

Informatik

Tentamen i JavaScript för webbutveckling 5 hp (ISGB15-provkod 1500) onsdagen den 24 april 08.15-13.15.

Ansvariga lärare:	Peter Bellström och Pierre Sjöberg		
Betygsgränser:	Max:	40	
	Godkänd:	20	
	Väl Godkänd:	30	

Maxpoäng på respektive uppgift anges inom parantes.

Inga hjälpmedel!

Förklara bilder och begrepp noggrant!

OBS! Nytt blad för varje uppgift och glöm inte skriva anonymitetskod.

LYCKA TILL!

Fråga 1. (6 p)

Författarna av kurslitteraturen *DOM Scripting Web Design with JavaScript and the Document Object Model* beskriver fyra olika varianter, två som är obtrusive och två som är unobtrusive, att skapa lyssnare för ett a-element. Redogör kortfattat för respektive variant samt skriv de instruktioner som krävs för att med respektive variant anropa funktionen `demo()` när användaren klickar på länken.

Fråga 2. (6 p)

Redogör exakt för vad du som användare ser innanför body-elementet (när du trycker F12 i Firefox och väljer fliken "Inspekterare" samt klickar på alla pilar för att visa all text efter att sidan har laddats och JavaScript-koden har exekverats (se HTML5-koden och JavaScript-koden följande sida). Observera att du skall redogöra för vad du ser och inte hur webbläsaren tolkar det du ser!

```

1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <title>Uppgift 2.</title>
5     <meta charset="utf-8">
6   </head>
7   <body>
8     <div>
9       <table>
10        <thead>
11          <tr>
12            <th></th><th>1</th><th>2</th><th>3</th>
13          </tr>
14        </thead>
15        <tbody>
16          <tr>
17            <th>1</th><td></td><td></td><td></td>
18          </tr>
19          <tr>
20            <th>2</th><td></td><td></td><td></td>
21          </tr>
22          <tr>
23            <th>3</th><td></td><td></td><td></td>
24          </tr>
25        </tbody>
26      </table>
27    </div>
28    <script src="uppgift2.js"></script>
29  </body>
30 </html>

```

```

1 "use strict";
2
3 function changeDOM() {
4   let trRefs = document.querySelectorAll( "tbody tr" );
5   let trRef = document.createElement( "tr" );
6   trRef.appendChild( document.createElement( "th" ) );
7   document.querySelector( "tbody" ).appendChild( trRef );
8
9   for( let i = 0; i < trRefs.length; i++ ) {
10     let tdRefs = trRefs.item(i).querySelectorAll( "td" );
11     for( let j = 0; j < tdRefs.length; j++ ) {
12       let txt = tdRefs.item( j ).appendChild( document.createTextNode( (i+1) * (j+1) ) );
13     }
14   }
15
16   for( let k = 0; k < trRefs.length; k++ ) {
17     let nbr = "";
18     let txt = null;
19     let tdRef = null;
20
21     for( let l = 0; l < trRefs.length; l++ ) {
22
23       let tdRefs = trRefs.item( l ).querySelectorAll( "td" );
24
25       nbr += tdRefs.item( k ).firstChild.nodeValue;
26     }
27
28     txt = document.createTextNode( nbr );
29     tdRef = document.createElement( "td" );
30     tdRef.appendChild( txt );
31     trRef.appendChild( tdRef );
32   }
33 }
34 changeDOM();

```

Fråga 3. (8 p)

- a) Skriv JavaScript-funktionen `modifyDOM()` i vilken du skall modifiera det användaren ser för alla p-element som finns i div-elementet. Efter att funktionen har exekverat skall det användaren ser i varje p-element, texten som finns mellan start och slut elementet, vara en siffra i vilken följd p-elementet återkommer i div-elementet. Din lösning skall också vara skriven så att den fungerar med p-element utan befintlig text och med p-element med befintlig text inklusive tab, enter och mellanslag. Att använda egenskaperna `textContent`, `innerHTML` eller `outerHTML` är inte tillåtet utan du skall använda dig av metoder och egenskaper som används och rekommenderas i kurslitteraturen. Bilden nedan kan vara till hjälp men skall bara ses som ett exempel på antalet p-element och dess innehåll. (6p)

```
1  <!DOCTYPE html>
2  <html lang="en">
3      <head>
4          <title>Uppgift 3.</title>
5          <meta charset="utf-8">
6      </head>
7      <body>
8          <p>Ett första p-element!</p>
9          <div>
10              <p></p>
11              <p> </p>
12              <p>String</p>
13              <p>Integer</p>
14              <p>
15              </p>
16          </div>
17          <script src="uppgift3.js"></script>
18      </body>
19  </html>
```

- b) Utgå från HTML5-koden ovan och skriv koden för att lägga till en lyssnare för klick-händelsen på div-elementet. Utgå ifrån att det finns text för alla p-element. När klick-händelsen inträffar skall du:
- Avbryta default-beteendet för klick-händelsen på div-elementet.
 - Avbryta att klick-händelsen på div-elementet propageras ("bubblar") till div-elementets förälder.
 - Till konsolen skriva ut texten på det p-element som användaren klickar på samt nodtypen som fångar händelsen. (2p)

Fråga 4. (5 p)

Utgå från HTML-koden nedan och skriv Javascript-koden till uppgift4.js som behövs för att få följande beteende:

- När sidan laddats klart skall taggen med id "sida" göras **osynlig**.
- När användaren klickar på något av menyvalen skall det aktuella valet **byta textfärg** till röd.
- Om användaren klickar på "Menyval 1" skall taggen med id "sida" **göras synlig**.
- Vid klick på "Menyval 2" skall innehållet i h1-taggen **byta innehåll** till "Sida 2".
- Vid klick på "Menyval 3" skall det **läggas in ny bild** under paragrafen i "sida". Bilden ligger i samma mapp som scriptet och heter "bild.jpg". bildtaggen skall även ha **alt-attributet** satt till "en bild".

Att använda egenskaperna **innerHTML** eller **outerHTML** är inte tillåtet utan du skall använda dig av metoder och egenskaper som används och rekommenderas i kurslitteraturen.

```
<!DOCTYPE html>
<html lang="sv">
  <head>
    <meta charset="utf-8">
    <title>Uppgift 4</title>
  </head>
  <body>
    <nav>
      <ul>
        <li value="1">Menyval 1</li>
        <li value="2">Menyval 2</li>
        <li value="3">Menyval 3</li>
      </ul>
    </nav>
    <main>
      <div id="sida">
        <h1>Sida 1</h1>
        <p>En bra text till första sidan...</p>
      </div>
    </main>
    <script type="text/javascript" src="uppgift4.js"></script>
  </body>
</html>
```

Fråga 5. (10 p)

HTML-dokumentet nedan innehåller ett formulär för att välja hur du vill bli kontaktad av en säljare.

Skriv Javascript-koden för funktionen `validate()` som körs när formuläret skickas.

Funktionen ska uppfylla följande krav:

1. Du ska validera fältvärdena enligt följande:
 - Någon av *radioknapparna* måste vara vald.
 - *Textarean* måste innehålla minst 5 tecken.
 - *Tid* måste väljas till något annat än "Välj en tid" om *rbLater* är vald
 - *Tid* får inte vara vald till annat än "Välj en tid" om *rbNow* är vald
2. Undantagshantering ska användas.
3. Om något av villkoren inte uppfylls ska ett undantagsobjekt kastas som innehåller ett lämpligt felmeddelande.
4. Om ett undantag kastas ska ett felmeddelande visas i `h1`-elementet med id *errmsg*.
5. Om ett undantag kastas ska funktionen returnera *false*.
6. Om inget undantag kastas ska funktionen returnera *true*.

```
<!DOCTYPE html>
<html lang="sv">
  <head>
    <meta charset="utf-8">
    <title>Uppgift 5</title>
  </head>
  <body>
    <form method="post" onsubmit="return validate();">
      <h1 id="errmsg"></h1>
      <h1>Boka kontakt med säljare</h1>
      <div>
        <input type="radio" name="rbWhen" id="rbNow">
        <label for="rbnow">Ring mig nu</label>
        <input type="radio" name="rbWhen" id="rbLater">
        <label for="rbnow">Ring mig senare</label>
      </div>
      <div>
        <label for="time">Välj klockslag</label>
        <select id="time">
          <option value="">Välj en tid</option>
          <option value="8">8:00</option>
          <option value="10">10:00</option>
          <option value="12">12:00</option>
        </select>
      </div>
      <div>
        <label for="txtsubject">Beskriv ditt ärende</label>
        <br>
        <textarea id="txtsubject" rows="10" cols="30"></textarea>
      </div>
      <input type="submit" value="Skicka">
    </form>
    <script type="text/javascript" src="uppgift5.js"></script>
  </body>
</html>
```

Fråga 6. (5 p)

När användaren öppnar HTML-dokumentet nedan skapas ett antal DOM-element. Redogör för hur innehållet i main-elementet ser ut efter att sidan laddats. Du ska alltså förklara hur DOM-strukturen ser ut, dvs. det som syns i inspekteraren/granskaren i webbläsarens utvecklarverktyg när du fäller ut main-elementet och eventuella underelement.

```
<!DOCTYPE html>
<html lang="sv">
  <head>
    <meta charset="utf-8">
    <title>Uppgift 6</title>
  </head>
  <body>
    <main></main>
    <script type="text/javascript" src="uppgift6.js"></script>
  </body>
</html>
```

```
'use strict';

let data = [
  {ras: 'Schäfer', namn: 'Urban'},
  {ras: 'Pudel', namn: 'Ulla'},
  {ras: 'Chiwawa', namn: 'Åke'},
  {ras: 'Blandras', namn: 'Gunnel'},
];

function fixHeading(ref, contents) {
  let header = document.createElement('h1');
  ref.appendChild(header);
  header.textContent=contents;
}

function fixParagraph(ref, contents) {
  let pararaph = document.createElement('p');
  let text = document.createTextNode(contents);
  ref.appendChild(pararaph);
  pararaph.appendChild(text);
}

function addBtn(ref) {
  let btn = document.createElement('button');
  btn.textContent='skicka';
  ref.appendChild(btn);
}

function init() {

  let main = document.querySelector('main');

  for(let i=0; i < data.length - 1; i++) {
    fixHeading(main, data[i].ras);
    fixParagraph(main, data[i+1].namn);
    if(i === data.length) {
      addBtn(main);
    }
  }
}

init();
```