



HealthOS Ltd.

Round 1: Basic Technical Task

Task 1 Overview

In this task you will manipulate the DOM tree. You will use JavaScript to take user input from a form and create and add new content inside a div element on the same page. This time you will have to use a *template* tag which defines the structure of the new content. Again the HTML is already given inside the *index.html* file. Your JavaScript code should be saved to *template.js* file inside the same directory.

Implementing the Task 1

In this exercise you will input contact information of people in a form and then append that information to the end of the contacts listing.

- Write your code into *template.js* file inside this directory.
- You can find the HTML in the *index.html* file inside this directory. Make note of the IDs and classes of the elements in the HTML. You can use them in your code to get hold of the elements. **DO NOT EDIT THIS FILE!**
- Contacts will be added inside the div element with id *contacts*
- Each individual contact will have the HTML structure of the template tag with id *contact-template*.
- Attach an event listener to the *form* element's *submit* event. The event listener is responsible for taking the input from values from the form and *appending* them inside the div with id *contacts*. The event listener should perform the following operations:
 1. First cancel the default action of submitting the form.
 2. Take the contents of the template tag and clone them to a new DOM node.
 3. Replace the contents of the cloned template contents with the contact information from the form.
 - Replace the h2 heading with the name of the contact.
 - Replace the text inside the paragraph element with class *email* with the email address of the contact.
 - The last paragraph with class *homepage* has an anchor (a) tag inside it. Replace the anchor's target address (href attribute) and the text inside the tag with the homepage address of the contact.All of the above updates should be done on the cloned node. **Do not change the original template element or its contents!**
- 4. Append **the cloned node** to the end of the div with id *contacts*
- 5. Reset the form to its initial state.

Task 2 Overview

Make an app using Next JS(Use any free API for fetching data) that displays a weekly weather forecast. The core requirement for the app is usable and clear user interface: users must get desired information (the weather forecast) just when they go to the app.

Users are more interested in two questions: what should I wear today and should I take an umbrella? So, it's good to start just with temperature and precipitation. Drop the information about wind speed, cloud cover, the time of sunset and sunrise (of course with cool animation that definitely will take a few hours of development), etc. Let's say you get it and now you need to show the weather for today.

The basic element to display weather forecast here is a day weather card. All cards with daily forecasts must be structured and well organized, so prepare to unleash your potential as an expert designer too.