INDIVIDUELL UPPGIFT 3

Dynamisk uppdatering av sidinnehåll

Syfte och mål

Syftet med denna uppgift är att lära sig använda JavaScript för att dynamiskt uppdatera delar av en webbsidas innehåll utan att hela sidan behöver laddas om. Detta är ett utmärkande drag av det som ofta kallas AJAX och som präglar moderna webbsidor.

Uppgiftsbeskrivning

Denna examinationsuppgift är individuell. Detta innebär att du ska lösa den självständigt.

Utgångspunkten för uppgiften är ett befintligt projekt som du hittar i filen **uppgift3.zip**. Tanken med uppgiften är att användaren ska kunna söka efter livsmedel och få information om livsmedlens energi- och näringsinnehåll.

En del arbete är redan gjort, men väsentliga delar är lämnade för att bli färdigställda av dig.

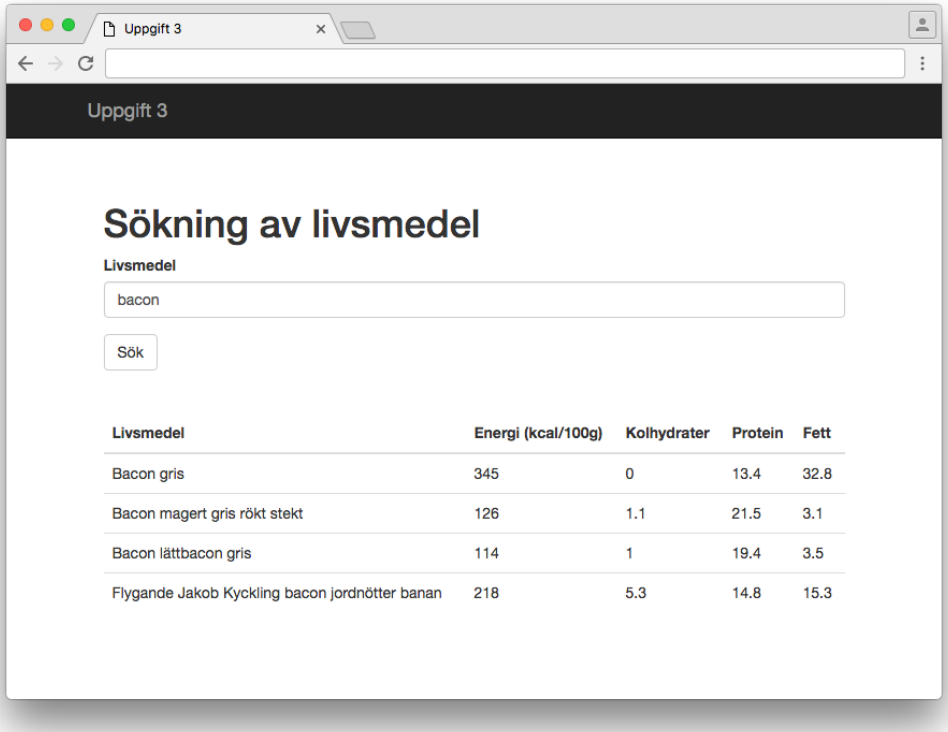
Du får sidan i ett skick där följande funktionalitet och innehåll är klart:

- **index.html** innehåller allt som behövs för uppgiften och ska inte förändras.
- Sidan innehåller ett sökformulär och en tom tabell där resultatet ska presenteras.
- Sidan innehåller en referens till filen **getLivsmedelsData.js**.
- **getLivsmedelsData.js** är tom sånär som på en rad som gömmer tabellen.

Din uppgift är att förändra **getLivsmedelsData.js** så att den uppfyller nedanstående krav:

- När användaren klickar på ”Sök-knappen” ska en sökning göras mot en livsmedelswebbtjänst med hjälp av så kallad JSONP (se särskild rubrik för detta) där den inmatade texten ska utgöra sökfras.
- Sökresultatet ska presenteras som rader i den tabell som finns i **index.html**.
- Resultatet ska utgöras av livsmedelsnamn, energi samt fördelningen av kolhydrater, protein och fett. Var noga att kontrollera så att rätt näringsvärden läggs i rätt kolumn i tabellen.
- När en ny sökning görs ska resultatet från föregående sökning tas bort.
- Om sökningen ger upphov till noll träffar ska tabellen inte visas (inget synligt tabellhuvud).
- Om sökningen ger upphov till någon eller några träffas ska tabellen vara synlig.

Följande bild visar hur ett resultat kan se ut:



Angående webbtjänsten

Sökningen av livsmedelsdata ska ske från en webbtjänst som finns tillgänglig på denna URL:

https://webservice.informatik.umu.se/webservice_livsmedel/getlivsmedel.php

Antag att vi vill söka efter alla livsmedel som innehåller texten ”bacon” i sitt namn. Vi kan då utforma vår URL på följande sätt:

https://webservice.informatik.umu.se/webservice_livsmedel/getlivsmedel.php?namn=bacon&callback=getLivsmedel

Parametern ”namn=bacon” avgör att det är just ”bacon” vi söker efter. Genom att även ange ”callback=getLivsmedel” instruerar vi webbtjänsten att svara på förfrågan genom att skicka data som ett argument i ett anrop till den funktion som angetts som värde (getLivsmedel).

```
getLivsmedel({
   "livsmedel":[
      {
         "namn":"Bacon gris",
         "energi":"345",
         "protein":"13.4",
         "fett":"32.8",
         "kolhydrater":"0"
      },
```

```
      {
         "namn":"Bacon magert gris rökt stekt",
         "energi":"126",
         "protein":"21.5",
         "fett":"3.1",
         "kolhydrater":"1.1"
      },
      {
         "namn":"Bacon lättbacon gris",
         "energi":"114",
         "protein":"19.4",
         "fett":"3.5",
         "kolhydrater":"1"
      },
      {
         "namn":"Flygande Jakob Kyckling bacon jordnötter banan",
         "energi":"218",
         "protein":"14.8",
         "fett":"15.3",
         "kolhydrater":"5.3"
      }
   ],
   "responseStatus":200
})
```


I kursens femte tema finns ett exempel som mer konkret visar hur användning av JSONP fungerar, men som arbetar mot en annan webbtjänst.

Att tänka på

- Examinationen är ett enskilt arbete, vilket innebär att likartade svar kan komma att underkännas. Svar i vilka plagiat upptäcks, kommer att underkännas.

Bedömningskriterier

Uppgiften kan betygsättas godkänd eller underkänd. För att klara uppgiften skall studenten ha fullföljt uppgiften i sin helhet enligt beskrivningen ovan. Det innebär att studenten har:

✓ Utvecklat en JavaScript-baserad lösning som uppfyller samtliga uppsatta krav.

✓ Lämnat in JavaScript-filen **getLivsmedelsData.js** via Cambro innan angiven sluttid för uppgiften.