are required to make it look 'designed'. That means, elements should be properly aligned, colors should be consistent across the pages, and needs to be pleasant to the eye. You can look for inspiration online.
- For that reason, we attach an example to provide guidance (see 'Example' folder in the zip provided). It doesn't mean it needs to look exactly that way, rather it is provided 'as is'.

Your first assignment is to build the front-end of your personal portfolio website. We provide you with a starter code base and a list of tasks to complete, each of which require you to add to or modify the starter code. The starter code is located in this folder:

<https://drive.google.com/file/d/1im6TXY2rTsBQszxeMGUuLAQ1IIPiuji/view>

To prepare for the assignment, we suggest you watch these videos and resources:

>> for HTML

<https://www.htmldog.com/guides/html/beginner/>

<https://www.w3schools.com/html/>

▶ HTML & CSS Crash Course Tutorial #1 - Introduction

>> for CSS

<https://www.w3schools.com/css/>

<https://css-tricks.com/css-beginner-mistakes-1/>

▶ HTML and CSS for Beginners Part 1: Introduction to HTML

>> for Javascript

<https://www.w3schools.com/js/>

▶ Learn JavaScript - Full Course for Beginners

>> Interacting with the DOM (How to manipulate HTML/CSS using Javascript)

https://www.w3schools.com/js/js_htmlDOM.asp

▶ [Learn DOM Manipulation In 18 Minutes](#)

>> Chrome developer tools

▶ [Chrome Developer Tools \(Devtools\) Tutorial](#)

The resources teach the fundamentals for front-end development, particularly HTML, CSS, and JavaScript. **The tasks in the ITC assignment are based on topics covered here.** You don't have to learn only from these resources and not everything... use this as a toolkit you can use and will help you out. You can find more content suitable for you, and use these resources to complete your assignments.

Although we encourage you to actively build more stuff based on topics you will learn but you do not need to submit them. The only work you need to submit is the ITC project. For the ITC project you submit to us, please **make sure you understand all of the code** (line-by-line) in the assignment. You will have to explain your code to an ITC staff member over live video in a technical interview. You may want to practice explaining your code out loud before your technical interview.

Please name the folder as follows: **LastName_FirstName_ITCPrecourse** (e.g., Doe_Jane_ITCPrecourse).

Please do not share your code. You need to keep it entirely confidential. Moreover, although you may use the internet to look up information, you are not permitted to work with others; all work must be done independently (remember, you're going to have to explain your work).

Good luck and have fun!

FREQUENTLY ASKED QUESTIONS (FAQ)

1. How do I fix the layout and styles on my page?

The resources we provided to you, will give you a great cover for HTML and CSS. If the content in those videos do not solve your problems, find examples on other places on the web, just a simple search on google can get you far (BTW, StackOverflow is a wonderful friend). We suggest starting with those and playing around with it before you start the assignment.

To find the answers you're looking for, try Googling a description of your problem. For instance, "how to create a fixed navbar with items centered in HTML CSS" or "how to horizontally align HTML elements using CSS."

Many ways exist for organizing and styling HTML elements. You might need to wrap some of your HTML elements inside of other elements, use certain style properties to direct the browser to override an HTML element's default style properties, or maybe some combination of those and other things. Remember to keep it simple and focus on the things that you know how to do.

2. How do I fix my JavaScript code that's not working?

As you write your JavaScript code, at each step along the way, use **console.log** to better understand what is happening. Make sure you remove those before submitting. You can also learn how to use the debugger (search online). For instance, if you're using a variable as a counter each time a button is clicked, console log the variable so that you can see the value change with each button click. Or if you're pulling an HTML element into a JS file, console log it to make sure it's working like you think it should. Once you finish, remove those.

INSTRUCTIONS

Download the starter code, and begin your work using your favorite editor or IDE (throughout the program we will primarily be using [Microsoft Visual Studio Code](#), but you are free to use other tools such as WebStorm or Atom). For a quick walkthrough on installing Visual Studio Code, you can watch the operating system-appropriate "Installing Visual Studio Code" video [here](#).

INDEX.HTML

1. In index.html, change the text of the element with id="profile-name" from "Israel Tech Challenge Pre-Course" to your full name. Next, add a subtitle below the title. The text of the subtitle should say "ITC Fullstack Development." For the font family of your project, choose a font from Google fonts (<https://fonts.google.com/>) or attach one of your choice to the styles. Otherwise, add styling of your choosing to the page. Throughout this assignment, please add your code for the styling to the "style.css" file. Also, throughout this assignment, we expect you to make your styling look **professional** and **eye-catching**.
2. In index.html, remove the "Visit ITC" link. Add a navigation bar at the top of the page. This navigation bar should be fixed to the top, always taking 100% of the width, and no matter if you scroll, it should always be visible. The navigation bar must have three links: "Home", "About", and "Contact". Use an <a> tag to link to the pages. Please add the style of your choosing to the navigation bar and links.
3. Add three clickable elements nested inside the element that has a 'card' class. These three clickable elements should be inside an unordered list as list elements. Each one of this list elements should be clickable. Link one to your LinkedIn account. Link another to your Github account. Link the third to your StackOverflow account. If you don't have accounts for each of those, please create them (www.linkedin.com, www.github.com, www.stackoverflow.com). Make sure you have a photo uploaded to your account profile for Github. For the clickable elements, use icons from Font Awesome (www.fontawesome.com).
4. In main.js, update the value of GITHUB_URL to your Github profile. Make sure you write the proper link. Then, change the code so that it displays your Github profile picture instead of the ITC Github profile picture.
5. On your home page, add functionality to the card containing your profile photo and buttons. When the user hovers over the card, have the card respond with some sort of animation. Trigger a second, more elaborate animation, upon clicking the card.

See the image 'index.png' in the 'Examples' folder for reference.

There should be two animations in total, one on hover, and one on click.

CONTACT.HTML AND ABOUT.HTML

6. In these two files, add the same navbar from your home page to each of the new pages you created. Add some basic HTML to each page so that the about page has a title that says "About" and the contact page has a title of "Contact".
7. In your navbar, connect the navbar links to their respective HTML pages. When you click on the links, your browser should display the corresponding HTML page.

ABOUT.HTML

8. On the "About" page, create a section that includes the following information: describe why you are interested in coding, your educational and career experiences, and your personal interests. Please style and layout the page as you choose.
9. Add another section below what you created in the previous step. In this section, write where you were born and the cities in which you have visited. There should be at least 3 cities in total. If you have visited two cities or less, complete the list with other cities you would like to visit. (you can also use google maps and embed a map onto your webpage (!!)) to show us exactly where you were - you can search online how to do this)
10. On all of your pages, put a footer at the bottom of the page. The footer should contain the same links as your navbar. It also should contain a sentence saying: "This page was built using: HTML, CSS and JavaScript". The words "HTML", "CSS", "JavaScript" should be stored in an array like so:

Generate the sentence on the website rather than hard-coding it. Ensure that the word "and" is included in the sentence but not in your array. Structure the array so that it accommodates additional languages without requiring extra code modifications. Your solution should seamlessly work with the following arrays:

- a. `const codingLanguages = ["HTML"]`
- b. `const codingLanguages = ["HTML", "CSS", "JavaScript"]`
- c. `const codingLanguages = ["HTML", "CSS", "JavaScript", "Python", "Java", "C++"]`

The original array should remain unaltered, without any added spaces, commas, or other characters. The generated sentence should include spaces after each comma, except for the last element, which should be preceded by " and." For example, `["HTML", "CSS", "JavaScript"]` should yield: "This website has been created with: HTML, CSS, and JavaScript."

CONTACT.HTML

11. Create a new HTML file named `Contact.html` and also a JavaScript file `Contact.js`.
12. Create a form on the Contact page with the following fields:
 - a. Full Name (input)
 - b. Email Address (input)
 - c. Message (textarea)
 - d. Submit Button (button)
13. Style the form to make it visually appealing and user-friendly. Ensure that all form elements are properly aligned and have suitable spacing.
14. Implement basic client-side validation for the form:
 - Ensure the Full Name field is not empty.
 - Validate that the Email Address field contains a valid email address format (e.g., `user@example.com`).
 - Display an error message if any of the above criteria are not met.

BEFORE YOU SUBMIT

Before you submit, make sure that you check all the following boxes:

- ☐ You have read the FAQ.
- ☐ Submission is compressed in zip format (not .rar)
- ☐ Your submission file name has the following format: **LastName_FirstName_ITCPrecourse.zip**
- ☐ In your HTML files, you don't have any undesired overflowing content. (What is overflow? [link](#))
- ☐ All your files are properly formatted (indented)

If you don't know how to do it, Visual Studio Code has a tool for it:



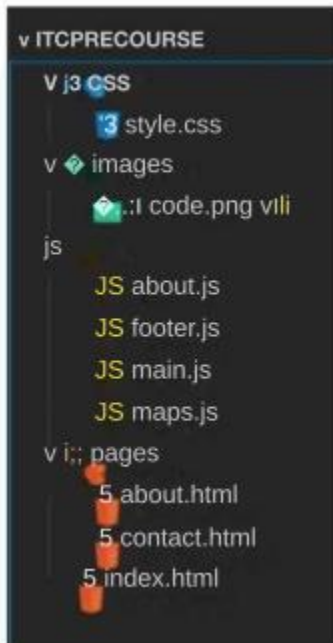
- ☐ All your links are relative rather than absolute: (Want to know more? [link](#))

```
<div class="navlinks">
  <a href="./index.html">Home</a>
  <a href="./about.html">About</a>
  <a href="./contact.html">Contact</a>
</div>
```


- ❑ Your HTML files don't contain errors. You can use [this website](#) to check that your HTML files are valid, as well as any other tool of your convenience.
- ❑ You are able to explain in depth the totality of your code.
- ❑ Your JS files don't create errors in the console (F12 on browser):

✖ ▶ Uncaught TypeError: Cannot read property 'addEventListener' of null [contact.html:194](#)
at [contact.html:194](#)

- ❑ Your files and folders are structured in an organized way, like the following (but not exclusively):



SUBMIT YOUR WORK

Check your work, zip it up (remember the specified format), [submit it and schedule your technical review](#)! Please ensure you are prepared to explain and answer questions on the code submitted.

Link to submit and schedule review:

<https://app.acuityscheduling.com/schedule.php?owner=12116384&appointmentType=1155132>

Looking forward to hearing from you!