# JS Advanced - Exam: 08.04.2020

Exam problems for the [“JavaScript Advanced” course @ SoftUni](https://softuni.bg/courses/javascript-advanced). Submit your solutions in the SoftUni Judge system at <https://judge.softuni.bg/Contests/xxxx/>.

## Problem 1. Task Manager

Use the **index.html** and **app.js** files to solve this problem. You have **NO permission** to directly change the given HTML code (index.html file).



### Your task

**Write the missing JavaScript code** to make the **Task Manager Functionality** works as follows:

When the Add button is clicked, first you need to validate the inputs. If any of the input fields is empty, the function doesn’t make anything.

After validating the input fields, you need to add the new task (article) in “Open” section.

The HTML structure looks like this:



The article should have two buttons “Start” and “Delete”. Be careful to set the classes for the buttons and the parent-div.

When the “Start” button is clicked, you need to move the Task in the section “In Progress”. Be careful with the buttons! The HTML structure looks like this:



When the “Delete” button is clicked, the Task (whole article) should be removed from the HTML.

After clicking the “Finish” button, the Task will be completed, and you should move the article in the section “Complete”. The buttons with their parent div-element should be removed.





### Submission

Submit your solution into a function.



### // без 7-ми тест

function solve() {

    const [addTask, open, inProgress, complete] = document.querySelectorAll('section');

    document.getElementById('add').addEventListener('click', onSubmit);

    const task = document.getElementById('task');

    const description = document.getElementById('description');

    const date = document.getElementById('date');

    function onSubmit(event) {

        event.preventDefault();

        if (task.value !== "" && description.value !== "" && date.value !== "") {

            const article = document.createElement('article');

            const h3 = document.createElement('h3');

            h3.textContent = task.value;

            const pDescr = document.createElement('p');

            pDescr.textContent = `Description: ${description.value}`;

            const pDate = document.createElement('p');

            pDate.textContent = `Due Date: ${date.value}`;

            const divClass = document.createElement('div');

            divClass.className = "flex";

            const buttonGreen = document.createElement('button');

            buttonGreen.className = "green";

            buttonGreen.textContent = "Start";

            //start button eventListener

            buttonGreen.addEventListener('click', startOnClick)

            const buttonRed = document.createElement('button');

            buttonRed.className = "red";

            buttonRed.textContent = "Delete";

            //delete button eventListener

            buttonRed.addEventListener('click', redOnClick)

            divClass.appendChild(buttonGreen);

            divClass.appendChild(buttonRed);

            article.appendChild(h3);

            article.appendChild(pDescr);

            article.appendChild(pDate);

            article.appendChild(divClass);

            const openDivSection = open.querySelectorAll('div')[1]; //the second div in section Open

            openDivSection.appendChild(article);

        }

        function startOnClick(event) {

            const article = event.target.parentNode.parentNode;

            const firstBtn = article.querySelectorAll('button')[0];

            const secondBtn = article.querySelectorAll('button')[1];

            firstBtn.className = "red";

            firstBtn.textContent = "Delete"

            firstBtn.addEventListener('click', (event) => {

                const article = event.target.parentNode.parentNode;

                article.remove();

            })

            secondBtn.className = "orange";

            secondBtn.textContent = "Finish"

            secondBtn.addEventListener('click', (event) => {

                const divClassFlex = event.target.parentNode;

                const redBtn = divClassFlex.querySelector('.red');

                const orangeBtn = divClassFlex.querySelector('.orange');

                redBtn.remove();

                orangeBtn.remove();

                complete.children[1].appendChild(article);

            });

            inProgress.children[1].appendChild(article);

        }

        function redOnClick(event) {

            const article = event.target.parentNode.parentNode;

            article.remove();

        }

    }

}

### // 7 тест излиза

function solve() {

    const [addTask, open, inProgress, complete] = document.querySelectorAll('section');

    const task = document.getElementById('task');

    const description = document.getElementById('description');

    const date = document.getElementById('date');

    const addBtn = document.getElementById('add');

    addBtn.addEventListener('click', add);

    function add(e) {

        e.preventDefault();

        if (task.value === '' || date.value === '' || isNaN(Date.parse(date.value))) { // check if all fields are valid

            return;

        }

        let article = factory('article', {}, '');

        open.children[1].appendChild(article); // open.children[1] is the second div in section Open

        let setTitle = factory('h3', {}, `${task.value}`);

        article.appendChild(setTitle);

        let setDescription = factory('p', {}, `Description: ${description.value}`);

        article.appendChild(setDescription);

        let setDate = factory('p', {}, `Due Date: ${date.value}`);

        article.appendChild(setDate);

        let btnsDiv = factory('div', { class: 'flex' }, '');

        article.appendChild(btnsDiv);

        let btnStart = factory('button', { class: 'green' }, 'Start');

        btnsDiv.appendChild(btnStart);

        btnStart.addEventListener('click', start);

        let btnDelete = factory('button', { class: 'red' }, 'Delete');

        btnsDiv.appendChild(btnDelete);

        btnDelete.addEventListener('click', deleteTask);

        task.value = '';

        description.value = '';

        date.value = '';

    }

    function deleteTask(e) {

        e.target.parentNode.parentNode.parentNode.children[0].remove();

    }

    function start(e) {

        const currentArticle = e.target.parentNode.parentNode.parentNode.children[0];

        inProgress.children[1].appendChild(currentArticle); // inProgress.children[1] is the second div in section inProgress

        let btnStr = currentArticle.children[3].children[0];

        let btnDel = currentArticle.children[3].children[1];

        btnStr.remove();

        btnDel.remove();

        const btnDelNew = factory('button', { class: 'red' }, 'Delete');

        currentArticle.children[3].appendChild(btnDelNew);

        btnDelNew.addEventListener('click', deleteTask);

        const btnFinish = factory('button', { class: 'orange' }, 'Finish');

        currentArticle.children[3].appendChild(btnFinish);

        btnFinish.addEventListener('click', finish);

    }

    function finish(e) {

        const currentArticle = e.target.parentNode.parentNode.parentNode.children[0];

        complete.children[1].appendChild(currentArticle); // complete.children[1] is the second div in section Complete

        currentArticle.children[3].remove();

    }

    // function for creating DOM elements with attributes

    function factory(type, attributes, text) {

        const el = document.createElement(type);

        for (let key in attributes) {

            el.setAttribute(key, attributes[key]);

        }

        if (text) {

            el.textContent = text;

        }

        return el;

    }

}