



---

# STÄDER OCH LÄNDER

---

Inlämningsuppgift – JavaScript och dataformatet JSON



DEN 2 OKTOBER 2020

WIVIANNE GRAPENHOLT  
grwi20ql@student.ju.se

## Innehåll

<b>Länk</b> .....	2
<b>Beskrivning</b> .....	2
meny( ) .....	2
Huvudmenyn .....	2
Undermeny.....	2
Stadsinformation vyn .....	2
Mina resmål vyn .....	2
<b>Testning och kvalitetssäkring</b> .....	3
<b>Uppfyllda krav</b> .....	3
<b>Förbättringar &amp; buggfixar</b> .....	3
<b>Sammanfattande reflektion</b> .....	4
Planering & Arbetssätt .....	4

## Länk

<https://github.com/wu20-Webbapplikation-MVP/stader-lander-WiviWonderWoman>

## Beskrivning

Själva html filen innehåller bara den grundläggande semantiska strukturen. Först i JavaScript filen kontrolleras om det finns något sparat i localStorage och om så är fallet sparas det även i en array med JSON.parse. Sedan hämtas all data med fetch från de båda JSON filerna och sparas i varsin array. De tre arrayerna har jag valt att spara som properties till ett objekt för att kunna nå och ändrar värde på dem varsomhelst. Objektet har även en property för att spara totala antalet invånare i besökta städer. När länderna hämtas med fetch från JSON filen anropas huvud-funktionen i sista .then, faktum är att all kod ligger inom detta scope.

### meny( )

Här deklarerar jag några variabler som jag behövde nå inom hela scopet men som inte var nödvändiga att ha globalt. Vidare så loopas arrayen med länder igenom och renderar en flexbox meny med länderna. Efter det skapar jag och skriver ut knappen för "mina resmål" (städer jag besökt) och även en instruktion till användaren om hur appen fungerar för att öka både användarvänlighet och tillgänglighet.

### Huvudmenyn

Jag fångar upp hela land-menyn med addEventListener och sparar landets id så att jag kan söka upp rätt land med hjälp av .find i arrayen.

### Undermeny

Med hjälp av landets id använder jag .filter på arrayen med städer för att hitta städer med matchande lands-id och sparar dom i en tillfällig array. Då startar en for loop som går igenom de filtrerade städerna och som genererar en undermeny med alla städer i valda land. Jag fångar upp klick på stad på samma sätt som för länderna, det vill säga jag lyssnar efter stadens id och letar upp staden i den filtrerade arrayen med hjälp av .find.

### Stadsinformation vyn

När rätt stad är hittad sparar jag värdet för stadens namn och invånarantal i lokala variabler för att använda värdena i informationen om staden som presenteras för klienten. Jag passar även på att rendera "besökt"-knappen. När användaren klickar på knappen lägger jag till stadens id i den globala property arrayen med hjälp av .push. Därefter sparar jag hela den uppdaterade arrayen i localStorage med hjälp av JSON.stringify.

### Mina resmål vyn

Denna vy initieras av att användaren klickar på "mina resmål" som har en egen eventListener eftersom den inte har något land-id. Här presenteras alla städer besökaren har valt att spara. Detta åstadkommer jag genom att loopa igenom arrayen med sparade stad id för att hitta rätt stad med .find. Inne i loopen passar jag även på att addera antalet invånare i hittad stad till den totala summan. Listan med städer presenteras samt ett meddelande med resultatet av uträkningen. När det är klart nollställer jag värdet för invånare för att det inte ska dubblas för varje klick på "mina resmål". Undertiden har även knappen för att rensa historiken skapats och renderats i footern. Vid eventuellt klick rensas localStorage med .clear och arrayen nollställs. För användaren visas då en ny vy med ett meddelande som bekräftar att minnet har tömts.

## Testning och kvalitetssäkring

Jag har testat applikationen i flera olika webbläsare och simulerat diverse skärmstorlekar. Funktionaliteten är intakt oavsett däremot är jag inte helt nöjd med landsmenyn på små skärmar eftersom länderna då lägger sig under varandra. Eftersom det endast är en "kosmetisk flaw" så väljer jag att låta det vara. Jag har validerat den genererade html-koden från sidkällan på WC3 Markup Validation Service och får endast två varning:

**Warning:** Article lacks heading. Consider using `h2`-`h6` elements to [add identifying headings to all articles](#).

[From line 23, column 5: to line 23, column 26](#)

```
<article id="article">
```

**Warning:** The `type` attribute is unnecessary for JavaScript resources.

[From line 39, column 1: to line 39, column 31](#)

```
erver --><script type="text/javascript">
```

Jag väljer att inte agera på de varningarna dels för att jag inte behöver någon rubrik och varningen om typ är ingen kod som jag själv skrivit och därför utom min makt.

## Uppfyllda krav

Jag hämtar all data med fetch från JSON-filerna och har därför inte haft någon anledning att modifierat dessa. Som jag beskriver här ovan så visas meny med länderna och ett klick genererar undermenyn med städer som tillhör landet. Informationen om staden samt invånarantal renderas med hjälp av ett klick på vald stad. Att undermenyn dröjer sig kvar tills besökaren klickar på annat land eller mina resmål är ett medvetet val för att användaren lättare ska kunna navigera. Besökt knappen visas nedanför stadens fakta och utför de krav på att spara data som anges i kravspecifikationen. Jag valde en grön färg till knappen för att förstärka att knappen sparar. Jag har valt att ha en "mina resmål" knapp som har samma gröna färg som besökt knappen för att signalera att de hänger ihop. Eftersom jag hade en eventListener på hela land-menyn men ändå ville ge intrycket av att mina resmål ingår i samma meny skapade jag en egen ul då stannar knappen på höger sida oavsett skärmstorlek. Så med hjälp av både färg och placering sticker knappen ut. Jag visar listan på besökta städer och summan av invånare i en ny vy på samma sida. Jag har en knapp för att rensa minnet och har valt att ge den en röd färg för att förtydliga för användaren att knappen raderar all historik.

## Förbättringar & buggfixar

Alla klick på besökt sparar stadens id som ett nytt värde i arrayen och localStorage. Tyvärr visas då även eventuella dubbletter och även den uträknade summan blir då felaktig i vyn för mina resmål. Som jag nämnt tidigare så sparas alla klick på "besökt" som ett nytt värde i arrayen och localStorage. Om jag hade haft mer tid så skulle jag lagt in en for loop där jag kollar så att valda stad inte redan är sparad och på så vis avhjälpa både att det skrivs ut samma stad flera gånger samt att summan av invånare blir korrekt.

Jag upptäckte att eftersom alla klick i meny fälten registreras så får jag felmeddelande i consolen att land / stad id inte hittades. Användaren märker bara detta om hen är i stadsinfo-vyn eller mina resmål-vyn och klickar i landmenyn för då töms vyn på innehåll. I vyn för mina resmål samt minnet är tömt dröjer sig eventListener för stad-menyn kvar vilket endast registreras i consolen.

En annan mindre trevlig feature är att mina resmål-vyn visas även om besökaren inte sparat något så här vill jag lägga till en if sats som kontrollerar om några resmål är sparade och lägga till ett meddelande som visas om det visar sig att minnet är tomt. Det hade också varit bra att lägga till en alert när man klickar på radera knappen för att undvika att historiken raderas av misstag.

Det kilar i mina fingrar att effektivisera (korta ner) koden genom att hämta alla getElementById som properties till det globala objektet men inser att tiden är för knapp.

## Sammanfattande reflektion

Jag är inte helt nöjd med kodstrukturen, känns rörigt när allt är nästlat inuti olika funktioner men å andra sidan är jag väldigt nöjd med själva funktionaliteten. Jag upplever också att koden inte kommer i logisk (ur ett användarperspektiv) ordning. Jag har gjort försök att strukturera om men då tappar jag genast någon funktion. Min förhoppning är väl att jag bara inte är van och att min kod är logisk ur kodperspektiv.

### Planering & Arbetssätt

Inför denna uppgift slarvade jag lite med själva planeringen. Det är också så att när jag väl sätter mig och kodar så hamnar jag lätt i ett hyperfokus flow som gör att jag ibland glömmer spara, ibland glömmer vad branchen / issue handlade om när jag upptäcker nya eller följd problem. Jag har inte heller en bra rutin för att kommentera min kod direkt vilket blir jobbigt att hitta i koden och mycket svårare att gå igenom och kommentera allt i efterhand. Detta är ju egentligen inga problem när jag jobbar på ett projekt själv men kan ställa till det i framtiden när jag samarbetar med andra. Som strategi för att motverka detta måste jag planera tydligare, blockera tid för ett specifikt problem i taget och bara jobba med det och använda påminnelser för att slita mig från skärmen med jämna mellanrum.

I denna uppgift har jag verkligen fått känna på the *"two stages of every programmer"*! Samtidigt som det är väldigt frustrerande att inte förstå vad jag sysslar med, till och med ifrågasätta varför jag ens försöker lära mig, så är den tillfredställande känslan när polletten ramlar ner och allt fungerar så värt allt jobb. Återigen är jag på det stora hela mer än nöjd med resultatet.