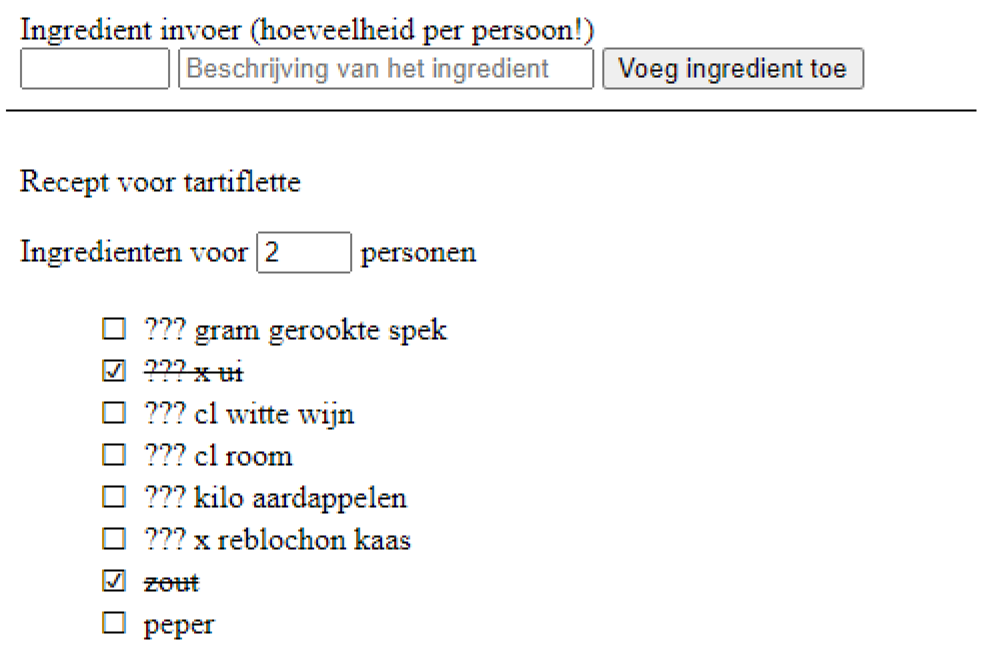
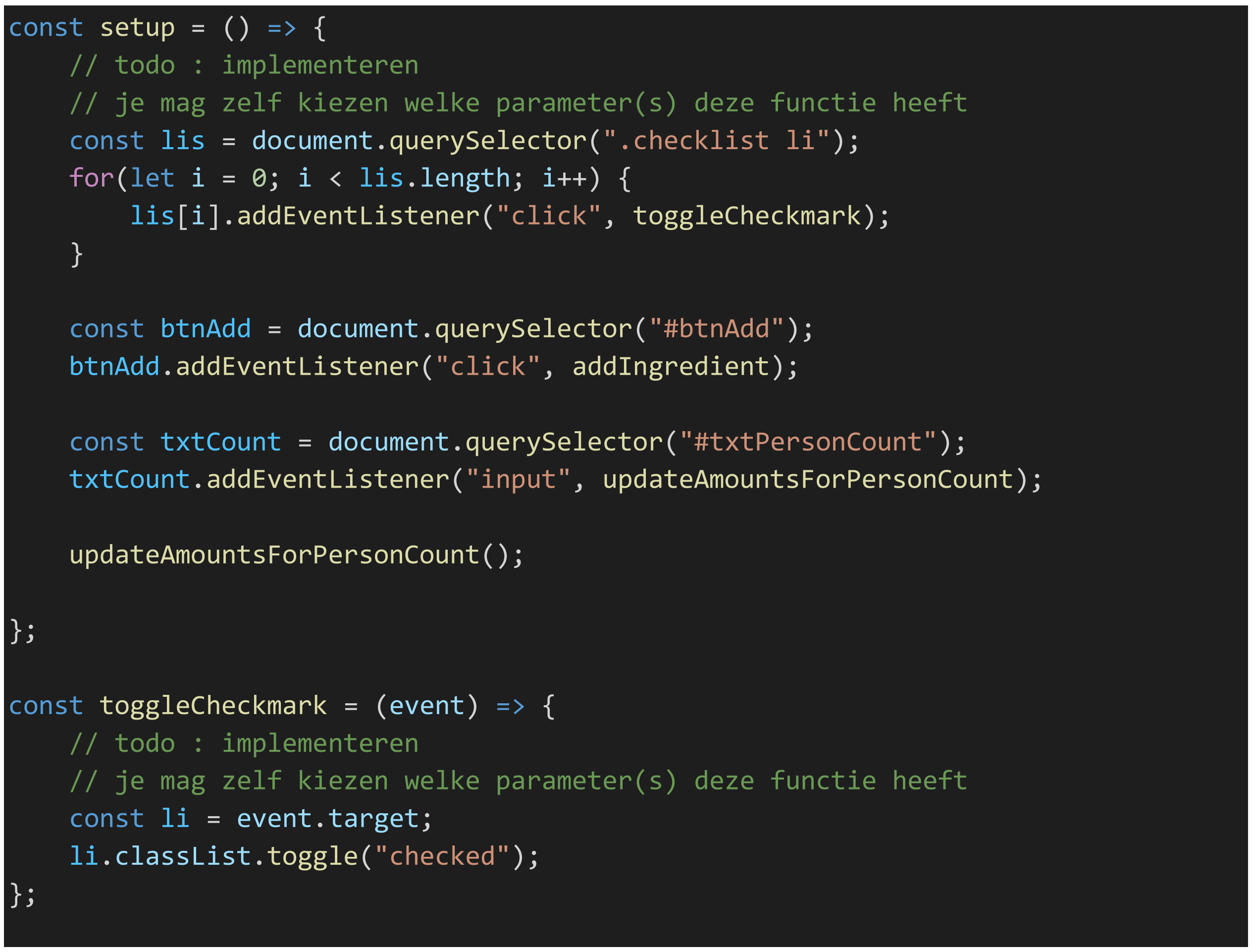


Ingredient invoer (hoeveelheid per persoon!)

Recept voor tartiflette

Ingredienten voor 2 personen

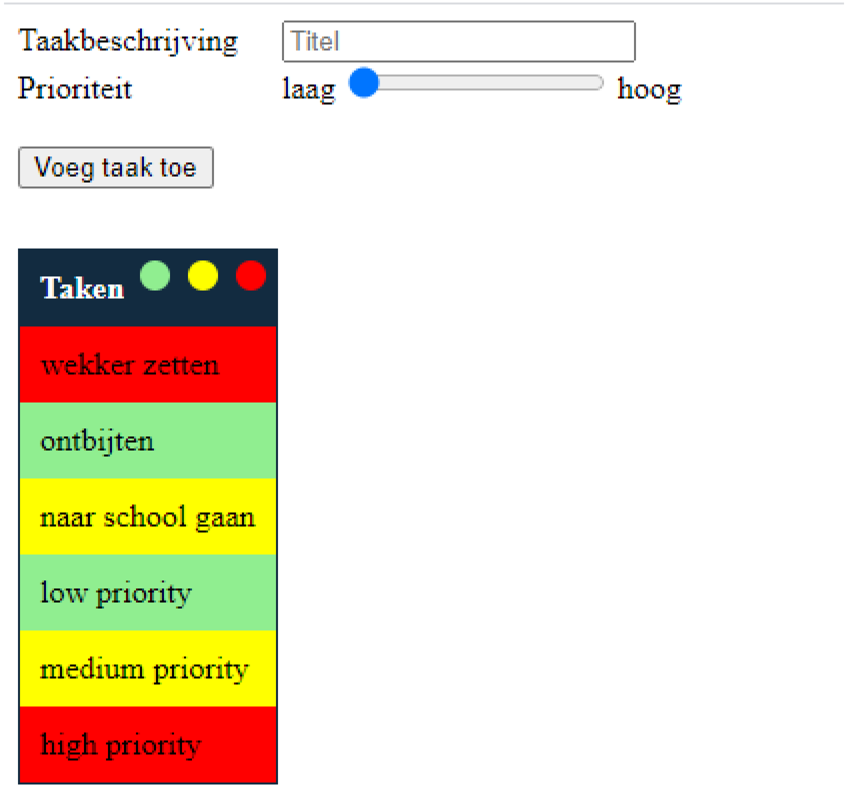
- ☐ ??? gram gerookte spek
- ☑ ~~??? x ui~~
- ☐ ??? cl witte wijn
- ☐ ??? cl room
- ☐ ??? kilo aardappelen
- ☐ ??? x reblochon kaas
- ☑ ~~zout~~
- ☐ peper



```javascript
const setup = () => {
    // todo : implementeren
    // je mag zelf kiezen welke parameter(s) deze functie heeft
    const lis = document.querySelector(".checklist li");
    for(let i = 0; i < lis.length; i++) {
        lis[i].addEventListener("click", toggleCheckmark);
    }

    const btnAdd = document.querySelector("#btnAdd");
    btnAdd.addEventListener("click", addIngredient);

    const txtCount = document.querySelector("#txtPersonCount");
    txtCount.addEventListener("input", updateAmountsForPersonCount);

    updateAmountsForPersonCount();

};

const toggleCheckmark = (event) => {
    // todo : implementeren
    // je mag zelf kiezen welke parameter(s) deze functie heeft
    const li = event.target;
    li.classList.toggle("checked");
};
```

```javascript
const addIngredient = () => {
    // todo : implementeren
    // je mag zelf kiezen welke parameter(s) deze functie heeft
    let amountPerPersontxt = document.querySelector("#txtAmountPerPerson").value;
    let description = document.querySelector("#txtDescription").value;

    let lstIngredients = document.querySelector(".lstIngredients");

    if(amountPerPersontxt.length>0){
        lstIngredients.insertAdjacentHTML("beforeend", `<li><span class ="amount"
 data-amount-per-person="${amountPerPersontxt}">???</span> ${description}</li>`);
    }else{
        lstIngredients.insertAdjacentHTML("beforeend", `<li>${description}</li>`)
;
    }
    let li = lstIngredients.lastElementChild.addEventListener("click", toggleChec
kmark);
    updateAmountsForPersonCount();
};

const updateAmountsForPersonCount = () => {
    // todo : implementeren
    // je mag zelf kiezen welke parameter(s) deze functie heeft
    let personCountAsText = document.querySelector("#txtPersonCount").value;
    let personCount = Number.parseInt(personCountAsText,10 );

    if(!Number.isNaN(personCount)){
        let spans = document.querySelectorAll(".lstIngredients .amount");
        let span = spans[i];
        let amountAsText ="###";

        let amountPerPersonAsText = span.getAttribute("data-amount-per-person");
        let amountPerPerson = Number.parseFloat(amountPerPersonAsText);
        if(amountPerPersontxt!=null && amountPerPersontxt.length>0){
            if(!Number.isNaN(amountPerson)){
                let amount = amountPerPerson*personCount;
                amountAsText = Number.isInteger(amount)?amount.toString(): amount
.toFixed(2);
            }
        }
        span.textContent=amountAsText;
    }else{

    }
};
```

```
// todo : zorg dat setup wordt opgeroepen zodra de DOM-tree is opgebouwd
window.addEventListener("load", setup);
```



Taakbeschrijving Titel

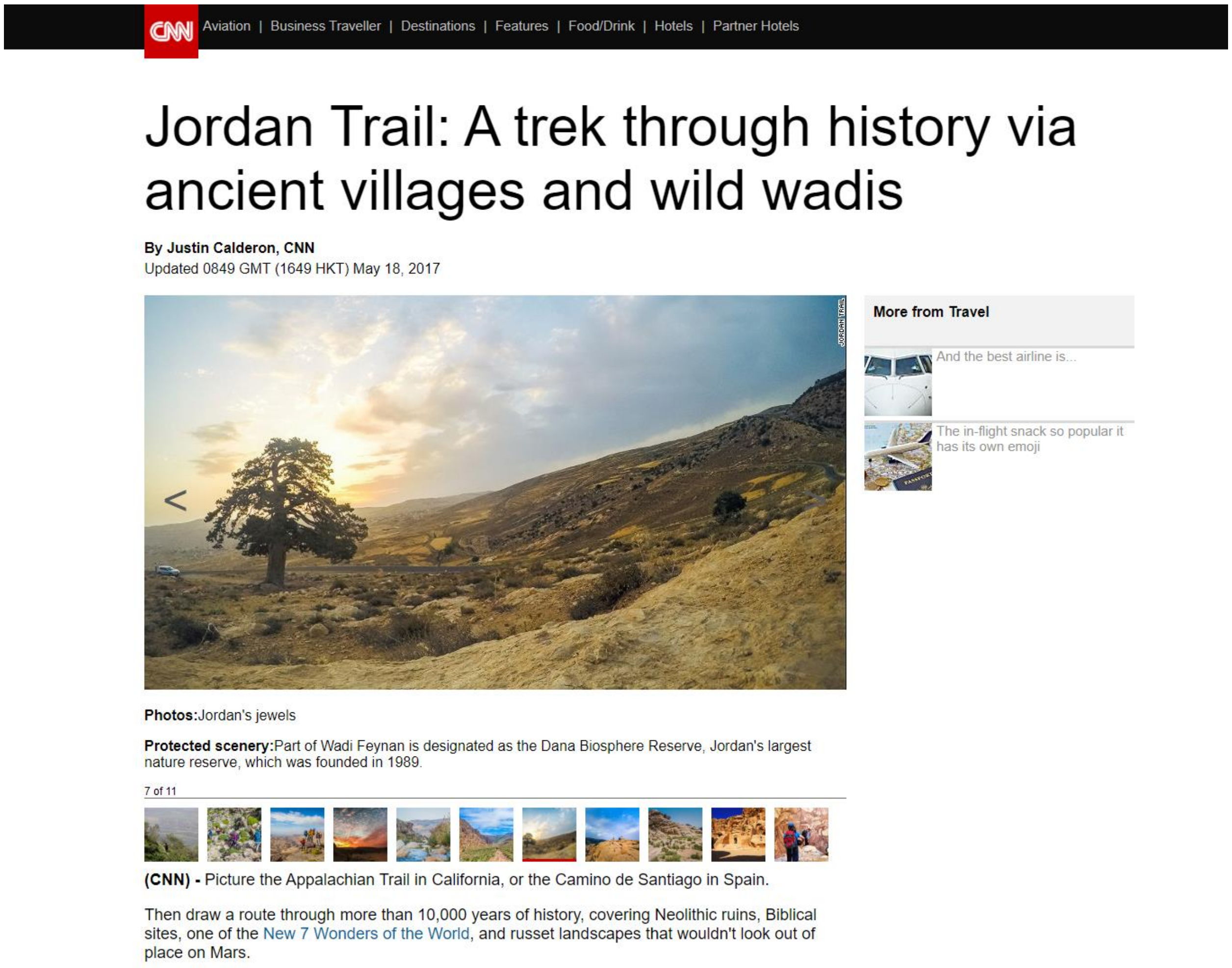
Prioriteit laag hoog

Voeg taak toe

Taken 🟩 🟨 🟥
wekker zetten
ontbijten
naar school gaan
low priority
medium priority
high priority

```
/*let priorities=["low","medium","high"]; // vervang null door een array met drie
 string waarden, resp. low, medium en high*/
    let piorities = ["low","medium","high"];

const getTextForPriorityLevel = (level) => {
    // geef de tekst terug voor dit priority level (bv. 0 is "low" en 2 is "high"
    return piorities[level];
};

const getPriorityLevelForText = (text) => {
    // geef het level terug voor deze priority tekst (bv. "low" is 0 en "high" is
 2)
    // of -1 indien de tekst geen geldige priority tekst is.
   return priorities.indexOf(text);
```

```
};

const setup = () => {
    // Zorg ervoor dat een klik op de #btnAdd button onze 'addTask' event listene
r oproept
    let btnAdd = document.getElementById("btnAdd");
    btnAdd.addEventListener("click", addTask);

    // Zorg ervoor dat een klik op een .dot element onze 'filterTasks' event list
ener oproept
    let dotElement = document.getElementsByClassName("dot");
    for(let i = 0; i < dotElement.length; i++){
        dotElement[i].addEventListener("click", filterTasks);
    }
    // Voeg enkele tasks toe, om snel te kunnen testen
    insertTaskHTML(0, "low priority");
    insertTaskHTML(1, "medium priority");
    insertTaskHTML(2, "high priority");
};

const addTask= () =>{
    // haal de titel op van de task
    let txtDescription = document.getElementById("txtDescription");
    let description = txtDescription.value.trim();

    // haal het priority level op van de task en zet om naar een getal
    let sldLevel = document.getElementById("sldLevel");
    let level = parseInt(sldLevel.value, 10);

    if(!isNaN(level) && description.length >0){
        insertTaskHTML(level, description);
    }
    // maak titel inputveld leeg
    txtDescription.value="";
};

const insertTaskHTML = (level, description) => {
    // Voeg de HTML code toe aan .tasks voor deze task (level is een Number, desc
ription is een string)
    // De task krijgt ook een class volgens het priority level (.low, medium of .
high)
    let levelText = getTextForPriorityLevel(level);
    let html = `<p class = "task ${levelText}" >${description}</p>`;
    // Je hoeft hierbij geen rekening te houden met de actuele filter level!
    // (maw indien wegens de filter enkel 'high' getoond wordt en je voegt een 'l
ow' toe, dan mag deze 'low' zichtbaar zijn)
   let divTask = document.getElementsByClassName("tasks")[0];
   divTask.innerHTML+=html;
};

const filterTasks = (event) => {
    // achterhaal op welke .dot geklikt werd
    let elemFilterDot = event.target;

    // haal de (onzichtbare) tekst op in deze .dot
   let filterText = elmfilterdot.textContent;

    // zet de tekst om naar een priority level (zodat je een Number hebt)
    let filterLevel = getPriorityLevelForText(filterText);
    if(filterLevel!=-1){
        // pas de classes aan van de .task elementen op basis van filterLevel
        adjustForFilter(filterLevel);
    }

};

const adjustForFilter = (filterLevel) => {
    // pas de CSS classes aan van de .task elementen (filterLevel is een Number)
    let txtTasks = document.getElementsByClassName("task");
    for(let i = 0; i< txtTasks.length; i++){
        let txtTask = txtTasks[i];
        txtTasks.classList.remove("hidden");
    }


        // voor elk level dat kleiner is dan wat gevraagd is
       for(let level = 0; level < filterLevel; level++){
           let levelText = getTextForPriorityLevel(level);
       }
            // indien task dit level heeft, verberg ze
        if(txtTask.classList.contains(levelText)){
            txtTask.classList.add("hidden");
        }
};

window.addEventListener("load", setup);
```


CNN
Aviation | Business Traveller | Destinations | Features | Food/Drink | Hotels | Partner Hotels
<
JORDAN TRAIL

Jordan Trail: A trek through history via ancient villages and wild wadis

By Justin Calderon, CNN
Updated 0849 GMT (1649 HKT) May 18, 2017

More from Travel

And the best airline is...

The in-flight snack so popular it has its own emoji

Photos: Jordan's jewels

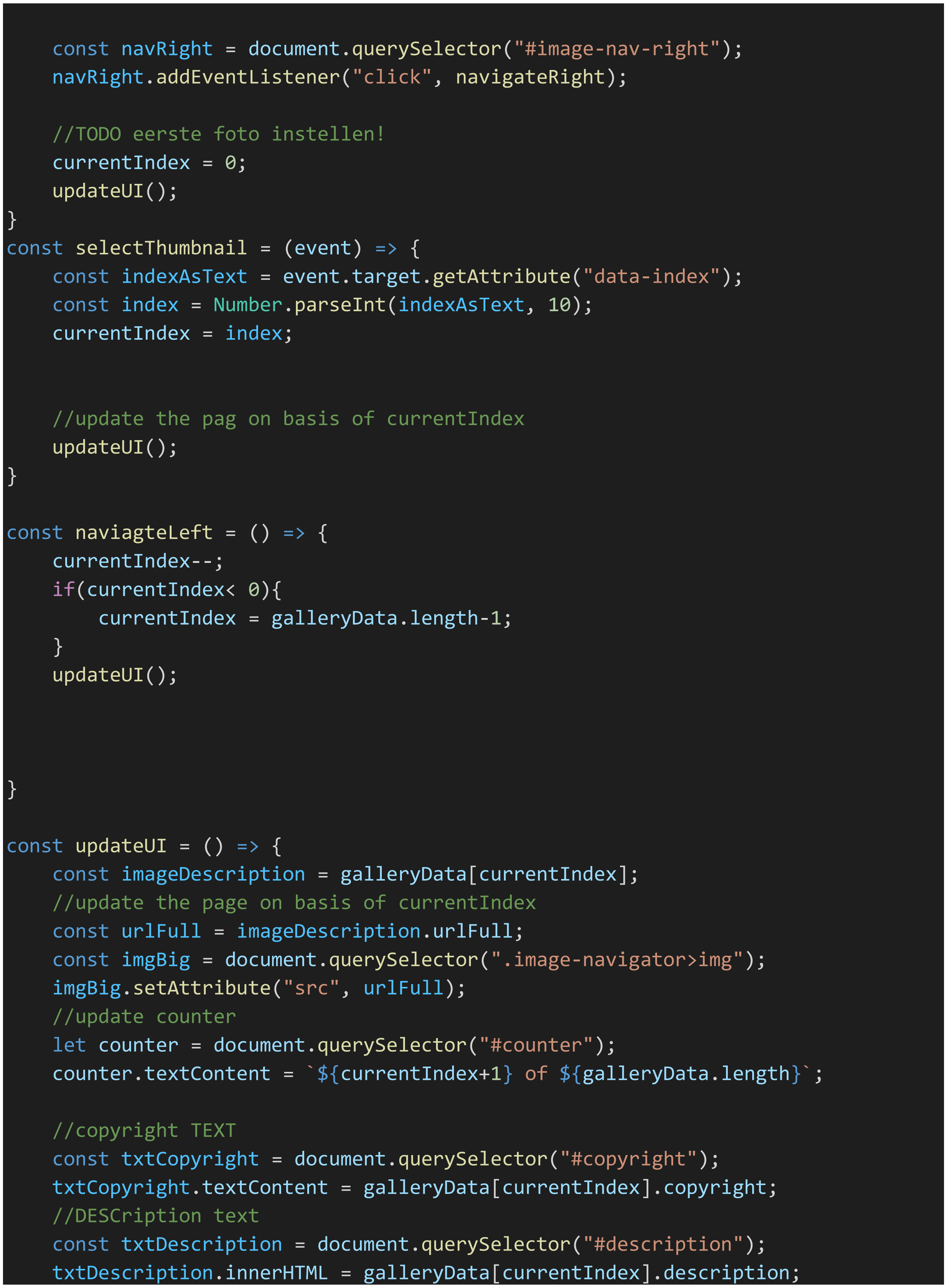
Protected scenery: Part of Wadi Feynan is designated as the Dana Biosphere Reserve, Jordan's largest nature reserve, which was founded in 1989.

7 of 11

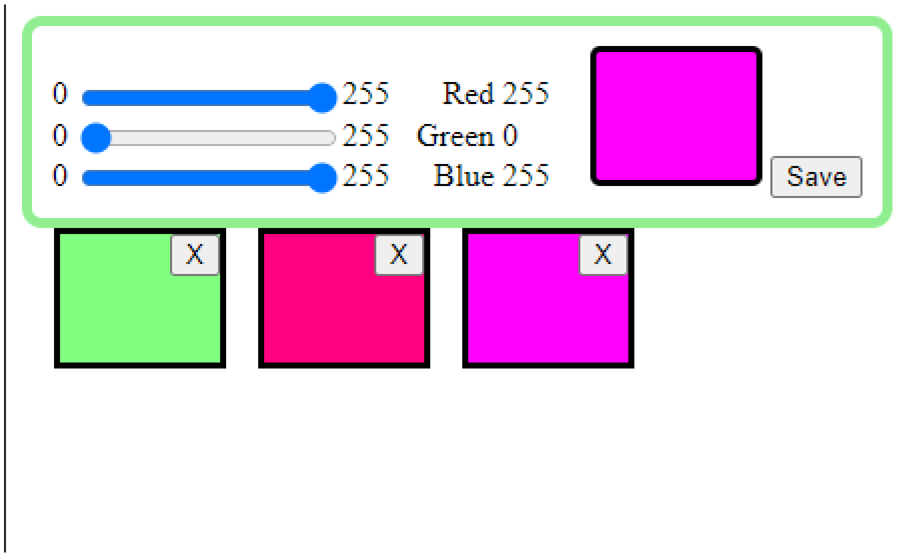
(CNN) - Picture the Appalachian Trail in California, or the Camino de Santiago in Spain.

Then draw a route through more than 10,000 years of history, covering Neolithic ruins, Biblical sites, one of the New 7 Wonders of the World, and russet landscapes that wouldn't look out of place on Mars.

```
let currentIndex = 0;

const setup = () => {
    console.log( galleryData[3].urlFull);

    const thumbnailList = document.querySelector("#thumbnail-list");

    for(let i = 0; i < galleryData.length; i++){
        const imageDescription = galleryData[i];
        const html = `<img src="${imageDescription.urlThumb}" data-
index = ${i}" class="thumbnail">`;
        thumbnailList.insertAdjacentHTML("beforeend", html);
    }

    thumbnailList.addEventListener("click", selectThumbnail);

    const navLeft = document.querySelector("#image-nav-left");
    navLeft.addEventListener("click", naviagteLeft);
```



```
    const navRight = document.querySelector("#image-nav-right");
    navRight.addEventListener("click", navigateRight);

    //TODO eerste foto instellen!
    currentIndex = 0;
    updateUI();
}
const selectThumbnail = (event) => {
    const indexAsText = event.target.getAttribute("data-index");
    const index = Number.parseInt(indexAsText, 10);
    currentIndex = index;


    //update the pag on basis of currentIndex
    updateUI();
}

const naviagteLeft = () => {
    currentIndex--;
    if(currentIndex< 0){
        currentIndex = galleryData.length-1;
    }
    updateUI();



}

const updateUI = () => {
    const imageDescription = galleryData[currentIndex];
    //update the page on basis of currentIndex
    const urlFull = imageDescription.urlFull;
    const imgBig = document.querySelector(".image-navigator>img");
    imgBig.setAttribute("src", urlFull);
    //update counter
    let counter = document.querySelector("#counter");
    counter.textContent = `${currentIndex+1} of ${galleryData.length}`;

    //copyright TEXT
    const txtCopyright = document.querySelector("#copyright");
    txtCopyright.textContent = galleryData[currentIndex].copyright;
    //DESCription text
    const txtDescription = document.querySelector("#description");
    txtDescription.innerHTML = galleryData[currentIndex].description;
    //Thumbnail underline under pic
    const thumbnailList = document.querySelectorAll("#thumbnail-list>img");
    for(let i = 0; i<thumbnailList.length; i++){
        const img = thumbnailList[i];
        const datIndexAsText = img.getAttribute("data-index");
        const index = Number.parseInt(datIndexAsText, 10);
        if(index === currentIndex){
            img.classList.add("activeThumbnail");

        }else{
            img.classList.remove("activeThumbnail");
        }
    }


}
const navigateRight = (event) => {
    currentIndex++;
    if(currentIndex> galleryData.length-1){
        currentIndex = 0;
    }
    updateUI();

    event.preventDefault(); //onlink click stays on the same place

}
window.addEventListener("load", setup);
```



```
const setup = () => {



    let slider = document.getElementsByClassName("slider");
    for(let i = 0; i < slider.length; i++) {
        slider[i].addEventListener("change", update);
        slider[i].addEventListener("input",update);
    }
    update();
    let btnSave = document.querySelector("#btnSave");
    btnSave.addEventListener("click", saveSwatch);
}

const saveSwatch = () => {
    let div = document.createElement("div");
    div.classList.add("swatch");

    let red = document.getElementById("sldRed").value;
    let green = document.getElementById("sldGreen").value;
    let blue = document.getElementById("sldBlue").value;
    div.style.backgroundColor = 'rgb('+ red+ ','+ green+ ','+ blue+')';

    let color = {
        red: red,
        green : green,
        blue : blue
    };
    let colorText = JSON.stringify(color);
    div.setAttribute("data-color", colorText);

    input = document.createElement("input");
    input.setAttribute("type", "button");
    input.value="X";
    input.addEventListener("click", deleteSwatch);

    div.appendChild(input);

    div.addEventListener("click", setColorPickerFromswatch);

    let swatchComponents = document.querySelector("#swatchComponents");
    swatchComponents.appendChild(div);
}

const deleteSwatch = (event) => {
    let input = event.target;
    let div = input.parentNode;
    let section = div.parentNode;
    section.removeChild(div);
    event.stopPropagation();
}

const setColorPickerFromswatch = (event) => {
    let swatch = event.target;
    let JsoncolorText = swatch.getAttribute("data-color");
    let color = JSON.parse(JsoncolorText);


    let sldRed = document.querySelector("#sldRed");
    let sldGreen = document.querySelector("#sldGreen");
    let sldBlue = document.querySelector("#sldBlue");

    sldRed.value = color.red;
    sldGreen.value = color.green;
    sldBlue.value = color.blue;
    update();
}

const update = () => {
    let red = document.getElementById("sldRed").value;
    let green = document.getElementById("sldGreen").value;
    let blue = document.getElementById("sldBlue").value;
    document.getElementById("lblRed").innerHTML = red;
    document.getElementById("lblGreen").innerHTML = green;
    document.getElementById("lblBlue").innerHTML = blue;

    let swatch = document.querySelector("#swatch");
    swatch.style.backgroundColor = 'rgb(' + red + ',' + green + ',' + blue + ')';
}
window.addEventListener("load", setup);
```